Administration Guide

FortiOS 7.2.0

FORTINET DOCUMENT LIBRARY
https://docs.fortinet.com

FORTINET VIDEO LIBRARY
https://video.fortinet.com

FORTINET BLOG
https://blog.fortinet.com

CUSTOMER SERVICE & SUPPORT
https://support.fortinet.com

FORTINET TRAINING & CERTIFICATION PROGRAM
https://www.fortinet.com/training-certification

FORTINET TRAINING INSTITUTE
https://training.fortinet.com

FORTIGUARD LABS
https://www.fortiguard.com

END USER LICENSE AGREEMENT
https://www.fortinet.com/doc/legal/EULA.pdf

FEEDBACK
Email: techdoc@fortinet.com

December 13, 2022
FortiOS 7.2.0 Administration Guide
01-720-791905-20221213

TABLE OF CONTENTS

Change Log

Date	Change Description
2022-03-31	Initial release.
2022-04-08	Added NAT46 and NAT64 policy and routing configurations on page 838.
2022-04-13	Added Configuring OS and host check on page 1745.
2022-04-14	Added Protocol options on page 915.
2022-04-21	Added Configuring FortiVoice on page 2269.
2022-04-25	Updated Using FortiSandbox inline scanning with antivirus on page 1113, Retail environment guest access on page 1799, Outbound firewall authentication with Azure AD as a SAML IdP on page 1869, and the DNS filter on page 1164 section.
2022-04-27	Added Configuring a proxy server for FortiGuard updates on page 2142. Updated Phase 1 configuration on page 1329.
2022-04-29	Updated SD-WAN rules on page 538.
2022-05-02	Updated Virtual Domains on page 1994 section. Moved Inter-VDOM routing configuration example: Internet access on page 2007.
2022-05-03	Added Advanced and specialized logging on page 2564. Updated Configuring FortiWeb on page 2262.
2022-05-04	Updated Configure FortiGate with FortiExplorer using BLE on page 48.
2022-05-12	Added Virtual VLAN switch on page 170.
2022-05-13	Added Basic RIPng example on page 336.
2022-05-17	Added Migrating from SSL VPN to ZTNA on page 1069.
2022-05-20	Added VDOM overview on page 1995, Inter-VDOM routing configuration example: Partial-mesh VDOMs on page 2017. Updated Virtual Domains on page 1994, General configurations on page 2000, Inter-VDOM routing configuration example: Internet access on page 2007.
2022-05-25	Added ZTNA session-based form authentication on page 1062.
2022-05-27	Added VIP groups on page 896. Updated ZTNA TCP forwarding access proxy example on page 1002.
2022-05-30	Updated Abbreviated TLS handshake after HA failover on page 2080 and Using configuration save mode on page 1991.
2022-05-31	Added SSL VPN with Azure AD SSO integration on page 1736.
2022-06-02	Added Physical interface on page 162 and Loopback interface on page 183.

Date	Change Description
	Updated Interfaces on page 135, VDOM overview on page 1995, General configurations on page 2000, Inter-VDOM routing configuration example: Internet access on page 2007, and Inter-VDOM routing configuration example: Partial-mesh VDOMs on page 2017.
2022-06-06	Updated Out-of-band management with reserved management interfaces on page 2047, ZTNA proxy access with SAML authentication example on page 1013, DDNS on page 208, and Proxy mode stream-based scanning on page 1090.
2022-06-10	Updated FGSP session synchronization between different FortiGate models or firmware versions on page 2106, Routing concepts on page 295, and VM license on page 2615. Added Include usernames in logs on page 1924.
2022-06-13	Added Trusted platform module support on page 1992.
2022-06-15	Updated Restricted SaaS access on page 251.
2022-06-17	Updated Health check options on page 512.
2022-06-20	Updated Phase 1 configuration on page 1329, and IKE Mode Config clients on page 1428.
2022-06-22	Added Traffic shaping policies on page 920 and VRRP on page 2112. Updated Traffic shaping profiles on page 924.
2022-06-30	Added Troubleshooting BGP on page 384.
2022-07-05	Updated Text strings on page 26.
2022-07-11	Added SSL VPN split DNS on page 1633. Updated HA active-active cluster setup on page 2037.
2022-07-12	Added Captive portals on page 160.
2022-07-15	Updated Configure FortiGate with FortiExplorer using BLE on page 48.
2022-07-20	Added LEDs on page 69.
2022-07-22	Added SSL VPN with Okta as SAML IdP on page 1730.
2022-07-27	Added Disabling the FortiGuard IP address rating on page 1307. Updated Administrators on page 1937.
2022-07-28	Added BFD on page 388 and BFD for multihop path for BGP on page 393.
2022-08-04	Updated System Events log page on page 2541 and Security Events log page on page 2544.
2022-09-27	Added ICAP scanning with SCP and FTP on page 1288.
2022-11-04	Updated Enhanced hashing for LAG member selection on page 181 and Access control lists on page 797.
2022-12-13	Updated NetFlow templates on page 435.

Getting started

This section explains how to get started with a FortiGate.

Differences between models

Not all FortiGates have the same features, particularly entry-level models (models 30 to 90). A number of features on these models are only available in the CLI.

Consult your model's QuickStart Guide, hardware manual, or the Feature / Platform Matrix for further information about features that vary by model.

FortiGate models differ principally by the names used and the features available:

- Naming conventions may vary between FortiGate models. For example, on some models the hardware switch interface used for the local area network is called *lan*, while on other units it is called *internal*.
- Certain features are not available on all models. Additionally, a particular feature may be available only through the CLI on some models, while that same feature may be viewed in the GUI on other models.
  If you believe your FortiGate model supports a feature that does not appear in the GUI, go to *System > Feature Visibility* and confirm that the feature is enabled. For more information, see Feature visibility on page 2163.

Low encryption models

Some FortiGate models support a low encryption (LENC) license. With an LENC license, FortiGate devices are considered low encryption models and are identified by *LENC*, for example FG-100E-LENC.

LENC models cannot use or inspect high encryption protocols, such as 3DES and AES. LENC models only use 56-bit DES encryption to work with SSL VPN and IPsec VPN, and they are unable to perform SSL inspection.

For a list of FortiGate models that support an LENC license, see FortiGate LENC Models.

Using the GUI

This section presents an introduction to the graphical user interface (GUI) on your FortiGate.

The following topics are included in this section:

- Connecting using a web browser
- Menus
- Tables

- Entering values
- GUI-based global search
- Loading artifacts from a CDN on page 29

For information about using the dashboards, see Dashboards and Monitors on page 76.

Connecting using a web browser

In order to connect to the GUI using a web browser, an interface must be configured to allow administrative access over HTTPS or over both HTTPS and HTTP. By default, an interface has already been set up that allows HTTPS access with the IP address 192.168.1.99.

Browse to https://192.168.1.99 and enter your username and password. If you have not changed the admin account's password, use the default user name, `admin`, and leave the password field blank.

The GUI will now display in your browser, and you will be required to provide a password for the administrator account.

To use a different interface to access the GUI:

1. Go to *Network > Interfaces* and edit the interface you wish to use for access. Take note of its assigned IP address.
2. In *Administrative Access*, select *HTTPS*, and any other protocol you require. You can also select *HTTP*, although this is not recommended as the connection will be less secure.
3. Click *OK*.
4. Browse to the IP address using your chosen protocol.
   The GUI will now be displayed in your browser.

Menus

If you believe your FortiGate model supports a menu that does not appear in the GUI, go to *System > Feature Visibility* and ensure the feature is enabled. For more information, see Feature visibility on page 2163.

The GUI contains the following main menus, which provide access to configuration options for most FortiOS features:

Dashboard	The dashboard displays various widgets that display important system information and allow you to configure some system options. For more information, see Dashboards and Monitors on page 76.
Network	Options for networking, including configuring system interfaces and routing options. For more information, see Network on page 135.
Policy & Objects	Configure firewall policies, protocol options, and supporting content for policies, including schedules, firewall addresses, and traffic shapers. For more information, see Policy and Objects on page 758.
Security Profiles	Configure your FortiGate's security features, including Antivirus, Web Filter, and Application Control.

	For more information, see Security Profiles on page 1084.
VPN	Configure options for IPsec and SSL virtual private networks (VPNs). For more information, see IPsec VPNs on page 1328 and SSL VPN on page 1617.
User & Authentication	Configure user accounts, groups, and authentication methods, including external authentication and single sign-on (SSO).
WiFi & Switch Controller	Configure the unit to act as a wireless network controller, managing the wireless Access Point (AP) functionality of FortiWiFi and FortiAP units. On certain FortiGate models, this menu has additional features allowing for FortiSwitch units to be managed by the FortiGate. For more information, see Wireless configuration on page 1932 and Switch Controller on page 1933.
System	Configure system settings, such as administrators, HA, FortiGuard, and certificates. For more information, see System on page 1934.
Security Fabric	Access the physical topology, logical topology, automation, and settings of the Fortinet Security Fabric. For more information, see Fortinet Security Fabric on page 2204.
Log & Report	Configure logging and alert email as well as reports. For more information, see Log and Report on page 2539.

Tables

Many GUI pages contain tables of information that can be filtered and customized to display specific information in a specific way. Some tables allow content to be edited directly on that table, or rows to be copied and pasted.

Navigation

Some tables contain information and lists that span multiple pages. Navigation controls will be available at the bottom of the page.

Filters

Filters are used to locate a specific set of information or content in a table. They can be particularly useful for locating specific log entries. The filtering options vary, depending on the type of information in the log.

Depending on the table content, filters can be applied using the filter bar, using a column filter, or based on a cell's content. Some tables allow filtering based on regular expressions.

Administrators with read and write access can define filters. Multiple filters can be applied at one time.

To manually create a filter:

1. Click *Add Filter* at the top of the table. A list of the fields available for filtering is shown.
2. Select the field to filter by.
3. Enter the value to filter by, adding modifiers as needed.
4. Press *Enter* to apply the filter.

To create a column filter:

1. Click the filter icon on the right side of the column header
2. Choose a filter type from the available options.
3. Enter the filter text, or select from the available values.
4. Click *Apply*.

To create a filter based on a cell's content:

1. Right click on a cell in the table.
2. Select a filtering option from the menu.

Column settings

Columns can be rearranged, resized, and added or removed from tables.

To add or remove columns:

1. Right a column header, or click the gear icon on the left side of the header row that appears when hovering the cursor over the headers.
2. Select columns to add or remove.
3. Click *Apply*.

To rearrange the columns in a table:

1. Click and drag the column header.

To resize a column:

1. Click and drag the right border of the column header.

To resize a column to fit its contents:

1. Click the dots or filter icon on the right side of the column header and select *Resize to Contents*.

To resize all of the columns in a table to fit their content:

1. Right a column header, or click the gear icon on the left side of the header row that appears when hovering the cursor over the headers.
2. Click *Best Fit All Columns*.

To reset a table to its default view:

1. Right a column header, or click the gear icon on the left side of the header row that appears when hovering the cursor over the headers.
2. Click *Reset Table*.
   Resetting a table does not remove filters.

Editing objects

In some tables, parts of a configuration can be edited directly in the table. For example, security profiles can be added to an existing firewall policy by clicking the edit icon in a cell in the *Security Profiles* column.

Copying rows

In some tables, rows can be copied and pasted using the right-click menu. For example, a policy can be duplicated by copying and pasting it.

Entering values

Numerous fields in the GUI and CLI require text strings or numbers to be entered when configuring the FortiGate. When entering values in the GUI, you will be prevented from entering invalid characters, and a warning message will be shown explaining what values are not allowed. If invalid values are entered in a CLI command, the setting will be rejected when you apply it.

- Text strings on page 26
- Numbers on page 27

Text strings

Text strings are used to name entities in the FortiGate configuration. For example, the name of a firewall address, administrator, or interface are all text strings.

The following characters cannot be used in text strings, as they present cross-site scripting (XSS) vulnerabilities:

- `“` - double quotes
- `'` - single quote
- `>` - greater than
- `<` - less than

Most GUI text fields prevent XSS vulnerable characters from being added.

VDOM names and hostnames can only use numbers (0-9), letters (a-z and A-Z), dashes, and underscores.

The `tree` CLI command can be used to view the number of characters allowed in a name field. For example, entering the following commands show that a firewall address name can contain up to 80 characters, while its FQDN can contain 256 characters:

```
tree firewall address
-- [address] --*name    (80)
             |- uuid
             |- subnet
             |- type
             |- sub-type
             |- clearpass-spt
             |- [macaddr] --*macaddr    (128)
             |- start-ip
             |- end-ip
             |- fqdn    (256)
             |- country         (3)
             |- wildcard-fqdn   (256)
             |- cache-ttl       (0,86400)
             |- wildcard
             |- sdn     (36)
             |- [fsso-group] --*name    (512)
             |- interface       (36)
             |- tenant  (36)
             |- organization    (36)
             |- epg-name        (256)
             |- subnet-name     (256)
             |- sdn-tag         (16)
             |- policy-group    (16)
             |- obj-tag         (256)
             |- obj-type
             |- tag-detection-level     (16)
             |- tag-type        (64)
             |- dirty
             |- comment
             |- associated-interface    (36)
             |- color   (0,32)
             |- filter
             |- sdn-addr-type
             |- node-ip-only
             |- obj-id
             |- [list] --*ip    (36)
                       |- obj-id        (128)
                       +- net-id        (128)
             |- [tagging] --*name       (64)
                          |- category   (64)
                          +- [tags] --*name     (80)
             |- allow-routing
             +- fabric-object
```

Numbers

Numbers are used to set sizes, rated, addresses, port numbers, priorities, and other such numeric values. They can be entered as a series of digits (without commas or spaces), in a dotted decimal format (such as IP addresses), or separated by colons (such as MAC addresses). Most numeric values use base 10 numbers, while some use hexadecimal values.

Most GUI and CLI fields prevent invalid numbers from being entered. The CLI help text includes information about the range of values allowed for applicable settings.

GUI-based global search

The global search option in the GUI allows users to search for keywords appearing in objects and navigation menus to quickly access the object and configuration page. Click the magnifying glass icon in the top-left corner of the banner to access the global search.

The global search includes the following features:

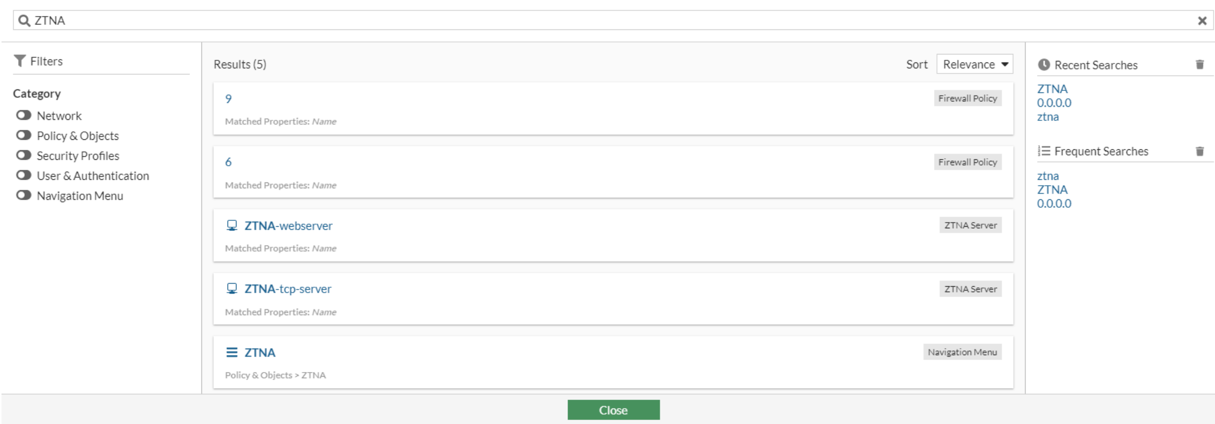
- Keep a history of frequent and recent searches
- Sort results alphabetically by increasing or decreasing order, and relevance by search weight
- Search by category
- Search in Security Fabric members (accessed by the Security Fabric members dropdown menu in the banner)

Examples

In this example, searching for the word *ZTNA* yields the following results:

- Firewall policy object *9*, which contains ZTNA in the property value, *Name*. The name of the policy is ZTNA-TCP.
- ZTNA server object *ZTNA-webserver*, which contains ZTNA in the property value, *Name*.
- ZTNA navigation menu item under *Policy & Objects* > *ZTNA*.

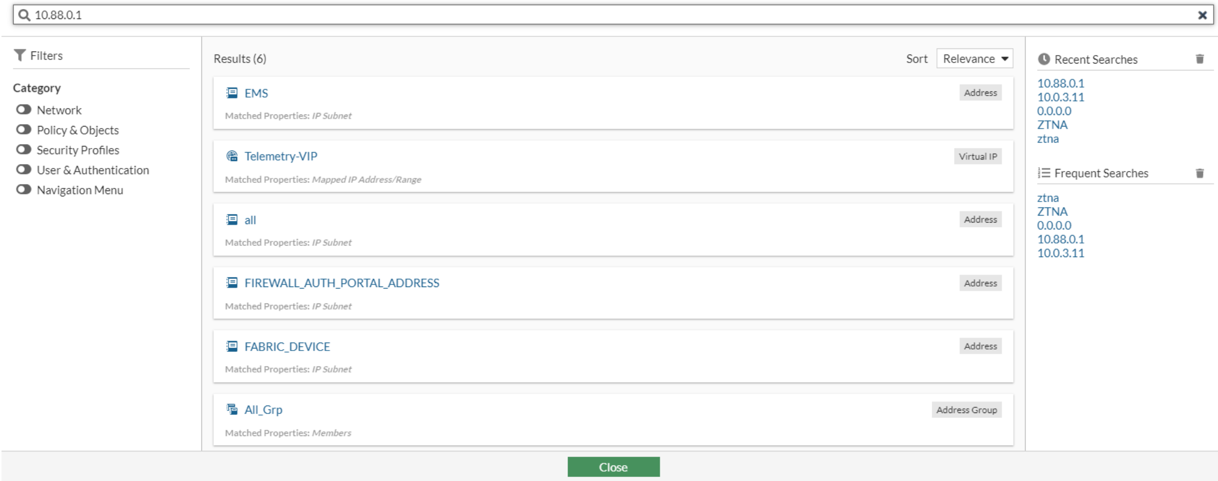
Since CMDB objects have a higher search weight (50) than navigation objects (20), the navigation menu result appears at the bottom.



In this example, searching for the address *10.88.0.1* yields the following results:

- Address object *EMS* that has a subnet of 10.88.0.1/32, which matches the search term.
- Virtual IP object *Telemetry-VIP* that has a mapped IP range of 10.88.0.1, which matches the search term.
- Address objects *all*, *FIREWALL_AUTH_PORTAL_ADDRESS*, and *FABRIC_DEVICE* that have IP subnets of 0.0.0.0/0, which the searched term falls into.
- Address group object *All_Grp* that contains members addresses that have IP subnets of 0.0.0.0/0, which the searched term falls into.

Sorting by *Relevance* will display address objects that are more closely matched at the top (10.88.0.1), and more loosely matched at the bottom (0.0.0.0).



Loading artifacts from a CDN

To improve GUI performance, loading static GUI artifacts cached in CDN (content delivery network) servers closer to the user instead of the FortiGate can be enabled. This allows the GUI to load more quickly with less latency for administrators who are accessing the FortiGate remotely. Upon failure, the files fall back to loading from the FortiGate. The CDN is only used after successful administrator logins.

To configure loading static GUI files from a CDN:

```
config system global
    set gui-cdn-usage {enable | disable}
end
```

Using the CLI

The Command Line Interface (CLI) can be used in lieu of the GUI to configure the FortiGate. Some settings are not available in the GUI, and can only be accessed using the CLI.

This section briefly explains basic CLI usage. For more information about the CLI, see the FortiOS CLI Reference.

- Connecting to the CLI on page 29
- CLI basics on page 32
- Command syntax on page 38
- Subcommands on page 41
- Permissions on page 43

Connecting to the CLI

You can connect to the CLI using a direct console connection, SSH, the FortiExplorer app on your iOS device, or the CLI console in the GUI.

You can access the CLI outside of the GUI in three ways:

- Console connection: Connect your computer directly to the console port of your FortiGate.
- SSH access: Connect your computer through any network interface attached to one of the network ports on your FortiGate.
- FortiExplorer: Connect your device to the FortiExplorer app on your iOS device to configure, manage, and monitor your FortiGate. See FortiExplorer management on page 43 for details.

To open a CLI console, click the _> icon in the top right corner of the GUI. The console opens on top of the GUI. It can be minimized and multiple consoles can be opened.

To edit policies and objects directly in the CLI, right-click on the element and select *Edit in CLI*.

Console connection

A direct console connection to the CLI is created by directly connecting your management computer or console to the FortiGate using its DB-9 or RJ-45 console port.

Direct console access to the FortiGate may be required if:

- You are installing the FortiGate for the first time and it is not configured to connect to your network.
- You are restoring the firmware using a boot interrupt. Network access to the CLI will not be available until after the boot process has completed, making direct console access the only option.

To connect to the FortiGate console, you need:

- A console cable to connect the console port on the FortiGate to a communications port on the computer. Depending on your device, this is one of:
  - null modem cable (DB-9 to DB-9)
  - DB-9 to RJ-45 cable (a DB-9-to-USB adapter can be used)
  - USB to RJ-45 cable
- A computer with an available communications port
- Terminal emulation software

To connect to the CLI using a direct console connection:

1. Using the console cable, connect the FortiGate unit’s console port to the serial communications (COM) port on your management computer.
2. Start a terminal emulation program on the management computer, select the COM port, and use the following settings:

Bits per second	9600
Data bits	8
Parity	None
Stop bits	1
Flow control	None

3. Press *Enter* on the keyboard to connect to the CLI.
4. Log in to the CLI using your username and password (default: *admin* and no password).
   You can now enter CLI commands, including configuring access to the CLI through SSH.

SSH access

SSH access to the CLI is accomplished by connecting your computer to the FortiGate using one of its network ports. You can either connect directly, using a peer connection between the two, or through any intermediary network.

If you do not want to use an SSH client and you have access to the GUI, you can access the CLI through the network using the CLI console in the GUI.

SSH must be enabled on the network interface that is associated with the physical network port that is used.

If your computer is not connected either directly or through a switch to the FortiGate, you must also configure the FortiGate with a static route to a router that can forward packets from the FortiGate to the computer. This can be done using a local console connection, or in the GUI.

To connect to the FortiGate CLI using SSH, you need:

- A computer with an available serial communications (COM) port and RJ-45 port
- An appropriate console cable
- Terminal emulation software
- A network cable
- Prior configuration of the operating mode, network interface, and static route.

To enable SSH access to the CLI using a local console connection:

1. Using the network cable, connect the FortiGate unit's port either directly to your computer's network port, or to a network through which your computer can reach the FortiGate.
2. Note the number of the physical network port.
3. Using direct console connection, connect and log into the CLI.
4. Enter the following command:

```
config system interface
    edit <interface_str>
        append allowaccess ssh
    next
end
```

   Where <interface_str> is the name of the network interface associated with the physical network port, such as port1.

5. Confirm the configuration using the following command to show the interface's settings:

```
show system interface <interface_str>
```

   For example:

```
show system interface port1
    config system interface
        edit "port1"
            set vdom "root"
            set ip 192.168.1.99 255.255.255.0
            set allowaccess ping https ssh
            set type hard-switch
            set stp enable
            set role lan
```

```
            set snmp-index 6
        next
    end
```

Connecting using SSH

Once the FortiGate is configured to accept SSH connections, use an SSH client on your management computer to connect to the CLI.

The following instructions use PuTTy. The steps may vary in other terminal emulators.

To connect to the CLI using SSH:

1. On your management computer, start PuTTy.
2. In the *Host Name (or IP address)* field, enter the IP address of the network interface that you are connected to and that has SSH access enabled.
3. Set the port number to 22, if it is not set automatically.
4. Select *SSH* for the *Connection type*.
5. Click *Open*. The SSH client connect to the FortiGate.
   The SSH client may display a warning if this is the first time that you are connecting to the FortiGate and its SSH key is not yet recognized by the SSH client, or if you previously connected to the FortiGate using a different IP address or SSH key. This is normal if the management computer is connected directly to the FortiGate with no network hosts in between.
6. Click *Yes* to accept the FortiGate's SSH key.
   The CLI displays the log in prompt.
7. Enter a valid administrator account name, such as `admin`, then press *Enter*.
8. Enter the administrator account password, then press *Enter*.
   The CLI console shows the command prompt (FortiGate hostname followed by a `#`). You can now enter CLI commands.

If three incorrect log in or password attempts occur in a row, you will be disconnected. If this occurs, wait for one minute, then reconnect and attempt to log in again.

CLI basics

Basic features and characteristics of the CLI environment provide support and ease of use for many CLI tasks.

Help

Press the question mark (?) key to display command help and complete commands.

- Press the question mark (?) key at the command prompt to display a list of the commands available and a description of each command.
- Enter a command followed by a space and press the question mark (?) key to display a list of the options available for that command and a description of each option.
- Enter a command followed by an option and press the question mark (?) key to display a list of additional options available for that command option combination and a description of each option.

- Enter a question mark after entering a portion of a command to see a list of valid complete commands and their descriptions. If there is only one valid command, it will be automatically filled in.

Shortcuts and key commands

Shortcut key	Action
?	List valid complete or subsequent commands. If multiple commands can complete the command, they are listed with their descriptions.
Tab	Complete the word with the next available match. Press multiple times to cycle through available matches.
Up arrow or Ctrl + P	Recall the previous command. Command memory is limited to the current session.
Down arrow, or Ctrl + N	Recall the next command.
Left or Right arrow	Move the cursor left or right within the command line.
Ctrl + A	Move the cursor to the beginning of the command line.
Ctrl + E	Move the cursor to the end of the command line.
Ctrl + B	Move the cursor backwards one word.
Ctrl + F	Move the cursor forwards one word.
Ctrl + D	Delete the current character.
Ctrl + C	Abort current interactive commands, such as when entering multiple lines. If you are not currently within an interactive command such as `config` or `edit`, this closes the CLI connection.
\ then Enter	Continue typing a command on the next line for a multiline command. For each line that you want to continue, terminate it with a backslash (\ ). To complete the command, enter a space instead of a backslash, and then press *Enter*.

Command tree

Enter `tree` to display the CLI command tree. To capture the full output, connect to your device using a terminal emulation program and capture the output to a log file. For some commands, use the `tree` command to view all available variables and subcommands.

Command abbreviation

You can abbreviate words in the command line to their smallest number of non-ambiguous characters.

For example, the command `get system status` could be abbreviated to `g sy stat`.

Adding and removing options from lists

When configuring a list, the `set` command will remove the previous configuration.

For example, if a user group currently includes members A, B, and C, the command `set member D` will remove members A, B, and C. To avoid removing the existing members from the group, the command `set members A B C D` must be used.

To avoid this issue, the following commands are available:

append	Add an option to an existing list. For example, `append member D` adds user D to the user group without removing any of the existing members.
select	Clear all of the options except for those specified. For example, `select member B` removes all member from the group except for member B.
unselect	Remove an option from an existing list. For example, `unselect member C` removes only member C from the group, without affecting the other members.

Environment variables

The following environment variables are support by the CLI. Variable names are case-sensitive.

$USERFROM	The management access type (`ssh`, `jsconsole`, and so on) and the IPv4 address of the administrator that configured the item.
$USERNAME	The account name of the administrator that configured the item.
$SerialNum	The serial number of the FortiGate.

For example, to set a FortiGate device's host name to its serial number, use the following CLI command:

```
config system global
    set hostname $SerialNum
end
```

Special characters

The following characters cannot be used in most CLI commands: `<`, `>`, `(`, `)`, `#`, `'`, and `"`

If one of those characters, or a space, needs to be entered as part of a string, it can be entered by using a special command, enclosing the entire string in quotes, or preceding it with an escape character (backslash, \).

To enter a question mark (`?`) or a tab, *Ctrl* + *V* or *Ctrl* + *Shift* + - must be entered first.

Question marks and tabs cannot be copied into the CLI Console or some SSH clients. They must be typed in.

Character	Keys
?	Ctrl + V or Ctrl + Shift + - then ?
Tab	Ctrl + V then Tab
Space (as part of a string value, not to end the string)	Enclose the string in single or double quotation marks: `"Security Administrator"` or `'Security Administrator'`. Precede the space with a backslash: `Security\ Administrator`.
' (as part of a string value, not to begin or end the string)	\'
" (as part of a string value, not to begin or end the string)	\"
\	\\

Using grep to filter command output

The `get`, `show`, and `diagnose` commands can produce large amounts of output. The `grep` command can be used to filter the output so that it only shows the required information.

The grep command is based on the standard UNIX grep, used for searching text output based on regular expressions.

For example, the following command displays the MAC address of the internal interface:

```
get hardware nic internal | grep Current_HWaddr
    Current_HWaddr        00:09:0f:cb:c2:75
```

The following command will display all TCP sessions that are in the session list, including the session list line number in the output:

```
get system session list | grep -n tcp
```

The following command will display all of the lines in the HTTP replacement message that contain `URL` or `url`:

```
show system replacemsg http | grep -i url
```

The following options can also be used:

```
-A <num> After
-B <num> Before
-C <num> Context
```

The `-f` option is available to support contextual output, in order to show the complete configuration. The following example shows the difference in the output when `-f` is used versus when it is not used:

Without `-f`:

```
show | grep ldap-group1
    edit "ldap-group1"
        set groups "ldap-group1"
```

With `-f`:

```
show | grep -f ldap-group1
    config user group
        edit "ldap-group1"
```

```
        set member "pc40-LDAP"
    next
end
config firewall policy
    edit 2
        set srcintf "port31"
        set dstintf "port32"
        set srcaddr "all"
        set action accept
        set identity-based enable
        set nat enable
        config identity-based-policy
            edit 1
                set schedule "always"
                set groups "ldap-group1"
                set dstaddr "all"
                set service "ALL"
            next
        end
    next
end
```

Language support and regular expressions

Characters such as ñ and é, symbols, and ideographs are sometimes acceptable input. Support varies depending on the type of item that is being configured. CLI commands, objects, field names, and options must use their exact ASCII characters, but some items with arbitrary names or values can be input using your language of choice. To use other languages in those cases, the correct encoding must be used.

Input is stored using Unicode UTF-8 encoding, but is not normalized from other encodings into UTF-8 before it is stored. If your input method encodes some characters differently than in UTF-8, configured items may not display or operate as expected.

Regular expressions are especially impacted. Matching uses the UTF-8 character values. If you enter a regular expression using a different encoding, or if an HTTP client sends a request in a different encoding, matches may not be what is expected.

For example, with Shift-JIS, backslashes could be inadvertently interpreted as the symbol for the Japanese yen (¥), and vice versa. A regular expression intended to match HTTP requests containing monetary values with a yen symbol may not work it if the symbol is entered using the wrong encoding.

For best results:

- use UTF-8 encoding, or
- use only characters whose numerically encoded values are the same in UTF-8, such as the US-ASCII characters that are encoded using the same values in ISO 8859-1, Windows code page 1252, Shift-JIS, and other encoding methods, or
- for regular expressions that must match HTTP requests, use the same encoding as your HTTP clients.

HTTP clients may send requests in encodings other than UTF-8. Encodings usually vary based on the client's operating system or input language. If the client's encoding method cannot be predicted, you might only be able to match the parts of the request that are in English, as the values for English characters tend to be encoded identically, regardless of the encoding method.

If the FortiGate is configured to use an encoding method other than UTF-8, the management computer's language may need to be changed, including the web browse and terminal emulator. If the FortiGate is configured using non-ASCII characters, all the systems that interact with the FortiGate must also support the same encoding method. If possible, the same encoding method should be used throughout the configuration to avoid needing to change the language settings on the management computer.

The GUI and CLI client normally interpret output as encoded using UTF-8. If they do not, configured items may not display correctly. Exceptions include items such as regular expression that may be configured using other encodings to match the encoding of HTTP requests that the FortiGate receives.

To enter non-ASCII characters in a terminal emulator:

1. On the management computer, start the terminal client.
2. Configure the client to send and receive characters using UTF-8 encoding.
   Support for sending and receiving international characters varies by terminal client.
3. Log in to the FortiGate.
4. At the command prompt, type your command and press *Enter*.
   Words that use encoded characters may need to be enclosed in single quotes (').
   Depending on your terminal client's language support, you may need to interpret the characters into character codes before pressing *Enter*. For example, you might need to enter: `edit '\743\601\613\743\601\652'`
5. The CLI displays the command and its output.

Screen paging

By default, the CLI will pause after displaying each page worth of text when a command has multiple pages of output. this can be useful when viewing lengthy outputs that might exceed the buffer of terminal emulator.

When the display pauses and shows `--More--`, you can:

- Press *Enter* to show the next line,
- Press *Q* to stop showing results and return to the command prompt,
- Press an arrow key, *Insert*, *Home*, *Delete*, *End*, *Page Up*, or *Page Down* to show the next few pages,
- Press any other key to show the next page, or
- Wait for about 30 seconds for the console to truncate the output and return to the command prompt.

When pausing the screen is disabled, press *Ctrl* + *C* to stop the output and log out of the FortiGate.

To disable pausing the CLI output:

```
config system console
    set output standard
end
```

To enable pausing the CLI output:

```
config system console
    set output more
end
```

Changing the baud rate

The baud rate of the local console connection can be changed from its default value of 9600.

To change the baud rate:

```
config system console
    set baudrate {9600 | 19200 | 38400 | 57600 | 115200}
end
```

Editing the configuration file

The FortiGate configuration file can be edited on an external host by backing up the configuration, editing the configuration file, and then restoring the configuration to the FortiGate.

Editing the configuration file can save time is many changes need to be made, particularly if the plain text editor that you are using provides features such as batch changes.

To edit the configuration file:

1. Backup the configuration. See Configuration backups on page 65 for details.
2. Open the configuration file in a plain text editor that supports UNIX-style line endings.
3. Edit the file as needed.

   Do not edit the first line of the configuration file.

   This line contains information about the firmware version and FortiGate model. If you change the model number, the FortiGate will reject the configuration when you attempt to restore it.

4. Restore the modified configuration to the FortiGate. See Configuration backups on page 65 for details.
   The FortiGate downloads the configuration file and checks that the model information is correct. If it is correct, the configuration file is loaded and each line is checked for errors. If a command is invalid, that command is ignored. If the configuration file is valid, the FortiGate restarts and loads the downloaded configuration.

Command syntax

When entering a command, the CLI console requires that you use valid syntax and conform to expected input constraints. It rejects invalid commands. Indentation is used to indicate the levels of nested commands.

Each command line consists of a command word, usually followed by configuration data or a specific item that the command uses or affects.

Notation

Brackets, vertical bars, and spaces are used to denote valid syntax. Constraint notations, such as `<address_ipv4>`, indicate which data types or string patterns are acceptable value input.

All syntax uses the following conventions:

Angle brackets < >	Indicate a variable of the specified data type.
Curly brackets { }	Indicate that a variable or variables are mandatory.
Square brackets [ ]	Indicate that the variable or variables are optional. For example: `show system interface [<name_str>]` To show the settings for all interfaces, you can enter `show system interface` To show the settings for the Port1 interface, you can enter `show system interface port1`.
Vertical bar |	A vertical bar separates alternative, mutually exclusive options. For example: `set protocol {ftp | sftp}` You can enter either `set protocol ftp` or `set protocol sftp`.

Space	A space separates non-mutually exclusive options.
	For example:
	`set allowaccess {ping https ssh snmp http fgfm radius-acct probe-response capwap ftm}`
	You can enter any of the following:
	`set allowaccess ping`
	`set allowaccess https ping ssh`
	`set allowaccess http https snmp ssh ping`
	In most cases, to make changes to lists that contain options separated by spaces, you need to retype the entire list, including all the options that you want to apply and excluding all the options that you want to remove.

Optional values and ranges

Any field that is optional will use square-brackets. The overall config command will still be valid whether or not the option is configured.

Square-brackets can be used is to show that multiple options can be set, even intermixed with ranges. The following example shows a field that can be set to either a specific value or range, or multiple instances:

```
config firewall service custom
    set iprange <range1> [<range2> <range3> ...]
end
```

next

The `next` command is used to maintain a hierarchy and flow to CLI commands. It is at the same indentation level as the preceding `edit` command, to mark where a table entry finishes.

The following example shows the next command used in the subcommand `entries`:

```
config dlp filepattern
    edit <1>
        set name <name>
        set comment [comment]
        config entries
            edit <2>
                set filter-type {pattern | type}
            next
```

After configuring table entry `<2>` then entering `next`, the `<2>` table entry is saved and the console returns to the `entries` prompt:

```
FGT60E1Q23456789 (entries) #
```

You can now create more table entries as needed, or enter `end` to save the table and return to the `filepattern` table element prompt.

end

The `end` command is used to maintain a hierarchy and flow to CLI commands.

The following example shows the same command and subcommand as the `next` command example, except `end` has been entered instead of `next` after the subcommand:

```
config dlp filepattern
    edit <1>
        set name <name>
        set comment [comment]
        config entries
            edit <2>
                set filter-type {pattern | type}
    end
```

Entering `end` will save the `<2>` table entry and the table, and exit the `entries` subcommand entirely. The console returns to the filepattern table element prompt:

```
FGT60E1Q23456789 (1) #
```

Subcommands

Subcommands are available from within the scope of some commands. When you enter a subcommand level, the command prompt changes to indicate the name of the current command scope. For example, after entering:

```
config system admin
```

the command prompt becomes:

```
(admin)#
```

Applicable subcommands are available until you exit the command, or descend an additional level into another subcommand. Subcommand scope is indicated by indentation.

For example, the `edit` subcommand is only available in commands that affects tables, and the `next` subcommand is available only in the `edit` subcommand:

```
config system interface
   edit port1
      set status up
   next
end
```

The available subcommands vary by command. From a command prompt under the `config` command, subcommands that affect tables and fields could be available.

Table subcommands

edit <table_row>	Create or edit a table value.
	In objects such as security policies, `<table_row>` is a sequence number. To create a new table entry without accidentally editing an existing entry, enter `edit 0`. The CLI will confirm that creation of entry 0, but will assign the next unused number when the entry is saved after entering `end` or `next`.
	For example, to create a new firewall policy, enter the following commands:
	`config firewall policy`
	   `edit 0`
	      `....`
	   `next`
	`end`
	To edit an existing policy, enter the following commands:
	`config firewall policy`
	   `edit 27`
	      `....`
	   `next`
	`end`
	The `edit` subcommand changes the command prompt to the name of the table value that is being edited.

delete <table_row>	Delete a table value.
	For example, to delete firewall policy 30, enter the following commands:
	`config firewall policy`
	   `delete 30`
	`end`

purge	Clear all table values. The `purge` command cannot be undone. To restore purged table values, the configuration must be restored from a backup.

move	Move an ordered table value.
	In the firewall policy table, this equivalent to dragging a policy into a new position. It does not change the policy's ID number.
	For example, to move policy 27 to policy 30, enter the following commands:
	`config firewall policy`
	`  move 27 to 30`
	`end`
	The `move` subcommand is only available in tables where the order of the table entries matters.

clone <table_row> to <table_row>	Make a clone of a table entry.
	For example, to create firewall policy 30 as a clone of policy 27, enter the following commands:
	`config firewall policy`
	`  clone 27 to 30`
	`end`
	The `clone` subcommand may not be available for all tables.

rename <table_row> to <table_row>	Rename a table entry.
	For example to rename an administrator from Flank to Frank, enter the following commands:
	`config system admin`
	`  rename Flank to Frank`
	`end`
	The `rename` subcommand is only available in tables where the entries can be renamed.

get	List the current table entries.
	For example, to view the existing firewall policy table entries, enter the following commands:
	`config firewall policy`
	`  get`

show	Show the configuration. Only table entries that are not set to default values are shown.
end	Save the configuration and exit the current `config` command.

Purging the `system interface` or `system admin` tables does not reset default table values. This can result in being unable to connect to or log in to the FortiGate, requiring the FortiGate to be formatted and restored.

Field subcommands

set <field> <value>	Modify the value of a field.

	For example, the command `set fsso enable` sets the `fsso` field to the value `enable`.
unset	Set the field to its default value.
select	Clear all of the options except for those specified. For example, if a group contains members A, B, C, and D, to remove all members except for B, use the command `select member B`.
unselect	Remove an option from an existing list. For example, if a group contains members A, B, C, and D, to remove only member B, use the command `unselect member B`.
append	Add an option to an existing multi-option table value.
clear	Clear all the options from a multi-option table value.
get	List the configuration of the current table entry, including default and customized values.
show	Show the configuration. Only values that are not set to default values are shown.
next	Save changes to the table entry and exit the `edit` command so that you can configure the next table entry.
abort	Exit the command without saving.
end	Save the configuration and exit the current `config` command.

Permissions

Administrator (or access) profiles control what CLI commands an administrator can access by assigning read, write, or no access to each area of FortiOS. For information, see Administrator profiles on page 1949.

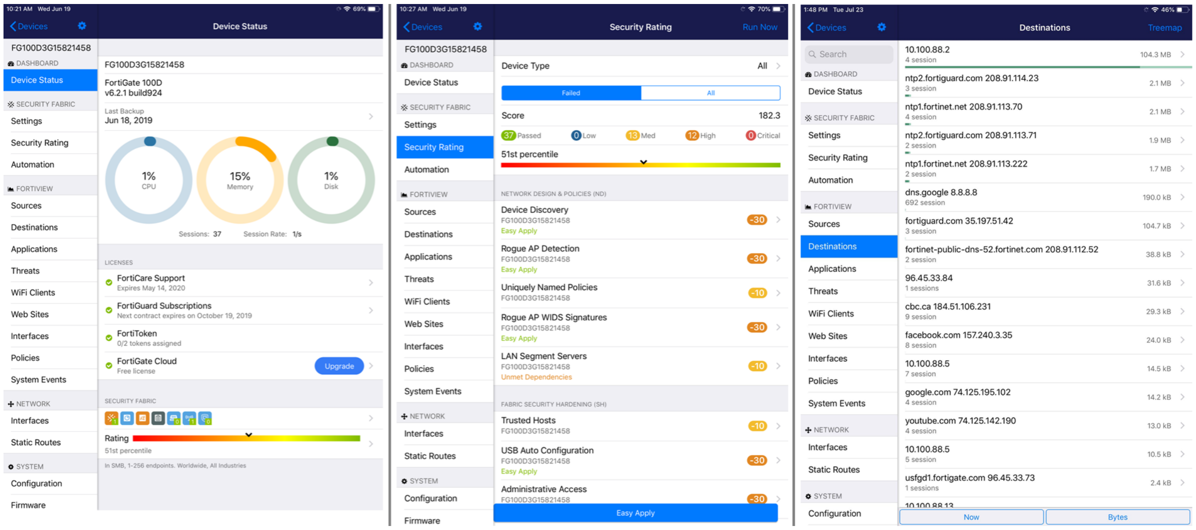
Read access is required to view configurations. Write access is required to make configuration changes. Depending on your account's profile, you may not have access to all CLI commands. To have access to all CLI commands, an administrator account with the *super_admin* profile must be used, such as the *admin* account.

Accounts assigned the *super_admin* profile are similar to the root administrator account. They have full permission to view and change all FortiGate configuration options, including viewing and changing other administrator accounts.

To increase account security, set strong passwords for all administrator accounts, and change the passwords regularly.

FortiExplorer management

FortiExplorer for iOS is a user-friendly application that helps you to rapidly provision, deploy, and monitor Security Fabric components from your iOS device.



FortiExplorer for iOS requires iOS 10.0 or later and is compatible with iPhone, iPad, and Apple TV. It is supported by FortiOS 5.6 and later, and is available on the App Store for iOS devices.

FortiExplorer is also available for support on Android on the Google Play Store. Steps for configuring FortiExplorer for Android may differ from what is included in the guide.

Advanced features are available with the purchase of FortiExplorer Pro. Paid features include the ability to add more than two devices, and firmware upgrades for devices with active licenses.

Up to six members can use this app with 'Family Sharing' enabled in the App Store.

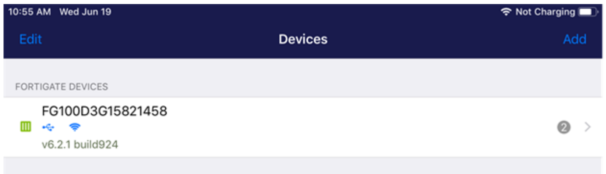
Firmware upload requires a valid firmware license. Users can download firmware for models with a valid support contract.

Getting started with FortiExplorer

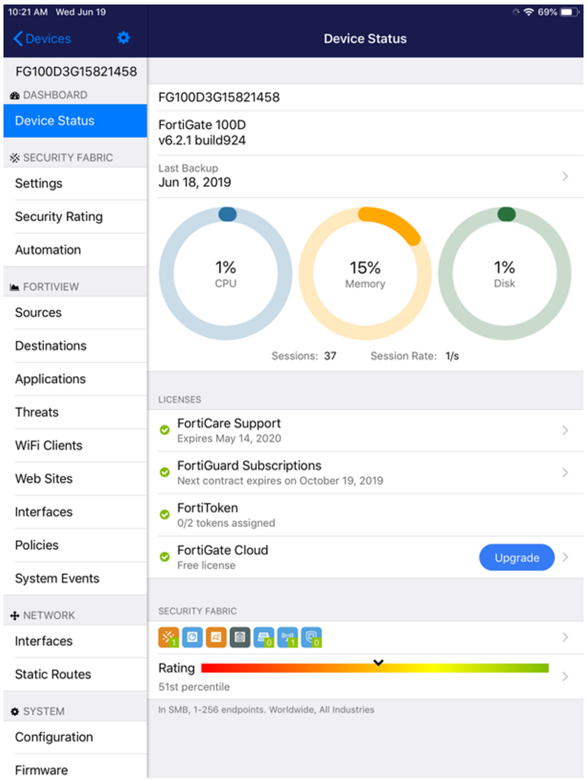
If your FortiGate is accessible on a wireless network, you can connect to it using FortiExplorer provided that your iOS device is on the same network. See Connecting FortiExplorer to a FortiGate with WiFi. If your 200F series or 80F series FortiGate is in close proximity, you can connect to it using FortiExplorer using Bluetooth Low Energy (BLE). See Configure FortiGate with FortiExplorer using BLE on page 48. Otherwise, you will need to physically connect your iOS device to the FortiGate using a USB cable.

To connect and configure a FortiGate with FortiExplorer using a USB connection:

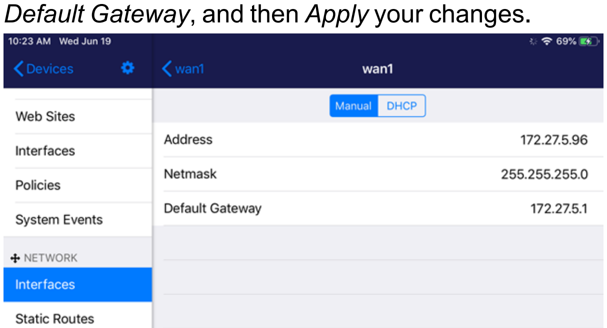
1. Connect your iOS device to your FortiGate USB A port. If prompted on your iOS device, *Trust* this computer.
2. Open FortiExplorer and select your FortiGate from the *FortiGate Devices* list . A blue USB icon will indicate that you are connected over a USB connection.



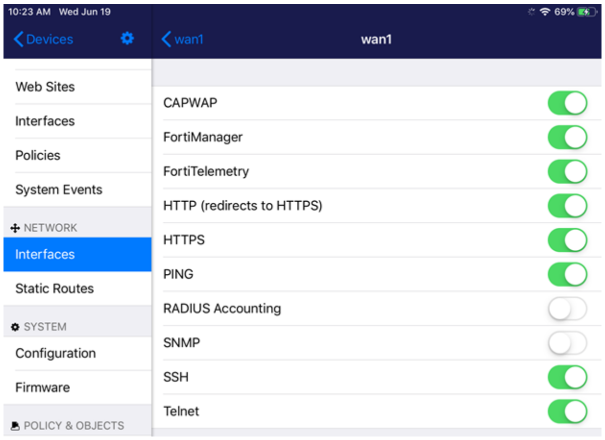
3. On the *Login* screen, select *USB*.
4. Enter the default *Username* (`admin`) and leave the *Password* field blank.
5. Optionally, select *Remember Password*.
6. Tap *Done* when you are ready.
   FortiExplorer opens the FortiGate management interface to the *Device Status* page:



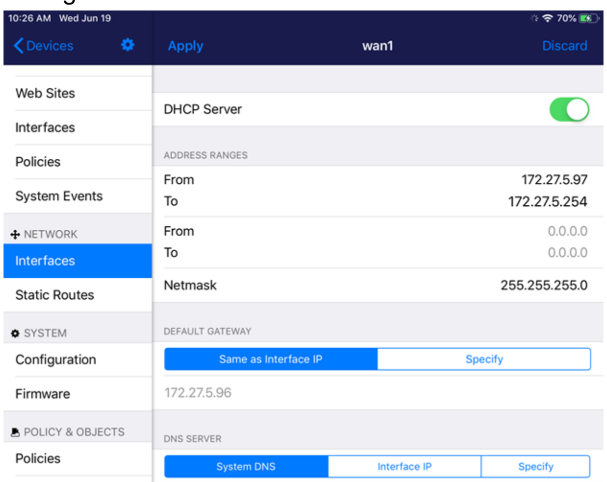
7. Go to *Network > Interfaces* and configure the WAN interface or interfaces.
8. The *wan1* interface *Address* mode is set to *DHCP* by default. Set it to *Manual* and enter its *Address*, *Netmask*, and *Default Gateway*, and then *Apply* your changes.



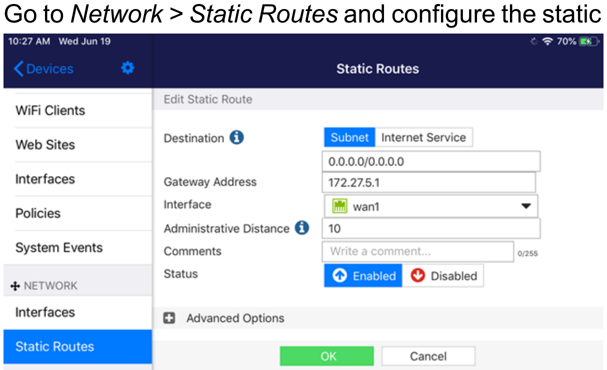
9. Optionally, configure *Administrative Access* to allow *HTTPS* access. This will allow administrators to access the FortiGate GUI using a web browser.



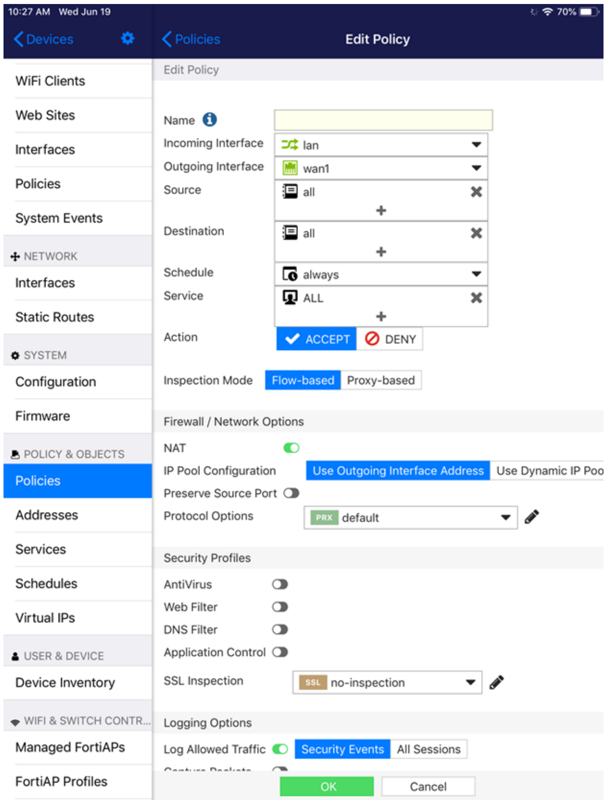
10. Go to *Network* > *Interfaces* and configure the local network (internal) interface.
11. Set the *Address* mode as before and configure *Administrative Access* if required.
12. Configure a *DHCP Server* for the internal network subnet.



13. Return to the internal interface using the < button at the top of the screen.
14. Go to *Network* > *Static Routes* and configure the static route to the gateway.



15. Go to *Policy & Objects* > *Firewall Policy* and edit the Internet access policy. Enter a *Name* for the policy, enable the required *Security Profiles*, configure *Logging Options*, then tap *OK*.

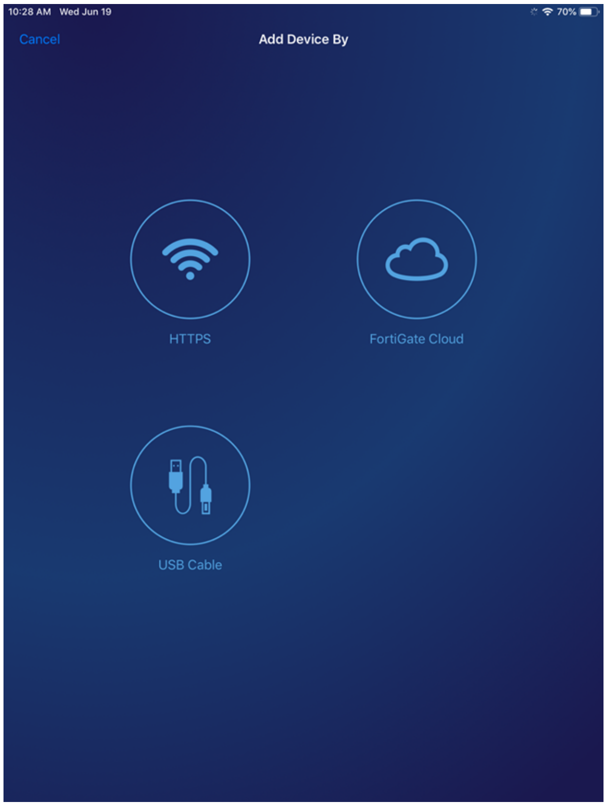


Connecting FortiExplorer to a FortiGate with WiFi

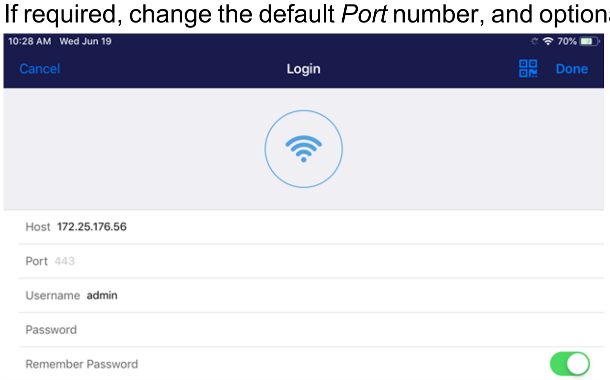
You can wirelessly connect to the FortiGate if your iOS device and the FortiGate are both connected to the same wireless network.

To connect and configure a FortiGate with FortiExplorer wirelessly:

1. Open the FortiExplorer app and tap *Add* on the *Devices* page.
2. On the *Add Device By* page, tap *HTTPS*.



3. Enter the *Host* information, *Username*, and *Password*.
4. If required, change the default *Port* number, and optionally enable *Remember Password*.



5. Tap *Done*.
6. If the FortiGate device identity cannot be verified, tap *Connect* at the prompt.
   FortiExplorer opens the FortiGate management interface to the *Device Status* page.

Configure FortiGate with FortiExplorer using BLE

FortiGate 200F series and 80F series devices can be initially configured in FortiExplorer using Bluetooth Low Energy (BLE).

The state of the status LED on the device shows if BLE is enabled. See the device QuickStart guides for more information about LED states: FortiGate 200F Series QuickStart Guide and FortiGate 80F Series QuickStart Guide.

When the status LED is flashing green, pressing and holding the reset button for five seconds or longer will reset the device to factory default settings.

BLE is enabled or disabled in the following scenarios after the FortiGate boots up:

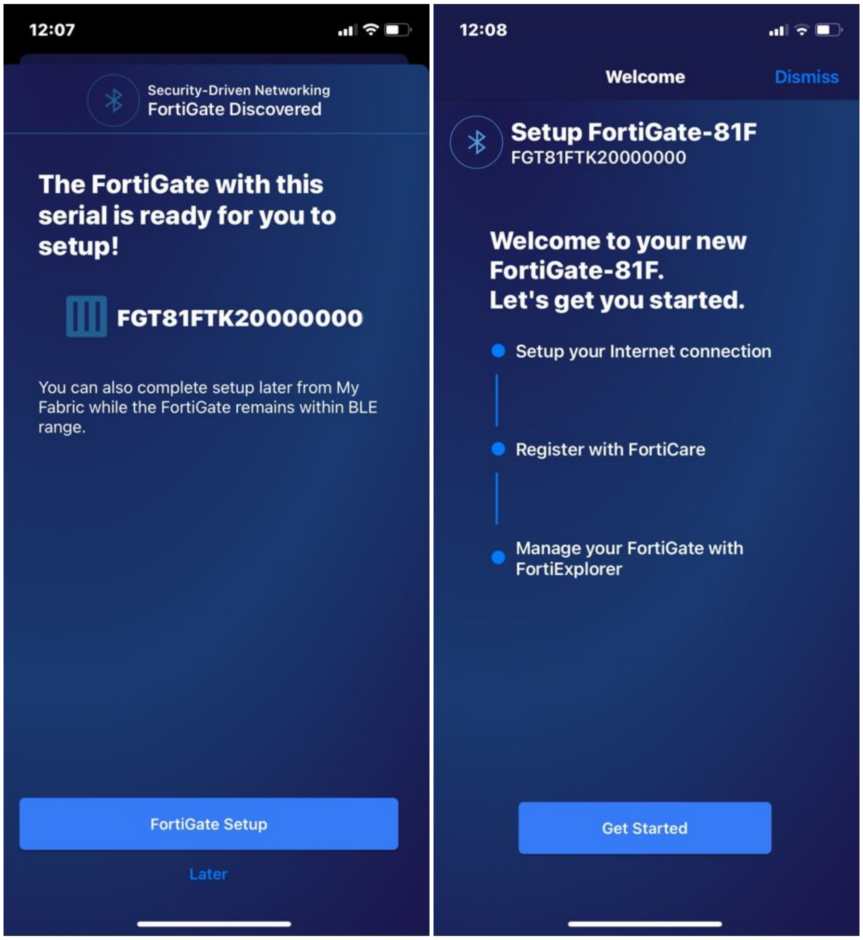
- In factory default settings:
  - After the FortiGate has finished booting up (when the console login prompt is shown), the status LED will be flashing amber or red to indicate that BLE is enabled.
  - If the FortiGate is configured without using BLE, BLE will immediately be disabled and the status LED will turn solid green.
  - If the FortiGate is configured using BLE, the LED will continue flashing until the configuring device disconnects from BLE, after which BLE is disabled and the status LED turns sold green.
- Not in factory default configuration:
  - One minute after the FortiGate has finished booting up (when the console login prompt is shown), the status LED will turn solid green. Press and hold the reset button for one second. The status LED will start flashing to indicate that BLE is enabled.
  - If no BLE connection is made with the FortiGate, BLE will be disabled after one minute and the status LED will turn solid green.
  - If the FortiGate is configured without using BLE, BLE will immediately be disabled and the status LED will turn solid green.
  - If the FortiGate is configured using BLE, the LED will continue flashing until the configuring device disconnects from BLE, after which BLE is disabled and the status LED turns sold green.

To enable BLE for one minute when the FortiGate is running and not in factory default configuration:

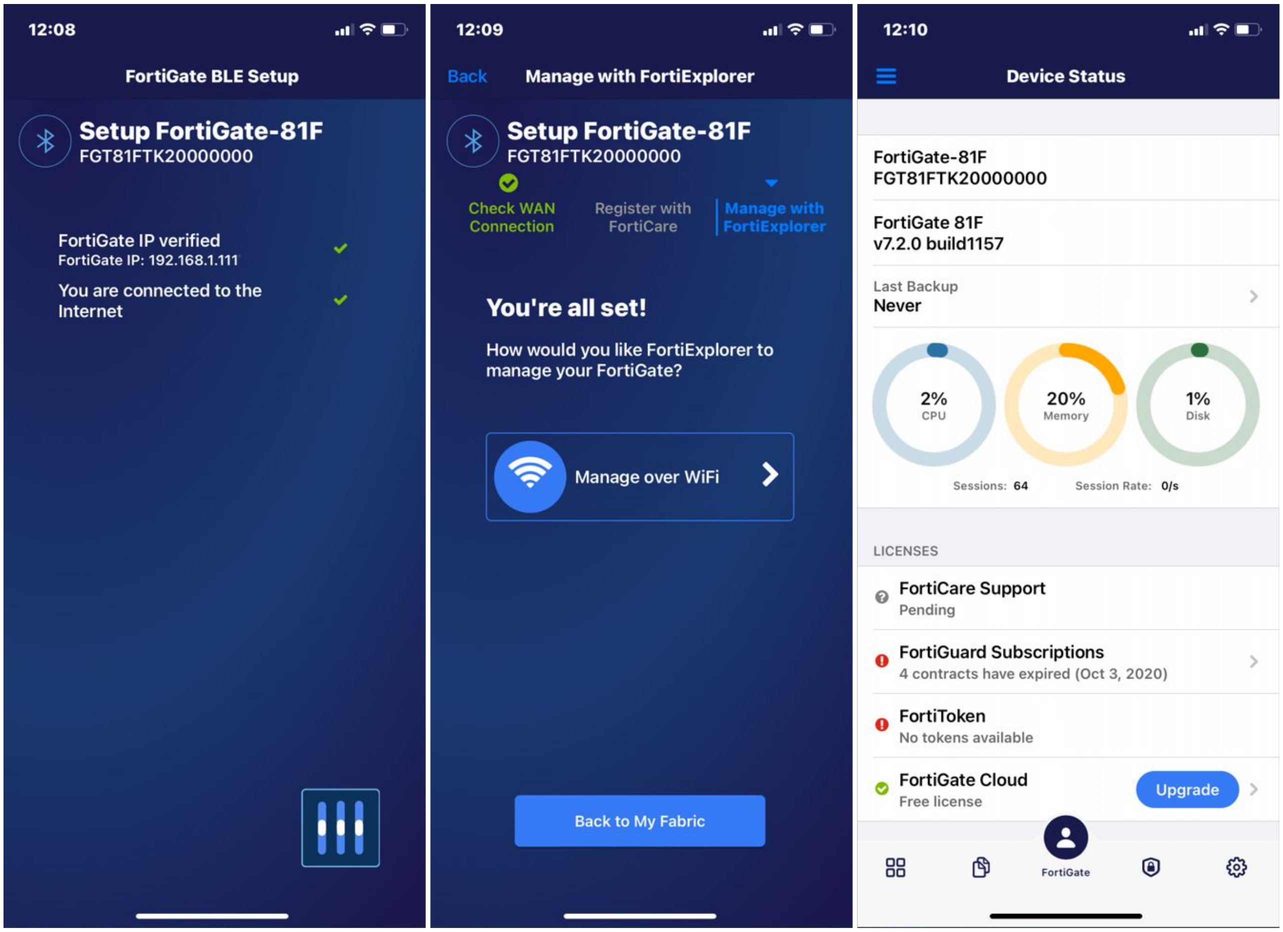
```
# diagnose bluetooth enable 1
```

To connect to and configure a FortiGate with FortiExplorer using BLE:

1. Ensure that BLE is enabled on the FortiGate device.
2. Enable Bluetooth on your iOS device and open the FortiExplorer app.
   If the app has detected the FortiGate device, the device's serial number will be shown.



3. Log into the FortiGate in the app using the default credentials: *admin* and no password.
4. If this is the first time logging into the device, set a password.
5. Optionally, register with FortiCare.
6. Configure the FortiGate, including the WAN and internal interfaces, static routes, and other required settings.



After configuring the FortiGate and disconnecting, BLE is disabled.

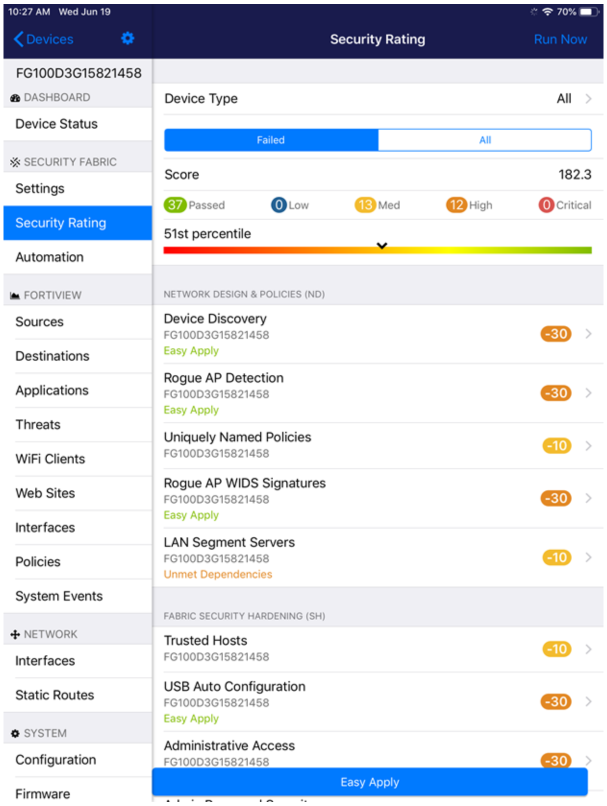
To check the status of BLE on the FortiGate:

```
diagnose hardware test ble
diagnose bluetooth status
diagnose bluetooth get_bt_version
diagnose bluetooth clean_bt_mode
```

Running a security rating

After configuring your network, run a security rating check to identify vulnerabilities and highlight best practices that could improve your network's security and performance.

Go to *Security Fabric > Security Rating* and follow the steps to determine the score. See Security rating on page 2335 for more information.



Upgrading to FortiExplorer Pro

FortiExplorer Pro allows you to add unlimited devices, and download firmware images for devices with active licenses.

To upgrade to FortiExplorer Pro:

1. In FortiExplorer, go to *Settings*.
2. Tap *Manage Subscription*.
3. Follow the on-screen prompts.

Basic administration

This section contains information about basic FortiGate administration that you can do after you installing the unit in your network.

- Basic configuration on page 53
- Registration on page 55
- FortiCare and FortiGate Cloud login on page 60
- Transfer a device to another FortiCloud account on page 62
- Configuration backups on page 65

Basic configuration

This topic will help you configure a few basic settings on the FortiGate as described in the Using the GUI on page 22 and Using the CLI on page 29 sections, including:

- Configuring an interface on page 53
- Configuring the hostname on page 53
- Configuring the default route on page 54
- Ensuring internet and FortiGuard connectivity on page 54

Configuring an interface

It is unlikely the default interface configuration will be appropriate for your environment and typically requires some effort of the administrator to use these settings, such as being physically near the FortiGate to establish a serial connection. Therefore, the first step is to configure an interface that can be used to complete the FortiGate configuration.

To configure an interface in the GUI:

1. Go to *Network > Interfaces*. Select an interface and click *Edit*.
2. Enter an *Alias*.
3. In the *Address* section, enter the *IP/Netmask*.
4. In *Administrative Access* section, select the access options as needed (such as *PING*, *HTTPS*, and *SSH*).
5. Optionally, enable *DHCP Server* and configure as needed.
6. Click *OK*.

To configure an interface in the CLI:

```
config system interface
    edit "port2"
        set ip 203.0.113.99 255.255.255.0
        set allowaccess ping https ssh
        set alias "Management"
    next
end
```

Configuring the hostname

Setting the FortiGate’s hostname assists with identifying the device, and it is especially useful when managing multiple FortiGates. Choose a meaningful hostname as it is used in the CLI console, SNMP system name, device name for FortiGate Cloud, and to identify a member of an HA cluster.

To configure the hostname in the GUI:

1. Go to *System > Settings*.
2. Enter a name in the *Host name* field.
3. Click *Apply*.

To configure the hostname in the CLI:

```
config system global
    set hostname 200F_YVR
end
```

Configuring the default route

Setting the default route enables basic routing to allow the FortiGate to return traffic to sources that are not directly connected. The gateway address should be your existing router or L3 switch that the FortiGate is connected to. If you are directly connecting to the FortiGate, you may choose your endpoint’s IP address as the gateway address. Set the interface to be the interface the gateway is connected to.

To configure the default route in the GUI:

1. Go to *Network > Static Routes* and click *Create New*.
2. Leave the destination subnet as *0.0.0.0/0.0.0.0*. This is known as a default route, since it would match any IPv4 address.
3. Enter the *Gateway Address*.
4. Select an *Interface*.
5. Click *OK*.

To configure the default route in the CLI:

```
config router static
    edit 0
        set gateway 192.168.1.254
        set device port1
    next
end
```

Ensuring internet and FortiGuard connectivity

This step is not necessary for the configuration; however, it is necessary in order to keep your FortiGate up to date against the latest threats. Updates are provided to FortiGates that are registered and make a request to the FortiGuard network to verify if there are any more recent definitions.

Use `execute ping <domain.tld>` to ensure the DNS resolution is able to resolve the following FortiGuard servers:

- fds1.fortinet.com
- service.fortiguard.net
- update.fortiguard.net

You also need to ensure the necessary ports are permitted outbound in the event your FortiGate is behind a filtering device. Refer to the Ports and Protocols document for more information.

Registration

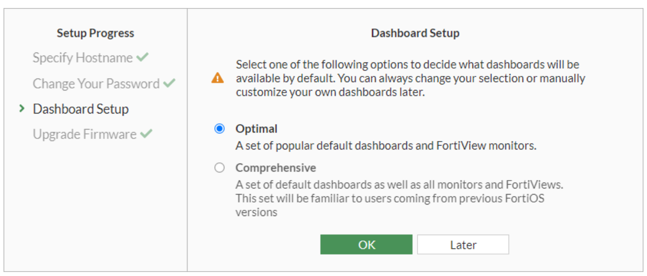
The FortiGate, and then its service contract, must be registered to have full access to Fortinet Customer Service and Support, and FortiGuard services. The FortiGate can be registered in either the FortiGate GUI or the FortiCloud support portal. The service contract can be registered from the FortiCloud support portal.

> The service contract number is needed to complete registrations on the FortiCloud support portal. You can find this 12-digit number in the email that contains your service registration document (sent from do-not-reply-contract@fortinet.com) in the service entitlement summary.

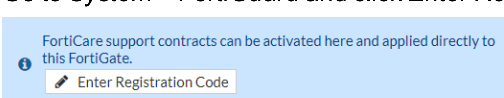
To register your FortiGate in the GUI:

1. Connect to the FortiGate GUI. A dialog box appears, which indicates the steps you should take to complete the setup of your FortiGate. These steps include:
   a. *Specify Hostname*
   b. *Change Your Password*
   c. *Dashboard Setup*
   d. *Upgrade Firmware*

   If you completed the Basic configuration on page 53, the hostname and password steps are already marked as complete (checkmark). If you chose to deploy the latest firmware, the *Upgrade Firmware* step is marked as complete.
2. Click *Begin* to complete the dashboard setup. Two options appear (*Optimal* and *Comprehensive*).



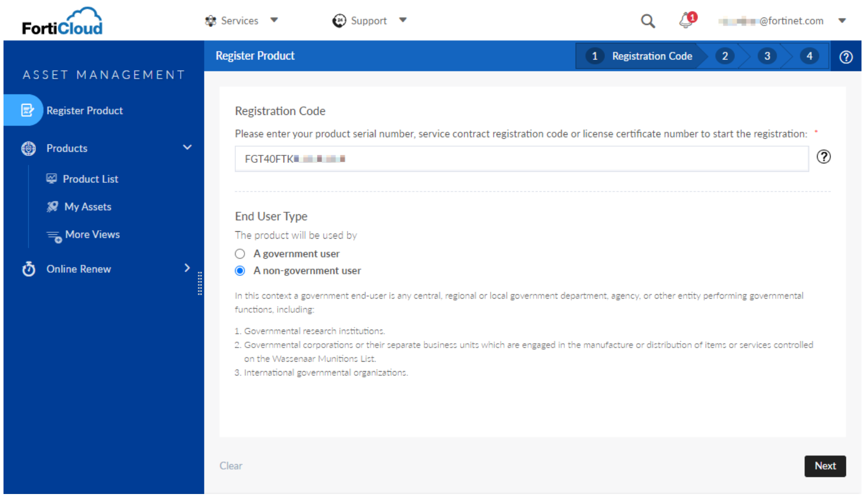
3. Select the desired setting and click *OK*. The *Dashboard* > *Status* page opens. Note that the licenses are grayed out because the device or virtual machine is not registered.
4. Go to *System* > *FortiGuard* and click *Enter Registration Code*.



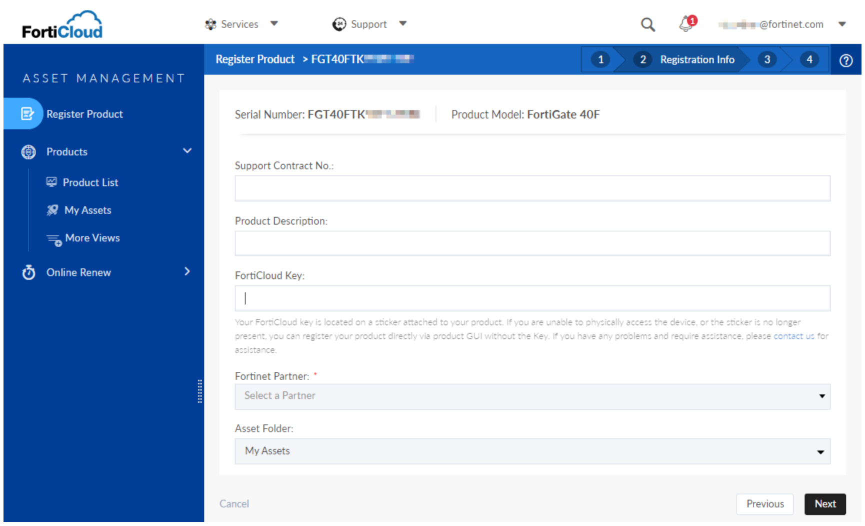
5. Enter the contract registration code from your service registration document.
6. Click *OK*.

To register the FortiGate on the FortiCloud support portal:

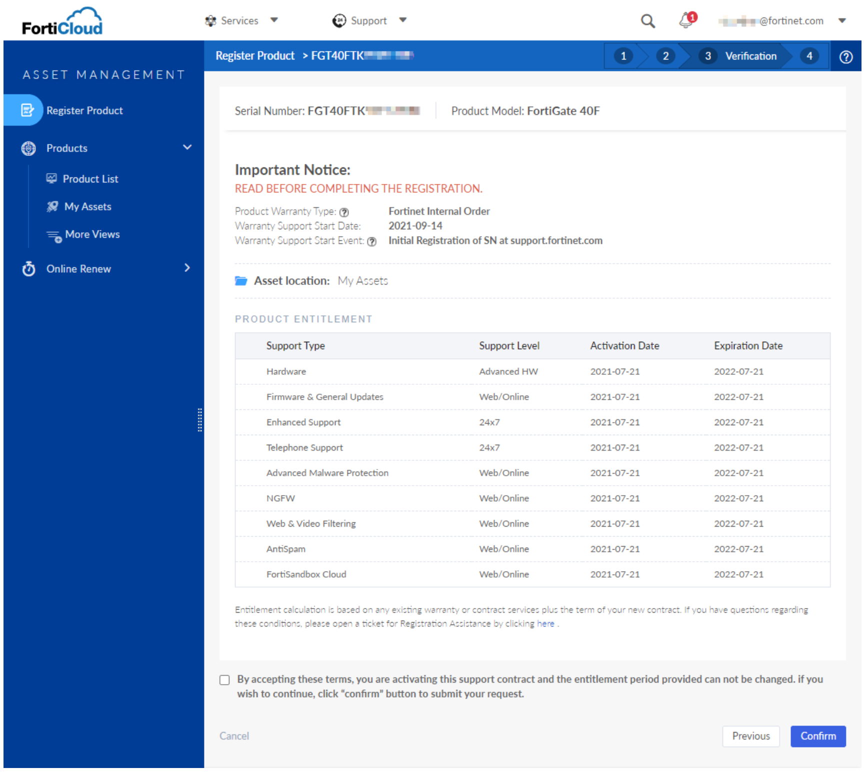
1. Go to support.fortinet.com and log in using your FortiCloud account credentials. If you do not have an account, click *Register* to create one.
2. In the left-side menu, click *Register Product*.
3. Enter the product serial number or license certificate number for a VM, select an end user type, then click *Next*.



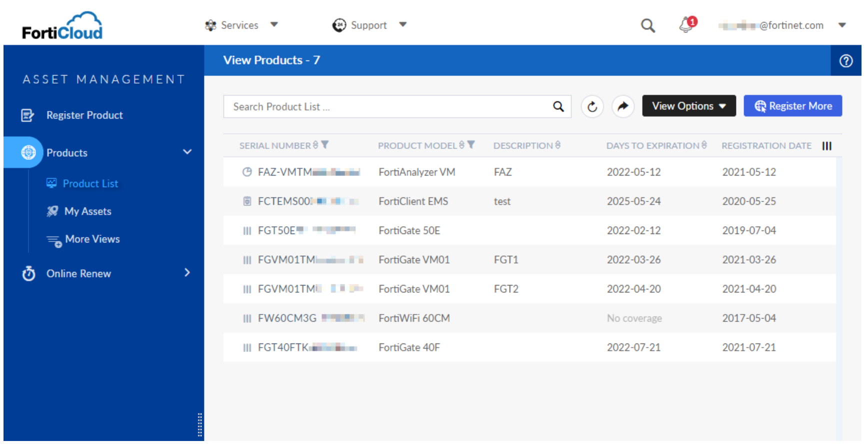
4. Enter the *Support Contract* number and *FortiCloud Key* (optionally, enter a product description), then click *Next*.



5. Review the product entitlement information, select the checkbox to accept the terms, then click *Confirm*.



6. Go to *Products* > *Product List*. The FortiGate is now visible in the product list.



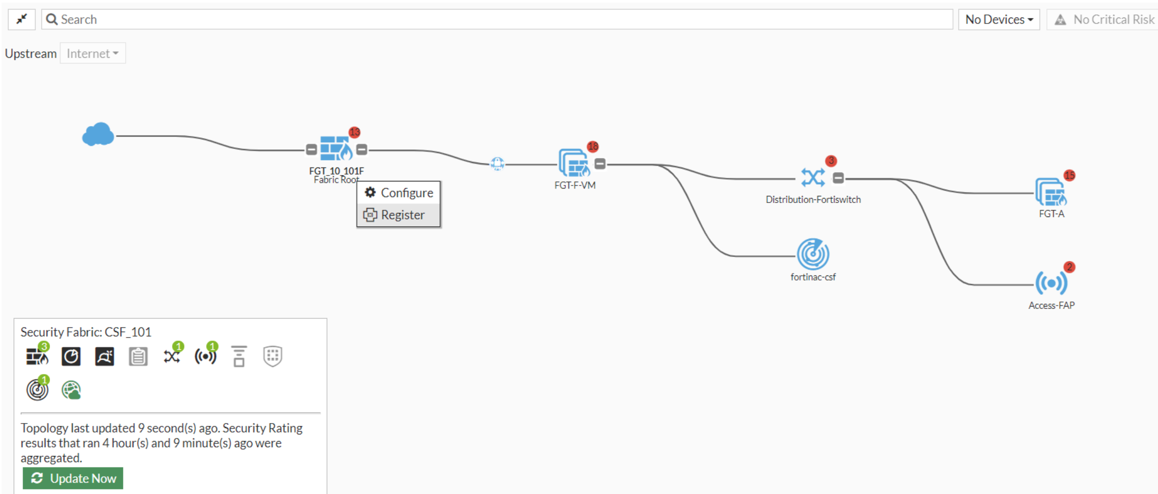
FortiCare Register button

The FortiCare *Register* button is displayed in the GUI on various Fabric and device related pages and widgets available for FortiGates.

There are two methods to access the *Register* button:

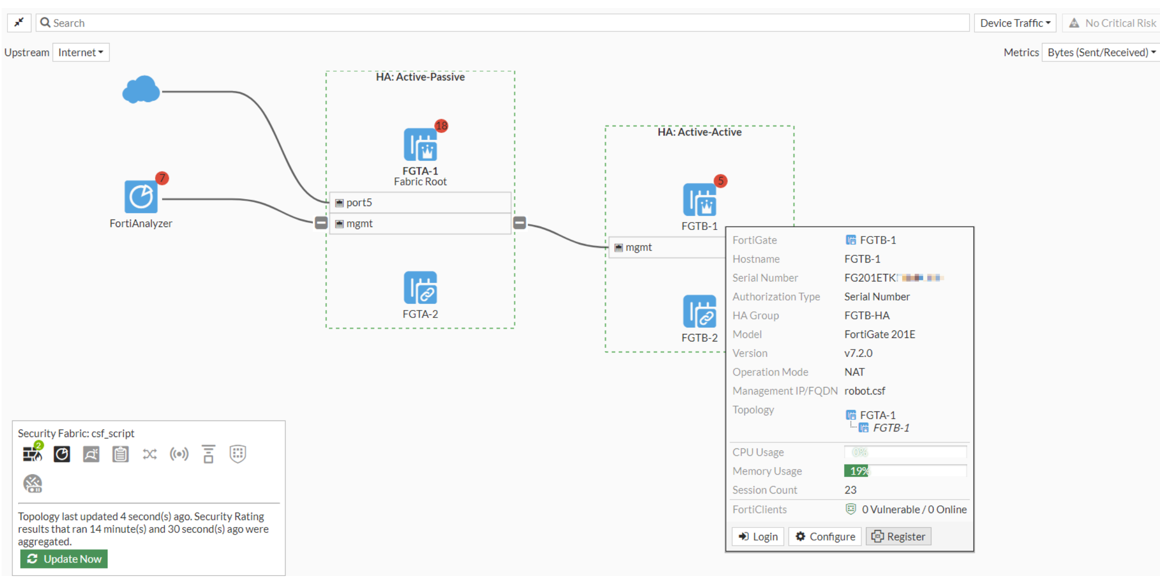
- Right-click on a device in a topology.

  Security Fabric > *Physical Topology* page:

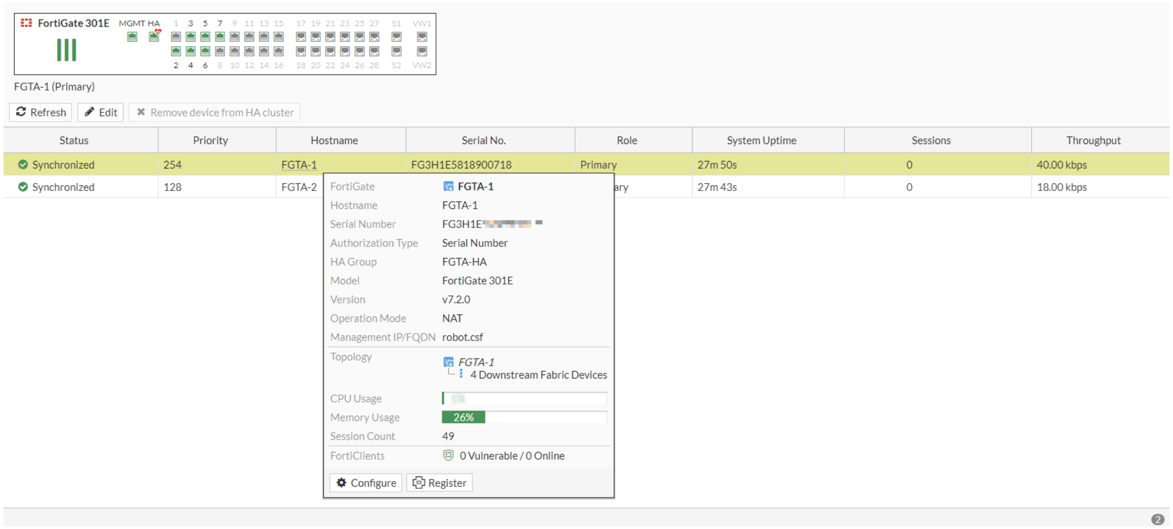


- Hover over a device to display the tooltip.

  Security Fabric > Logical Topology page:

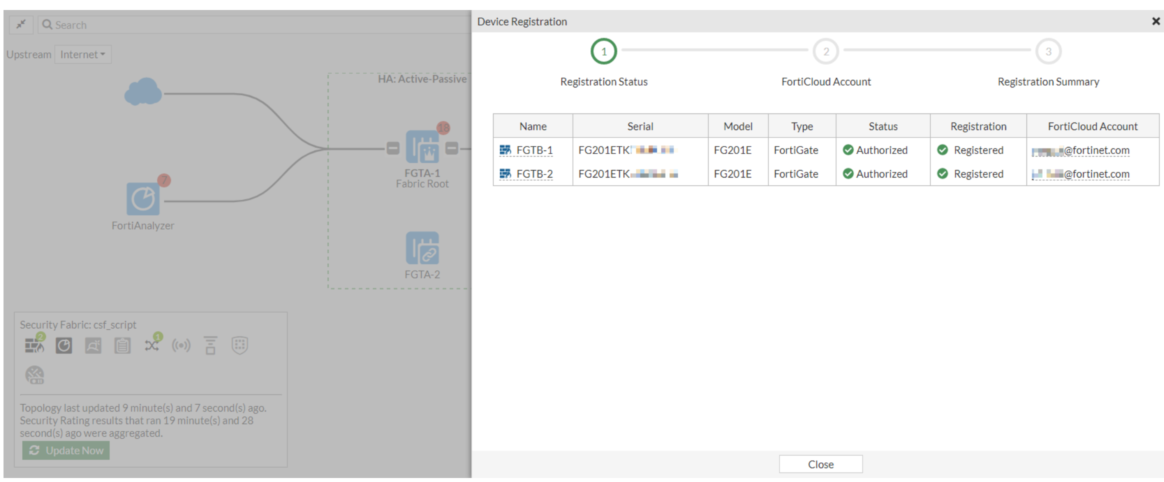


  System > HA page:



The *Register* button is also accessible from tooltips for devices on the *Managed FortiAPs* and *Managed FortiSwitches pages*.

Clicking *Register* opens the *Device Registration* pane. If a device is already registered, the pane still opens and displays the device information.



Primary and secondary HA members can be registered to FortiCare at the same time from the primary unit by using the *Register* button. The secondary unit will register through the HA proxy. In this example, a HA member is registered from the *Physical Topology* page.

To register a HA member to FortiCare:

1. On the primary unit, go to *Security Fabric > Physical Topology*.
2. Hover over the HA member and click *Register*. The *Device Registration* pane opens.
3. Select the device and click *Register*.
4. Enter the required FortiCloud account information (password, country or region, reseller) and click *Submit*.
5. Once the registration is complete, click *Close*.

FortiCare and FortiGate Cloud login

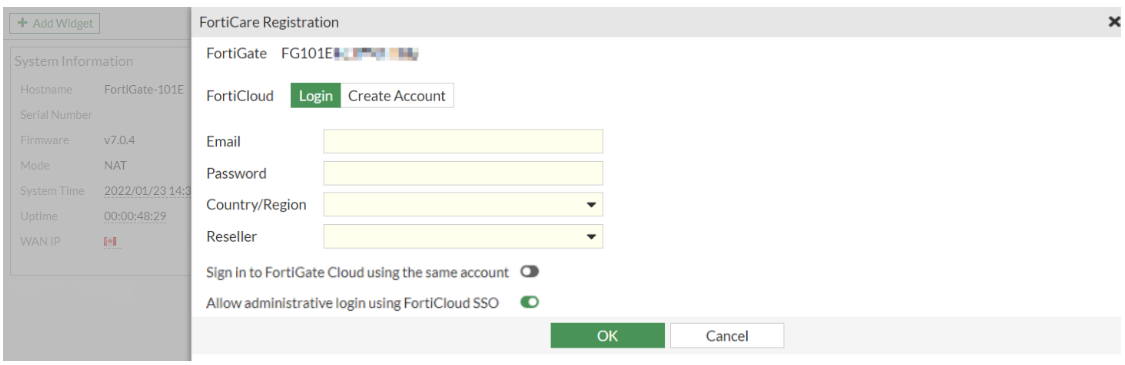
With FortiCloud, FortiOS supports a unified login to FortiCare and FortiGate Cloud. The FortiGate Cloud setup is a subset of the FortiCare setup.

- If the FortiGate is not registered, activating FortiGate Cloud will force you to register with FortiCare.
- If a FortiGate is registered in FortiCare using a FortiCloud account, then only that FortiCloud account can be used to activate FortiGate Cloud.
- If a different FortiCloud account was already used to activate FortiGate Cloud, then a notification asking you to migrate to FortiCloud is shown in the GUI after upgrading FortiOS.

The CLI can be used to activate FortiGate Cloud without registration, or with a different FortiCloud account.

To activate FortiGate Cloud and register with FortiCare at the same time:

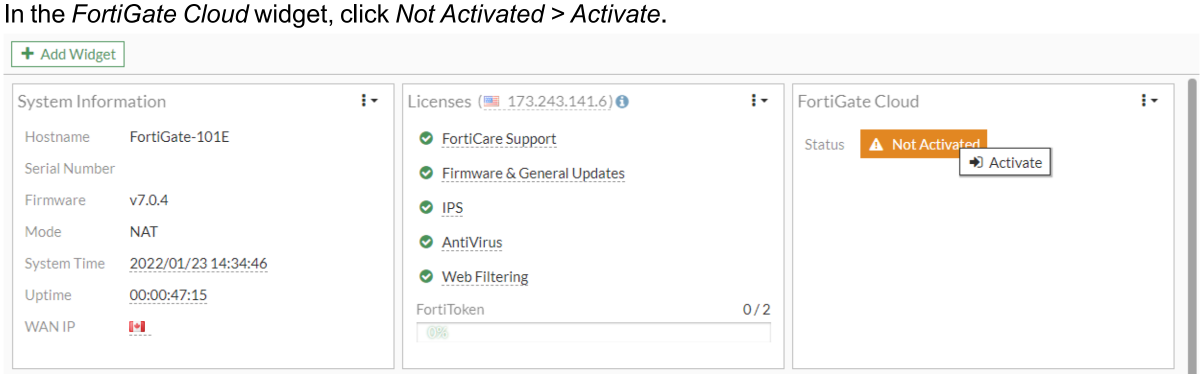
1. Go to *Dashboard* > *Status*.
2. In the FortiGate Cloud widget, click *Not Activated* > *Activate*.
   You must register with FortiCare before activating FortiGate Cloud.



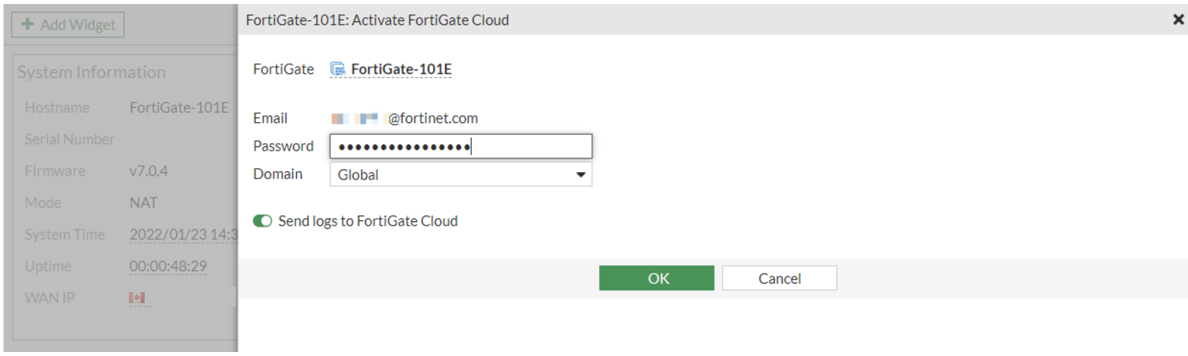
3. Enter your FortiCare *Email* address and *Password*.
4. Select your *Country/Region* and *Reseller*.
5. Enable *Sign in to FortiGate Cloud using the same account*.
6. Click *OK*.

To activate FortiGate Cloud on an already registered FortiGate:

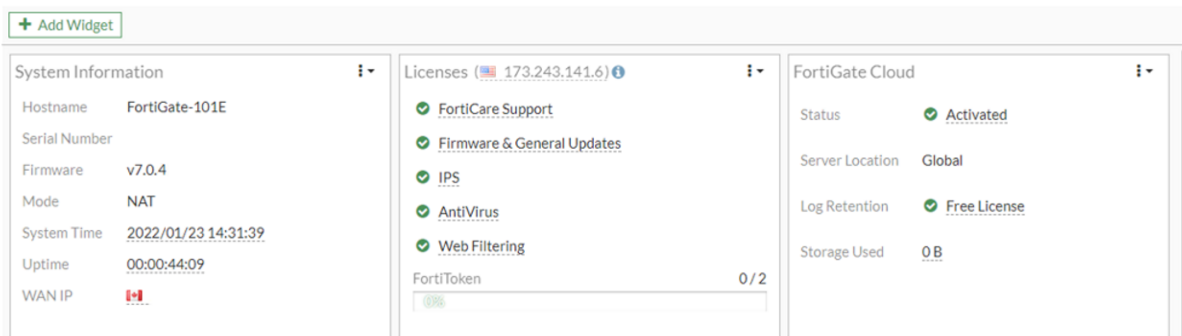
1. Go to *Dashboard* > *Status*.
2. In the *FortiGate Cloud* widget, click *Not Activated* > *Activate*.



3. Enter the password for the account that was used to register the FortiGate.

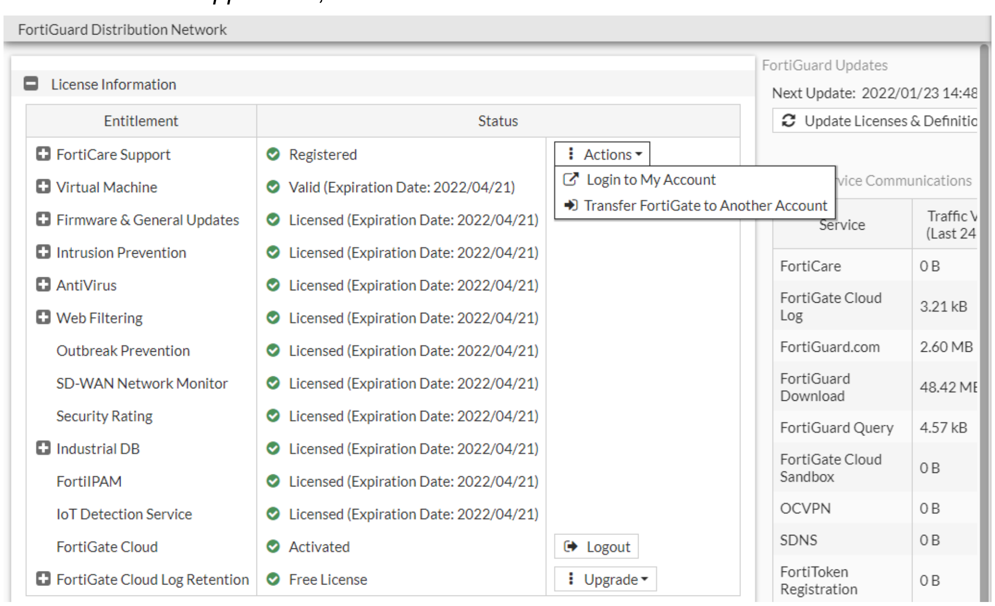


4. Click *OK*.
   The *FortiGate Cloud* widget now shows the activated FortiCloud account.

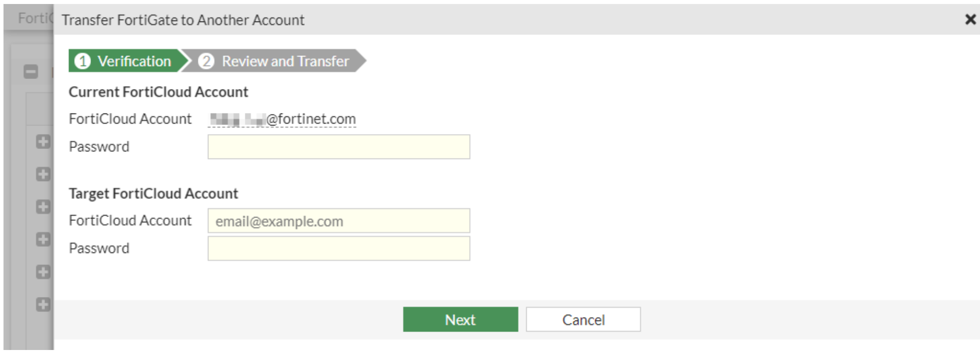


To migrate from the activated FortiGate Cloud account to the registered FortiCloud account:

1. Go to *System > FortiGuard*.
2. In the *FortiCare Support* row, click *Actions > Transfer FortiGate to Another Account*.



3. Enter the *Password* of the current FortiCloud account.



4. Enter the target *FortiCloud Account* name and *Password*, then click *Next*.
5. Review the information in the *From* and *To* fields, then click *Transfer*.

To activate FortiGate Cloud using an account that is not used for registration:

1. Enter the following with the credentials for the account being used to activate FortiGate Cloud:

```
# execute fortiguard-log login <account_id> <password>
```

2. Check the account type:

```
# diagnose fdsm contract-controller-update
Protocol=2.0|Response=202|Firmware=FAZ-4K-FW-2.50-
100|SerialNumber=FAMS000000000000|Persistent=false|ResponseItem=HomeServer:172.16.95.151
:443*AlterServer:172.16.95.151:443*Contract:20200408*NextRequest:86400*UploadConfig:Fals
e*ManagementMode:Local*ManagementID:737941253*AccountType:multitenancy

Result=Success
```

A FortiCloud account that is not used for the support portal account cannot be used to register FortiGate. Attempting to activate FortiGate Cloud with this type of account will fail.

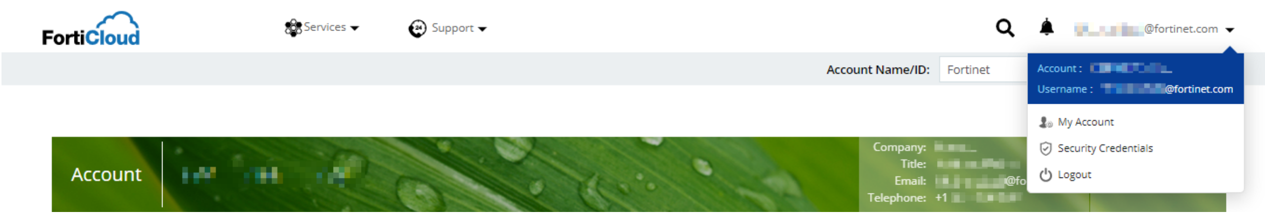
Transfer a device to another FortiCloud account

Master account users can transfer a device from one FortiCloud/FortiCare account to another. Users can transfer a device up to three times within a twelve-month time period. If more transfers are required within the twelve-month time period, contact Technical Support to request the transfer.

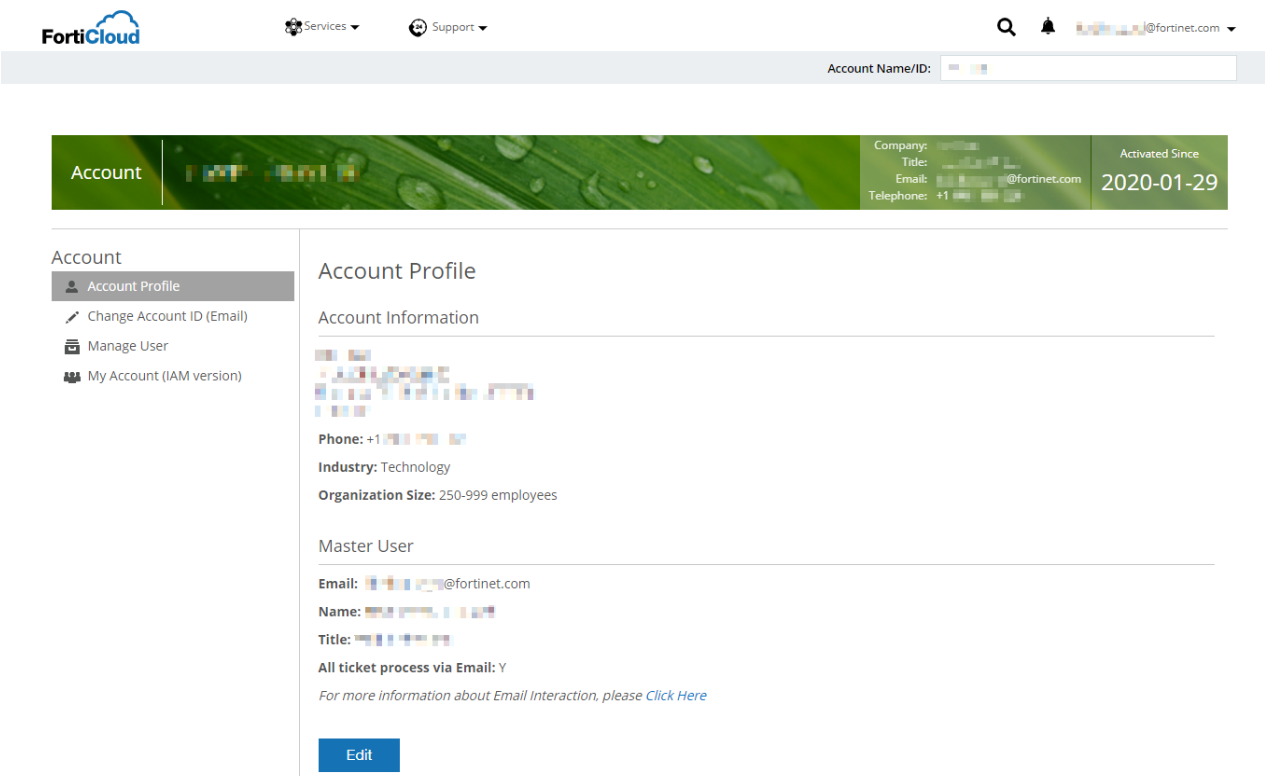
Requirements:

To transfer an account, you must:

- Have access to the FortiGate, as well as both the FortiCloud and FortiCare accounts.
- Be a master account user.
  To verify if you are the master account user, log in to support.fortinet.com. Click the username, then select *My Account*.



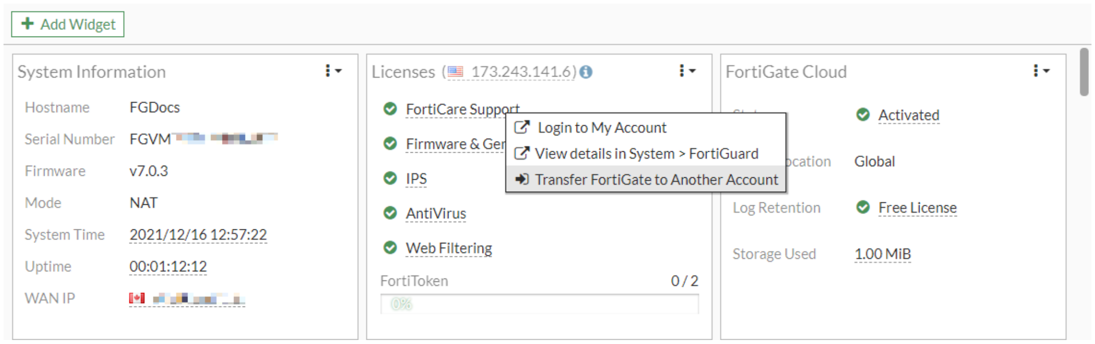
The *Account Profile* page opens.



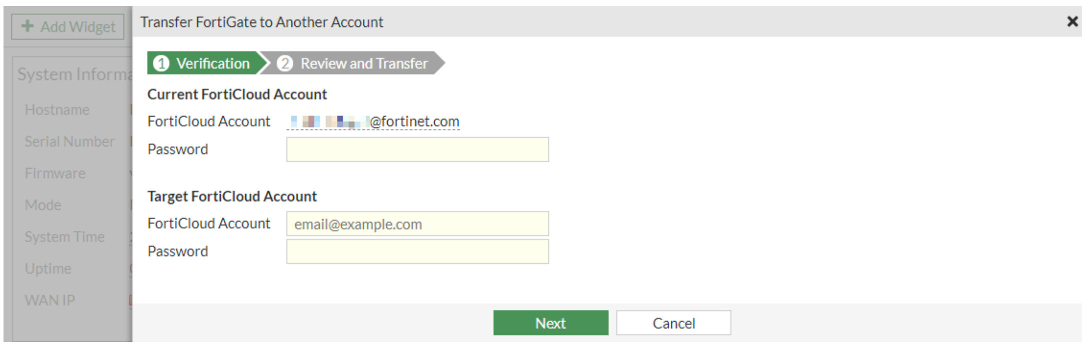
To transfer an account in the GUI:

1. Go to *Dashboard > Status*.
2. In the *Licenses* widget, click the *FortiCare Support* link, then click *Transfer FortiGate to Another Account*.

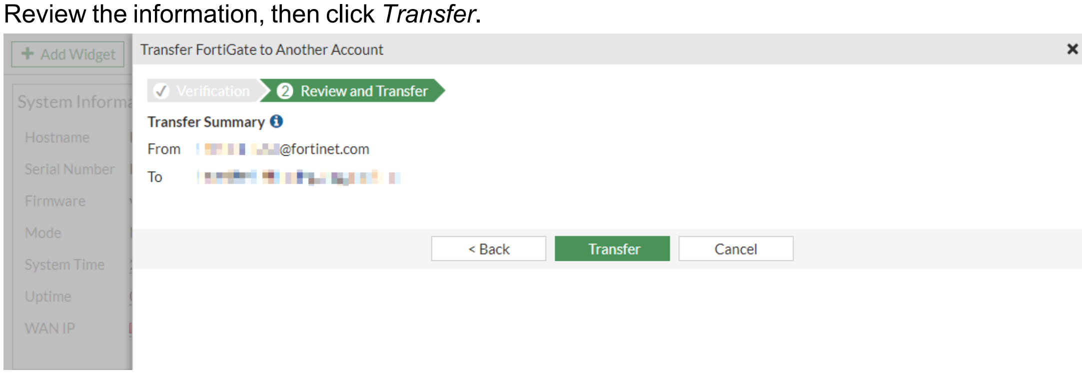
You can also transfer an account from *System > FortiGuard*.



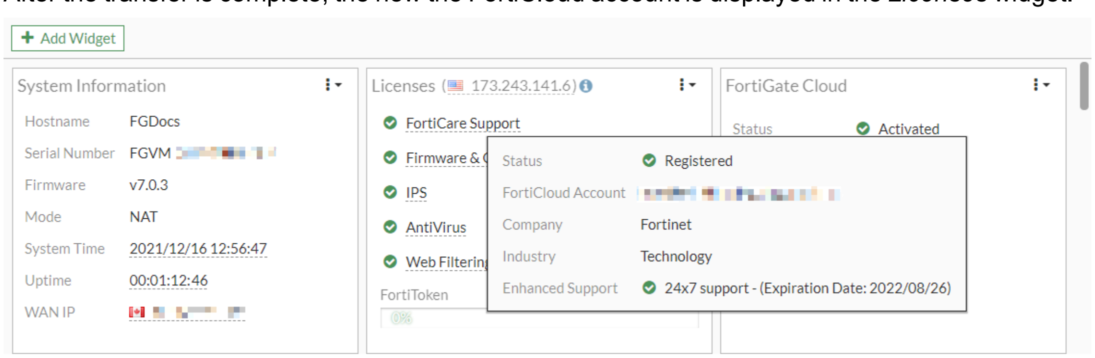
3. In the *Current FortiCloud Account* fields, enter the username and password for the current account. In the *Target FortiCloud Account* fields, enter the new username and password.
4. Click *Next*.



5. Review the information, then click *Transfer*.



After the transfer is complete, the new the FortiCloud account is displayed in the *Licenses* widget.



Configuration backups

Once you successfully configure the FortiGate, it is extremely important that you back up the configuration. You can use the GUI or CLI to back up the configuration in FortiOS format, or you can use the CLI to back up the configuration in YAML format. In some cases, you may need to reset the FortiGate to factory defaults or perform a TFTP upload of the firmware, which will erase the existing configuration. In these instances, the configuration on the device will have to be recreated, unless a backup can be used to restore it. You should also back up the local certificates, as the unique SSL inspection CA and server certificates that are generated by your FortiGate by default are not saved in a system backup.

We also recommend that you back up the configuration after *any* changes are made, to ensure you have the most current configuration available. Also, back up the configuration before any upgrades of the FortiGate's firmware. Should anything happen to the configuration during the upgrade, you can easily restore the saved configuration.

Always back up the configuration and store it on the management computer or off-site. You have the option to save the configuration file in FortiOS format to various locations including the local PC, USB key, FTP, and TFTP server. FTP and TFTP are only configurable through the CLI. In YAML format, configuration files can be backed up or restored on an FTP or TFTP server through the CLI.

If you have VDOMs, you can back up the configuration of the entire FortiGate or only a specific VDOM. Note that if you are using FortiManager or FortiGate Cloud, full backups are performed and the option to back up individual VDOMs will not appear.

You can also back up and restore your configuration using Secure File Copy (SCP). See How to download/upload a FortiGate configuration file using secure file copy (SCP).

You enable SCP support using the following command:

```
config system global
    set admin-scp enable
end
```

For more information about this command and about SCP support, see config system global.

Backing up the configuration

To back up the configuration in FortiOS format using the GUI:

1. Click on the user name in the upper right-hand corner of the screen and select *Configuration > Backup*.
2. Direct the backup to your *Local PC* or to a *USB Disk*.
   The *USB Disk* option will not be available if no USB drive is inserted in the USB port. You can also back up to FortiManager using the CLI.
3. If VDOMs are enabled, indicate whether the scope of the backup is the entire FortiGate configuration (*Global*) or only a specific VDOM configuration (*VDOM*).
   If backing up a VDOM configuration, select the VDOM name from the list.
4. Enable *Encryption*. Encryption must be enabled on the backup file to back up VPN certificates.
5. Enter a password, and enter it again to confirm it. This password will be required to restore the configuration.
6. Click *OK*.
7. When prompted, select a location on the PC or USB disk to save the configuration file. The configuration file will have a .conf extension.

To back up the configuration in FortiOS format using the CLI:

Use one of the following commands:

```
execute backup config management-station <comment>
```

or:

```
execute backup config usb <backup_filename> [<backup_password>]
```

or for FTP, note that port number, username are optional depending on the FTP site:

```
execute backup config ftp <backup_filename> <ftp_server>[<:ftp_port>] [<user_name>]
      [<password>] [<backup_password>]
```

or for TFTP:

```
execute backup config tftp <backup_filename> <tftp_servers> [<backup_password>]
```

or for SFTP:

```
execute backup config sftp <backup_filename> <sftp_server>[<:sftp_port>] <user>
      <password> [<backup_password>]
```

Use the same commands to backup a VDOM configuration by first entering the commands:

```
config vdom
  edit <vdom_name>
```

In FortiOS format, the configuration can be backed up to IPv4 and IPv6 FTP, TFTP, and SFTP servers.

The configuration can be restored from IPv4 and IPv6 FTP and TFTP servers.

To back up the configuration in YAML format using the CLI:

```
# execute backup yaml-config {ftp | tftp} <filename> <server> [username] [password]
```

For example:

```
# execute backup yaml-config  tftp  301E.yaml 172.16.200.55
    Please wait...
    Connect to tftp server 172.16.200.55 ...
    #
    Send config file to tftp server OK.
```

In YAML format, the configuration can be backed up to and restored from an FTP or TFTP server.

Restoring a configuration

To restore the FortiGate configuration in FortiOS format using the GUI:

1. Click on the user name in the upper right-hand corner of the screen and select *Configuration* > *Restore*.
2. Identify the source of the configuration file to be restored: your *Local PC* or a *USB Disk*.
   The *USB Disk* option will not be available if no USB drive is inserted in the USB port. You can restore from the FortiManager using the CLI.
3. Click Upload, locate the configuration file, and click *Open*.
4. Enter the password if required.
5. Click *OK*.

To restore the FortiGate configuration in FortiOS format using the CLI:

```
execute restore config management-station normal 0
```

or:

```
execute restore config usb <backup_filename> [<backup_password>]
```

or for FTP, note that port number, username are optional depending on the FTP site:

```
execute restore config ftp <backup_filename> <ftp_server>[<:port>] [<user_name>]
      [<password>] [<backup_password>]
```

or for TFTP:

```
execute restore config tftp <backup_filename> <tftp_server> [<backup_password>]
```

The FortiGate will load the configuration file and restart. Once the restart has completed, verify that the configuration has been restored.

To restore configuration files in YAML format:

```
# execute restore yaml-config {ftp | tftp} <filename> <server> [username] [password]
```

For example:

```
# execute restore  yaml-config  ftp  301E-1.yaml 172.16.200.55 root sys@qa123456
    This operation will overwrite the current setting and could possibly reboot the system!
    Do you want to continue? (y/n) y
    Please wait...
    Connect to ftp server 172.16.200.55 ...
    Get config file from ftp server OK.
    File check OK.
    #
    The system is going down NOW !!
```

Troubleshooting

When restoring a configuration, errors may occur, but the solutions are usually straightforward.

Error message	Reason and Solution
Configuration file error	This error occurs when attempting to upload a configuration file that is incompatible with the device. This may be due to the configuration file being for a different model or being saved from a different version of firmware. **Solution**: Upload a configuration file that is for the correct model of FortiGate device and the correct version of the firmware.
Invalid password	When the configuration file is saved, it can be protected by a password. The password entered during the upload process is not matching the one associated with the configuration file. **Solution**: Use the correct password if the file is password protected.

Configuration revision

You can manage multiple versions of configuration files on models that have a 512MB flash memory and higher. Revision control requires either a configured central management server or the local hard drive, if your FortiGate has this feature. Typically, configuration backup to local drive is not available on lower-end models.

The central management server can either be a FortiManager unit or FortiGate Cloud.

If central management is not configured on your FortiGate unit, a message appears instructing you to either

- Enable central management, or
- Obtain a valid license.

When revision control is enabled on your FortiGate unit, and configuration backups have been made, a list of saved revisions of those backed-up configurations appears.

Configuration revisions are viewed by clicking on the user name in the upper right-hand corner of the screen and selecting *Configuration* > *Revisions*.

Back up and restore the local certificates

This procedure exports a server (local) certificate and private key together as a password protected PKCS12 file. The export file is created through a customer-supplied TFTP server. Ensure that your TFTP server is running and accessible to the FortiGate before you enter the command.

To back up the local certificates:

Connect to the CLI and use the following command:

```
execute vpn certificate local export tftp <cert_name> <filename> <tftp_ip>
```

where:

- `<cert_name>` is the name of the server certificate.
- `<filename>` is a name for the output file.
- `<tftp_ip>` is the IP address assigned to the TFTP server host interface.

To restore the local certificates using the GUI:

1. Move the output file from the TFTP server location to the management computer.
2. Go to *System* > *Certificates* and click *Import* > *Local*.
3. Select the certificate type, then click *Upload* in the *Certificate file* field.
4. On the management computer, browse to the file location, select it, and click *Open*.
5. If the *Type* is *Certificate*, upload the *Key file* as well.
6. If required, enter the *Password* that is required to upload the file or files.
7. Click *OK*.

To restore the local certificates using the CLI:

Connect to the CLI and use the following command:

```
execute vpn certificate local import tftp <filename> <tftp_ip>
```

Restore factory defaults

There may be a need to reset the FortiGate to its original defaults; for example, to begin with a fresh configuration. There are two options when restoring factory defaults. The first resets the entire device to the original out-of-the-box configuration.

You can reset the device with the following CLI command:

```
execute factoryreset
```

When prompted, type `y` to confirm the reset.

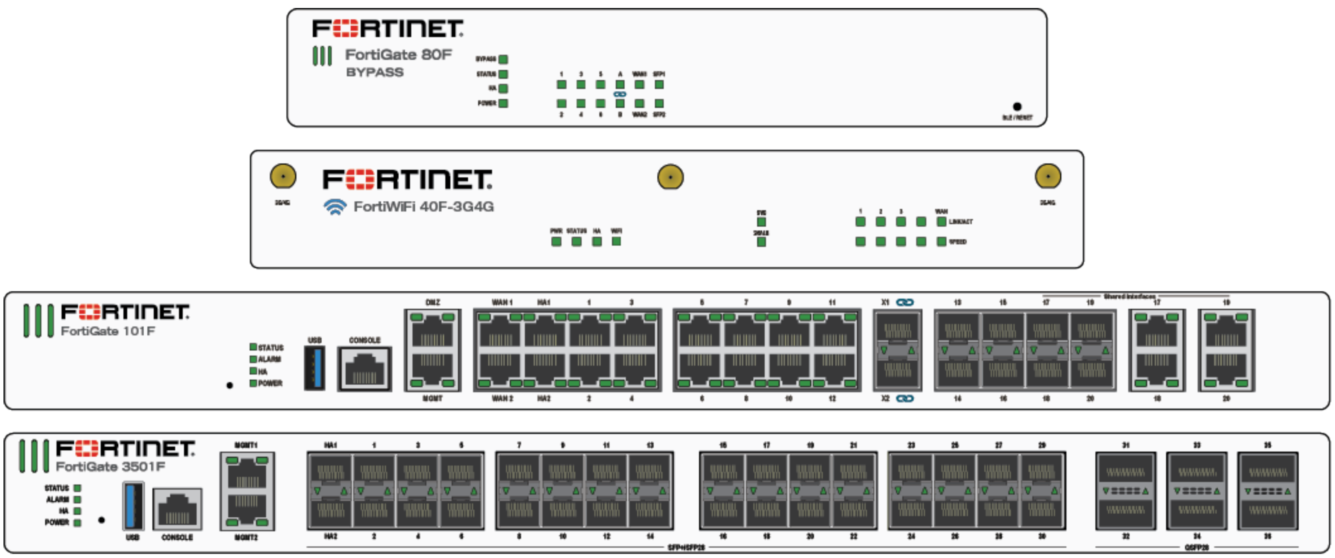
Alternatively, in the CLI you can reset the factory defaults but retain the interface and VDOM configuration with the following command:

```
execute factoryreset2
```

LEDs

Check your device's QuickStart guide for specific LED information: FortiGate QuickStart Guides.

The following faceplates show where the LEDs are typically found on FortiGate models:



LED	State	Description
Logo	Green	The unit is on
	Blue	The FortiWiFi unit is on
	Off	The unit is off

LED	State	Description
Power (PWR)	Green	The unit is on and/or both power supplies are functioning
	Amber	One power supply is functioning
	Flashing Amber	Power supply failure
	Red	The unit is on, but only one power supply is functional
	Flashing Red	Power failure
	Off	The unit is off
Status (STA)	Green	Normal
	Flashing Green	Booting up
	Amber	Major or minor alarm
	Flashing Amber	BLE is on
	Red	Major alarm
	Flashing Red	BLE is on
	Off	The unit is off
Bypass (BYP)	Amber	Bypass Port Pair is active
	Off	Bypass Port Pair is off
Alarm	Red	Major alarm
	Amber	Minor alarm
	Off	No alarms
HA	Green	Operating in an HA cluster
	Amber or Red	HA failover
	Off	HA disabled
Max PoE	Green, Amber, or Red	Maximum PoE power allocated
	Off	PoE power available or normal
PoE	Green	Power delivered
	Flashing Green	Error or PoE device requesting power
	Off	No PoE device connected or no power delivered
SVC	Green	SVC is on
	Flashing Green	SVC activity
	Off	SVC is off

LED	State	Description
3G / 4G	Green	3G / 4G service is on
	Flashing Green	3G / 4G activity
	Off	3G / 4G service is off
WiFi	Green	WiFi connected
	Flashing Green	WiFi activity
	Off	WiFi is off
Power Supply	Green	Power supply operating normally
	Flashing Green	Power detected, but power supply not providing power or is in standby mode
	Amber	Power output is off, there is a power supply error, or there is no input power but the redundant supply is on
	Flashing Amber	Power supply error or warning events, or the power supply should be replaced
	Red	Cord unplugged or power lost
	Flashing Red	Power supply warning events
	Off	Power not detected
Power Supply OK	Green	Standby rail and main output on
	Flashing Green	Standby rail and main output off
	Off	Error or no AC power input
Power Supply Fail	Amber	Main output or fan error detected
	Flashing Amber	Power supply warning event detected
	Off	No errors or no power
Power Supply Input	Green	Input voltage within normal range
	Flashing Green	Over or under voltage warning
	Off	No input power
Power Supply Output	Green	Output voltage normal
	Flashing Green	Standby mode
	Amber	Critical error
	Flashing Amber	Warning
	Off	No output

LED	State	Description
Fan	Green	Fan(s) operating normally
	Flashing Green	Fan switching/initialization in progress
	Amber	Fan failure
	Red	Fan error, RPM too low or too high, or both fan sets have at least one alert
	Flashing Red	One fan set has at least one alert
	Off	Fan error or fan is off

Port LEDs

LED	State	Description
Ethernet	Green	Connected at 1 Gbps
	Flashing Green	Transmitting and receiving data at 1 Gbps
	Amber	Connected at 10/100 Mbps
	Flashing Amber	Transmitting and receiving data at 10/100 Mbps
	Off	No link established
Ethernet Link/Activity	Green	Connected
	Flashing Green	Transmitting data
	Off	No link established
Ethernet Speed	Green	Connected at 1 Gbps
	Amber	Connected at 100 Mbps
	Off	Not connected or connected at 10 Mbps
Ethernet 10G Link/Activity	Green	Connected
	Flashing Green	Transmitting data
	Off	No link established
Ethernet 10G Speed	Green	Connected at 10 Gbps
	Amber	Connected at 5 Gbps, 2.5 Gbps, or 1 Gbps
	Off	Not connected or connected at 100 Mbps

LED	State	Description
PoE	Green	PoE power on or PoE device receiving power
	Amber	Providing power
	Red	Connected but not powered
	Off	PoE power off or no device receiving power
SFP	Green	Connected at 1 Gbps
	Flashing Green	Data activity
	Off	No link established
SFP+	Green	Connected at 10 Gbps or 1 Gbps
	Flashing Green	Data activity
	Off	No link established
SFP28	Green	Connected at 25 Gbps, 10 Gbps, or 1 Gbps
	Flashing Green	Data activity
	Off	No link established
QSFP28	Green	Connected at 100 Gbps or 40 Gbps
	Flashing Green	Data activity
	Off	No link established

Alarm levels

Minor alarm

Also called an IPMI non-critical (NC) alarm, it indicates a temperature or power level outside of the normal operating range that is not considered a problem. For a minor temperature alarm, the system could respond by increasing the fan speed. A non-critical threshold can be an upper non-critical (UNC) threshold (for example, a high temperature or a high power level) or a lower non-critical (LNC) threshold (for example, a low power level).

Major alarm

Also called an IPMI critical or critical recoverable (CR) alarm, it indicates that the system is unable to correct the cause of the alarm, and that intervention is required. For example, the cooling system cannot provide enough cooling to reduce the temperature. It can also mean that the conditions are approaching the outside limit of the allowed operating range. A critical threshold can also be an upper critical (UC) threshold (such as a high temperature or high power level) or a lower critical (LC) threshold (such as a low power level).

Critical alarm

Also called an IPMI non-recoverable (NR) alarm, it indicates that the system has detected a temperature or power level that is outside of the allowed operating range and physical damage is possible.

Troubleshooting your installation

If your FortiGate does not function as desired after installation, try the following troubleshooting tips:

1. **Check for equipment issues**
   Verify that all network equipment is powered on and operating as expected. Refer to the QuickStart Guide for information about connecting your FortiGate to the network.
2. **Check the physical network connections**
   Check the cables used for all physical connections to ensure that they are fully connected and do not appear damaged, and make sure that each cable connects to the correct device and the correct Ethernet port on that device.
3. **Verify that you can connect to the internal IP address of the FortiGate**
   Connect to the GUI from the FortiGate's internal interface by browsing to its IP address. From the PC, try to ping the internal interface IP address; for example, `ping 192.168.1.99`. If you cannot connect to the internal interface, verify the IP configuration of the PC. If you can ping the interface but can't connect to the GUI, check the settings for administrative access on that interface. Alternatively, use SSH to connect to the CLI, and then confirm that HTTPS has been enabled for Administrative Access on the interface.
4. **Check the FortiGate interface configurations**
   Check the configuration of the FortiGate interface connected to the internal network (under *Network > Interfaces*) and check that *Addressing mode* is set to the correct mode.
5. **Verify the security policy configuration**
   Go to *Policy & Objects > Firewall Policy* and verify that the internal interface to Internet-facing interface security policy has been added and is located near the top of the policy list. Check the *Active Sessions* column to ensure that traffic has been processed (if this column does not appear, right-click on the table header and select *Active Sessions*). If you are using NAT mode, check the configuration of the policy to make sure that *NAT* is enabled and that *Use Outgoing Interface Address* is selected.
6. **Verify the static routing configuration**
   Go to *Network > Static Routes* and verify that the default route is correct. Go to *Monitor > Routing Monitor* and verify that the default route appears in the list as a static route. Along with the default route, you should see two routes shown as *Connected*, one for each connected FortiGate interface.
7. **Verify that you can connect to the Internet-facing interface's IP address**
   Ping the IP address of the Internet-facing interface of your FortiGate. If you cannot connect to the interface, the FortiGate is not allowing sessions from the internal interface to Internet-facing interface. Verify that PING has been enabled for *Administrative Access* on the interface.
8. **Verify that you can connect to the gateway provided by your ISP**
   Ping the default gateway IP address from a PC on the internal network. If you cannot reach the gateway, contact your ISP to verify that you are using the correct gateway.
9. **Verify that you can communicate from the FortiGate to the Internet**
   Access the FortiGate CLI and use the command `execute ping 8.8.8.8`. You can also use the `execute traceroute 8.8.8.8` command to troubleshoot connectivity to the Internet.
10. **Verify the DNS configurations of the FortiGate and the PCs**
    Check for DNS errors by pinging or using traceroute to connect to a domain name; for example: `ping www.fortinet.com`.

    If the name cannot be resolved, the FortiGate or PC cannot connect to a DNS server and you should confirm that the DNS server IP addresses are present and correct.

11. **Confirm that the FortiGate can connect to the FortiGuard network**
    Once the FortiGate is on your network, you should confirm that it can reach the FortiGuard network. First, check the *License Information* widget to make sure that the status of all FortiGuard services matches the services that you have purchased. Go to *System* > *FortiGuard*, and, in the Filtering section, click *Test Connectivity*. After a minute, the GUI should indicate a successful connection. Verify that your FortiGate can resolve and reach FortiGuard at `service.fortiguard.net` by pinging the domain name. If you can reach this service, you can then verify the connection to FortiGuard servers by running the command `diagnose debug rating`. This displays a list of FortiGuard IP gateways you can connect to, as well as the following information:
    - `Weight`: Based on the difference in time zone between the FortiGate and this server
    - `RTT`: Return trip time
    - `Flags`: D (IP returned from DNS), I (Contract server contacted), T (being timed), F (failed)
    - `TZ`: Server time zone
    - `Curr Lost`: Current number of consecutive lost packets
    - `Total Lost`: Total number of lost packets
12. **Consider changing the MAC address of your external interface**
    Some ISPs do not want the MAC address of the device connecting to their network cable to change. If you have added a FortiGate to your network, you may have to change the MAC address of the Internet-facing interface using the following CLI command:
    ```
    config system interface
      edit <interface>
        set macaddr <xx:xx:xx:xx:xx:xx>
      end
    end
    ```
13. **Check the FortiGate bridge table (transparent mode)**
    When a FortiGate is in transparent mode, the unit acts like a bridge sending all incoming traffic out on the other interfaces. The bridge is between interfaces on the FortiGate unit. Each bridge listed is a link between interfaces. Where traffic is flowing between interfaces, you expect to find bridges listed. If you are having connectivity issues and there are no bridges listed, that is a likely cause. Check for the MAC address of the interface or device in question. To list the existing bridge instances on the FortiGate, use the following CLI command:
    ```
    diagnose netlink brctl name host root.b
    show bridge control interface root.b host.
    fdb: size=2048, used=25, num=25, depth=1
    Bridge root.b host table
    port no device devname mac addr ttl attributes
    3 4 wan1 00:09:0f:cb:c2:77 88
    3 4 wan1 00:26:2d:24:b7:d3 0
    3 4 wan1 00:13:72:38:72:21 98
    4 3 internal 00:1a:a0:2f:bc:c6 6
    1 6 dmz 00:09:0f:dc:90:69 0 Local Static
    3 4 wan1 c4:2c:03:0d:3a:38 81
    3 4 wan1 00:09:0f:15:05:46 89
    3 4 wan1 c4:2c:03:1d:1b:10 0
    2 5 wan2 00:09:0f:dc:90:68 0 Local Static
    ```
14. **Use FortiExplorer if you cannot connect to the FortiGate over Ethernet**
    If you cannot connect to the FortiGate GUI or CLI, you may be able to connect using FortiExplorer. Refer to the QuickStart Guide or see the section on FortiExplorer for more details.
15. **Either reset the FortiGate to factory defaults or contact Fortinet Support for assistance**
    To reset the FortiGate to factory defaults, use the CLI command `execute factoryreset`. When prompted, type `y` to confirm the reset.

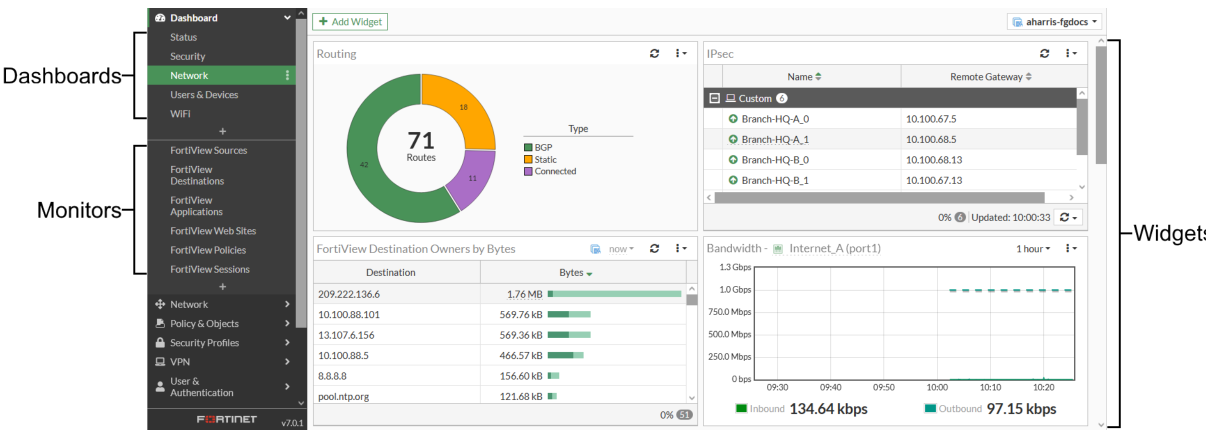
If you require further assistance, visit the Fortinet Support website.

Dashboards and Monitors

FortiOS includes predefined dashboards so administrators can easily monitor device inventory, security threats, traffic, and network health. You can customize the appearance of a default dashboard to display data pertinent to your Security Fabric or combine widgets to create custom dashboards. Many dashboards also allow you to switch views between fabric devices.

Each dashboard contains a set of widgets that allow you to view drilldown data and take actions to prevent threats. Use widgets to perform tasks such as viewing device inventory, creating and deleting DHCP reservations, and disconnecting dial-up users. You can add or remove widgets in a dashboard or save a widget as a standalone monitor.

Monitors display information in both text and visual format. Use monitors to change views, search for items, view drilldown information, or perform actions such as quarantining an IP address. FortiView monitors for the top categories are located below the dashboards. All of the available widgets can be added to the tree menu as a monitor.

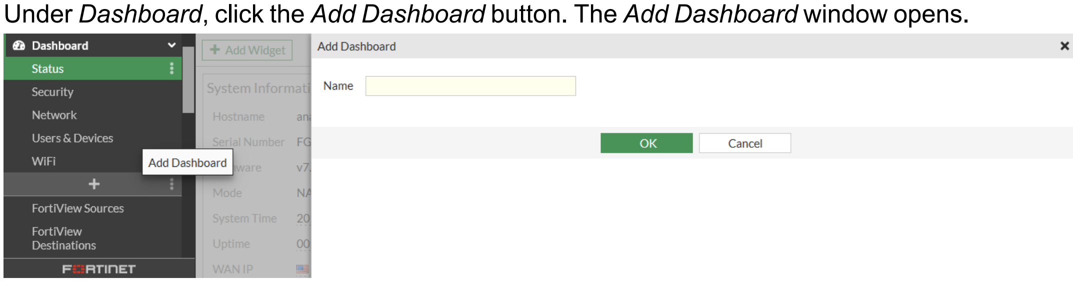


Using dashboards

You can combine widgets to create custom dashboards. You can also use the dropdown in the tree menu to switch to another device in the Security Fabric.

To create a new dashboard:

1. Under *Dashboard*, click the *Add Dashboard* button. The *Add Dashboard* window opens.

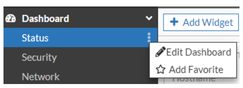


2. Enter a name in the *Name* field and click *OK*. The new dashboard opens.

To add a widget to a dashboard:

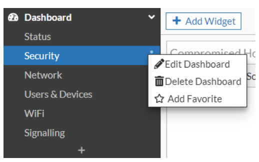
1. In the tree menu, select a dashboard.
2. In the banner, click *Add Widget*. The *Add Dashboard Widget* pane opens.
3. Click the *Add* button next to the widget. You can use the *Search* field to search for a widget. Enable *Show More* to view more widgets in a category.
4. Configure the widget settings, then click *Add Widget*.
5. Click *Close*.
6. (Optional) Click and drag the widget to the desired location in the dashboard.

To edit a dashboard:

1. Click the *Actions* menu next to the dashboard and select*Edit Dashboard*.


2. Edit the dashboard and click *OK*.

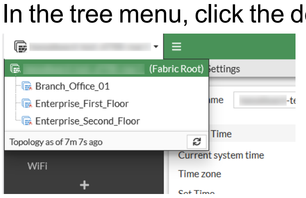
To delete a dashboard:

1. Click the *Actions* menu next to the dashboard and select *Delete Dashboard*.


2. Click *Delete Dashboard* . The *Confirm* dialog opens.
3. *Click OK.*

You cannot delete the *Status* dashboard.

To switch to another device in the Security Fabric:

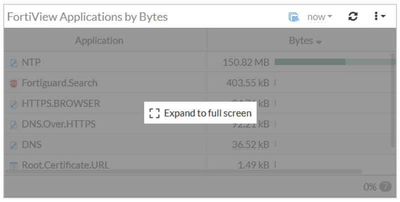
1. In the tree menu, click the device name and select a fabric device from dropdown.


Using widgets

You can convert a widget to a standalone monitor, change the view type, configure tables, and filter data.

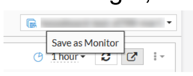
To save a dashboard widget as a monitor:

1. Hover over the widget and click *Expand to full screen*.



Full screen mode is not supported in all widgets.

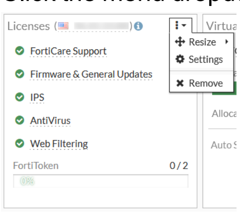
2. In the widget, click *Save as Monitor*. The *Add Monitor* window opens.



3. (Optional) Enter a new name for the monitor in the *Name* field.
4. Click *OK*.

To view the widget settings:

1. Click the menu dropdown at the right side of the widget and select *Settings*.



2. Configure the widget settings and click *OK*.

The settings will vary depending on the widget.

To configure a table in the widget:

1. Hover over the left side of the table header and click *Configure Table*.

2. Configure the table options:

Option	Description
Best Fit All Columns	Resizes all of the columns in a table to fit their content.
Reset Table	Resets the table to the default view.
Select Columns	Adds or removes columns from the view.

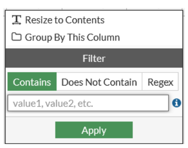
3. Click *Apply*.

To filter or configure a column in a table:

1. Hover over a column heading, and click *Filter/Configure Column*.

2. Configure the column options.

Option	Description
Resize to Contents	Resizes the column to fit the content.
Group by this Column	Groups the table rows by the contents in the selected column.



3. Click *Apply*.
4. To filter a column, enter a value in the *Filter* field, and click *Apply*.

Filtering is not supported in all widgets.

Widgets

Dashboards are created per VDOM when VDOM mode is enabled.For information about VDOM mode, see Virtual Domains on page 1994.

Some dashboards and widgets are not available in Multi-VDOM mode.

The following table lists the available widgets in VDOM mode:

Category	Widgets
FortiView	• FortiView Application Bandwidth FortiView

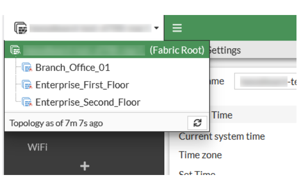
Category	Widgets
	• Applications FortiView Cloud Applications • FortiView Destination Interfaces FortiView • Destination Owners FortiView Destinations • FortiView Policies FortiView Sessions • FortiView Source Interfaces FortiView • Sources FortiView VPN FortiView Web • Categories FortiView Countries/Regions • FortiView Destination Firewall Objects • FortiView Interface Pairs FortiView Search • Phrases FortiView Servers FortiView Source • Firewall Objects FortiView Sources - WAN • FortiView Traffic Shaping
Security Fabric	• Fabric Device • FortiGate Cloud • Security Fabric Status
Network	• DHCP • Interface Bandwidth • IP Pool Utilization • IPsec • Routing • SD-WAN • SSL-VPN • Top IP Pools by Assigned IPs The *Interface Bandwidth* widget can monitor a maximum of 25 interfaces.
System	• Administrators • Botnet Activity • HA Status • License Status • System Information • Top System Events • Virtual Machine
Resource Usage	• CPU Usage • Disk Usage • Log Rate Memory Usage • Session Rate • Sessions
Security	• Advanced Threat Protection Statistics • Compromised Hosts

Category	Widgets
	• FortiClient Detected Vulnerabilities • GTP Tunnel Rate • GTP Tunnels • Host Scan Summary • Quarantine • Top Endpoint Vulnerabilities • Top Failed Authentication • Top FortiSandbox Files • Top Threats • Top Threats - WAN
User & Authentication	• Device Inventory • Firewall Users • FortiClient • FortiGuard Quota • FortiSwitch NAC VLANs • Top Admin Logins • Top Vulnerable Endpoint Devices • Top Cloud Users
WiFi	• Channel Utilization • Clients By FortiAP • FortiAP Status • Historical Clients • Interfering SSIDs • Login Failures • Rogue APs • Signal Strength • Top WiFi Clients

Viewing device dashboards in the Security Fabric

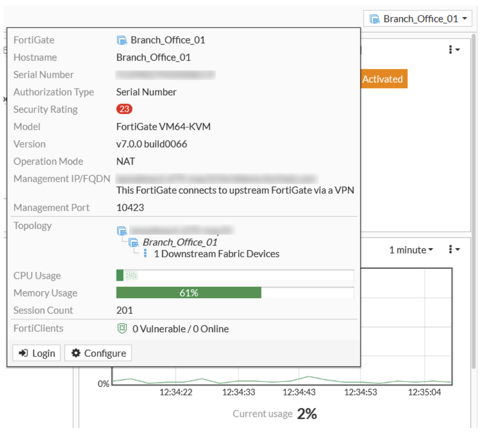
Use the device dropdown to view the dashboards in downstream fabric devices. You can also create dedicated device dashboards or log in and configure fabric devices.

To view the dashboards in fabric devices, click the device dropdown at the left side of the page, and select a device from the list.



The device dropdown is available in the *Status*, *Security*, *Network*, *Users & Devices*, and *WiFi* dashboards. You can also enable the dropdown when you create a dashboard.

To log in to or configure a fabric device, hover over the device name until the device dialog opens and then select *Login* or *Configure*.

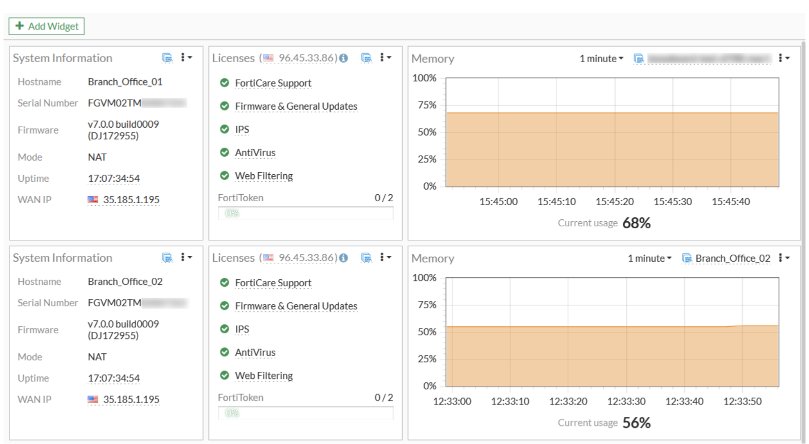


Creating a fabric system and license dashboard

Create a dashboard summary page to monitor all the fabric devices in a single view. You can use this dashboard to monitor aspects of the devices such as system information, VPN and routing.

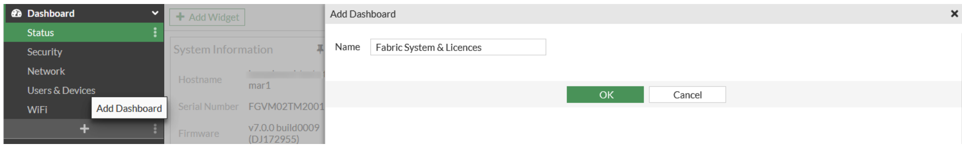
Example

The following image is an example of a *Fabric System & License* dashboard to monitor the *System Information*, *Licenses*, and *Memory* usage for *Branch_Office_01* and *Branch_Office_02*.

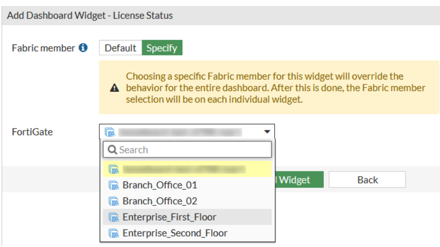


To create a system dashboard:

1. Click the *Add Dashboard* button. The *Add Dashboard* window opens.



2. In the *Name* field, enter a name such as *Fabric System & Licenses*, and click *OK*. The new dashboard appears.
3. In the banner, click *Add Widget*. The *Add Dashboard Widget* window opens. You can use the *Search* field to search for a specific widget (for example, *License Status*, *System Information*, and *Memory Usage*).
4. Click the *Add* button next to widget. The *Add Dashboard Widget* window opens.
5. In the *Fabric member* area, select *Specify* and select a device in the Security Fabric.



6. Click *Add Widget*. The widget is added to the dashboard.
   Repeat this step for all the devices you want to view in the dashboard.
7. (Optional) Arrange the widgets in the dashboard by fabric device.

Dashboards

A dashboard is a collection of widgets that show the status of your devices, network, and Security Fabric at a glance. Widgets are condensed monitors that display a summary of the key details about your FortiGate pertaining to routing, VPN, DHCP, devices, users, quarantine, and wireless connections.

The following dashboards are included in the dashboard templates:

Dashboard	Default Template	Use these widgets to:
Status	• Comprehensive • Optimal	• View the device serial number, licenses, and administrators • View the status of devices in the security fabric • Monitor CPU and Memory usage • Monitor IPv4 and IPv6 sessions • View VMs and Cloud devices
Security	• Optimal	• View compromised hosts and host scan summary • View top threats and vulnerabilities
Network	• Optimal	• Monitor DHCP clients • Monitor IPsec VPN connections • Monitor current routing table • Monitor SD-WAN status • Monitor SSL-VPN connections

Dashboard	Default Template	Use these widgets to:
Users & Devices	• Optimal	• View users and devices connected to the network • Identify threats from individual users and devices • View FortiGuard and FortiClient data • Monitor traffic bandwidth over time
WiFi	• Comprehensive • Optimal	• View FortiAP status, channel utilization, and clients • View login failures and signal strength • View the number of WiFi clients

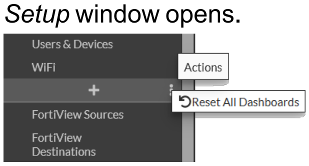
Resetting the default dashboard template

You can use the GUI to change the default dashboard template. The *Optimal* template contains a set of popular default dashboards and FortiView monitors. The *Comprehensive* template contains a set of default dashboards as well as all of the FortiView monitors.

Resetting the default template will delete any custom dashboards and monitors, and reset the widget settings.

To reset all dashboards:

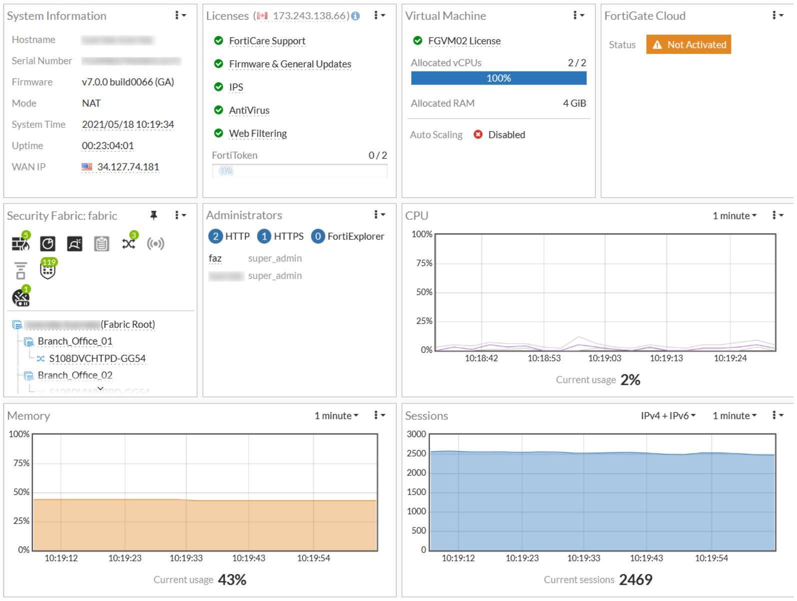
1. Click the *Actions* menu next to *Add Dashboard* or *Add Monitor* and click *Reset All Dashboards*. The *Dashboard Setup* window opens.



2. Select *Optimal* or *Comprehensive* and click *OK*.

Status dashboard

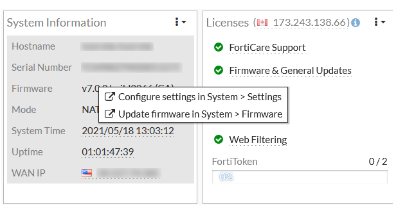
The *Status* dashboard provides an overview of your FortiGate device and the devices in your Security Fabric. If your FortiGate is a Virtual Machine, information about the Virtual Machine is also displayed in the dashboard.



Updating system information

The *System Information* widget contains links to the *Settings* module where you can update the *System Time*, *Uptime*, *and WAN IP*.

A notification will appear in the *Firmware* field when a new version of FortiOS is released. Click *Update firmware in System > Firmware* to view the available versions and update FortiOS.



Viewing fabric devices

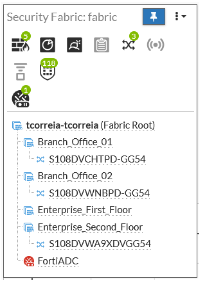
The *Security Fabric* widget provides a visual overview of the devices connected to the fabric and their connection status. Hover of a device icon to view more information about the device.

Click a device in the fabric to:

- View the device in the physical or logical topology
- Register, configure, deauthorize, or log in to the device
- Open *Diagnostics and Tools*
- View the *FortiClient Monitor*

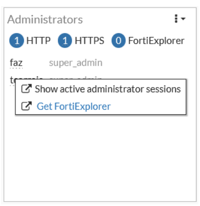
These options will vary depending on the device.

Click *Expand & Pin hidden content* to view all the devices in the fabric at once.



Viewing administrators

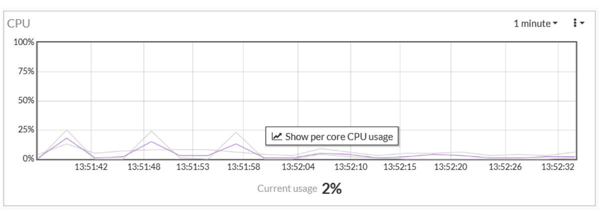
The *Administrators* widget displays the active administrators and their access interface. Click the username to view the *Active Administrator Sessions* monitor. You can use the monitor to end an administrator's session.



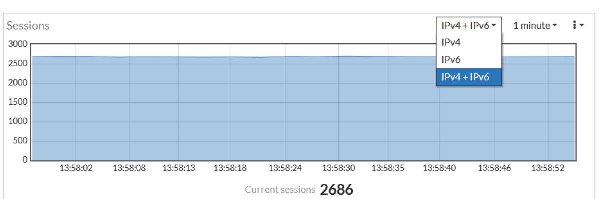
Resource widgets

The resource widgets show the current usage statistics for *CPU*, *Memory*, and *Sessions*.

Click the *CPU* monitor to show the per core CPU usage.



You can switch between *IPv4*, *IPv6*, or *IPv4+IPv6* in the *Sessions* monitor.



Security dashboard

The widgets in the *Security* dashboard provide a snapshot of the current threats and vulnerabilities targeting your Security Fabric.

The *Security* dashboard contains the following widgets:

Widget	Description
Compromised Hosts by Verdict	Shows the session information for a compromised host. See Viewing session information for a compromised host on page 87.
Top Threats by Threat Level	Shows the top traffic sessions aggregated by threat. You can expand the widget to view drilldown information about the *Threat*, *Threat Category*, *Threat Level*, *Threat Score* and *Sessions*.
FortiClient Detected Vulnerablities	Shows a summary of vulnerabilities detected by FortiClient. FortiClient must be enabled.
Host Scan Summary	Shows a summary of hosts scanned. Hover over a color in the chart to view the number of hosts by category. Click the chart to view the *FortiClient Monitor* or *Device Inventory* monitor.
Top Vulnerable Endpoint Devices by Detected Vulnerabilities	Shows a summary devices aggregated by vulnerabilities. Expand the widget to view drilldown information about the *Device*, *Source* and *Detected Vulnerablities*.

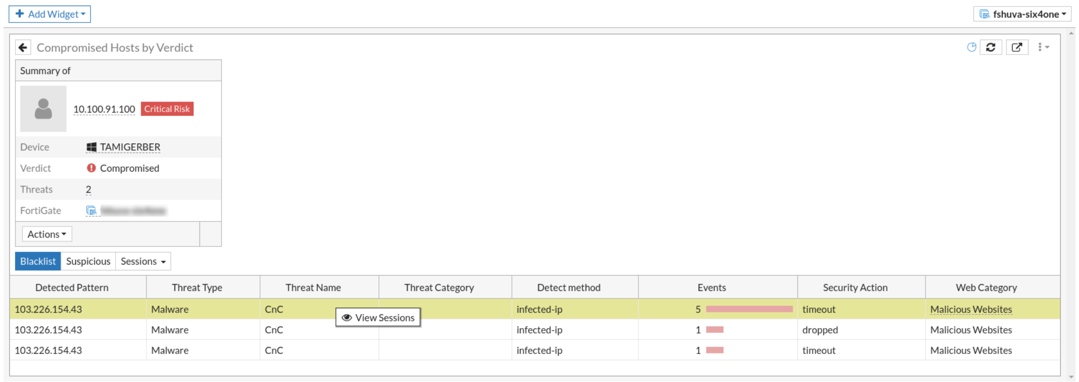
Viewing session information for a compromised host

You can use the *Compromised Hosts by Verdict* widget to view the session information for a compromised host.

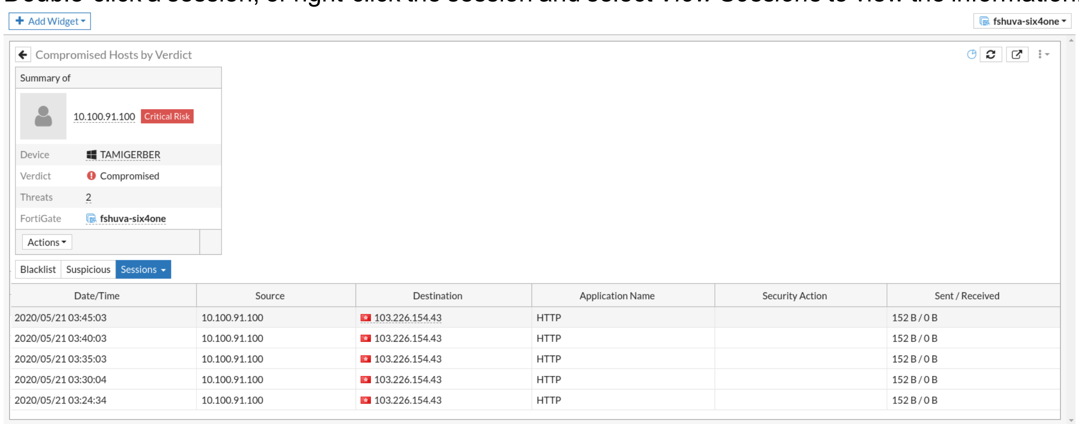
To view session information for a compromised host in the GUI:

1. Go to *Dashboard > Security* and expand the *Compromised Hosts by Verdict* widget.

2. Double-click a compromised host to view the session information. You can also right-click a compromised host, and select *View Sessions*.



3. Double-click a session, or right-click the session and select *View Sessions* to view the information.



Network dashboard

The widgets in the Network dashboard show information related to networking for this FortiGate and other devices connected to your Security Fabric. Use this dashboard to monitor the status of Routing, DHCP, SD-WAN, IPsec and SSL VPN tunnels. All of the widgets in the *Network* dashboard can be expanded to full screen and saved as a monitor.

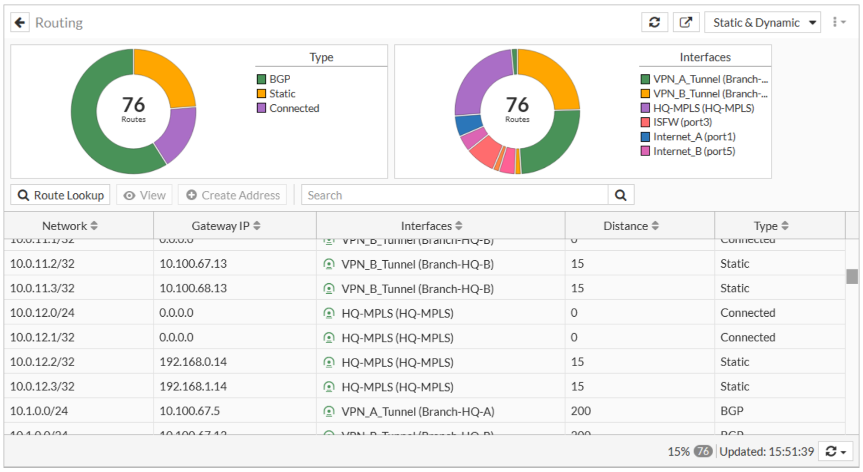
The *Network* dashboard contains the following widgets:

Widget	Description
Static & Dynamic Routing	Shows the static and dynamic routes currently active in your routing table. The widget also includes policy routes, BGP neighbors and paths, and OSPF neighbors. See Static & Dynamic Routing monitor on page 89.
DHCP	Shows the addresses leased out by FortiGate's DHCP servers. See DHCP monitor on page 92.
SD-WAN	Shows a summary of the SD-WAN status, including ADVPN shortcut information.
IPsec	Shows the connection statuses of your IPsec VPN site to site and dial-up tunnels. See IPsec monitor on page 93.
SSL-VPN	Shows a summary of remote active users and the connection mode. See SSL-VPN monitor on page 95.

Widget	Description
IP Pool Utilization	Shows IP pool utilization.

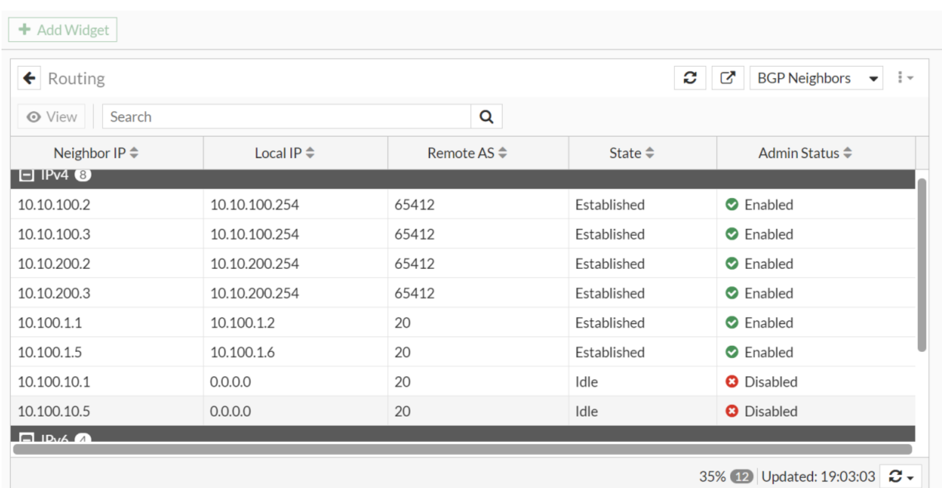
Static & Dynamic Routing monitor

The *Static & Dynamic Routing Monitor* displays the routing table on the FortiGate, including all static and dynamic routing protocols in IPv4 and IPv6. You can also use this monitor to view policy routes, BGP neighbors and paths, and OSPF neighbors.

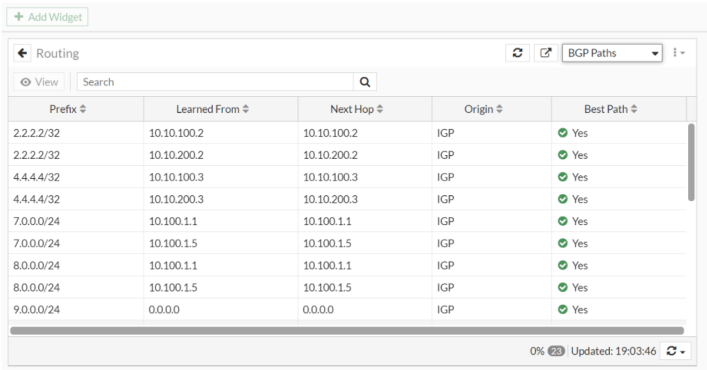


To view the routing monitor in the GUI:

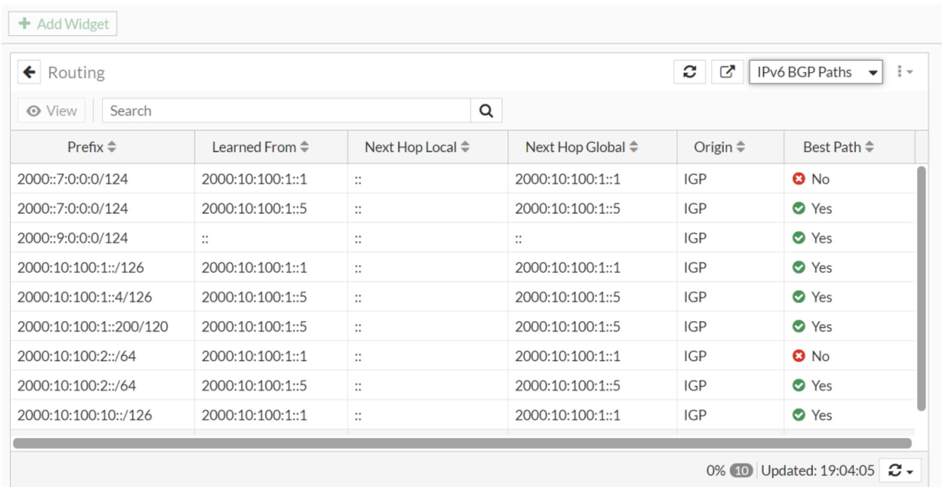
1. Go to *Dashboard > Network*.
2. Hover over the *Routing* widget, and click *Expand to Full Screen*. The *Routing* monitor is displayed.
3. To view neighbors and paths, click the monitors dropdown at the top of the page.
   BGP Neighbors

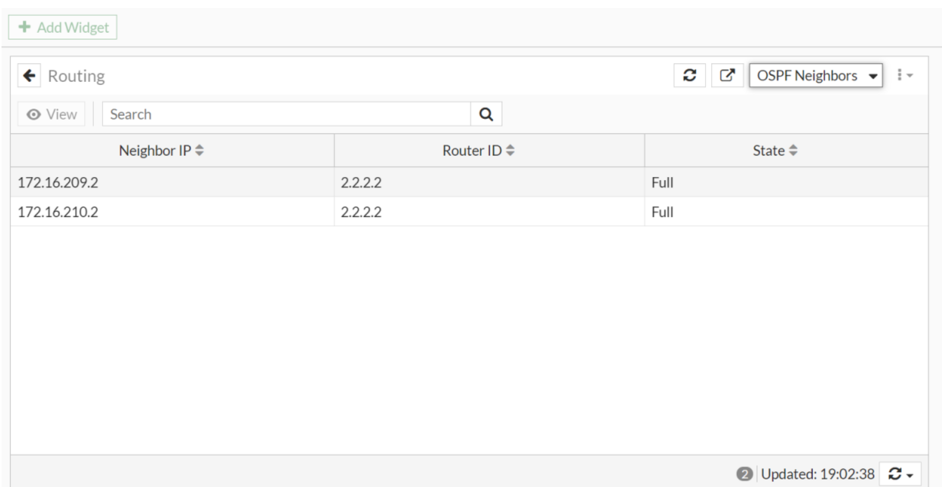
   BGP Paths



IPv6 BGP Paths



OSPF Neighbors



4. To filter the *Interfaces* and *Type* columns:
   a. Click the *Static & Dynamic* tab.
   b. Hover over the column heading, and click the *Filter/Configure Column* icon.

   c. Click *Group By This Column*, then click *Apply*.

5. (Optional) Click *Save as Monitor* to save the widget as monitor.

To look up a route in the GUI:

1. Click *Route Lookup*.
2. Enter an IP address in the *Destination* field, then click *Search*. The matching route is highlighted on the *Routing* monitor.

To view the routing table in the CLI:

```
# get route info routing-table all
```

Sample output:

```
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
* - candidate default

Routing table for VRF=0
S* 0.0.0.0/0 [1/0] via 10.0.10.1, To-HQ-A
[1/0] via 10.0.12.1, To-HQ-MPLS
[1/0] via 10.10.11.1, To-HQ-B
[1/0] via 10.100.67.1, port1
[1/0] via 10.100.67.9, port2
C 10.0.10.0/24 is directly connected, To-HQ-A
C 10.0.10.2/32 is directly connected, To-HQ-A
C 10.0.11.0/24 is directly connected, To-HQ-B
C 10.0.11.2/32 is directly connected, To-HQ-B
C 10.0.12.0/24 is directly connected, To-HQ-MPLS
C 10.0.12.2/32 is directly connected, To-HQ-MPLS
C 10.1.0.0/24 is directly connected, port3
C 10.1.0.2/32 is directly connected, port3
C 10.1.0.3/32 is directly connected, port3
C 10.1.100.0/24 is directly connected, vsw.port6
```

To look up a firewall route in the CLI:

```
# diagnose firewall proute list
```

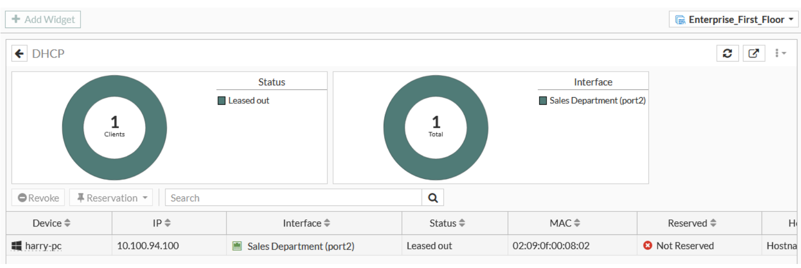
Sample output:

```
list route policy info(vf=root):


id=0x7f450002 vwl_service=2(BusinessCritialCloudApp) vwl_mbr_seq=4 5 3 dscp_tag=0xff 0xff
     flags=0x0 tos=0x00 tos_mask=0x00 protocol=0 sport=0:65535 iif=0 dport=1-65535 oif=3
     (port1) oif=4(port2) oif=18(To-HQ-MPLS)
source(1): 0.0.0.0-255.255.255.255
destination wildcard(1): 0.0.0.0/0.0.0.0
internet service(4): Microsoft.Office.365(4294837472,0,0,0, 33182) Microsoft.Office.Online
     (4294837475,0,0,0, 16177) Salesforce(4294837976,0,0,0, 16920) GoToMeeting
     (4294836966,0,0,0, 16354)
hit_count=0 last_used=2020-03-30 10:50:18
```

```
id=0x7f450003 vwl_service=3(NonBusinessCriticalCloudApp) vwl_mbr_seq=4 5 dscp_tag=0xff 0xff
      flags=0x0 tos=0x00 tos_mask=0x00 protocol=0 sport=0:65535 iif=0 dport=1-65535 oif=3
      (port1) oif=4(port2)
source(1): 0.0.0.0-255.255.255.255
destination wildcard(1): 0.0.0.0/0.0.0.0
internet service(2): Facebook(4294836806,0,0,0, 15832) Twitter(4294838278,0,0,0, 16001)
hit_count=0 last_used=2020-03-30 10:50:18

id=0x7f450004 vwl_service=4(Ping-Policy) vwl_mbr_seq=1 2 dscp_tag=0xff 0xff flags=0x0
      tos=0x00 tos_mask=0x00 protocol=1 sport=0:65535 iif=0 dport=1-65535 oif=16(To-HQ-A)
      oif=17(To-HQ-B)

To view neighbors and paths
```

DHCP monitor

The DHCP monitor shows all the addresses leased out by FortiGate's DHCP servers. You can use the monitor to revoke an address for a device, or create, edit, and delete address reservations.



To view the DHCP monitor:

1. Go to *Dashboard* > *Network*.
2. Hover over the *DHCP* widget, and click *Expand to Full Screen*.

To filter or configure a column in the table, hover over the column heading and click *Filter/Configure Column*.

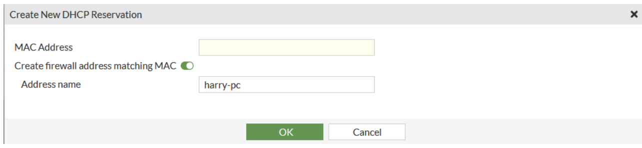
To revoke a lease:

1. Select a device in the table.
2. In the toolbar, click *Revoke*, or right-click the device, and click *Revoke Lease(s)*. The *Confirm* page is displayed.
3. Click *OK*.

A confirmation window opens only if there is an associated address reservation. If there is no address, the lease will be removed immediately upon clicking *Revoke*.

To create a DHCP reservation:

1. Select a server in the table.
2. In the toolbar, click *Reservation*, or right-click the device and click *Create DHCP Reservation*. The *Create New DHCP Reservation* page is displayed.
3. Configure the DHCP reservation settings.



4. Click *OK*.

To view top sources by bytes:

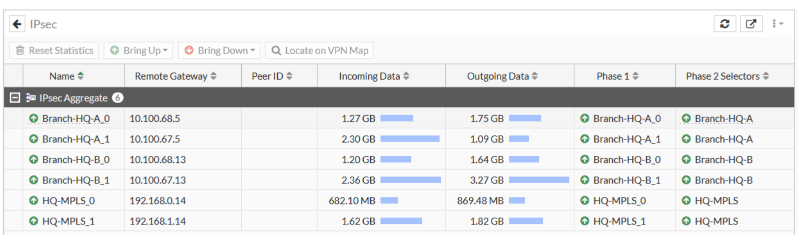
1. Right-click a device in the table and click *Show in FortiView*. The *FortiView Sources by Bytes* widget is displayed.

To view the DHCP lease list in the CLI:

```
# execute dhcp lease-list
```

IPsec monitor

The IPsec monitor displays all connected Site to Site VPN, Dial-up VPNs, and ADVPN shortcut tunnel information. You can use the monitor to bring a phase 2 tunnel up or down or disconnect dial-up users. A notification appears in the monitor when users have not enabled two-factor authentication.



To view the IPsec monitor in the GUI:

1. Go to *Dashboard* > *Network*.
2. Hover over the *IPsec* widget, and click *Expand to Full Screen*. A warning appears when an unauthenticated user is detected.

> To filter or configure a column in the table, hover over the column heading and click *Filter/Configure Column*.

3. Hover over a record in the table. A tooltip displays the *Phase 1* and *Phase 2* interfaces. A warning appears next to a user who has not enabled two-factor authentication.

To reset statistics:

1. Select a tunnel in the table.
2. In the toolbar, click *Reset Statistics* or right-click the tunnel, and click *Reset Statistics*. The *Confirm* dialog is displayed.
3. Click *OK*.

To bring a tunnel up:

1. Select a tunnel in the table.
2. Click *Bring Up*, or right-click the tunnel, and click *Bring Up*. The *Confirm* dialog is displayed.
3. Click *OK*.

To bring a tunnel down:

1. Select a tunnel in the table.
2. Click *Bring Down*, or right-click the tunnel, and click *Bring Down*. The *Confirm* dialog is displayed.
3. Click *OK*.

To locate a tunnel on the VPN Map:

1. Select a tunnel in the table.
2. Click *Locate on VPN Map*, or right-click the tunnel, and click *Locate on VPN Map*. The *VPN Location Map* is displayed.

To view the IPsec monitor in the CLI:

```
# diagnose vpn tunnel list
```

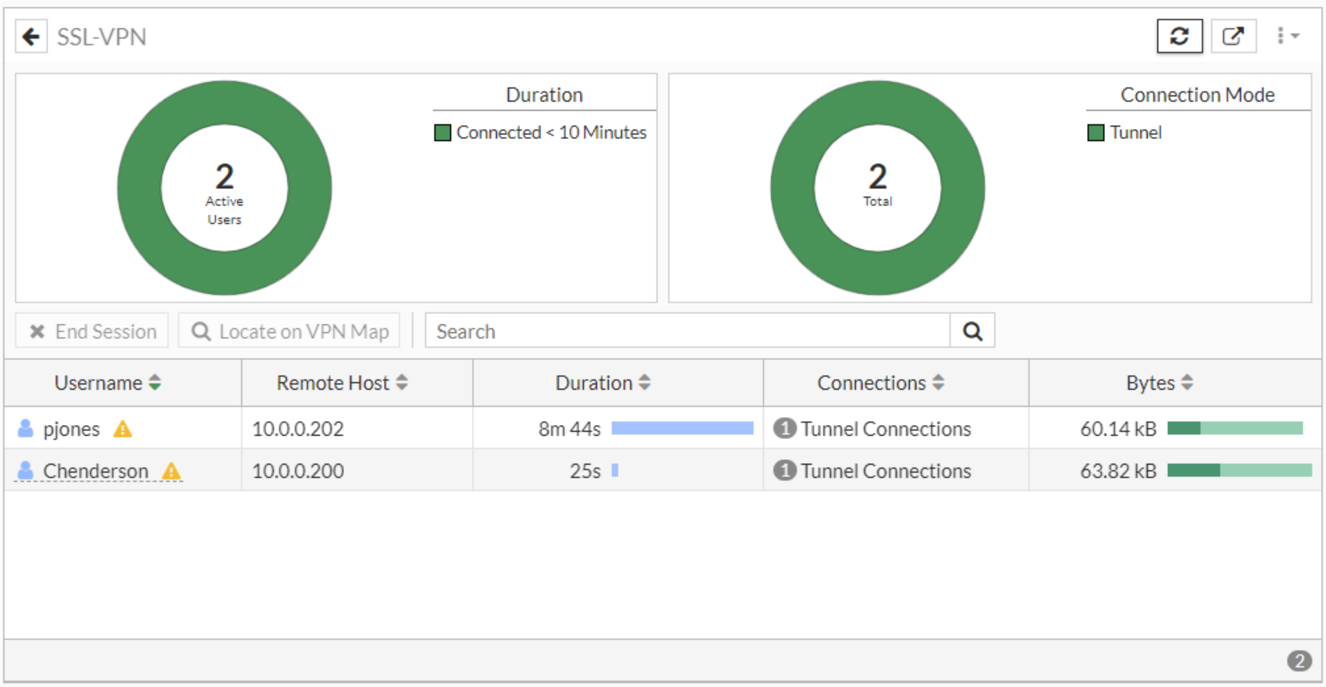
Sample output:

```
list all ipsec tunnel in vd 0
------------------------------------------------------
name=fct-dialup ver=1 serial=4 10.100.67.5:0->0.0.0.0:0 tun_id=0.0.0.0 dst_mtu=0
bound_if=3 lgwy=static/1 tun=intf/0 mode=dialup/2 encap=none/512 options[0200]=frag-rfc
     accept_traffic=1 overlay_id=0

proxyid_num=0 child_num=0 refcnt=12 ilast=5545 olast=5545 ad=/0
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-demand on=0 idle=20000ms retry=3 count=0 seqno=0
natt: mode=none draft=0 interval=0 remote_port=0
run_tally=0
------------------------------------------------------
name=To-HQ-MPLS ver=2 serial=3 192.168.0.14:0->192.168.0.1:0 tun_id=19.168.0.1 dst_mtu=1500
bound_if=7 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/528 options[0210]=create_dev
     frag-rfc accept_traffic=1 overlay_id=0

proxyid_num=1 child_num=0 refcnt=22 ilast=0 olast=0 ad=/0
stat: rxp=66693 txp=29183 rxb=33487128 txb=1908427
dpd: mode=on-demand on=1 idle=20000ms retry=3 count=0 seqno=0
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=To-HQ-MPLS proto=0 sa=1 ref=6 serial=1 adr
src: 0:0.0.0.0/0.0.0.0:0
dst: 0:0.0.0.0/0.0.0.0:0
SA: ref=3 options=32203 type=00 soft=0 mtu=1438 expire=266/0B replaywin=2048
```

```
seqno=2c5e esn=0 replaywin_lastseq=00002ea3 itn=0 qat=0 hash_search_len=1
life: type=01 bytes=0/0 timeout=1773/1800
dec: spi=700c9198 esp=aes key=16 ebd04605de6148c8a92ced48b30930fa
ah=sha1 key=20 5f0201f67d7c714a046025a1df41d40376437f6a
enc: spi=5aaccc20 esp=aes key=16 13d5d4b46e5e9c42eef509f2d9879188
ah=sha1 key=20 2dde67ef7a2a78b622d9a7ec6d75ad3c55d241e1
dec:pkts/bytes=11938/5226964, enc:pkts/bytes=11357/1312184
```

SSL-VPN monitor

The SSL-VPN monitor displays remote user logins and active connections. You can use the monitor to disconnect a specific connection. The monitor will notify you when VPN users have not enabled two-factor authentication.



To view the SSL-VPN monitor in the GUI:

1. Go *Dashboard* > *Network*.
2. Hover over the *SSL-VPN* widget, and click *Expand to Full Screen*.The *Duration* and *Connection Summary* charts are displayed at the top of the monitor.

To filter or configure a column in the table, hover over the column heading and click *Filter/Configure Column*.

To disconnect a user:

1. Select a user in the table.
2. In the table, right-click the user, and click *End Session*. The Confirm window opens.
3. Click *OK*.

To monitor SSL-VPN users in the CLI:

```
# get vpn ssl monitor
```

Sample output

```
SSL VPN Login Users:
Index User Group Auth Type Timeout From HTTP in/out HTTPS in/out
0 amitchell TAC 1(1) 296 10.100.64.101 3838502/11077721 0/0
1 mmiles Dev 1(1) 292 10.100.64.101 4302506/11167442 0/0

SSL VPN sessions:
Index User Group Source IP Duration I/O Bytes Tunnel/Dest IP
```

Users & Devices

The *Users & Devices* dashboard shows the current status of users and devices connected to your network. All of the widgets can be expanded to view as monitor. In monitor view, you can create firewall addresses, deauthenticate a user, or remove a device from the network.

The *User & Devices* dashboard contains the following widgets:

Widget	Description
Device Inventory	Shows a summary of the hardware and software that is connected to the network. See Device inventory on page 96.
FortiClient	Shows a summary of the FortiClient endpoints.
Firewall Users	Shows a summary of the users logged into the network.
Quarantine	Shows a summary of quarantined devices.
FortiSwitch NAC VLANs	Shows a summary of VLANs assigned to devices by FortiSwitch NAC policies.

Device inventory

You can enable device detection to allow FortiOS to monitor your networks and gather information about devices operating on those networks, including:

- MAC address
- IP address
- Operating system
- Hostname
- Username
- Endpoint tags
- When FortiOS detected the device and on which interface

You can enable device detection separately on each interface in *Network > Interfaces*.

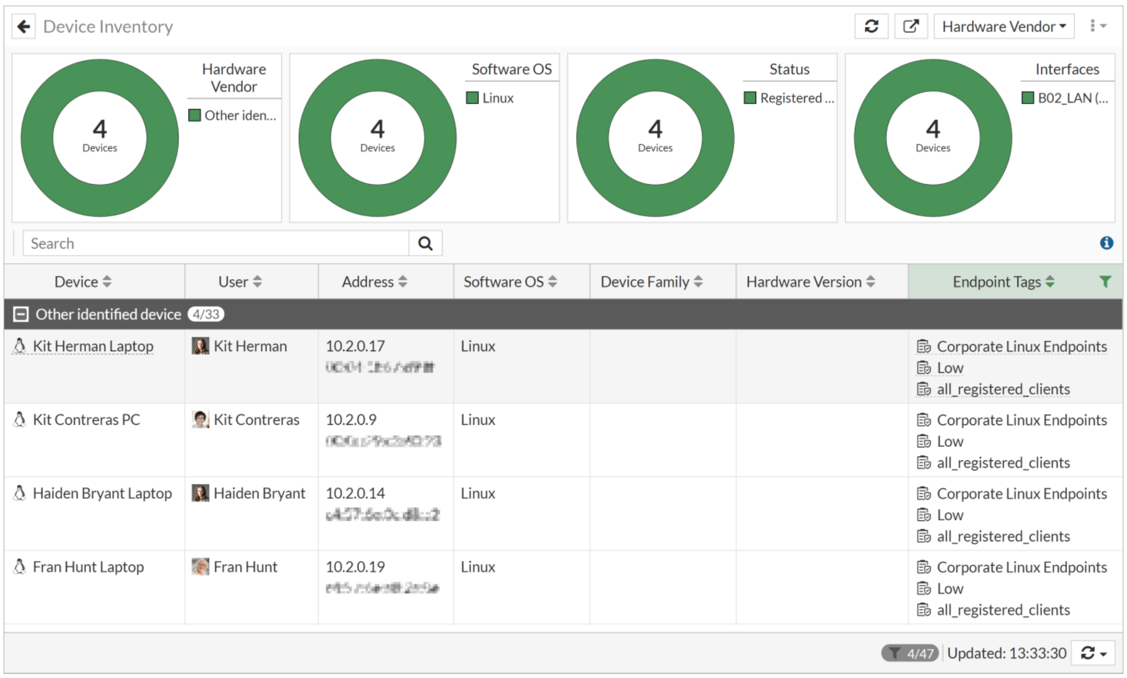
Device detection is intended for devices directly connected to your LAN and DMZ ports. The widget is only available when your *Interface Role* is *LAN*, *DMZ* or *Undefined*. It is not available when the role is WAN.

You can also manually add devices to Device Inventory to ensure that a device with multiple interfaces displays as a single device.

To view the device inventory monitor:

1. Go to *Dashboard* > *Users & Devices*.
   If you are using the Comprehensive dashboard template, go to *Dashboard* > *Device Inventory Monitor*.
2. Hover over the *Device Inventory* widget, and click *Expand to Full Screen*. The *Device Inventory* monitor is displayed.

To filter or configure a column in the table, hover over the column heading, and click *Filter/Configure Column*. See Device inventory and filtering on page 97.



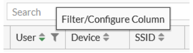
Device inventory and filtering

The *Device Inventory* widget contains a series of summary charts that provide an overview of the hardware, operating system, status, and interfaces. You can use these clickable charts to simplify filtering among your devices.

To view the device inventory and apply a filter:

1. Go to *Dashboard* > *Users & Devices*.
   If you are using the Comprehensive dashboard template, go to *Dashboard* > *Device Inventory Monitor*. See .
2. Hover over the *Device Inventory* widget, and click *Expand to Full Screen*. The *Device Inventory* monitor is displayed.
3. To filter the order of the charts by operating system, click the dropdown in the top menu bar and select *Software OS*.
4. To filter a chart, click an item in the legend or chart area. The table displays the filter results.

5. To combine filters, hover over a column heading and click *Filter/Configure Column*.

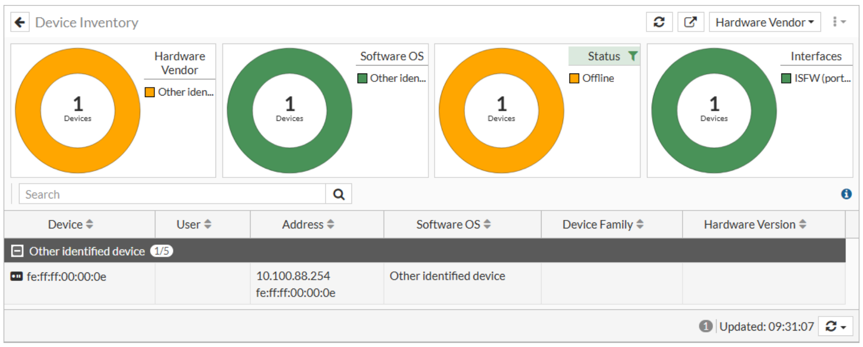


6. Click the filter icon in the top-right corner of the chart to remove the filter.

Filter examples

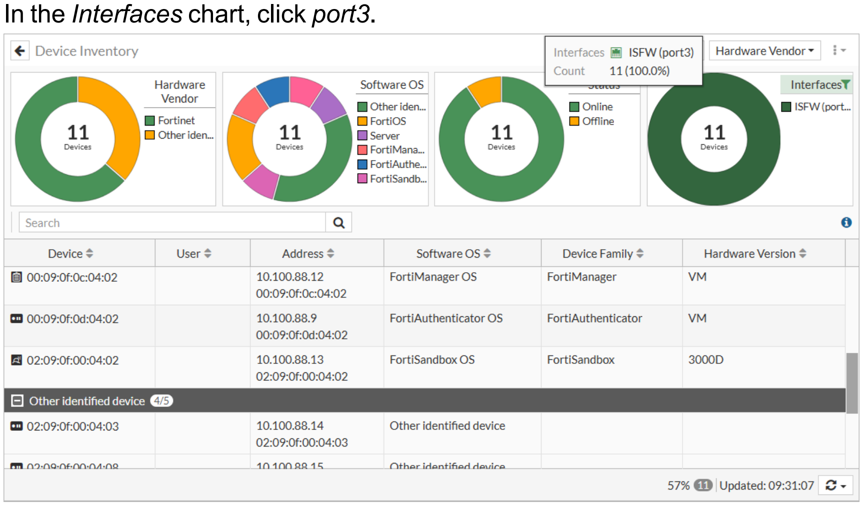
To filter all offline devices:

1. In the *Status* chart, click *Offline* in the legend or on the chart itself.



To filter all devices discovered on port3:

1. In the *Interfaces* chart, click *port3*.



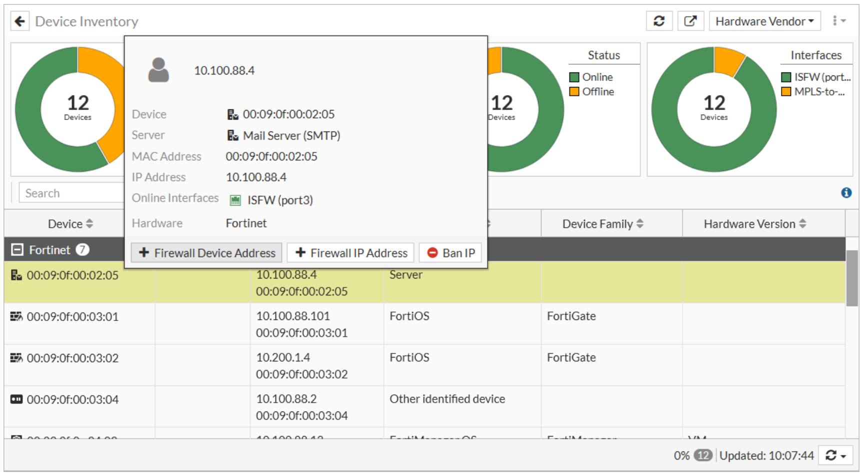
Adding MAC-based addresses to devices

Assets detected by device detection appear in the *Device Inventory* widget. You can manage policies around devices by adding a new device object (MAC-based address) to a device. Once you add the MAC-based address, the device can be used in address groups or directly in policies.

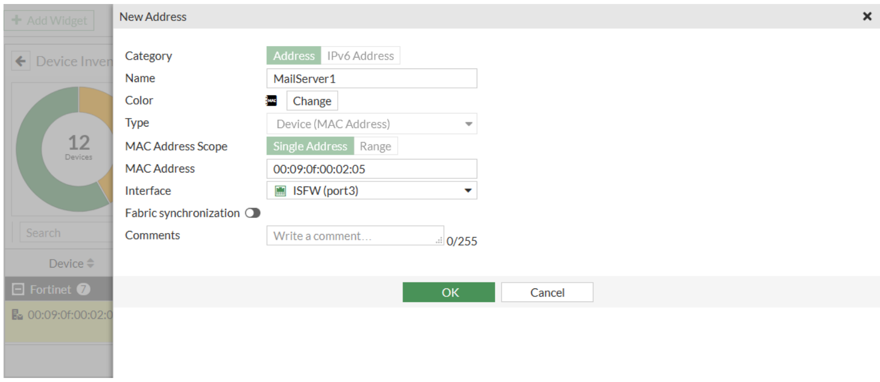
To add a MAC-based address to a device:

1. Go to *Dashboard > Users & Devices*.
   If you are using the Comprehensive dashboard template, go to *Dashboard > Device Inventory Monitor*. See .

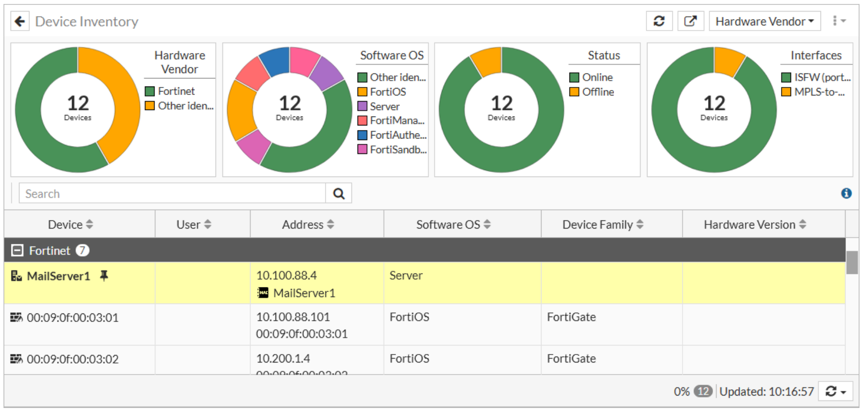
2. Hover over the *Device Inventory* widget, and click *Expand to Full Screen*. The *Device Inventory* monitor is displayed.
3. Click a device, then click *Firewall Device Address*. The *New Address* dialog is displayed.



4. In the *Name* field, give the device a descriptive name so that it is easy to in the *Device* column.
5. Configure the *MAC Address*.

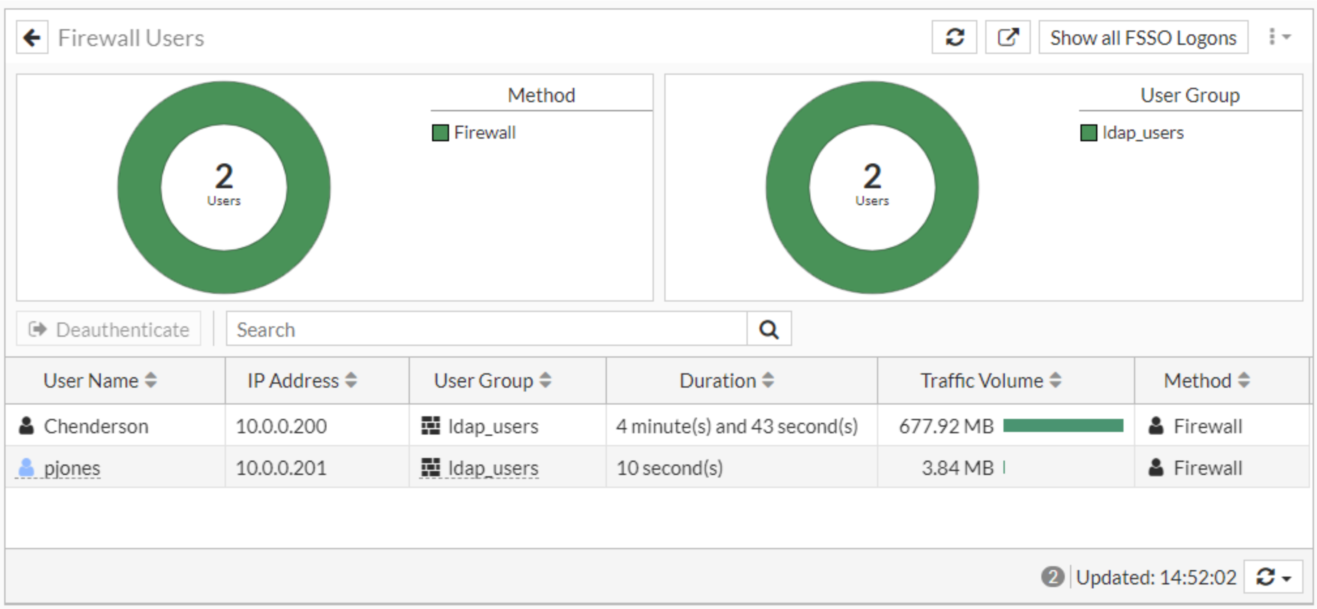


6. Click *OK*, then refresh the page. The MAC address icon appears in the *Address* column next to the device name.



Firewall Users monitor

The Firewall Users monitor displays all firewall users currently logged in. You can use the monitor to diagnose user-related logons or to highlight and deauthenticate a user.



To view the firewall monitor:

1. Go to *Dashboard > Users & Devices*.
   If you are using the Comprehensive dashboard template, go to *Dashboard > Firewall User Monitor*. See .
2. Hover over the *Firewall Users* widget, and click *Expand to Full Screen*.
3. To show FSSO logons, click *Show all FSSO Logons* at the top right of the page.

To filter or configure a column in the table, hover over the column heading and click *Filter/Configure Column*.

To deauthenticate a user:

1. Go to *Dashboard > Users & Devices*.
2. Hover over the *Firewall Users* widget, and click *Expand to Full Screen*.
3. (Optional) Use the *Search* field to search for a specific user.
4. In the toolbar, click *Deauthenticate*, or right-click the user, and click *Deauthenticate*. The *Confirm* dialog is displayed.
5. Click *OK*.

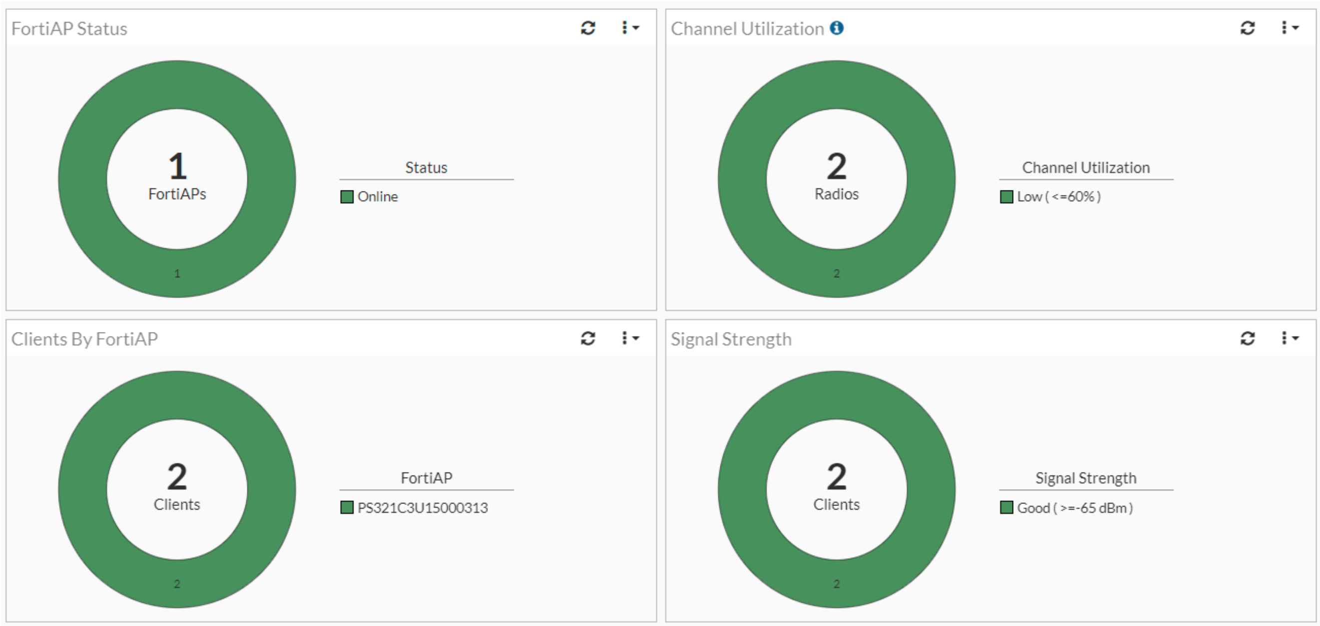
To view firewall users in the CLI:

```
# diagnose firewall auth list
```

WiFi dashboard

The *WiFi* dashboard provides an overview of your WiFi network's performance, including FortiAP status, channel utilization, WiFi clients and associated information, login failures, and signal strength.

To access the WiFi dashboard, go to *Dashboard* > *WiFi*.



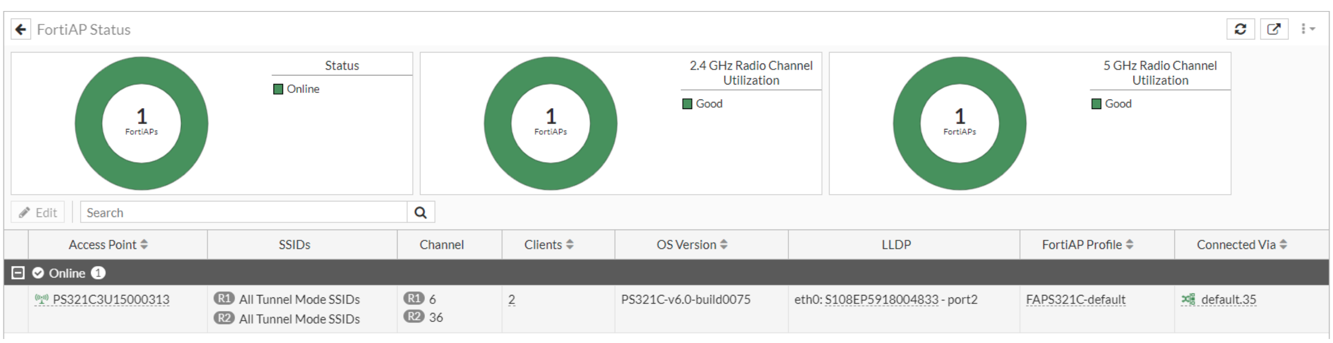
The WiFi dashboard can be customized per your requirements. To learn more about using and modifying dashboards and widgets, see Dashboards and Monitors on page 76.

This section describes the following monitors available for the WiFi Dashboard:

- FortiAP Status monitor on page 101
- Clients by FortiAP monitor on page 103

FortiAP Status monitor

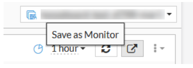
The *FortiAP Status* monitor displays the status and the channel utilization of the radios of FortiAP devices connected to a FortiGate. It also provides access to tools to diagnose and analyze connected APs.



To view the *FortiAP Status* monitor:

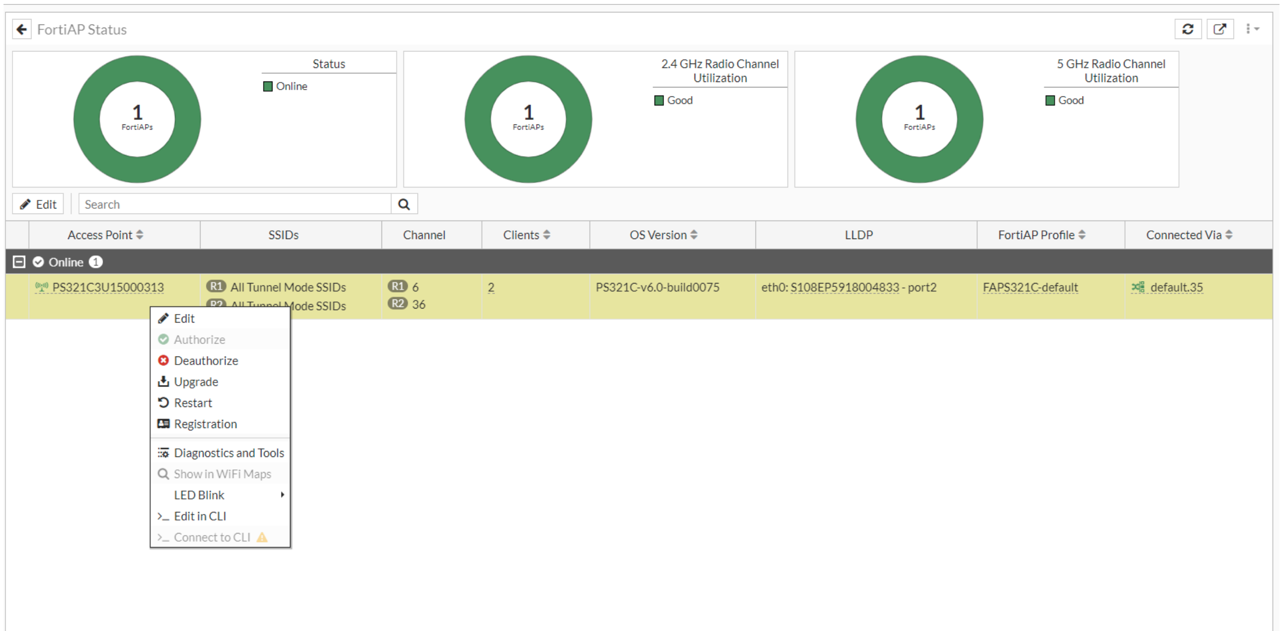
1. Go to *Dashboard* > *WiFi*.
2. Hover over the *FortiAP Status* widget, and click *Expand to Full Screen*. The *FortiAP Status* monitor opens.

3. (Optional) Click *Save as Monitor* to save the widget as monitor.

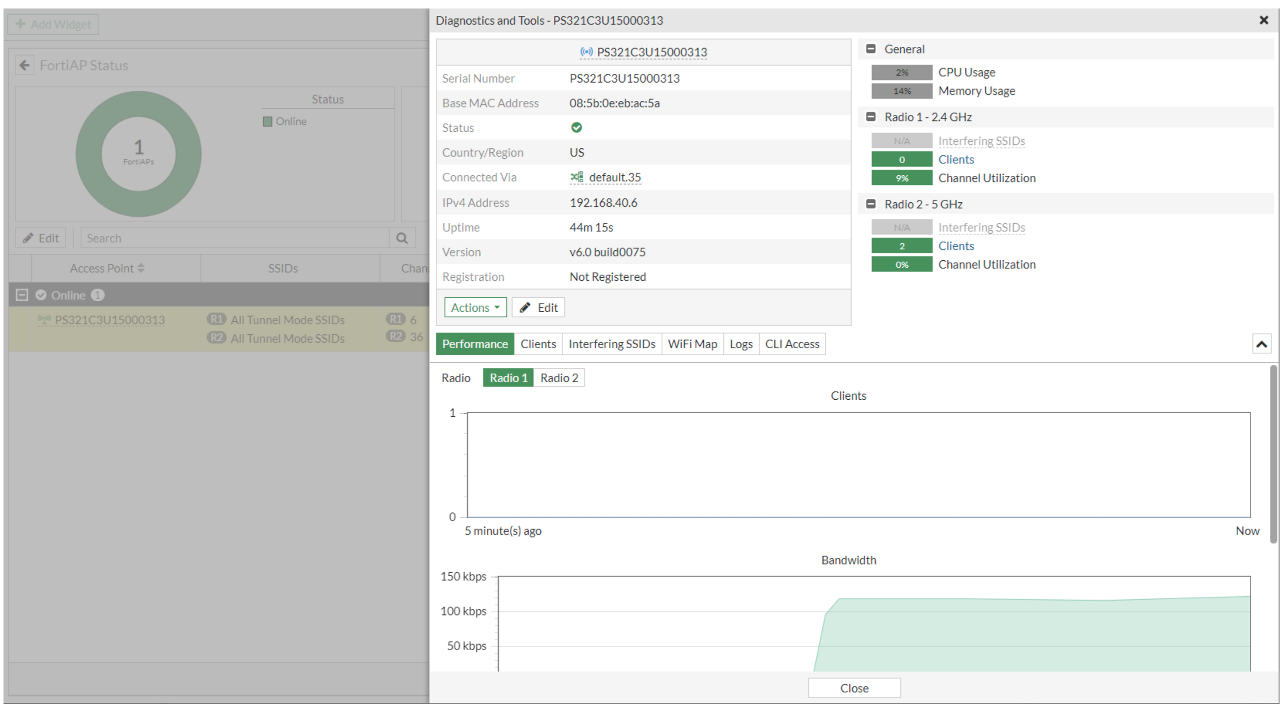


To view the Diagnostics and Tools menu:

1. Right-click an *Access Point* in the table, and click *Diagnostics and Tools*. The *Diagnostics and Tools* dialog opens.



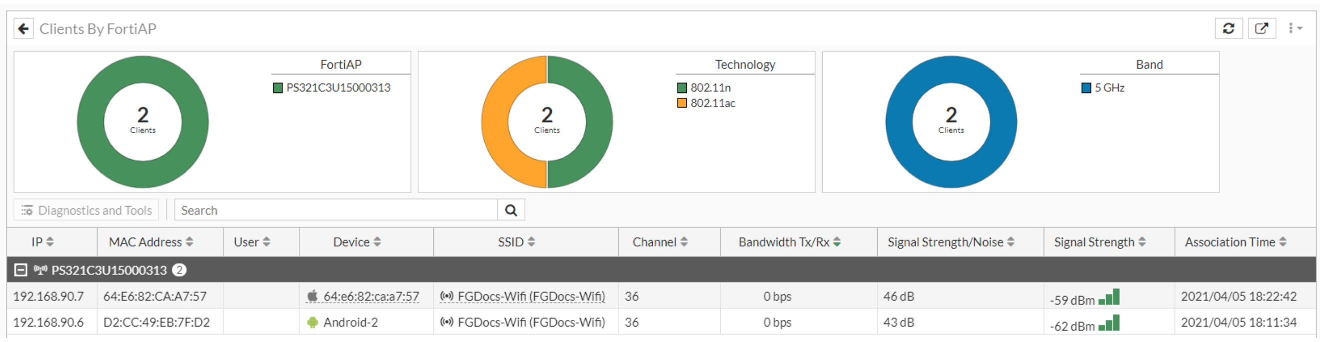
2. To monitor and analyze the FortiAP device, click on the tabs in the *Diagnostics and Tools* dialog, such as *Clients*, *Spectrum Analysis*, *VLAN Probe*, and so on.



The *Diagnostics and Tools* dialog is similar to the device dialog from *WiFi & Switch Controller > Managed FortiAPs*. To learn more about the various tabs and their functions, see Spectrum analysis of FortiAP E models, VLAN probe report, and Standardize wireless health metrics.

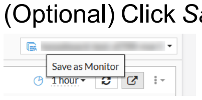
Clients by FortiAP monitor

The *Clients by FortiAP* monitor allows you to view detailed information about the health of individual WiFi connections in the network. It also provides access to tools to diagnose and analyze connected wireless devices.



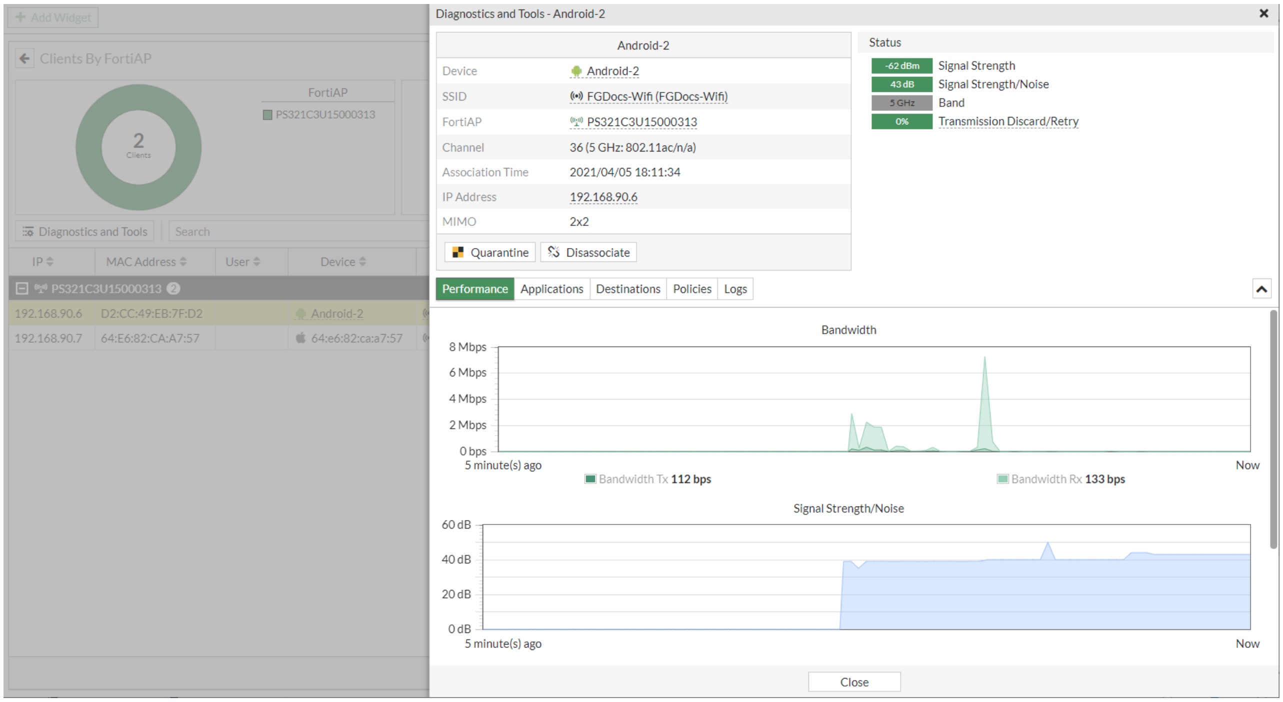
To view the *Clients by FortiAP* monitor:

1. Go to *Dashboard > WiFi*.
2. Hover over the *Clients by FortiAP* widget, and click *Expand to Full Screen*. The *Clients by FortiAP* monitor opens.
3. (Optional) Click *Save as Monitor* to save the widget as monitor.



To view the summary page for a wireless client:

1. Right-click a client in the table and select *Diagnostics and Tools*. The *Diagnostics and Tools - <device>* page is displayed.



2. (Optional) Click *Quarantine* to quarantine the client,
3. (Optional) Click *Disassociate* to disassociate the client.

Health status

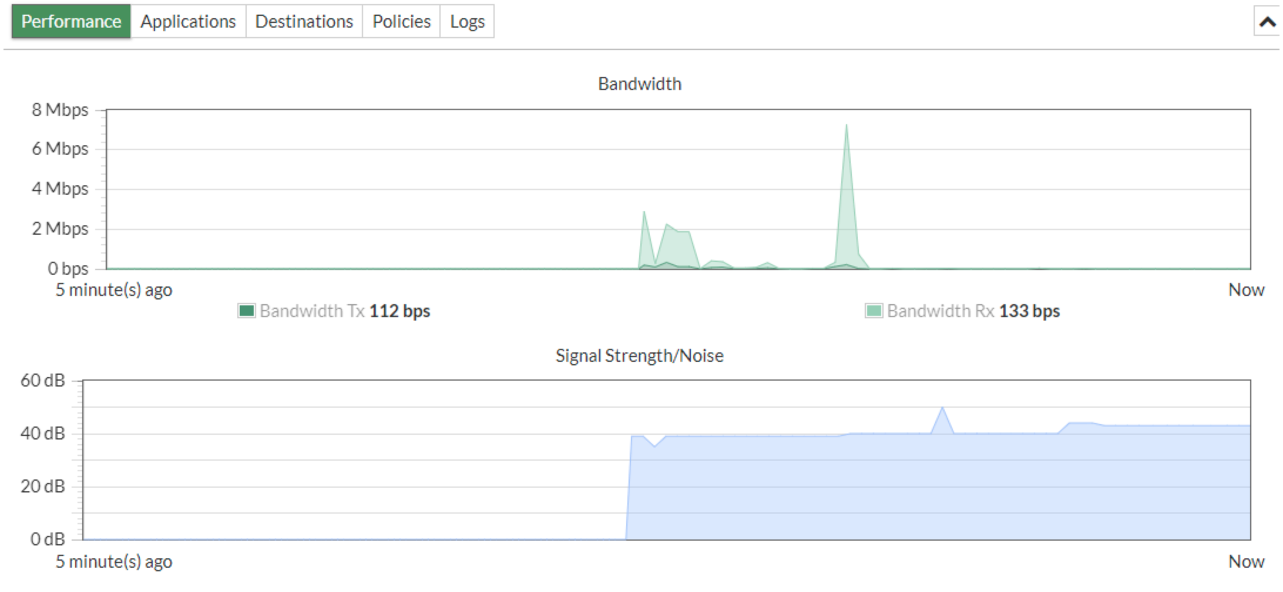
The *Status* section displays the overall health for the wireless connection. The overall health of the connection is:

- Good if the value range for all three conditions are *Good*
- Fair or poor if one of the three conditions is *Fair* or *Poor* respectively.

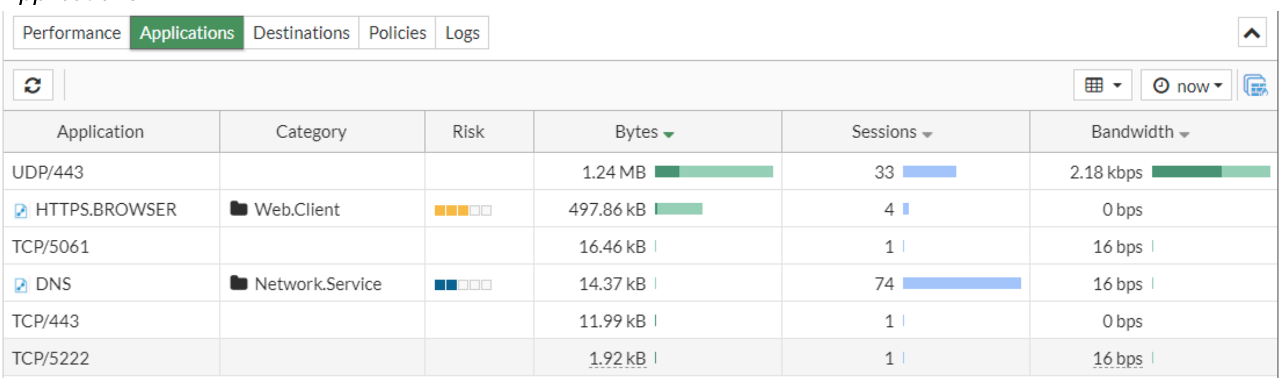
Condition	Value Range
Signal Strength	• *Good > -56dBm* • *-56dBm > Fair > -75dBm* • *Poor < -75dBm*
Signal Strength/Noise	• *Good > 39dBm* • *20dBm < Fair < 39dBm* • *Poor < 20dBm*
Band	• *Good = 5G band* • *Fair = 2.4G band*

The summary page also has the following FortiView tabs:

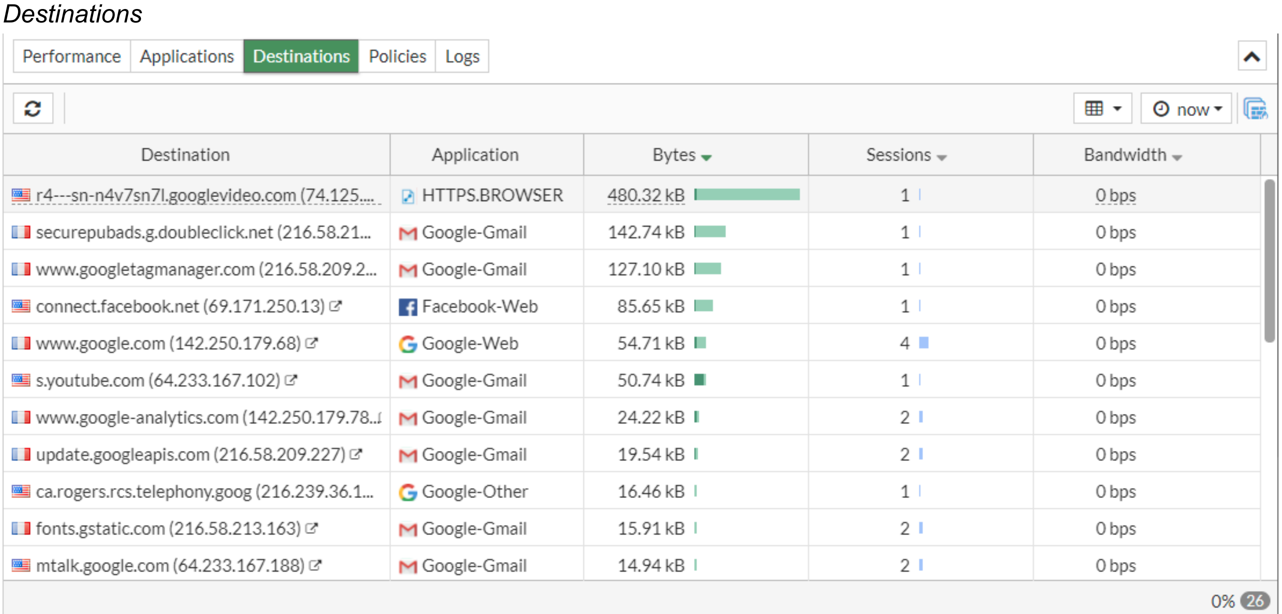
- *Performance*



- *Applications*

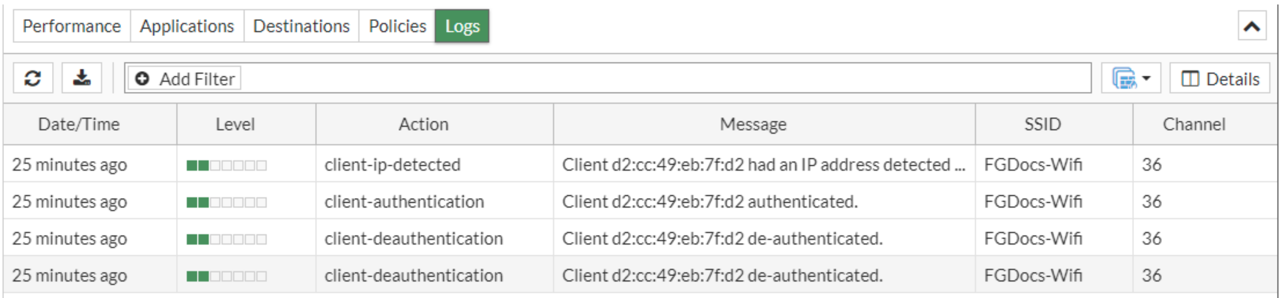


- *Destinations*



- *Policies*

- *Logs*



Monitors

FortiGate supports both FortiView and Non-FortiView monitors. FortiView monitors are driven by traffic information captured from logs and real-time data. Non-FortiView monitors capture information from various real-time state tables on the FortiGate.

Non-FortiView monitors

Non-FortiView monitors capture information on various state tables, such as the routes in the routing table, devices in the device inventory, DHCP leases in the DHCP lease table, connected VPNs, clients logged into the wireless network, and much more. These monitors are useful when troubleshooting the current state of the FortiGate, and to identify whether certain objects are in the state table or not. For more information, see Dashboards on page 83.

FortiView monitors

FortiView is the FortiOS log view tool and comprehensive monitoring system for your network. FortiView integrates real-time and historical data into a single view on your FortiGate. It can log and monitor network threats, keep track of administration activities, and more.

Use FortiView monitors to investigate traffic activity such as user uploads and downloads, or videos watched on YouTube. You can view the traffic on the whole network by user group or by individual. FortiView displays the information in both text and visual format, giving you an overall picture of your network traffic activity so that you can quickly decide on actionable items.

FortiView is integrated with many UTM functions. For example, you can quarantine an IP address directly in FortiView or create custom devices and addresses from a FortiView entry.

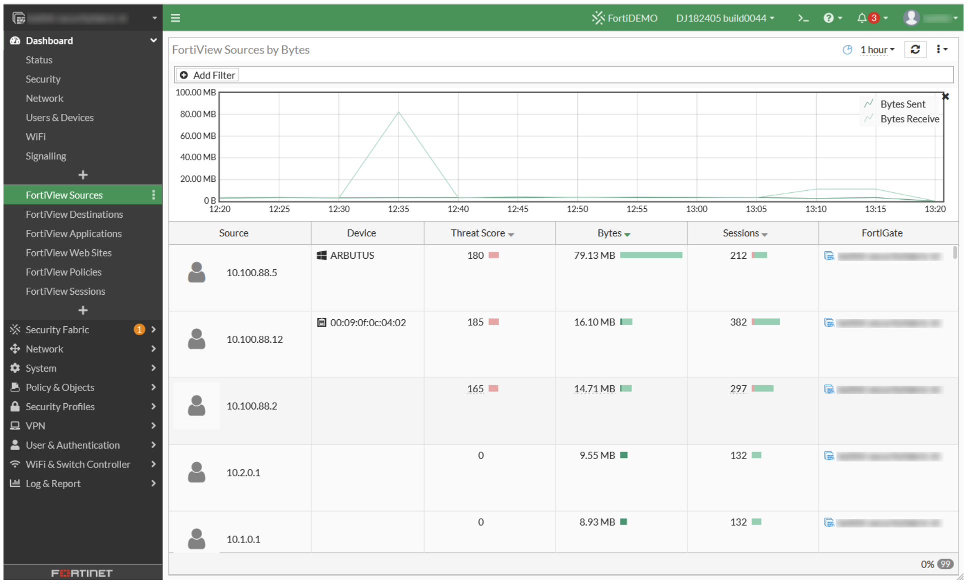
The logging range and depth will depend on the FortiGate model.

The *Optimal* template contains a set of popular default dashboards and FortiView monitors. The *Comprehensive* template contains a set of default dashboards as well as all of the FortiView monitors. See Dashboards on page 83.

Template	Monitors
Optimal	• FortiView Sources • FortiView Destinations • FortiView Applications • FortiView Web Sites • FortiView Policies • FortiView Sessions
Comprehensive	• FortiView Sources • FortiView Destinations • FortiView Applications • FortiView Web Sites • FortiView Threats • FortiView Compromised Hosts • FortiView Policies • FortiView Sessions • Device Inventory Monitor • Routing Monitor • DHCP Monitor • SD-WAN Monitor • FortiGuard Quota Monitor • IPsec Monitor • SSL-VPN Monitor • Firewall User Monitor • Quarantine Monitor • FortiClient Monitor • FortiAP Clients Monitor • Rogue APs Monitor

FortiView monitors and widgets

FortiView monitors are available in the tree menu under *Dashboards*. The menu contains several default monitors for the top categories. Additional FortiView monitors are available as widgets that can be added to the dashboards. You can also add FortiView monitors directly to the tree menu with the Add (**+**) button.



Core FortiView monitors

The following default monitors are available in the tree menu:

Dashboard	Usage
FortiView Sources	Displays Top Sources by traffic volume and drilldown by Source.
FortiView Destinations	Displays Top Destinations by traffic volume and drilldown by Destination.
FortiView Applications	Displays Top Applications by traffic volume and drilldown by Application.
FortiView Web Sites	Displays Top Websites by session count and drilldown by Domain.
FortiView Policies	Displays Top Policies by traffic volume and drilldown by Policy number
FortiView Sessions	Displays Top Sessions by traffic source and can be used to end sessions.

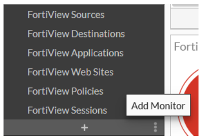
Usage is based on default settings. The pages may be customized further and sorted by other fields.

You can quarantine a host and ban an IP from all of the core FortiView monitors.

Adding FortiView monitors

Non-core FortiView monitors are available in the *Add monitor* pane. You can add a FortiView widget to a dashboard or the tree menu as a monitor.

To add a monitor to the tree menu:

1. In the tree menu, under the monitors section, click *Add Monitor* (+). The *Add Monitor* window opens.


2. Click *Add* next to a monitor. You can use the *Search* field to search for a specific monitor.
3. In the *FortiGate* area, select *All FortiGates* or *Specify* to select a FortiGate device in the security fabric.
4. (Optional) In the *Data Source* area, select *Specify* and select a source device.
5. From the *Time Period* dropdown, select the time period. This option is not available in all monitors.
6. In the *Visualization* area, select *Table View* or *Bubble Chart*.
7. From the *Sort By* dropdown, select the sorting method.
8. Click *Add Monitor*. The monitor is added to the tree menu.

Monitors by category

Usage is based on the default settings. The monitors may be customized further and sorted by other fields.

LANDMARK

Widget	Sort by	Usage
Applications	Bytes/Sessions/Bandwidth/Packets	Displays top applications and drilldown by application.
Application Bandwidth	Bytes/Bandwidth	Displays bandwidth for top applications and drilldown by application.
Cloud Applications	Bytes/Sessions/Files(Up/Down)	Displays top cloud applications and drilldown by application.
Cloud Users	Bytes/Sessions/Files(Up/Down)	Displays top cloud users and drilldown by cloud user.
Compromised Hosts	Verdict	Displays compromised hosts and drilldown by source.
Countries/Regions	Bytes/Sessions/Bandwidth/Packets	Displays top countries/regions and drilldown by countries/regions.
Destination Firewall Objects	Bytes/Sessions/Bandwidth/Packets	Displays top destination firewall objects and drilldown by destination objects.
Destination Owners	Bytes/Sessions/Bandwidth/Packets	Displays top destination owners and drilldown by destination.
Destinations	Bytes/Sessions/Bandwidth/Packets	Displays top destinations and drilldown by destination.

Widget	Sort by	Usage
Search Phrases	Count	Displays top search phrases and drilldown by search phrase.
Source Firewall Objects	Bytes/Sessions/Bandwidth/Packets	Displays top search phrases and drilldown by source object.
Sources	Bytes/Sessions/Bandwidth/Packets	Displays top sources and drilldown by source.
Threats	Threat level/Threat Score/Sessions	Displays top threats and drilldown by threat.
Traffic Shaping	Dropped Bytes/Bytes/Sessions/Bandwidth/Packets	Displays top traffic shaping and drilldown by shaper.
Web Categories	Bytes/Sessions/Bandwidth/Packets	Displays top web categories and drilldown by category.
Web Sites	Bytes/Sessions/Bandwidth/Packets	Displays top web sites and drilldown by domain.
WiFi Clients	Bytes/Sessions	Displays top WiFi clients and drilldown by source.

WAN

Widget	Sort by	Usage
Servers	Bytes/Sessions/Bandwidth/Packets	Displays top servers and drilldown by server address.
Sources	Bytes/Sessions/Bandwidth/Packets	Displays top sources and drilldown by device.
Threats	Threat Level/Threat Score/Sessions	Displays top threats and drilldown by threat.

All Segments

Widget	Sort by	Usage
Admin Logins	Configuration Changes/Logins/Failed Logins	Displays top admin logins by username.
Destination Interfaces	Bytes/Sessions/Bandwidth/Packets	Displays top destination interfaces by destination interface.
Endpoint Vulnerabilities	Severity	Displays top endpoint vulnerabilities by vulnerability name.
Failed Authentication	Failed Attempts	Displays top failed authentications by failed authentication source.
FortiSandbox Files	Submitted	Displays top FortiSandbox files by file name.

Widget	Sort by	Usage
Interface Pairs	Bytes/Sessions/Bandwidth/Packets	Displays top interface pairs by source interface.
Policies	Bytes/Sessions/Bandwidth/Packets	Displays top policies by policy.
Source Interfaces	Bytes/Sessions/Bandwidth/Packets	Displays top source interfaces by source interface.
System Events	Level/Events	Displays top system events by event name.
VPN	Connections/Bytes	Displays top VPN connections by user.
Vulnerable Endpoint Devices	Detected Vulnerabilities	Displays top vulnerable endpoint devices by device.

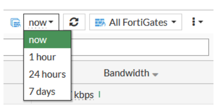
A maximum of 25 interfaces can be monitored at one time on a device.

Using the FortiView interface

Use the FortiView interface to customize the view and visualizations within a monitor to find the information you are looking for. The tools in the top menu bar allow you to change the time display, refresh or customize the data source, and filter the results. You can also right-click a table in the monitor to view drilldown information for an item.

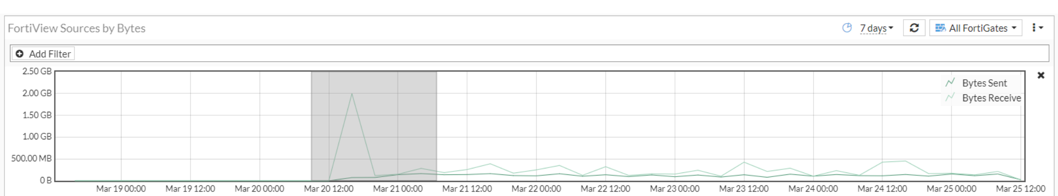
Real-time and historical charts

Use the *Time Display* dropdown to select the time period to display on the current monitor. Time display options vary depending on the monitor and can include real-time information (*now*) and historical information (*1 hour*, *24 hours*, and *7 days*).

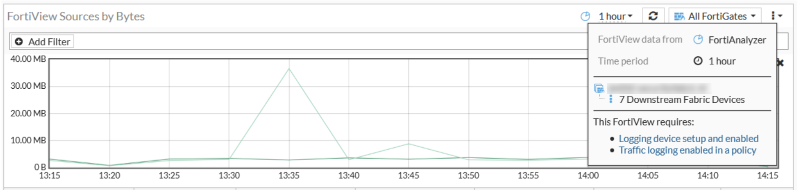


Disk logging or remote logging must be enabled to view historical information.

You can create a custom time range by selecting an area in table with your cursor.

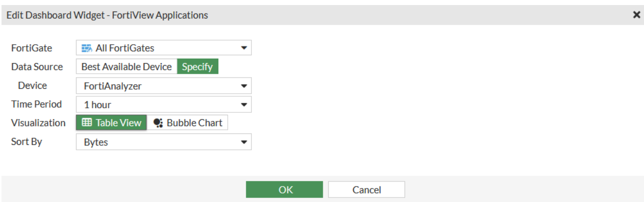


The icon next to the time period identifies the data source (FortiGate Disk, FortiAnalyzer, or FortiGate Cloud). You can hover over the icon to see a description of the device.



Data source

FortiView gathers information from a variety of data sources. If there are no log disk or remote logging configured, the data will be drawn from the FortiGate's session table, and the *Time Period* is set to *Now*.



Other data sources that can be configured are:

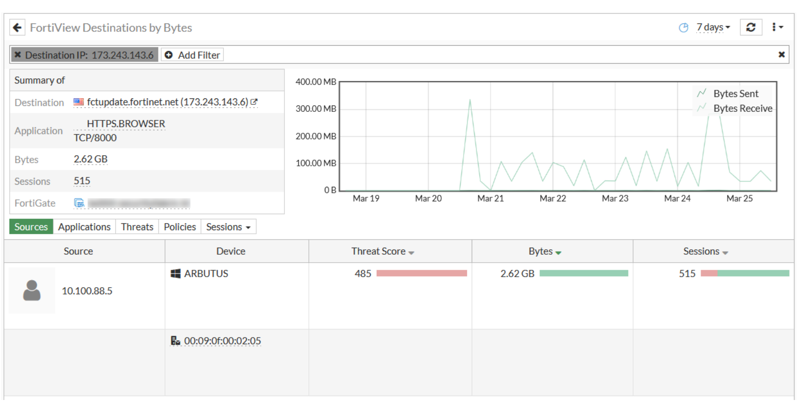
- FortiGates (disk)
- FortiAnalyzer
- FortiGate Cloud

When *Data Source* is set to *Best Available Device*, FortiAnalyzer is selected when available, then FortiGate Cloud, and then FortiGate Disk.

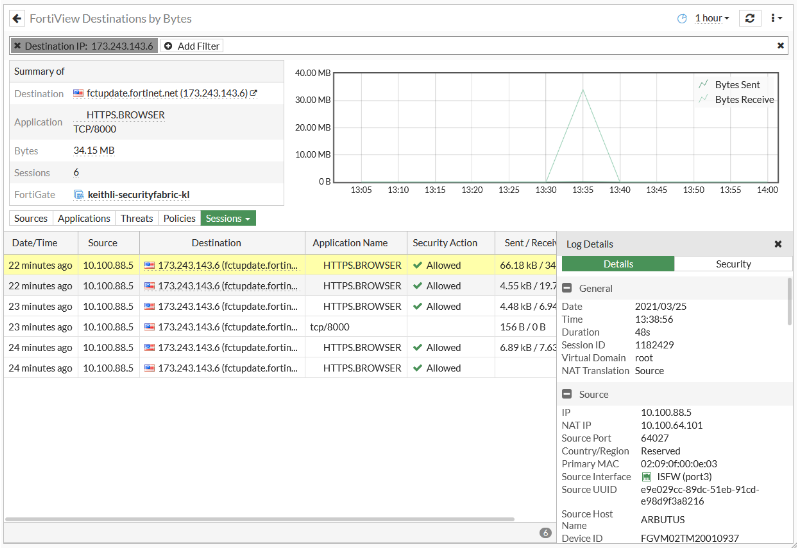
Drilldown information

Double-click or right-click an entry in a FortiView monitor and select *Drill Down to Details* to view additional details about the selected traffic activity. Click the *Back* icon in the toolbar to return to the previous view.

You can group drilldown information into different drilldown views. For example, you can group the drilldown information in the *FortiView Destinations* monitor by *Sources*, *Applications*, *Threats*, *Policies*, and *Sessions*.



Double-click an entry to view the logs in *Sessions* view. Double-click a session to view the logs.



Graph

- The graph shows the bytes sent/received in the time frame. real time does not include a chart.
- Users can customize the time frame by selecting a time period within the graph.

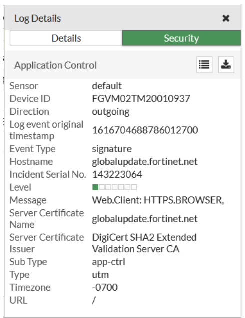
Summary of

- Shows information such as the user/avatar, avatar/source IP, bytes, and sessions total for the time period.
- Can quarantine host (access layer quarantine) if they are behind a FortiSwitch or FortiAP.
- Can ban IP addresses, adds the source IP address into the quarantine list.

Tabs

- Drilling down entries in any of these tabs (except sessions tab) will take you to the underlying traffic log in the sessions tab.
- *Applications* shows a list of the applications attributed to the source IP. This can include scanned applications (using Application Control in a firewall policy or unscanned applications.
  ```
  config log gui-display
     set fortiview-unscanned-apps enable
  end
  ```
- *Destinations* shows destinations grouped by IP address/FQDN.
- *Threats* lists the threats caught by UTM profiles. This can be from antivirus, IPS, Web Filter, Application Control, etc.
- *Web Sites* contains the websites which were detected either with webfilter, or through FQDN in traffic logs.
- *Web Categories* groups entries into their categories as dictated by the Web Filter Database.
- *Policies* groups the entries into which polices they passed through or were blocked by.
- *Sessions* shows the underlying logs (historical) or sessions (real time). Drilldowns from other tabs end up showing the underlying log located in this tab.
- *Search Phrases* shows entries of search phrases on search engines captured by a Web Filter UTM profile, with deep inspection enabled in firewall policy.
- More information can be shown in a tooltip while hovering over these entries.

To view matching logs or download a log, click the *Security* tab in the *Log Details* .



Enabling FortiView from devices

You can enable FortiView from SSD disk, FortiAnalyzer and FortiGate Cloud.

FortiView from disk

FortiView from disk is available on all FortiGates with an SSD disk.

Restrictions

Model	Supported view
Desktop models (100 series) with SSD	Five minutes and one hour
Medium models with SSD	Up to 24 hours

Large models (1500D and above) with SSD	Up to seven days
	To enable seven days view:
	`config log setting`
	`    set fortiview-weekly-data enable`
	`end`

Configuration

A firewall policy needs to be in place with traffic logging enabled. For optimal operation with FortiView, internal interface roles should be clearly defined as LAN. DMZ and internet facing or external interface roles should be defined as WAN.

To configure logging to disk:

```
config log disk setting
   set status enable
end
```

To include sniffer traffic and local-deny traffic when FortiView from Disk:

```
config report setting
   set report-source forward-traffic sniffer-traffic local-deny-traffic
end
```

This feature is only supported through the CLI.

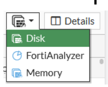
Troubleshooting

Use `execute report flush-cache` and `execute report recreate-db` to clear up any irregularities that may be caused by upgrading or cache issues.

Traffic logs

To view traffic logs from disk:

1. Go to *Log & Report*, and select either the *Forward Traffic*, *Local Traffic*, or *Sniffer Traffic* views.
2. In the top menu bar, click *Log location* and select *Disk*.



FortiView from FortiAnalyzer

Connect FortiGate to a FortiAnalyzer to increase the functionality of FortiView. Adding a FortiAnalyzer is useful when adding monitors such as the *Compromised Hosts*. FortiAnalyzer also allows you to view historical information for up to seven days.

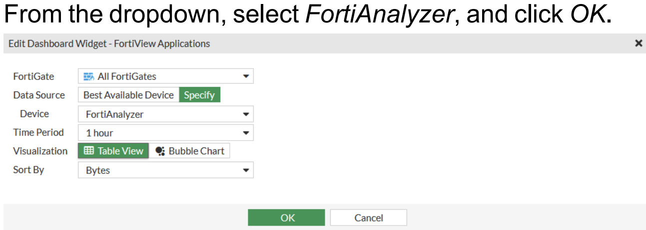
Requirements

- A FortiGate or FortiOS
- A compatible FortiAnalyzer (see Compatibility with FortiOS)

To configure logging to the FortiAnalyzer, see Configuring FortiAnalyzer on page 2215

To enable FortiView from FortiAnalyzer:

1. Go to *Dashboard > FortiView Sources*.
2. Select a time range other than *Now* from the dropdown list to view historical data.
3. In top menu, click the dropdown, and select *Settings*. The *Edit Dashboard Widget* dialog is displayed.
   a. In the *Data Source* area, click *Specify*.
   b. From the dropdown, select *FortiAnalyzer*, and click *OK*.



All the historical information now comes from the FortiAnalyzer.

When *Data Source* is set to *Best Available Device*, FortiAnalyzer is selected when available, then FortiGate Cloud, and then FortiGate Disk.

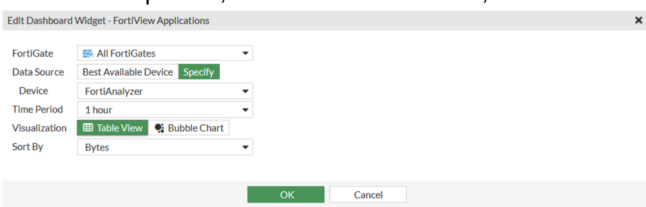
FortiView from FortiGate Cloud

This function requires a FortiGate that is registered and logged into a compatible FortiGate Cloud. When using FortiGate Cloud, the *Time Period* can be set to up to 24 hours.

To configure logging to FortiGate Cloud, see Configuring FortiGate Cloud on page 2217.

To enable FortiView with log source as FortiGate Cloud:

1. Go to *Dashboard* > *FortiView Sources*.
2. In the top menu, click the dropdown, and select *Settings*. The *Edit Dashboard Widget* window opens.
   a. In the *Data Source* area, click *Specify*.
   b. From the dropdown, select *FortiGate Cloud*, then click *OK*.



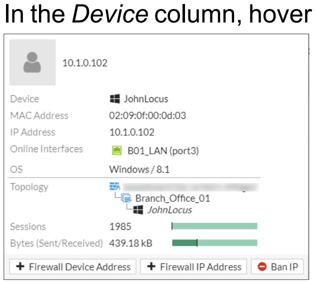
You can select FortiGate Cloud as the data source for all available FortiView pages and widgets.

FortiView sources

The *FortiView Sources* monitor displays top sources sorted by Bytes, Sessions or Threat Score. The information can be displayed in real time or historical views. You can use the monitor to create or edit a firewall device address or IP address definitions, and temporarily or permanently ban IPs.

To add a firewall device address:

1. In the *Device* column, hover over the device MAC address. An information window opens.

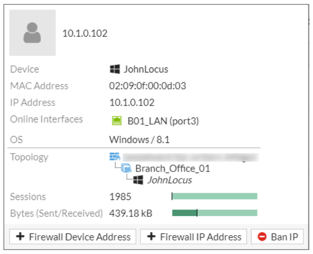


2. Click *Firewall Device Address*. The *New Address* dialog opens.
3. Configure the address settings, and click *Return*.

Use the *Name* field to assign a descriptive name to a device so it is easier to find it in the *Device* column. After you finish configuring the device, refresh the page to see the new name in the monitor.

To add a firewall IP address:

1. In the *Device* column, hover over the device MAC address. An information window opens.

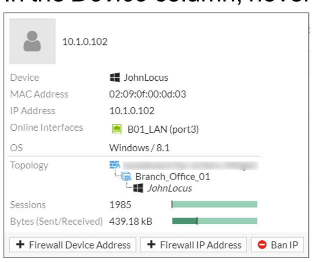


2. Click *Firewall IP Address*. The *New Address* window opens.
3. Configure the address settings, and click *Return*.

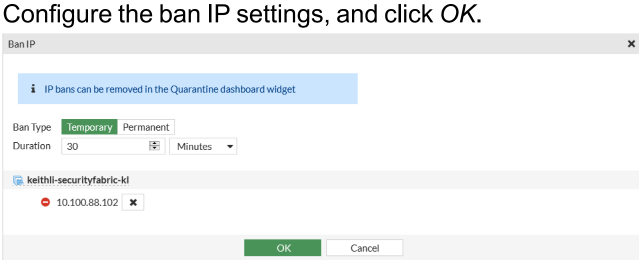
Use the *Name* field to assign a descriptive name to a device so it is easier to find it in the *Device* column. After you finish configuring the device, refresh the page to see the new name in the monitor.

To ban an IP address:

1. In the *Device* column, hover over the device MAC address. An information window opens.



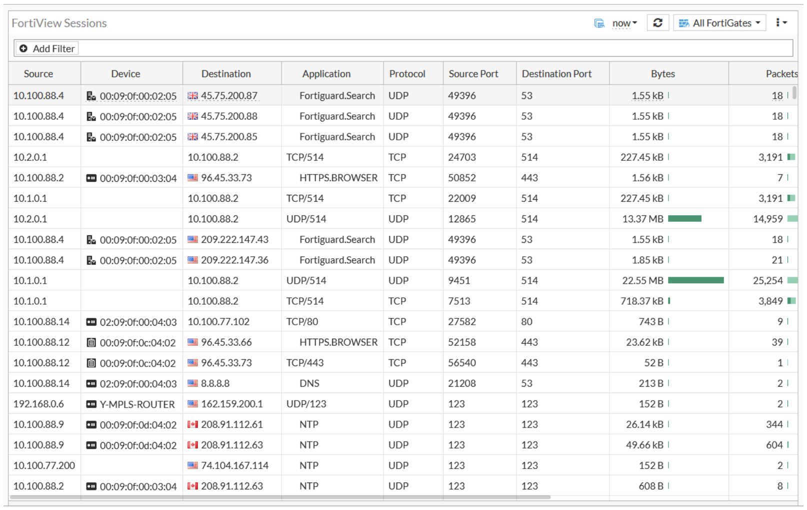
2. Click *Ban IP* . The *Ban IP* dialog is displayed.
3. Configure the ban IP settings, and click *OK*.



FortiView Sessions

The *FortiView Sessions* monitor displays *Top Sessions* by traffic source and can be used to end sessions.

To view the *FortiView Sessions* dashboard, go to *Dashboard* > *FortiView Sessions*.

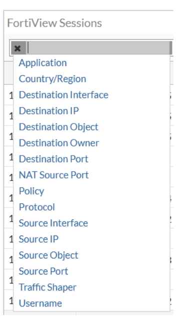


The session table displayed on the *FortiView Sessions* monitor is useful when verifying open connections. For example, if you have a web browser open to browse the Fortinet website, you would expect a session entry from your computer on port 80 to the IP address for the Fortinet website. You can also use a session table to investigate why there are too many sessions for FortiOS to process.

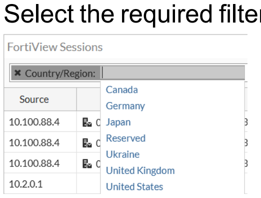
You can filter the sessions displayed in the session table by setting up the available filtering options.

To filter sessions in the session table:

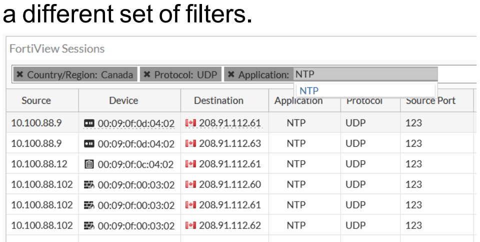
1. Click on the *Add Filter* button at the top of the session table.



2. Select the required filtering option. The session table updates to the filter selection.



3. You may add one or more filters depending upon your requirements. To add more filters, repeat the above steps for a different set of filters.



You can be very specific with how you use filters and target sessions based on different filter combinations. For example, you may want to view all sessions from a device with a particular IP by adding the *Source IP* filter. Similarly, you may need to target all the sessions having a particular *Destination IP* and *Destination Port*, and so on.

You may also view the session data in the CLI.

To view session data using the CLI:

```
# diagnose sys session list
```

The session table output in the CLI is very large. You can use the supported filters in the CLI to show only the data you need.

To view session data with filters using the CLI:

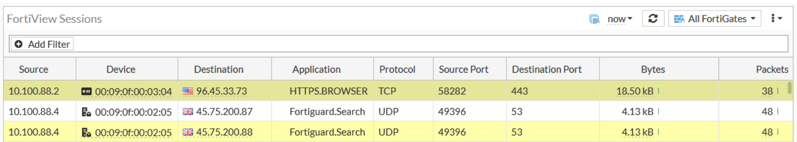
```
# diagnose sys session filter <option>
```

See to learn more about using the supported filters in the CLI.

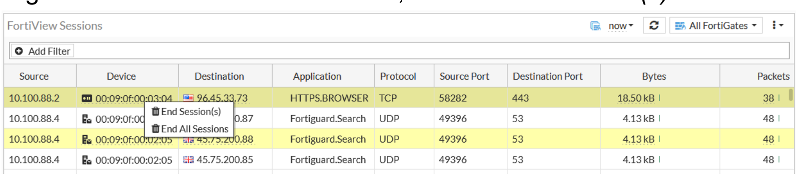
You may also decide to end a particular session or all sessions for administrative purposes.

To end sessions from the GUI:

1. Select the session you want to end. To select multiple sessions, hold the *Ctrl* or *Shift* key on your keyboard while clicking the sessions.



2. Right-click on the selected sessions, click on *End Session(s)* or *End All Sessions*.



3. Click *OK* in the confirmation dialog.

FortiView Top Source and Top Destination Firewall Objects monitors

The *FortiView Source Firewall Objects* and *FortiView Destination Firewall Objects* monitors leverage UUID to resolve firewall object address names for improved usability.

Requirements

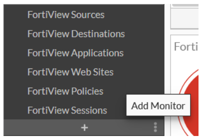
To have a historical *Firewall Objects*-based view, address objects' UUIDs need to be logged.

To enable address object UUID logging in the CLI:

```
config system global
   set log-uuid-address enable
```

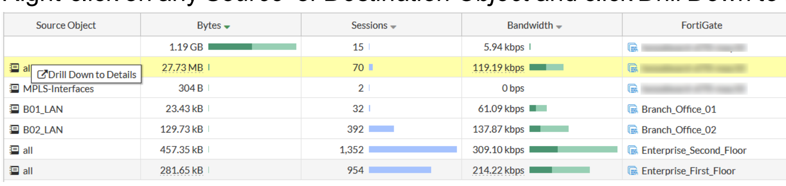
```
end
```

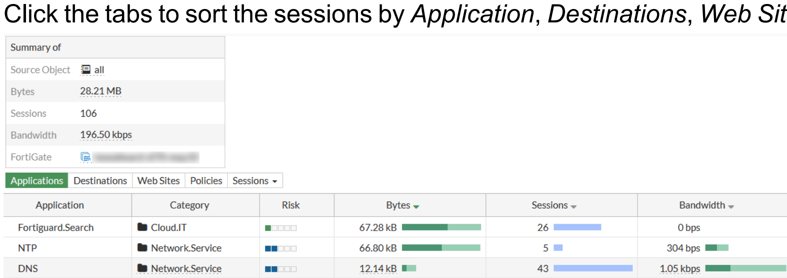
To add a firewall object monitor in the GUI:

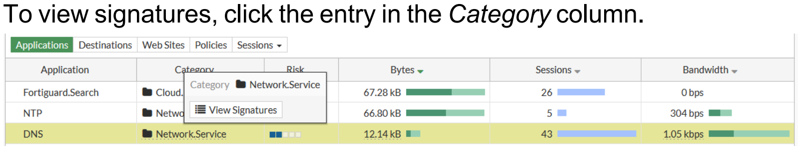
1. Click *Add Monitor*. The *Add Monitor* window opens.


2. In the *Search* field, type *Destination Firewall Objects* and click the *Add* button next to the dashboard name.
3. In the *FortiGate* area, select the FortiGate(s) from the dropdown.
4. In the *Data Source* area, select *Best Available Device* or *Specify*. For information, see Using the FortiView interface on page 111.
5. From the *Time Period* dropdown, select the time period. Select *now* for real-time information, or (*1 hour*, *24 hours*, and *7 days*) for historical information.
6. In the *Visualization* area, select *Table View* or *Bubble Chart*.
7. From the *Sort By* dropdown, select *Bytes*, *Sessions*, *Bandwidth*, or *Packets*.
8. Click *Add Monitor*. The monitor is added to the tree menu.

To drill down Firewall Objects:

1. Open the *FortiView Source Firewall Objects* or *FortiView Destination Firewall Objects* monitor.
2. Right-click on any *Source* or *Destination Object* and click *Drill Down to Details*.


3. Click the tabs to sort the sessions by *Application*, *Destinations*, *Web Sites*, or *Policies*.


4. To view signatures, click the entry in the *Category* column.


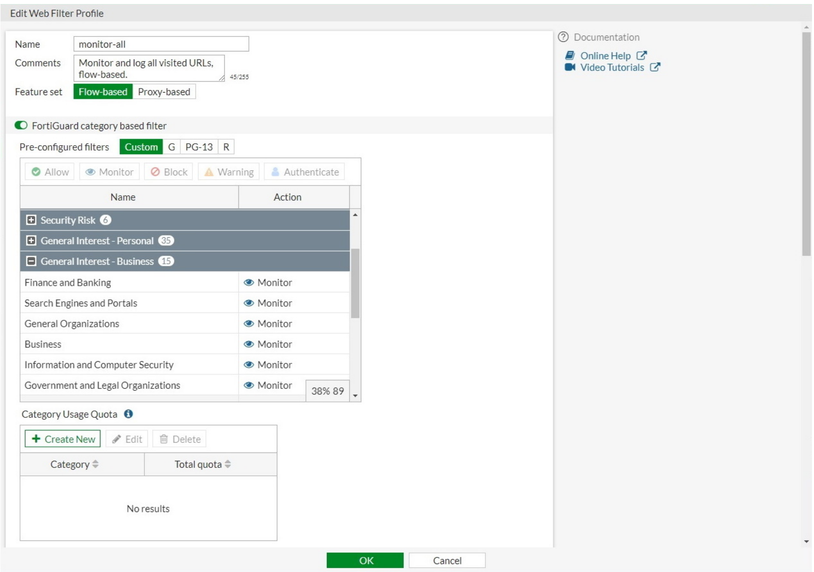
5. To views sessions, right-click an entry and click *View Sessions*, or click the *Sessions* tab.
6. To end a session, right-click an entry in the Sessions tab and select *End Sessions* or *End All Sessions*.

Viewing top websites and sources by category

You can use FortiGuard web categories to populate the category fields in various FortiView monitors such as *FortiView Web Categories*, *FortiView Websites* or *FortiView Sources*. To view the categories in a monitor, the web filter profile must be configured to at least monitor for a FortiGuard category based on a web filter and applied to a firewall policy for outbound traffic.

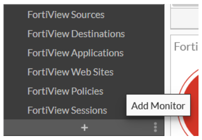
To verify the web filter profile is monitor-only:

1. Go to *Security Profiles > Web Filter*.
2. Double-click a web filter that is applied to an outbound traffic firewall policy. The *Edit Web Filter Profile* window opens.
3. Ensure *FortiGuard category based filter* is enabled.
   In the image below, the *General Interest - Business* categories are monitor-only.



To create a Web categories monitor:

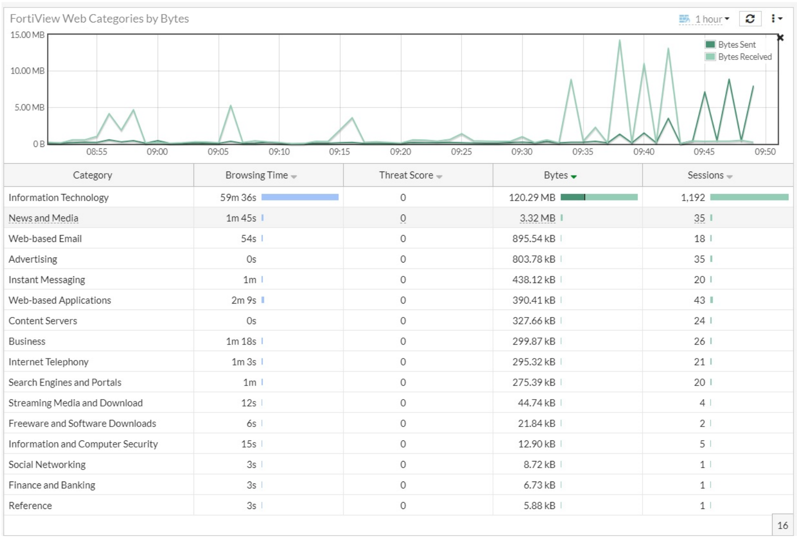
1. Click *Add Monitor*. The *Add Monitor* window opens.



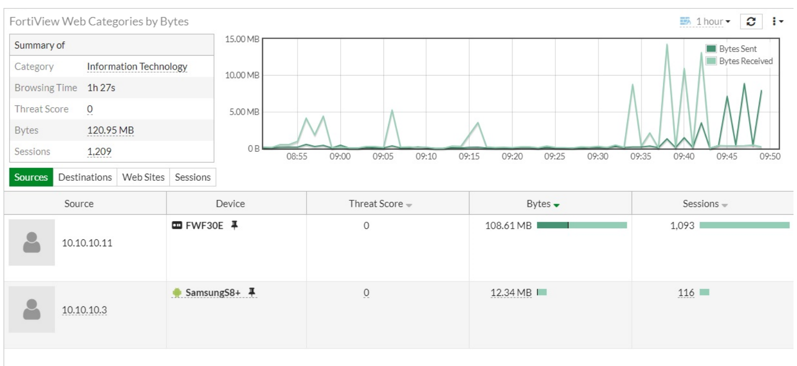
2. In the *Search* field, type *FortiView Web Categories* and click the *Add* button next to the monitor name.
3. In the *FortiGate* area, select the FortiGate(s) from the dropdown.
4. In the *Data Source* area, click *Best Available Device* or *Specify* to select a device in the security fabric.
5. From the *Time Period* dropdown, select a time period greater than *Now*.
6. From the *Sort By* dropdown, select *Bytes*, *Sessions*, *Bandwidth*, or *Packets*.
7. Click *Add Monitor*. The widget is added to the tree menu.

Viewing the web filter category

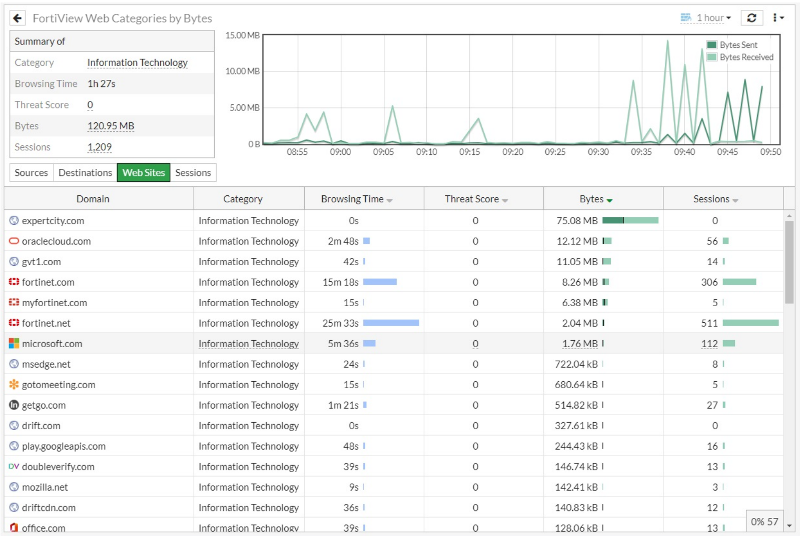
The web filter category name appears in the *Category* column of the dashboard.



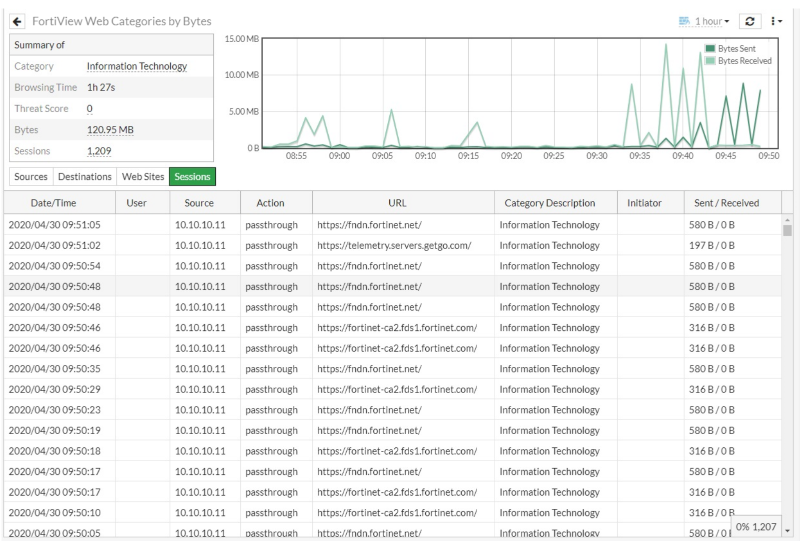
Click an entry in the table. The category name appears at the top of the *Summary of* box.



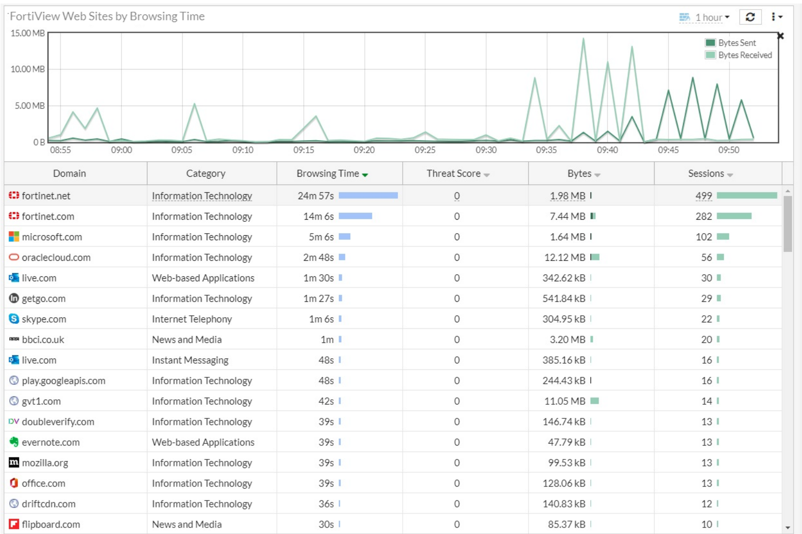
Click the *Web Sites* tab. The category name appears in the *Category* column.



Click the *Sessions* tab. The category name appears in the *Category Description* column.



The category name also appears in the *Category* column in the *FortiView Websites* and *FortiView Sources* monitors.



Domain	Category	Browsing Time	Threat Score	Bytes	Sessions
fortinet.net	Information Technology	24m 57s	0	1.98 MB	499
fortinet.com	Information Technology	14m 6s	0	7.44 MB	282
microsoft.com	Information Technology	5m 6s	0	1.64 MB	102
oraclecloud.com	Information Technology	2m 48s	0	12.12 MB	56
live.com	Web-based Applications	1m 30s	0	342.62 kB	30
getgo.com	Information Technology	1m 27s	0	541.84 kB	29
skype.com	Internet Telephony	1m 6s	0	304.95 kB	22
bbci.co.uk	News and Media	1m	0	3.20 MB	20
live.com	Instant Messaging	48s	0	385.16 kB	16
play.googleapis.com	Information Technology	48s	0	244.43 kB	16
gvt1.com	Information Technology	42s	0	11.05 MB	14
doubleverify.com	Information Technology	39s	0	146.74 kB	13
evernote.com	Web-based Applications	39s	0	47.79 kB	13
mozilla.org	Information Technology	39s	0	99.53 kB	13
office.com	Information Technology	39s	0	128.06 kB	13
driftcdn.com	Information Technology	36s	0	140.83 kB	12
flipboard.com	News and Media	30s	0	85.37 kB	10

Cloud application view

To see different cloud application views, set up the following:

- A FortiGate with a firewall policy that uses the *Application Control* security profile.
- A FortiGate with log data from the local disk or FortiAnalyzer.
- Optional but highly recommended: *SSL Inspection* set to *deep-inspection* in the related firewall policies.

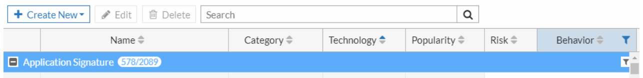
Viewing cloud applications

Cloud applications

All cloud applications require *SSL Inspection* set to *deep-inspection* on the firewall policy. For example, `Facebook_File.Download` can monitor Facebook download behavior which requires *SSL deep-inspection* to parse the deep information in the network packets.

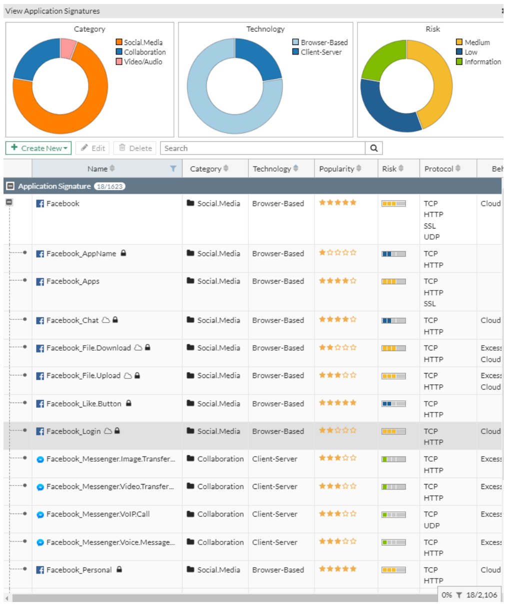
To view cloud applications:

1. Go to *Security Profiles > Application Control*.
2. Select a relative Application Control profile used by the firewall policy and click *Edit*.
3. On the *Edit Application Sensor* page, click *View Application Signatures*.
4. Hover over a column heading or the *Application Signature* bar. In the right gutter area, click the filter icon to filter the applications.



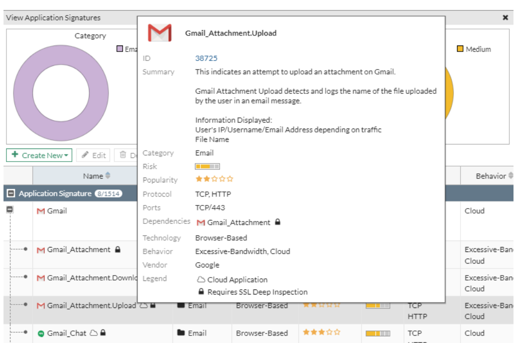
Cloud applications have a cloud icon beside them.

The lock icon indicates that the application requires SSL deep inspection.



5. Hover over an item to see its details.

   This example shows *Gmail_Attachment.Download*, a cloud application signature based sensor which requires SSL deep inspection. If any local network user behind the firewall logs into Gmail and downloads a Gmail attachment, that activity is logged.



Applications with cloud behavior

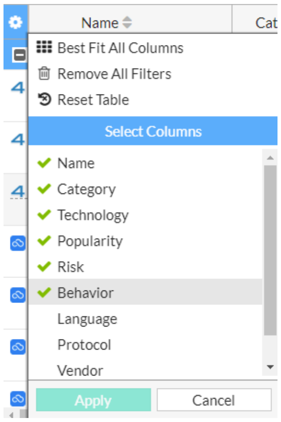
Applications with cloud behavior is a superset of cloud applications.

Some applications do not require SSL deep inspection, such as Facebook, Gmail, and YouTube. This means that if any traffic trigger application sensors for these applications, there is a FortiView cloud application view for that traffic.

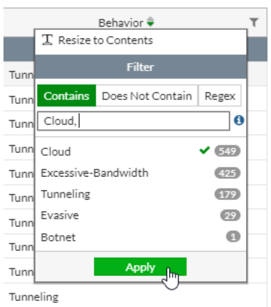
Other applications require SSL deep inspection, such as Gmail attachment, Facebook_Workplace, and so on.

To view applications with cloud behavior:

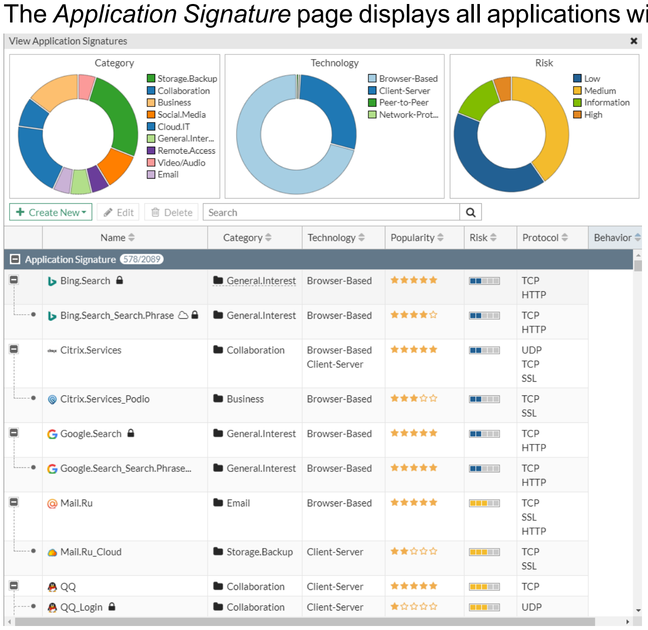
1. In the *Application Signature* page, ensure the *Behavior* column is displayed. If necessary, add the *Behavior* column.
   a. Hover over the left side of the table column headings to display the *Configure Table* icon.
   b. Click *Configure Table* and select *Behavior*.
   c. Click *Apply*.



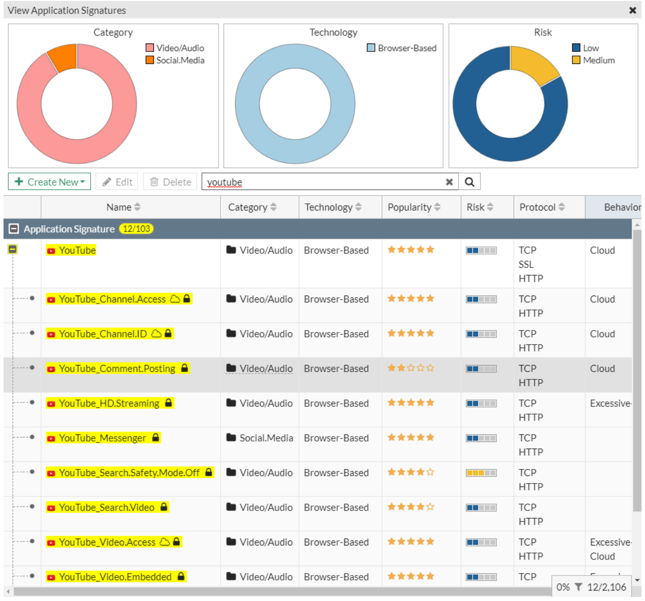
2. Click the filter icon in the *Behavior* column and select *Cloud* to filter by Cloud. Then click *Apply*.



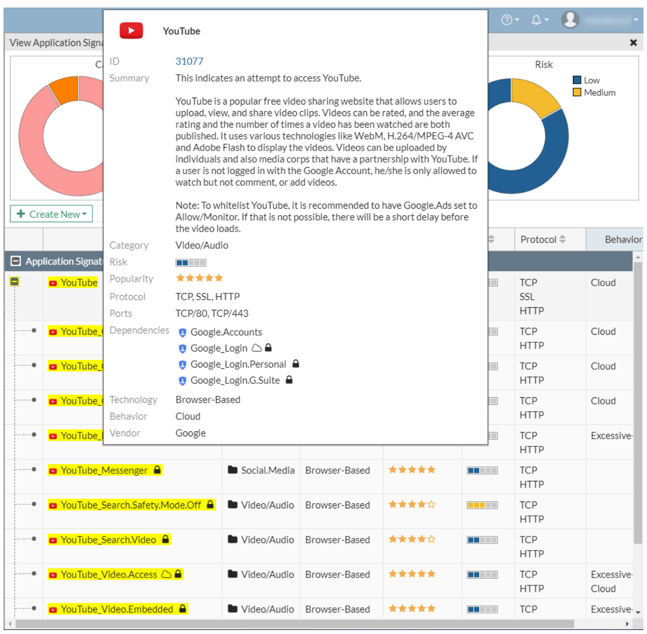
3. The *Application Signature* page displays all applications with cloud behavior.



4. Use the *Search* box to search for applications. For example, you can search for *youtube*.

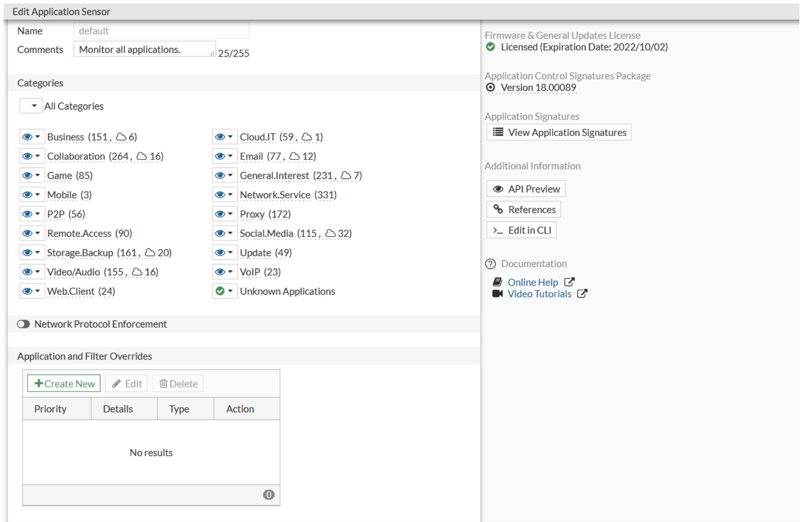


5. Hover over an item to see its details.
This example shows an application sensor with no lock icon which means that this application sensor does not require SSL deep inspection. If any local network user behind the firewall tries to navigate to the YouTube website, that activity is logged.



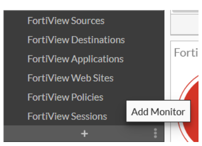
Configuring the Cloud Applications monitor

On the *Edit Application Sensor* page in the *Categories* section, the eye icon next to a category means that category is monitored and logged.

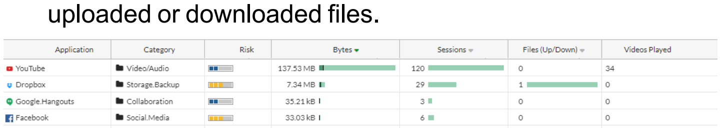


To add the Cloud Applications monitor in the GUI:

1. Click *Add Monitor*. The *Add monitor* window opens.



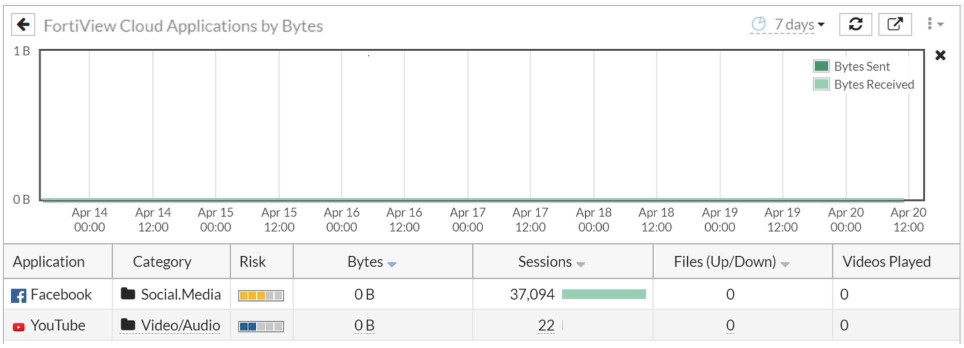
2. In the *Search* field, enter *FortiView Cloud Applications* and click the *Add* button next to the monitor.
3. In the *FortiGate* area, select the FortiGate(s) from the dropdown.
4. In the *Data Source* area, click *Best Available Device* or *Specify* to select a device in the security fabric.
5. From the *Time Period* dropdown, select a time period greater than *Now*.
6. From the *Sort By* dropdown, select *Bytes*, *Sessions*, or *Files (Up/Down)*.
7. Click *Add Monitor*. The monitor is added to the tree menu.
8. Open the monitor. If SSL deep inspection is enabled in the related firewall policy, then the monitor shows the additional details that are logged, such as *Files (Up/Down)* and *Videos Played*.
   - For YouTube, the *Videos Played* column is triggered by the *YouTube_Video.Play* cloud application sensor. This shows the number of local network users who logged into YouTube and played YouTube videos.
   - For Dropbox, the *Files (Up/Down)* column is triggered by *Dropbox_File.Download* and *Dropbox_File.Upload* cloud application sensors. This shows the number of local network users who logged into Dropbox and uploaded or downloaded files.



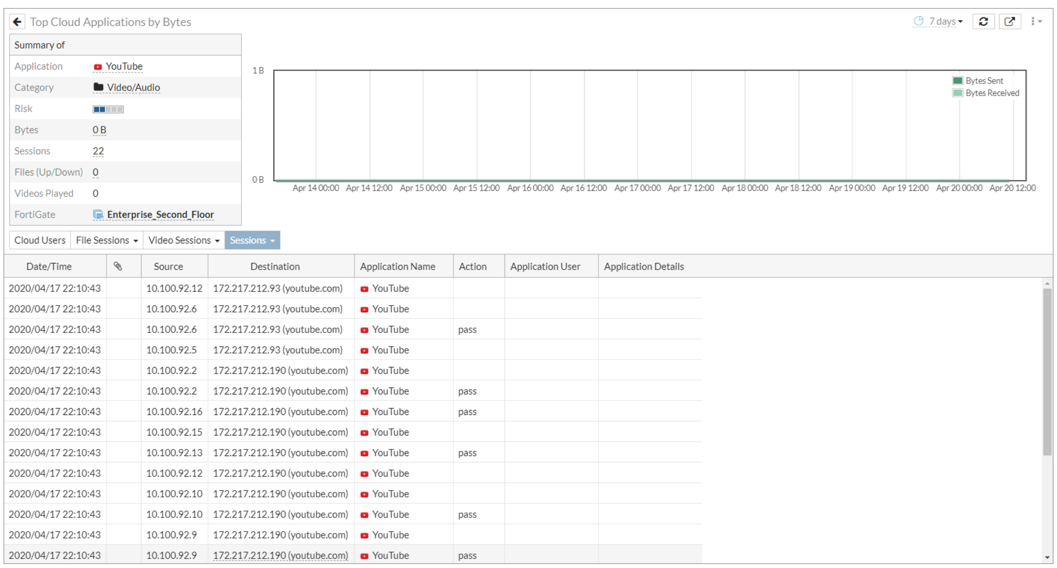
Using the Cloud Applications monitor

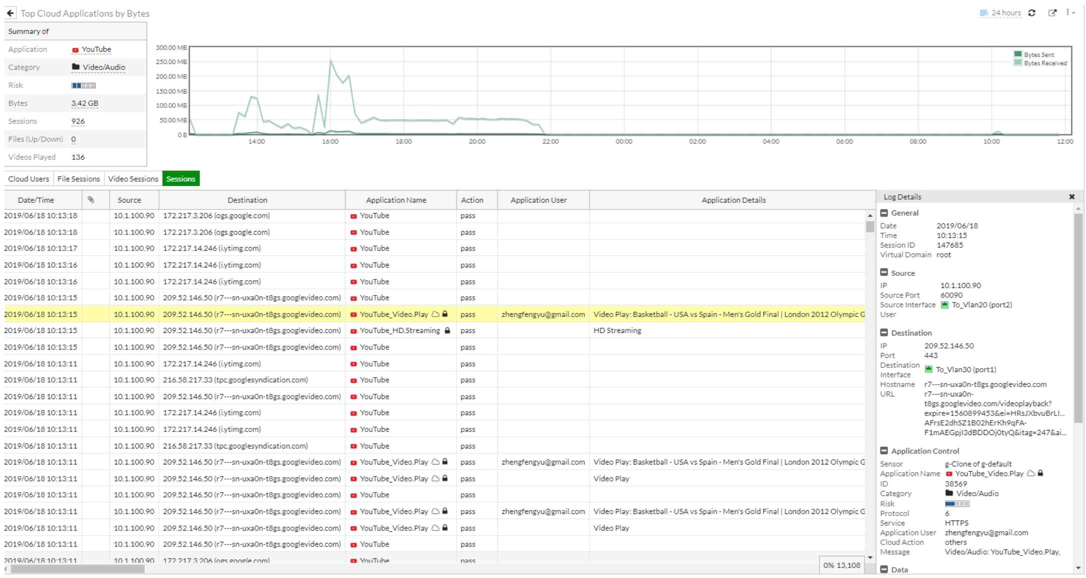
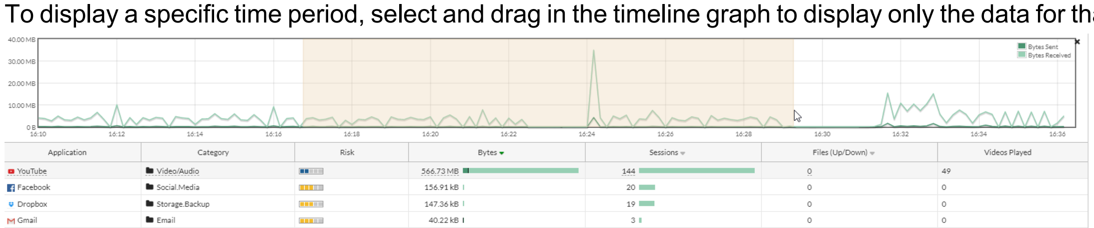
To see additional information in the Cloud Applications monitor:

1. In the tree menu, click the *FortiView Cloud Applications* monitor to open it.



2. For details about a specific entry, double-click the entry or right-click the entry and select *Drill Down to Details*.
3. To see all the sessions for an application, click *Sessions*.
   In this example, the *Application Name* column shows all applications related to YouTube.



4. To view log details, double-click a session to display the *Log Details* pane.
   Sessions monitored by SSL deep inspection (in this example, Youtube_Video.Play) captured deep information such as *Application User*, *Application Details*, and so on. The *Log Details* pane also shows additional deep information such as application *ID*, *Message*, and so on.
   Sessions not monitored by SSL deep inspection (YouTube) did not capture the deep information.
   
5. To display a specific time period, select and drag in the timeline graph to display only the data for that time period.
   

Top application: YouTube example

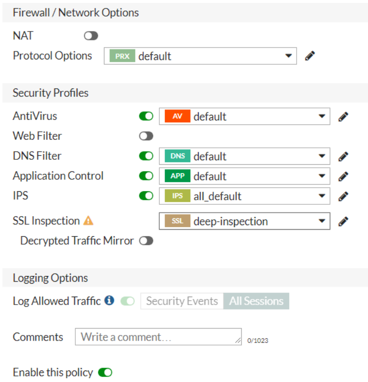
Monitoring network traffic with SSL deep inspection

This example describes how to monitor network traffic for YouTube using *FortiView Applications* view with SSL deep inspection.

To monitor network traffic with SSL deep inspection:

1. Create a firewall policy with the following settings:
   - *Application Control* is enabled.
   - *SSL Inspection* is set to *deep-inspection*.

- *Log Allowed Traffic* is set to *All Sessions*.



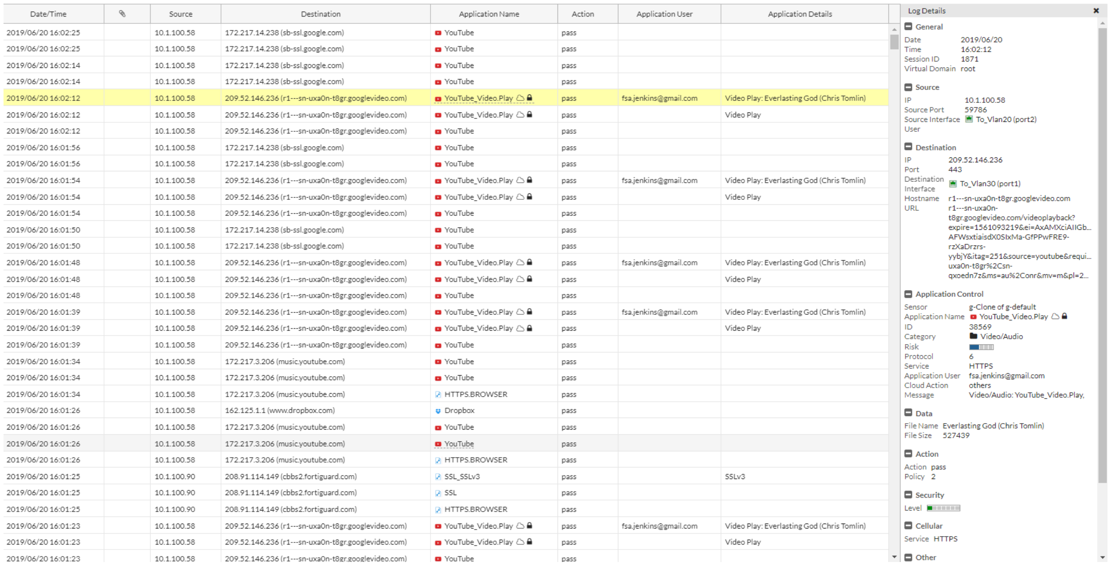
2. Go to *Security Profiles > Application Control*.
3. Select a relative Application Control profile used by the firewall policy and click *Edit*.
4. Because YouTube cloud applications are categorized into *Video/Audio*, ensure the *Video/Audio* category is monitored. Monitored categories are indicate by an eye icon.
5. Click *View Application Signatures* and hover over YouTube cloud applications to view detailed information about YouTube application sensors.
6. Expand *YouTube* to view the Application Signatures associated with the application.

Application Signature	Description	Application ID
YouTube_Video.Access	An attempt to access a video on YouTube.	16420
YouTube_Channel.ID	An attempt to access a video on a specific channel on YouTube.	44956
YouTube_Comment.Posting	An attempt to post comments on YouTube.	31076
YouTube_HD.Streaming	An attempt to watch HD videos on YouTube.	33104
YouTube_Messenger	An attempt to access messenger on YouTube.	47858
YouTube_Video.Play	An attempt to download and play a video from YouTube.	38569
YouTube_Video.Upload	An attempt to upload a video to YouTube.	22564
YouTube	An attempt to access YouTube. This application sensor does not depend on SSL deep inspection so it does not have a cloud or lock icon.	31077
YouTube_Channel.Access	An attempt to access a video on a specific channel on YouTube.	41598

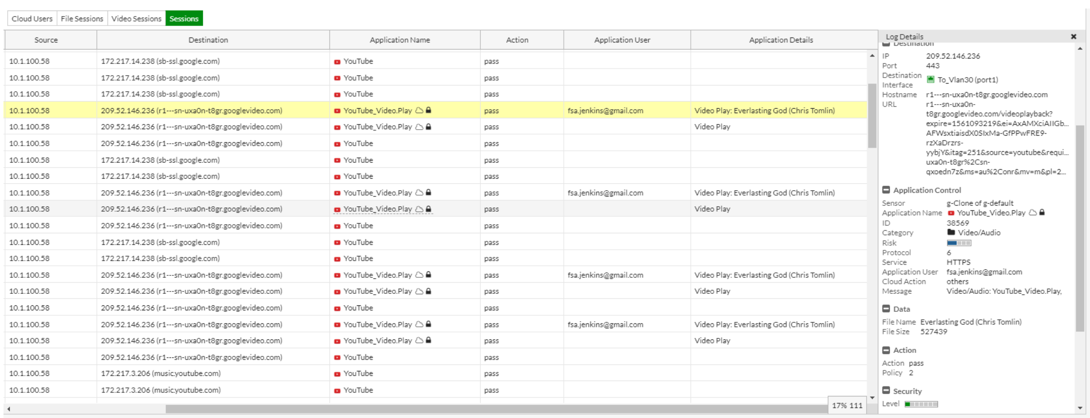
To view the application signature description, click the ID link in the information window.

7. On the test PC, log into YouTube and play some videos.
8. On the FortiGate, go to *Log & Report > Security Events* and look for log entries for browsing and playing YouTube videos in the *Application Control* card.

In this example, note the *Application User* and *Application Details*. Also note that the *Application Control ID* is *38569* showing that this entry was triggered by the application sensor *YouTube_Video.Play*.



9. Go to *Dashboard > FortiView Applications*.
10. In the *FortiView Applications* monitor, double-click *YouTube* to view the drilldown information.
11. Select the *Sessions* tab to see all the entries for the videos played. Check the sessions for *YouTube_Video.Play* with the ID *38569*.



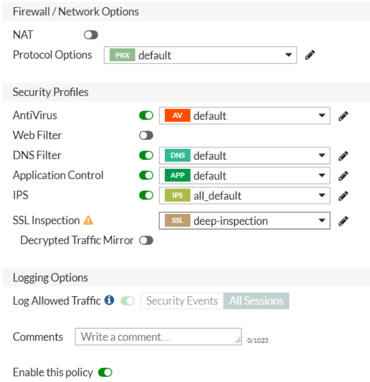
Monitoring network traffic without SSL deep inspection

This example describes how to monitor network traffic for YouTube using FortiView cloud application view without SSL deep inspection.

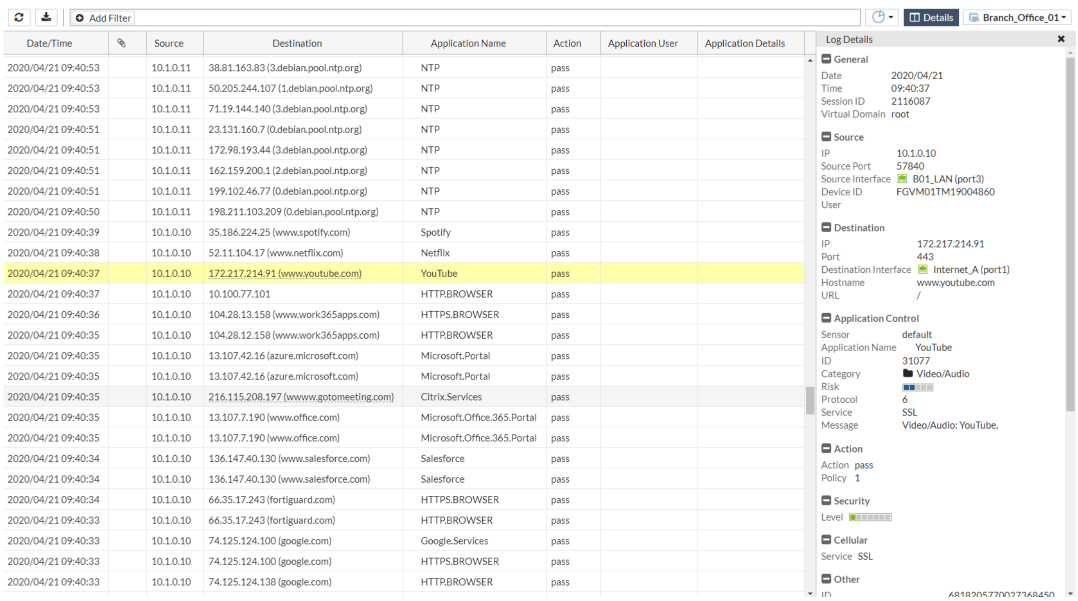
To monitor network traffic without SSL deep inspection:

1. Create a firewall policy with the following settings.
   - *Application Control* is enabled.
   - *SSL Inspection* is set to *certificate-inspection*.

- *Log Allowed Traffic* is set to *All Sessions*.

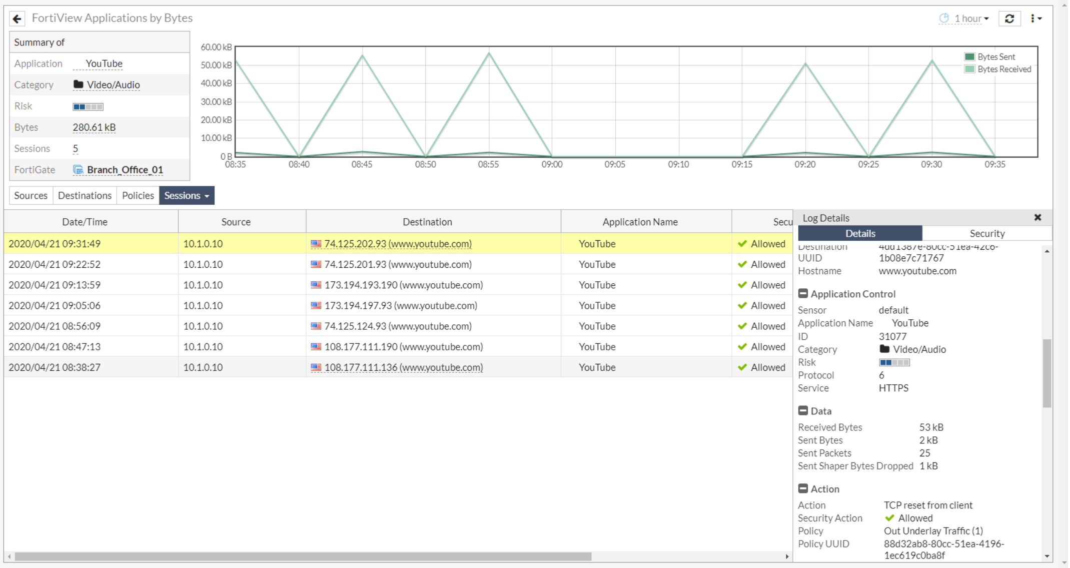


2. On the test PC, log into YouTube and play some videos.
3. On the FortiGate, go to *Log & Report > Security Events* and look for log entries for browsing and playing YouTube videos in the *Application Control* card.
   In this example, the log shows only applications with the name YouTube. The log cannot show YouTube application sensors which rely on SSL deep inspection.



4. Go to *Dashboard > FortiView Applications*.
   The *FortiView Application by Bytes* monitor shows the YouTube cloud application without the video played information that requires SSL deep inspection.

5. Double-click *YouTube* and click the *Sessions* tab.
These sessions were triggered by the application sensor *YouTube* with the ID *31077*. This is the application sensor with cloud behavior which does not rely on SSL deep inspection.



Network

The following topics provide information about network settings:

Interfaces

Physical and virtual interfaces allow traffic to flow between internal networks, and between the internet and internal networks. FortiOS has options for configuring interfaces and groups of sub-networks that can scale as your organization grows. The following table lists commonly used interface types.

Interface type	Description
Physical	A physical interface can be connected to with either Ethernet or optical cables. Depending on the FortiGate model, there is a varying number of Ethernet or optical physical interfaces. Some FortiGates have a grouping of interfaces labeled as *lan* that have a built-in switch functionality. See Physical interface on page 162 for more information.
VLAN	A virtual local area network (VLAN) logically divides a local area network (LAN) into distinct broadcast domains using IEEE 802.1Q VLAN tags. A VLAN interface supports VLAN tagging and is associated with a physical interface that can be connected to a device, such as a switch or a router that supports these tags. VLANs can be used on a FortiGate in NAT or transparent mode, and the FortiGate functions differently depending on the operation mode See VLAN on page 163 for more information.

Interface type	Description
Aggregate	An aggregate interface uses a link aggregation method to combine multiple physical interfaces to increase throughput and to provide redundancy. FortiOS supports a link aggregation (LAG) interface using the Link Aggregation Control Protocol (LACP) based on IEEE 802.3ad. See Aggregation and redundancy on page 179 for more information.
Redundant	A redundant interface combines multiple physical interfaces where traffic only uses one of the interfaces at a time. Its primary purpose is to provide redundancy. This interface is typically used with a fully-meshed HA configuration. See Aggregation and redundancy on page 179 for more information.
Loopback	A loopback interface is a logical interface that is always up because it has no physical link dependency, and the attached subnet is always present in the routing table. It can be accessed through several physical or VLAN interfaces. See Loopback interface on page 183 for more information.
Software switch	A software switch is a virtual switch interface implemented in firmware that allows member interfaces to be added to it. Devices connected to member interfaces communicate on the same subnet, and packets are processed by the FortiGate's CPU. A software switch supports adding a wireless SSID as a member interface. See Software switch on page 183 for more information.
Hardware switch	A hardware switch is a virtual switch interface implemented at the hardware level that allows member interfaces to be added to it. Devices connected to member interfaces communicate on the same subnet. A hardware switch relies on specific hardware to optimize processing and supports the Spanning Tree Protocol (STP). See Hardware switch on page 185 for more information.
Zone	A zone is a logical group containing one or more physical or virtual interfaces. Grouping interfaces in zones can simplify firewall policy configurations. See Zone on page 189 for more information.
Virtual wire pair	A virtual wire pair (VWP) is an interface that acts like a virtual wire consisting of two interfaces, with an interface at each of the wire. No IP addressing is configured on a VWP, and communication is restricted between the two interfaces using firewall policies. See Virtual wire pair on page 191 for more information.
FortiExtender WAN extension	A FortiExtender WAN extension is a managed interface that allows a connected FortiExtender to provide WAN connectivity to the FortiGate. See FortiExtender on page 401 for more information.
FortiExtender LAN extension	A FortiExtender LAN extension is a managed interface that allows a connected FortiExtender to provide LAN connectivity to the FortiGate. See FortiExtender on page 401 for more information.

Interface type	Description
Enhanced MAC VLAN	An enhanced media access control (MAC) VLAN, or EMAC VLAN, interface allows a physical interface to be virtually subdivided into multiple virtual interfaces with different MAC addresses. In FortiOS, the EMAC VLAN functionality acts like a bridge. See Enhanced MAC VLAN on page 195 for more information.
VXLAN	A Virtual Extensible LAN (VXLAN) interface encapsulates layer 2 Ethernet frames within layer 3 IP packets and is used for cloud and data center networks. See VXLAN on page 198 for more information.
Tunnel	A tunnel virtual interface is used for IPsec interface-based or GRE tunnels and are created when configuring IPsec VPN and GRE tunnels, respectively. The tunnel interface can be configured with IP addresses on both sides of the tunnel since this is a requirement when using a tunnel interface with a dynamic routing protocol. See OSPF with IPsec VPN for network redundancy on page 1459, GRE over IPsec on page 1374, and Cisco GRE-over-IPsec VPN on page 1405 for more information.
WiFi SSID	A WiFi SSID interface is used to control wireless network user access to a wireless local radio on a FortiWiFi or to a wireless access point using a FortiAP. The SSID is created using the *WiFi & Switch Controller > SSIDs page*, and it appears in the *Network > Interfaces* page once it is created. See Defining a wireless network interface (SSID) in the FortiWiFi and FortiAP Configuration Guide for more information.
VDOM link	A VDOM link allows VDOMs to communicate internally without using additional physical interfaces. See Inter-VDOM routing for more information.

Interface settings

Administrators can configure both physical and virtual FortiGate interfaces in *Network > Interfaces*. There are different options for configuring interfaces when FortiGate is in NAT mode or transparent mode.

The available options will vary depending on feature visibility, licensing, device model, and other factors. The following list is not comprehensive.

To configure an interface in the GUI:

1. Go to *Network > Interfaces*.
2. Click *Create New > Interface*.
3. Configure the interface fields:

Interface Name	Physical interface names cannot be changed.

Alias	Enter an alternate name for a physical interface on the FortiGate unit. This field appears when you edit an existing physical interface. The alias does not appear in logs. The maximum length of the alias is 25 characters.
Type	The configuration type for the interface, such as VLAN, Software Switch. 802.3ad Aggregate, and others.
Interface	This field is available when *Type* is set to *VLAN*. Select the name of the physical interface that you want to add a VLAN interface to. Once created, the VLAN interface is listed below its physical interface in the *Interface* list. You cannot change the physical interface of a VLAN interface.
VLAN ID	This field is available when *Type* is set to *VLAN*. Enter the VLAN ID. The VLAN ID can be any number between 1 and 4094 and must match the VLAN ID added by the IEEE 802.1Q-compliant router or switch that is connected to the VLAN subinterface. The VLAN ID can be edited after the interface is added.
VRF ID	Virtual Routing and Forwarding (VRF) allows multiple routing table instances to coexist on the same router. One or more interface can have a VRF, and packets are only forwarded between interfaces with the dame VRF.
Virtual Domain	Select the virtual domain to add the interface to. Only administrator accounts with the *super_admin* profile can change the *Virtual Domain*.
Interface Members	This section can have different formats depending on the *Type*. Members can be selected for some interface types: • *Software Switch* or *Hardware Switch*: Specify the physical and wireless interfaces joined into the switch. • *802.3ad Aggregate* or *Redundant Interface*: This field includes the available and selected interface lists.
Role	Set the role setting for the interface. Different settings will be shown or hidden when editing an interface depending on the role: • *LAN*: Used to connected to a local network of endpoints. It is default role for new interfaces. • *WAN*: Used to connected to the internet. When WAN is selected, the *Estimated bandwidth* setting is available, and the following settings are not: *DHCP server*, *Create address object matching subnet*, *Device detection*, *Security mode*, *One-arm sniffer*, *Dedicate to extension/fortiap modes*, and *Admission Control*.and will show Estimated Bandwidth settings. • *DMZ*: Used to connected to the DMZ. When selected, *DHCP server* and *Security mode* are not available. • *Undefined*: The interface has no specific role. When selected, *Create address object matching subnet* is not available.
Estimated bandwidth	The estimated WAN bandwidth.

	The values can be entered manually, or saved from a speed test executed on the interface. The values can be used in SD-WAN rules that use the Maximize Bandwidth or Best Quality strategy.
Traffic mode	This option is only available when *Type* is *WiFi SSID*. • *Tunnel*: Tunnel to wireless controller • *Bridge*: Local bridge with FortiAP's interface • *Mesh*: Mesh downlink
Address	
Addressing mode	Select the addressing mode for the interface. • *Manual*: Add an IP address and netmask for the interface. If IPv6 configuration is enabled, you can add both an IPv4 and an IPv6 address. • *DHCP*: Get the interface IP address and other network settings from a DHCP server. • *Auto-managed by IPAM*: Assign subnets to prevent duplicate IP addresses from overlapping within the same Security Fabric. See Configure IPAM locally on the FortiGate on page 143. • *PPPoE*: Get the interface IP address and other network settings from a PPPoE server. This option is only available on the low-end FortiGate models. • *One-Arm Sniffer*: Set the interface as a sniffer port so it can be used to detect attacks. See One-arm sniffer on page 151.
IP/Netmask	If *Addressing Mode* is set to *Manual*, enter an IPv4 address and subnet mask for the interface. FortiGate interfaces cannot have multiple IP addresses on the same subnet.
IPv6 addressing mode	Select the addressing mode for the interface: • *Manual*: Add an IP address and netmask for the interface. • *DHCP*: Get the interface IP address and other network settings from a DHCP server. • *Delegated*: Select an *IPv6 upstream interface* that has DHCPv6 prefix delegation enabled, and enter an *IPv6 subnet* if needed. The interface will get the IPv6 prefix from the upstream DHCPv6 server that is connected to the IPv6 upstream interface, and form the IPv6 address with the subnet configured on the interface.
IPv6 Address/Prefix	If *Addressing Mode* is set to *Manual* and IPv6 support is enabled, enter an IPv6 address and subnet mask for the interface. A single interface can have an IPv4 address, IPv6 address, or both.
Auto configure IPv6 address	Automatically configure an IPv6 address using Stateless Address Auto-configuration (SLAAC). This option is available when *IPv6 addressing mode* is set to *Manual*.
DHCPv6 prefix delegation	Enable/disable DHCPv6 prefix delegation, which can be used to delegate IPv6 prefixes from an upstream DHCPv6 server to another interface or downstream device.

	When enabled, there is an option to enable a *DHCPv6 prefix hint* that helps the DHCPv6 server provide the desired prefix.
Create address object matching subnet	This option is available when *Role* is set to *LAN* or *DMZ*. Enable this option to automatically create an address object that matches the interface subnet.
Secondary IP Address	Add additional IPv4 addresses to this interface.
Administrative Access	
IPv4 Administrative Access	Select the types of administrative access permitted for IPv4 connections to this interface. See Configure administrative access to interfaces on page 141.
IPv6 Administrative Access	Select the types of administrative access permitted for IPv6 connections to this interface. See Configure administrative access to interfaces on page 141.
DHCP Server	Enable a DHCP server for the interface. See DHCP server on page 287.
Stateless Address Auto-configuration (SLAAC)	Enable to provide IPv6 addresses to connected devices using SLAAC.
DHCPv6 Server	Select to enable a DHCPv6 server for the interface. When enabled, you can configure *DNS service* settings: *Delegated* (delegate the DNS received from the upstream server), *Same as System DNS*, or *Specify* (up to four servers). You can also enable *Stateful server*to configure the DHCPv6 server to be stateful. Manually enter the IP range, or use Delegated mode to delegate IP prefixes from an upstream DHCPv6 server connected to the upstream interface.
Network	
Device Detection	Enable/disable passively gathering device identity information about the devices on the network that are connected to this interface.
Security Mode	Enable/disable captive portal authentication for this interface. After enabling captive portal authentication, you can configure the authentication portal, user and group access, custom portal messages, exempt sources and destinations/services, and redirect after captive portal.
DSL Settings	
Physical mode	Set to *ADSL* or *VDSL*.
Transfer mode	Set to *PTM* or *ATM*. If the *Transfer mode* is set to *ATM*, the *Virtual channel identification*, *Virtual path identification*, *ATM protocol*, and *MUX type* can be configured.
Traffic Shaping	
Outbound shaping profile	Enable/disable traffic shaping on the interface. This allows you to enforce bandwidth limits on individual interfaces. See Interface-based traffic shaping profile on page 953 for more information.
Miscellaneous	

Comments	Enter a description of the interface of up to 255 characters.
Status	Enable/disable the interface. • *Enabled*: The interface is active and can accept network traffic. • *Disabled*: The interface is not active and cannot accept traffic.

4. Click *OK*.

To configure an interface in the CLI:

```
config system interface
    edit <name>
        set vdom <VDOM_name>
        set mode {static | dhcp | pppoe}
        set ip <IP_address/netmask>
        set security-mode {none | captive-portal | 802.1X}
        set egress-shaping-profile <profile>
        set device-identification {enable | disable}
        set allowaccess {ping https ssh http snmp telnet fgfm radius-acct probe-response
fabric ftm}
        set eap-supplicant {enable | disable}
        set eap-method {peap | tls}
        set eap-identity <identity>
        set eap-password <password>
        set eap-ca-cert <CA_cert>
        set eap-user-cert <user_cert>
        set secondary-IP enable
        config secondaryip
            edit 1
                set ip 9.1.1.2 255.255.255.0
                set allowaccess ping https ssh snmp http
            next
        end
    next
end
```

Configure administrative access to interfaces

You can configure the protocols that administrators can use to access interfaces on the FortiGate. This helps secure access to the FortiGate by restricting access to a limited number of protocols. It helps prevent users from accessing interfaces that you don't want them to access, such as public-facing ports.

As a best practice, you should configure administrative access when you're setting the IP address for a port.

To configure administrative access to interfaces in the GUI:

1. Go to *Network* > *Interfaces*.
2. Create or edit an interface.
3. In the *Administrative Access* section, select which protocols to enable for *IPv4* and *IPv6 Administrative Access*.

Speed Test	Allow this interface to listen to speed test sender requests. To allow the FortiGate to be configured as speed test server, configure the following:

	`config system global`
	`    set speedtest-server {enable | disable}`
	`end`
	
	For more detail, see Speed tests run from the hub to the spokes in dial-up IPsec tunnels on page 664.

HTTPS	Allow secure HTTPS connections to the FortiGate GUI through this interface. If configured, this option is enabled automatically.
HTTP	Allow HTTP connections to the FortiGate GUI through this interface. This option can only be enabled if HTTPS is already enabled.
PING	The interface responds to pings. Use this setting to verify your installation and for testing.
FMG-Access	Allow FortiManager authorization automatically during the communication exchanges between FortiManager and FortiGate devices.
SSH	Allow SSH connections to the CLI through this interface.
SNMP	Allow a remote SNMP manager to request SNMP information by connecting to this interface.
FTM	Allow FortiToken Mobile Push (FTM) access.
RADIUS Accounting	Allow RADIUS accounting information on this interface.
Security Fabric Connection	Allow Security Fabric access. This enables FortiTelemetry and CAPWAP.

FEC implementations on 10G, 25G, 40G, and 100G interfaces

Only supported FEC (forward error correction) implementations are allowed to be configured on 10G, 25G, 40G, and 100G interfaces based on the speed that is selected.

- For 1000M, 10G, or 40G interfaces, FEC is not supported and the option is disabled.
- For 25G and 100G interfaces, FEC is automatically set to `cl91-rs-fec` by default.

To configure an interface for FEC:

```
config system interface
    edit <name>
        set speed {10000full | 1000full | 100Gauto | 100Gfull | 25000auto | 25000full |
40000full}
        set mediatype {sr4 | lr4 | cr4}
        set forward-error-correction {disable | cl91-rs-fec | cl74-fc-fec}
    next
end
```

`speed {10000full | 1000full | 100Gauto | 100Gfull | 25000auto | 25000full | 40000full}`	Set the interface speed: • 10000full: 10G full-duplex • 1000full: 1000M full-duplex • 100Gauto: 100G auto-negotiation • 100Gfull: 100G full-duplex • 25000auto: 25G auto-negotiation • 25000full: 25G full-duplex • 40000full: 40G full-duplex
`mediatype {sr4 | lr4 | cr4}`	Set the media type to use: • sr4: short-range transceiver (4-lane) • lr4: long-range transceiver (4-lane) • cr4: copper transceiver (4-lane)
`forward-error-correction {disable | cl91-rs-fec | cl74-fc-fec}`	Set the forward error correction type: • disable: disable forward error correction • cl91-rs-fec: Reed-Solomon (FEC CL91) • cl74-fc-fec: Firecode (FEC CL74)

To change the interface speed from 40G to 100G:

```
config system interface
    edit port26
        set speed 100Gfull
    next
end

The speed/mediatype/FEC of port26 will be changed from 40000full/sr4/disable to
100Gfull/sr4/cl91-rs-fec.
Do you want to continue? (y/n) y
```

Since the speed changed to 1000G, the `mediatype` setting automatically changes to `sr4`, and the `forward-error-correction` setting automatically changes to `cl91-rs-fec`. When the speed was 40G, the `forward-error-correction` setting was disabled.

Configure IPAM locally on the FortiGate

IPAM (IP address management) is available locally on the FortiGate. A standalone FortiGate, or a Fabric root in the Security Fabric, can act as the IPAM server. Interfaces configured to be auto-managed by IPAM will receive an address from the IPAM server's address/subnet pool. *DHCP Server* is automatically enabled in the GUI, and the address range is populated by IPAM. Users can customize the address pool subnet and the size of a subnet that an interface can request.

To configure IPAM settings:

```
config system ipam
    set pool-subnet <class IP and netmask>
    set status {enable | disable}
end
```

`pool-subnet <class IP and netmask>`	Set the IPAM pool subnet, class A or class B subnet.
`status {enable | disable}`	Enable/disable IP address management services.

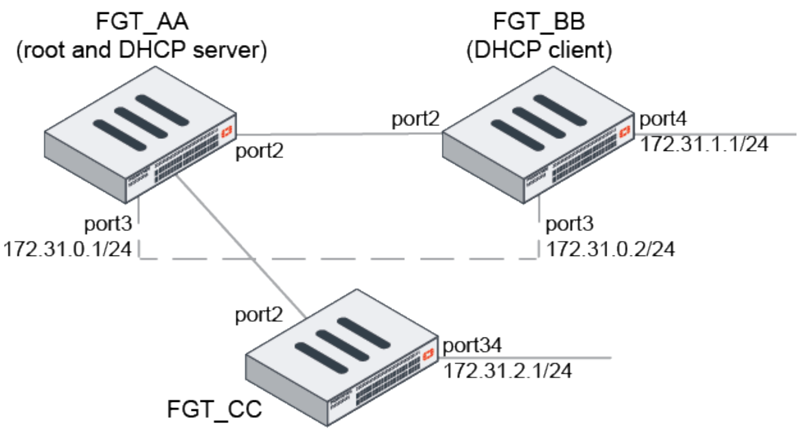
In previous FortiOS versions, the `set fortiipam-integration` option was configured under `config system global`.

The following options are available for allocating the subnet size:

```
config system interface
    set managed-subnetwork-size {32 | 64 | 128 | 256 |512 | 1024 | 2048 | 4096 | 8192 |
16384 | 32768 | 65536}
end
```

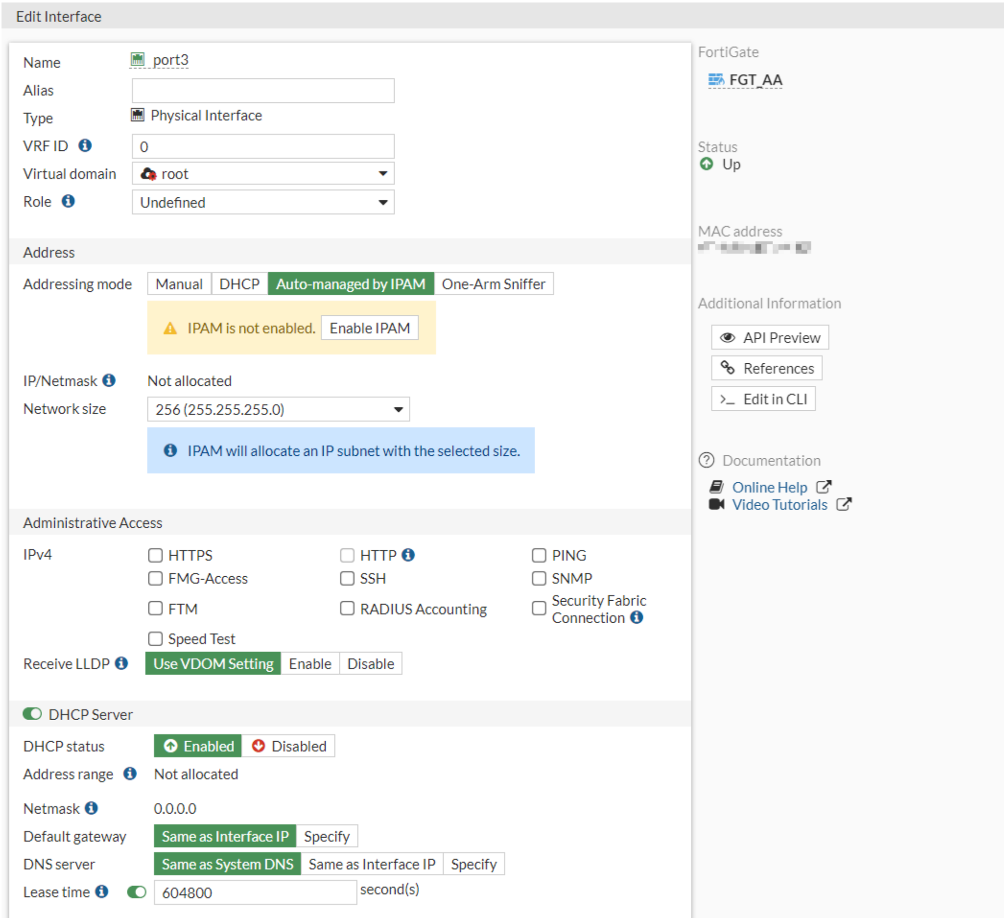
Example

In this example, FGT_AA is the Security Fabric root with IPAM enabled. FGT_BB and FGT_CC are downstream Fabric devices and retrieve IPAM information from FGT_AA. The Fabric interface on all FortiGates is port2. FGT_AA acts as the DHCP server, and FGT_BB acts as the DHCP client.

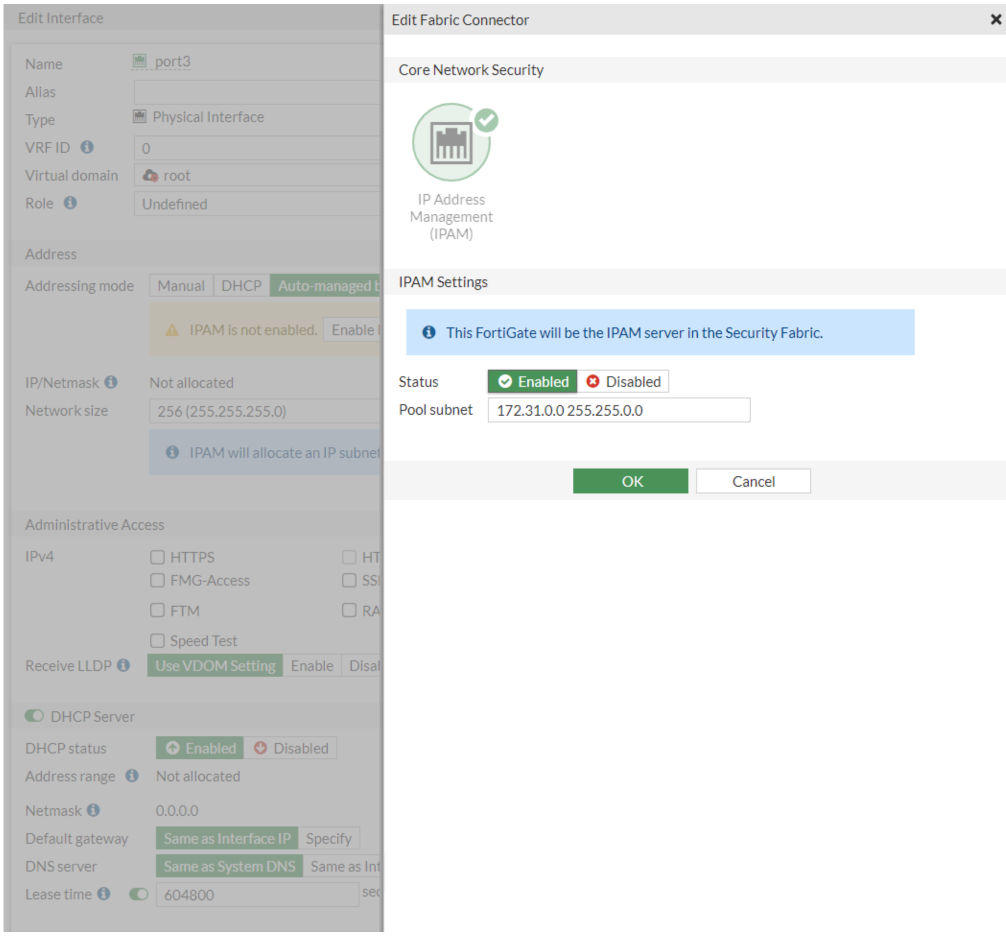


To configure IPAM locally in the Security Fabric:

1. On the root FortiGate, go to *Network > Interfaces* and edit port3.
2. For *Addressing Mode*, select *Auto-Managed by IPAM*. *DHCP Server* is automatically enabled.



3. In this example, IPAM is not enabled yet. Click *Enable IPAM*. The *Edit Fabric Connector* pane opens.

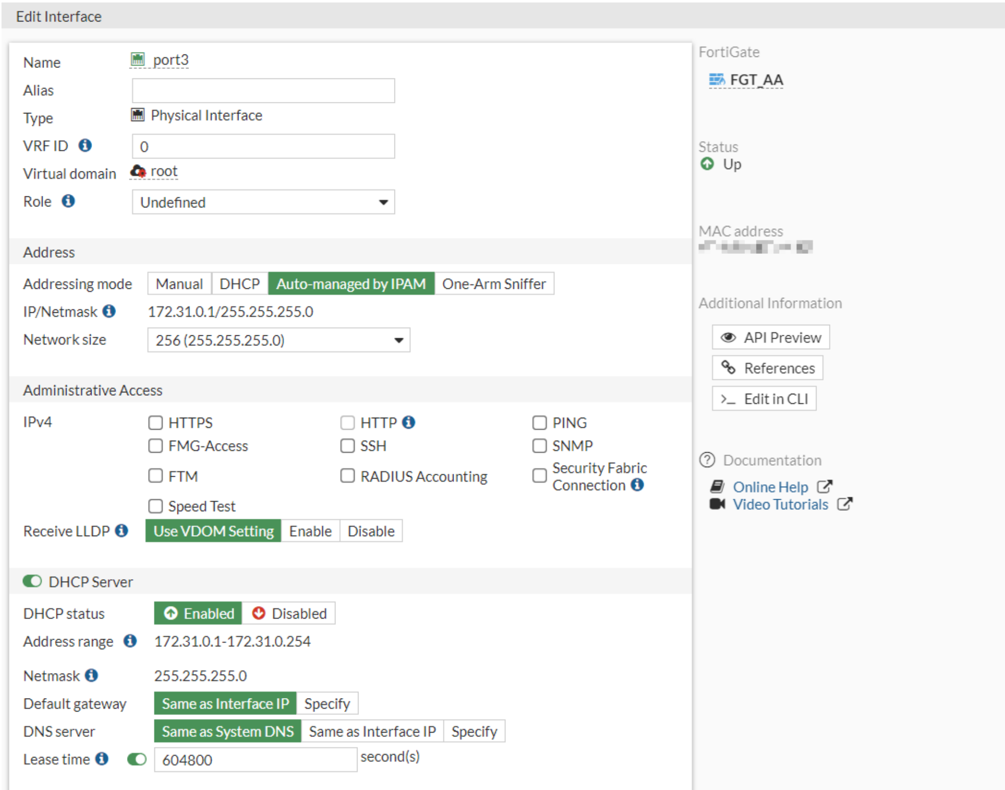


4. Enter the *Pool subnet* (only class A and B are allowed) and click *OK*. The root FortiGate is now the IPAM server in the Security Fabric. The following is configured in the backend:

```
config system interface
    edit "port3"
        set vdom "root"
        set ip 172.31.0.1 255.255.255.0
        set type physical
        set device-identification enable
        set snmp-index 5
        set ip-managed-by-fortiipam enable
        end
    next
end

config system ipam
    set status enable
end
```

IPAM is managing a 172.31.0.0/16 network and assigned port3 a /24 network by default.

The *IP/Netmask* field in the *Address* section has been automatically assigned a class C IP by IPAM. The *Address range* and *Netmask* fields in the *DHCP Server* section have also been automatically configured by IPAM.

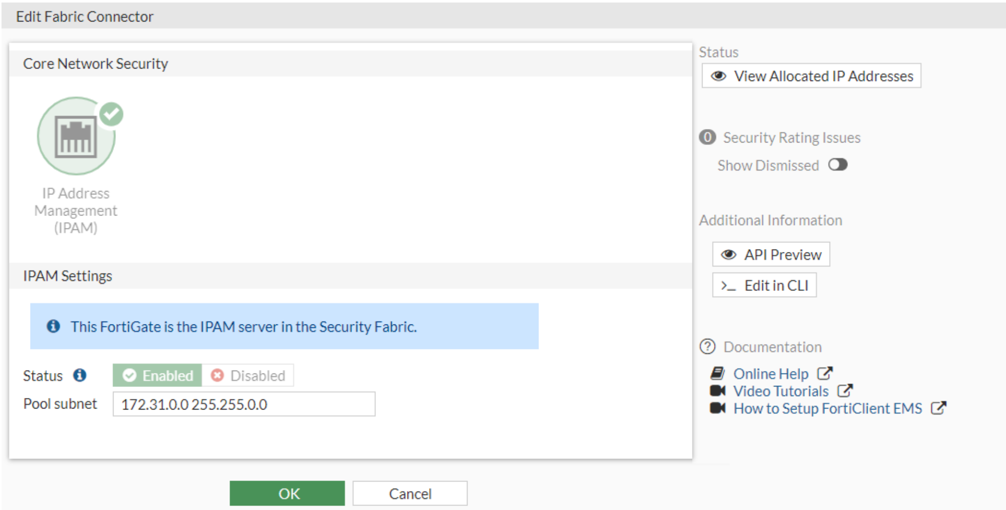


5. Click *OK*.
6. Log in to FGT-BB and set the *Addressing Mode* of port4 to *Auto-Managed by IPAM*. The subnet assigned from the pool on the root is 172.31.1.1/24.
7. Log in to FG_CC and set the *Addressing Mode* of port34 to *Auto-Managed by IPAM*. The subnet assigned from the pool on the root is 172.31.2.1/24.

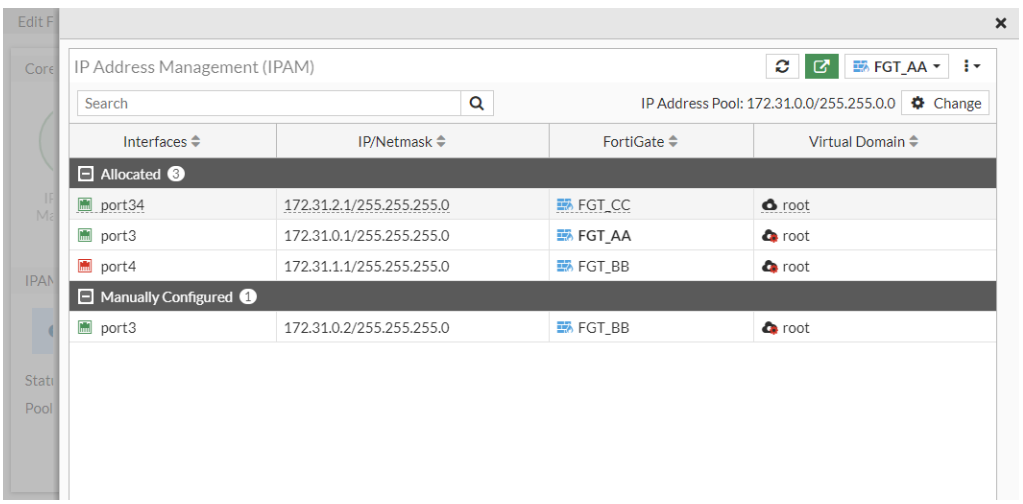
Any interface on a downstream FortiGate can be managed by the IPAM server. The interface does not have to be directly connected to the Fabric root FortiGate.

To edit the IPAM subnet:

1. Go to *Security Fabric > Fabric Connectors* and double-click the *IP Address Management (IPAM)* card.
2. Edit the pool subnet if needed.



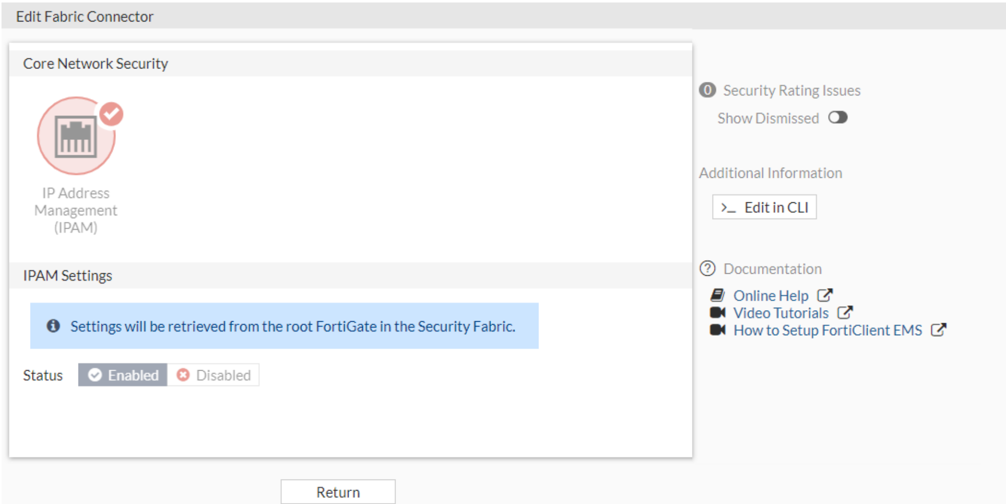
3. In the right-side pane, click *View Allocated IP Addresses* to view the subnet allocations (port34, port3, and port3) and DHCP lease information. On FGT_BB, port3 is a DHCP client and the DHCP server interface (FGT_AA port3) is managed by IPAM, so it is displayed in the *Manually Configured* section.



The same allocated IP address information is available in the *IP Address Management (IPAM)* widget that can be added to the *Dashboard > Status* page.

4. Click *OK*.

On downstream FortiGates, the settings on the *IP Address Management (IPAM)* card cannot be changed if IPAM is enabled on the root FortiGate.



Diagnostics

Use the following commands to view IPAM related diagnostics.

To view the largest available subnet size:

```
# diagnose sys ipam largest-available-subnet
Largest available subnet is a /17.
```

To verify IPAM allocation information:

```
# diagnose sys ipam dump-ipams-entries
IPAM Entries: (sn, vdom, interface, subnet/mask, flag)
  F140EP4Q17000000 root port34 172.31.2.1/24 0
  FG5H1E5818900001 root port3 172.31.0.1/24 0
  FG5H1E5818900002 root port4 172.31.1.1/24 0
  FG5H1E5818900003 root port3 172.31.0.2/24 1
```

To verify the available subnets:

```
# diagnose sys ipam dump-ipams-free-subnets
IPAM free subnets: (subnet/mask)
  172.31.3.0/24
  172.31.4.0/22
  172.31.8.0/21
  172.31.16.0/20
  172.31.32.0/19
  172.31.64.0/18
  172.31.128.0/17
```

To remove a device from IPAM in the Security Fabric:

```
# diagnose sys ipam delete-device-from-ipams F140EP4Q17000000
Successfully removed device F140EP4Q17000000 from ipam
```

Interface MTU packet size

Changing the maximum transmission unit (MTU) on FortiGate interfaces changes the size of transmitted packets. Most FortiGate device's physical interfaces support jumbo frames that are up to 9216 bytes, but some only support 9000 or 9204 bytes.

To avoid fragmentation, the MTU should be the same as the smallest MTU in all of the networks between the FortiGate and the destination. If the packets sent by the FortiGate are larger than the smallest MTU, then they are fragmented, slowing down the transmission. Packets with the DF flag set in the IPv4 header are dropped and not fragmented .

On many network and endpoint devices, the path MTU is used to determine the smallest MTU and to transmit packets within that size.

- ASIC accelerated FortiGate interfaces, such as NP6, NP7, and SOC4 (np6xlite), support MTU sizes up to 9216 bytes.
- FortiGate VMs can have varying maximum MTU sizes, depending on the underlying interface and driver.
- Virtual interfaces, such as VLAN interfaces, inherit their MTU size from their parent interface.

To verify the supported MTU size:

```
config system interface
    edit <interface>
        set mtu-override enable
        set mtu <integer>
    next
end
```

To change the MTU size:

```
config system interface
    edit <interface>
        set mtu-override enable
        set mtu <max bytes>
    next
end
```

Maximum MTU size on a path

To manually test the maximum MTU size on a path, you can use the ping command on a Windows computer.

For example, you can send ICMP packets of a specific size with a DF flag, and iterate through increasing sizes until the ping fails.

- The `-f` option specifies the Do not Fragment (DF) flag.
- The `-l` option specifies the length, in bytes, of the Data field in the echo Request messages. This does not include the 8 bytes for the ICMP header and 20 bytes for the IP header. Therefore, if the maximum MTU is 1500 bytes, then the maximum supported data size is: 1500 - 8 - 20 = 1472 bytes.

To determine the maximum MTU size on a path:

1. In Windows command prompt, try a likely MTU size:

   ```
   >ping 4.2.2.1 -l 1472 -f
   Pinging 4.2.2.1 with 1472 bytes of data:
   ```

```
Reply from 4.2.2.1: bytes=1472 time=41ms TTL=52
Reply from 4.2.2.1: bytes=1472 time=42ms TTL=52
Reply from 4.2.2.1: bytes=1472 time=103ms TTL=52
Reply from 4.2.2.1: bytes=1472 time=38ms TTL=52

Ping statistics for 4.2.2.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 38ms, Maximum = 103ms, Average = 56ms
```

2. Increase the size and try the ping again:

```
>ping 4.2.2.1 -l 1473 -f
Pinging 4.2.2.1 with 1473 bytes of data:
Request timed out.

Ping statistics for 4.2.2.1:
    Packets: Sent = 1, Received = 0, Lost = 1 (100% loss),
```

The second test fails, so the maximum MTU size on the path is 1472 bytes + 8-byte ICMP header + 20-byte IP header = 1500 bytes

Maximum segment size

The TCP maximum segment size (MSS) is the maximum amount of data that can be sent in a TCP segment. The MSS is the MTU size of the interface minus the 20 byte IP header and 20 byte TCP header. By reducing the TCP MSS, you can effectively reduce the MTU size of the packet.

The TCP MSS can be configured in a firewall policy (see Configurations in the CLI on page 765), or directly on an interface.

To configure the MSS on an interface:

```
config system interface
    edit "wan2"
        set vdom "root"
        set mode dhcp
        set allowaccess ping fgfm
        set type physical
        set tcp-mss 1448
        set role wan
    next
end
```

One-arm sniffer

You can use a one-arm sniffer to configure a physical interface as a one-arm intrusion detection system (IDS). Traffic sent to the interface is examined for matches to the configured security profile. The matches are logged, and then all received traffic is dropped. Sniffing only reports on attacks; it does not deny or influence traffic.

You can also use the one-arm sniffer to configure the FortiGate to operate as an IDS appliance to sniff network traffic for attacks without actually processing the packets. To configure a one-arm IDS, enable sniffer mode on a physical interface and connect the interface to the SPAN port of a switch or a dedicated network tab that can replicate the traffic to the FortiGate.

If the one-arm sniffer option is not available, this means the interface is in use. Ensure that the interface is not selected in any firewall policies, routes, virtual IPs, or other features where a physical interface is specified. The option also does not appear if the role is set to WAN. Ensure the role is set to LAN, DMZ, or undefined.

The following table lists some of the one-arm sniffer settings you can configure:

Field	Description
Security Profiles	The following profiles are configurable in the GUI and CLI: • Antivirus • Web filter • Application control • IPS • File filter The following profiles are only configurable in the CLI: • Email filter • DLP • IPS DoS

CPU usage and packet loss

Traffic scanned on the one-arm sniffer interface is processed by the CPU, even if there is an SPU, such as NPU or CP, present. The one-arm sniffer may cause higher CPU usage and perform at a lower level than traditional inline scanning, which uses NTurbo or CP to accelerate traffic when present.

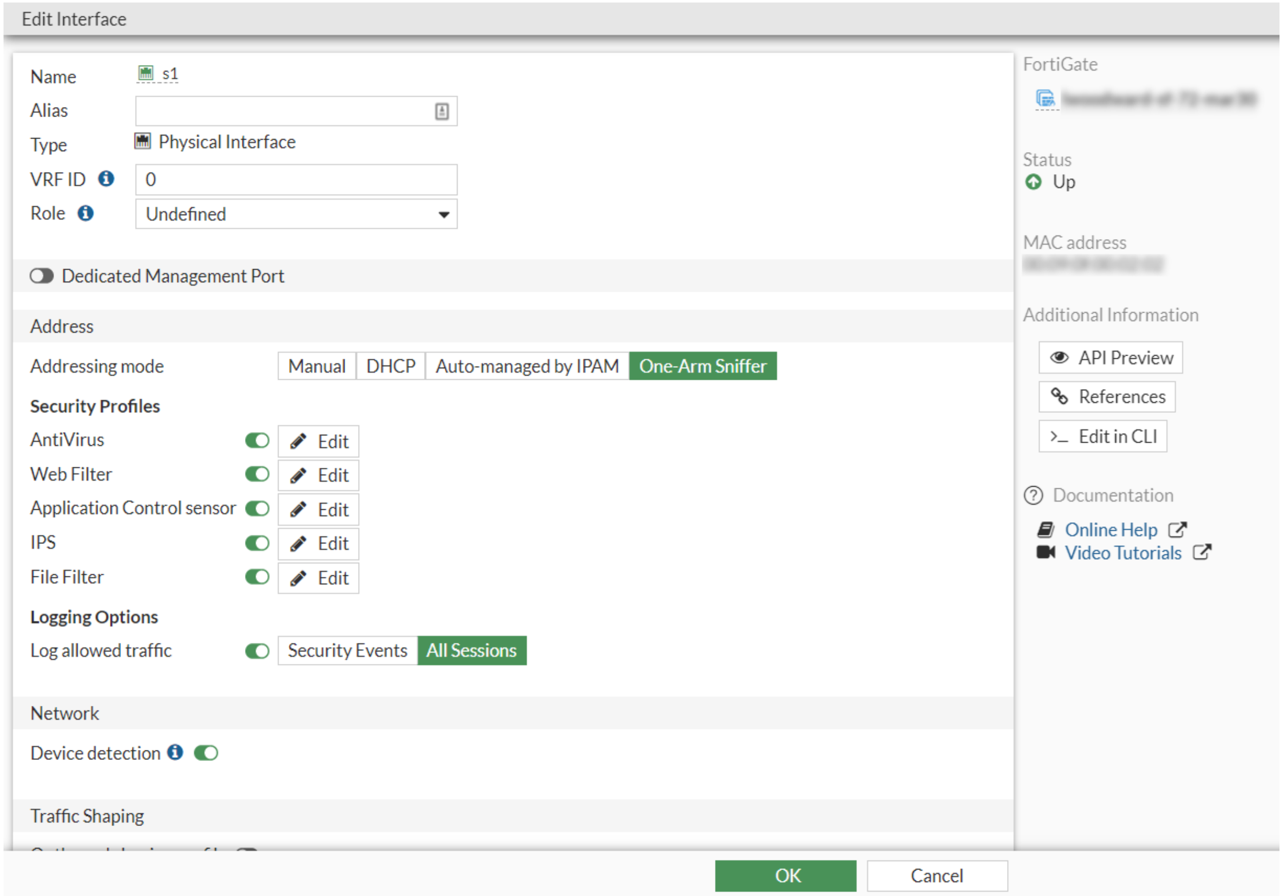
The absence of high CPU usage does not indicate the absence of packet loss. Packet loss may occur due to the capacity of the TAP devices hitting maximum traffic volume during mirroring, or on the FortiGate when the kernel buffer size is exceeded and it is unable to handle bursts of traffic.

Example configuration

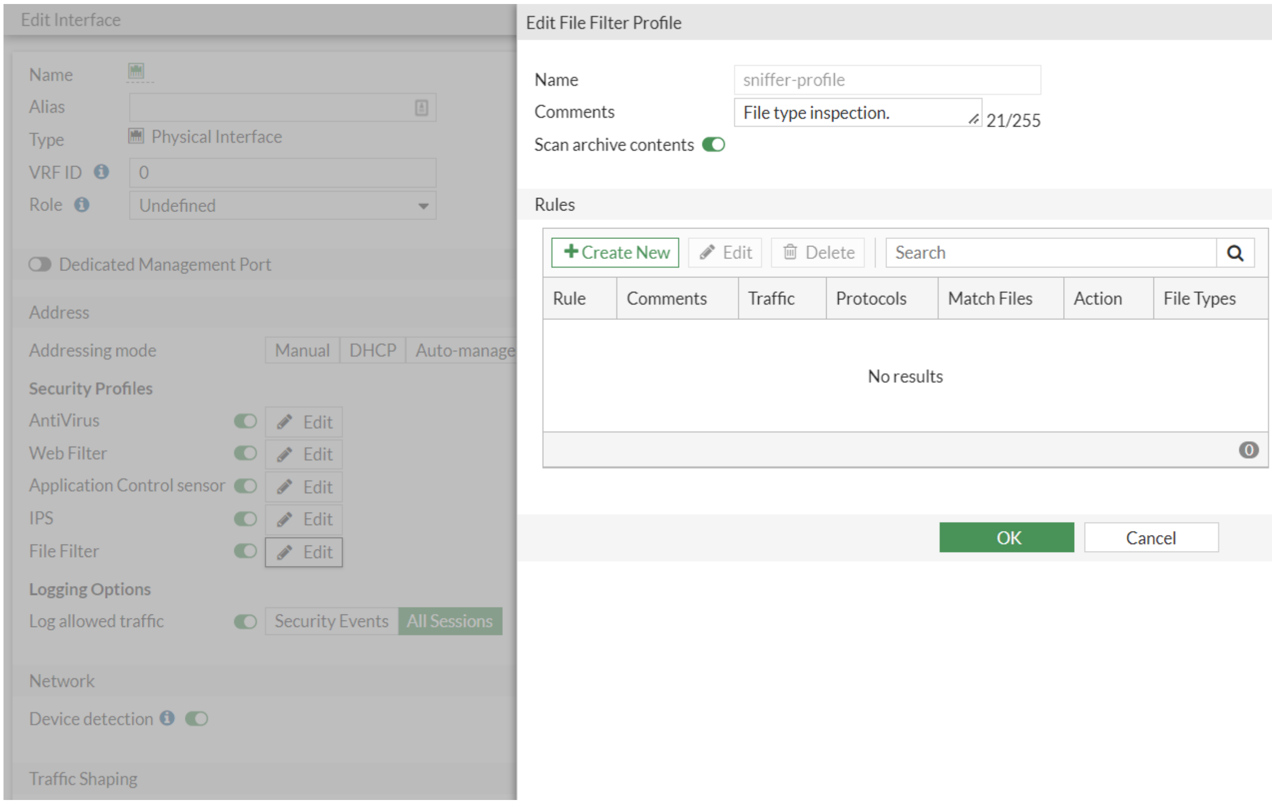
The following example shows how to configure a file filter profile that blocks PDF and RAR files used in a one-arm sniffer policy.

To configure a one-arm sniffer policy in the GUI:

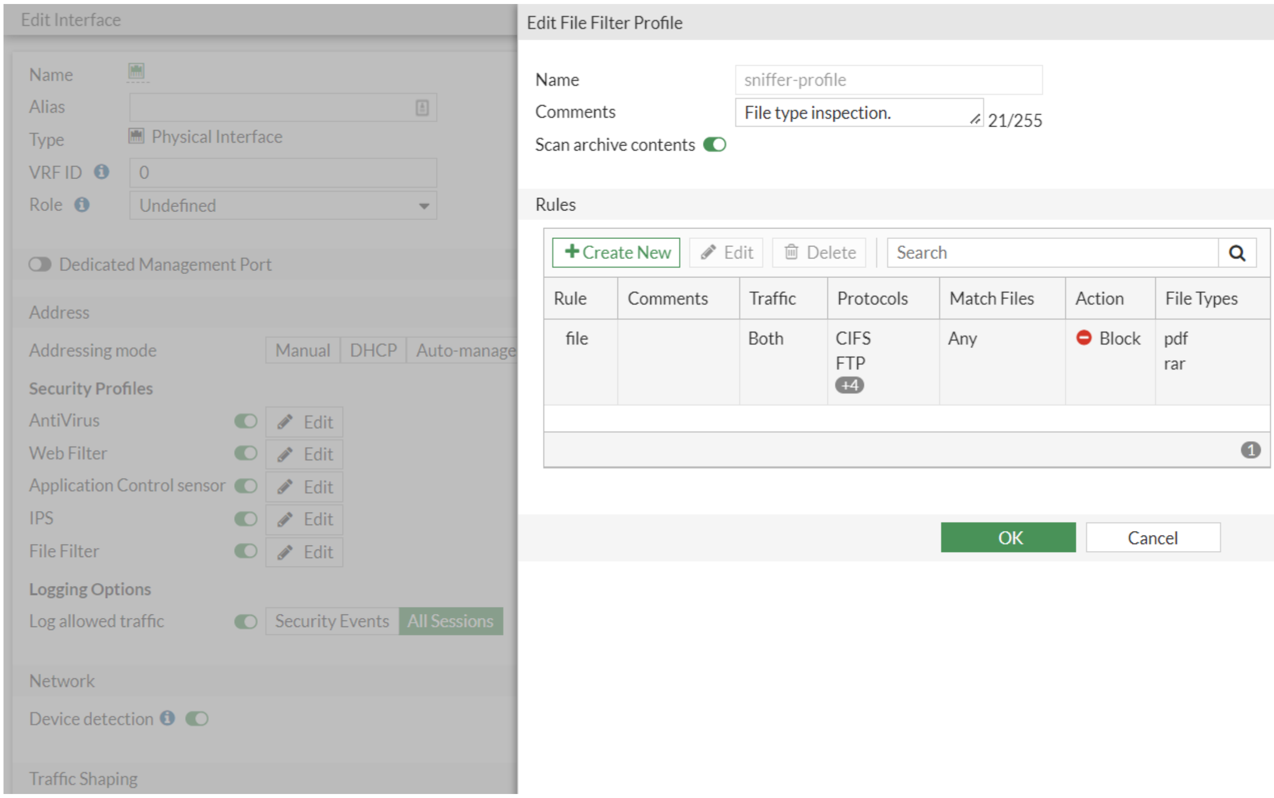
1. Go to *Network > Interfaces* and double-click a physical interface to edit it.
2. For *Role*, select either *LAN*, *DMZ*, or *Undefined*.
3. For *Addressing Mode*, select *One-Arm Sniffer*.



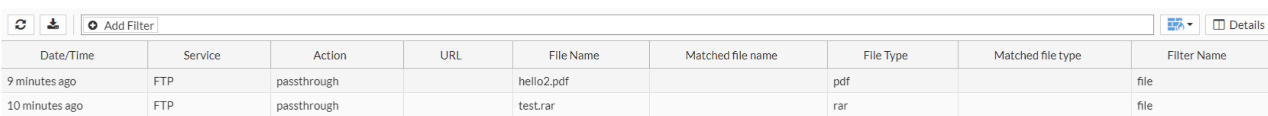
4. In the *Security Profiles* section, enable *File Filter* and click *Edit*. The *Edit File Filter Profile* pane opens.
5. In the *Rules* table, click *Create New*.



6. Configure the rule:
   a. For *File types*, click the + and select *pdf* and *rar*.
   b. For *Action*, select *Block*.
   c. Click *OK* to save the rule.
7. Click *OK* to save the file filter profile.



8. Click *OK* to save the interface settings.
9. Go to *Log & Report > Security Events* to view the *File Filter* logs.



To configure a one-arm sniffer policy in the CLI:

1. Configure the interface:

```
config system interface
    edit "s1"
        set vdom "root"
        set ips-sniffer-mode enable
        set type physical
        set role undefined
        set snmp-index 31
    next
end
```

2. Configure the file filter profile:

```
config file-filter profile
    edit "sniffer-profile"
        set comment "File type inspection."
        config rules
            edit "1"
                set protocol http ftp smtp imap pop3 cifs
```

```
                set action block
                set file-type "pdf" "rar"
            next
        end
    next
end
```

3. Configure the firewall sniffer policy:

```
config firewall sniffer
    edit 1
        set interface "s1"
        set file-filter-profile-status enable
        set file-filter-profile "sniffer-profile"
    next
end
```

4. View the log:

```
# execute log filter category  19
# execute log display
1 logs found.
1 logs returned.

1: date=2020-12-29 time=09:14:46 eventtime=1609262086871379250 tz="-0800"
logid="1900064000" type="utm" subtype="file-filter" eventtype="file-filter"
level="warning" vd="root" policyid=1 sessionid=792 srcip=172.16.200.55 srcport=20
srcintf="s1" srcintfrole="undefined" dstip=10.1.100.11 dstport=56745 dstintf="s1"
dstintfrole="undefined" proto=6 service="FTP" profile="sniffer-profile"
direction="outgoing" action="blocked" filtername="1" filename="hello.pdf" filesize=9539
filetype="pdf" msg="File was blocked by file filter."
```

Interface migration wizard

The *Integrate Interface* option on the *Network* > *Interfaces* page helps migrate a physical port into another interface or interface type such as aggregate, software switch, redundant, zone, or SD-WAN zone. The FortiGate will migrate object references either by replacing the existing instance with the new interface, or deleting the existing instance based on the user's choice. Users can also change the VLAN ID of existing VLAN sub-interface or FortiSwitch VLANs.

The interface migration wizard does not support turning an aggregate, software switch, redundant, zone, or SD-WAN zone interface back into a physical interface.

Integrating an interface

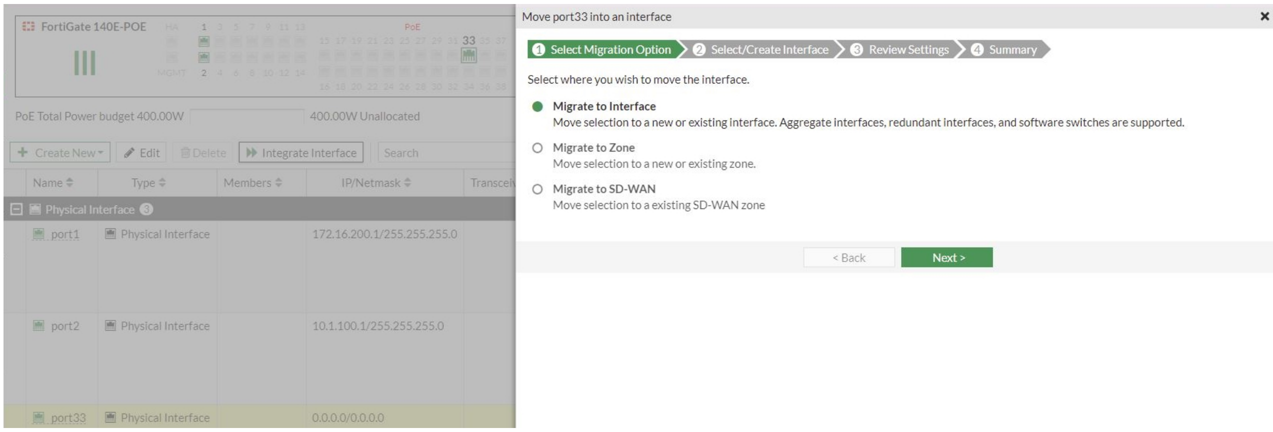
In this example, a DHCP server interface is integrated into a newly created redundant interface, which transfers the DHCP server to a redundant interface.

To integrate an interface:

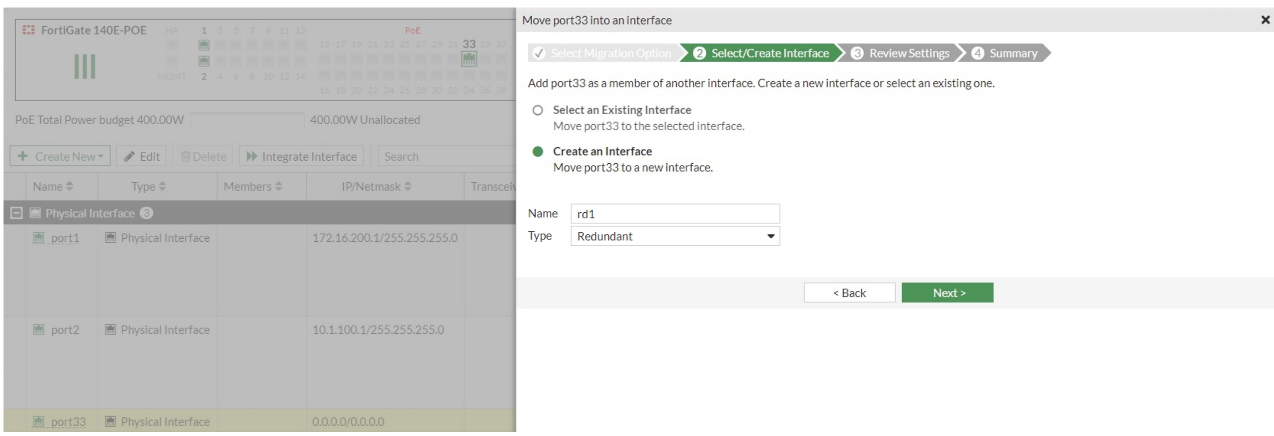
1. Go to *Network* > *Interfaces* and select an interface in the list.
2. Click *Integrate Interface*. The wizard opens.

Alternatively, select an interface in the list. Then right-click and select *Integrate Interface*.

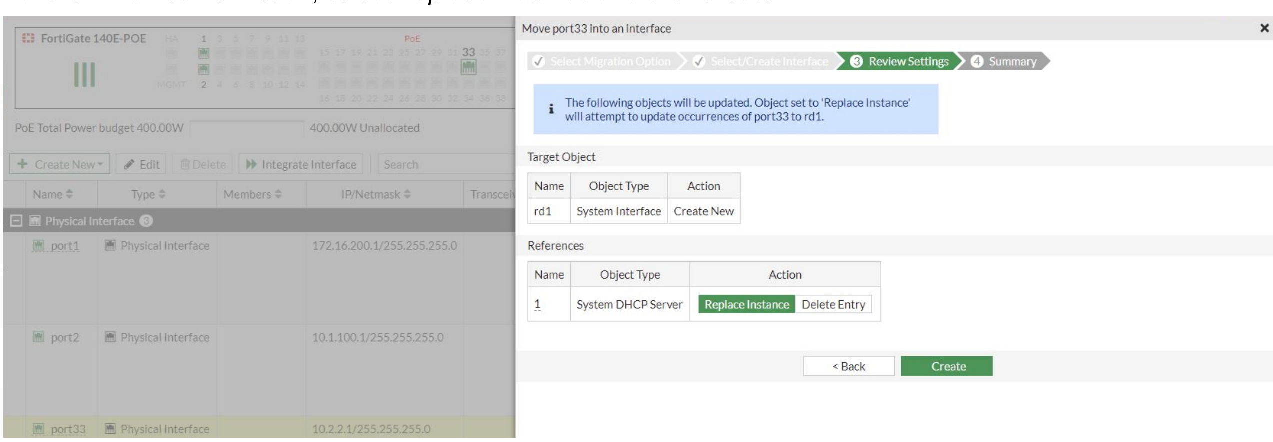
3. Select *Migrate to Interface* and click *Next*.



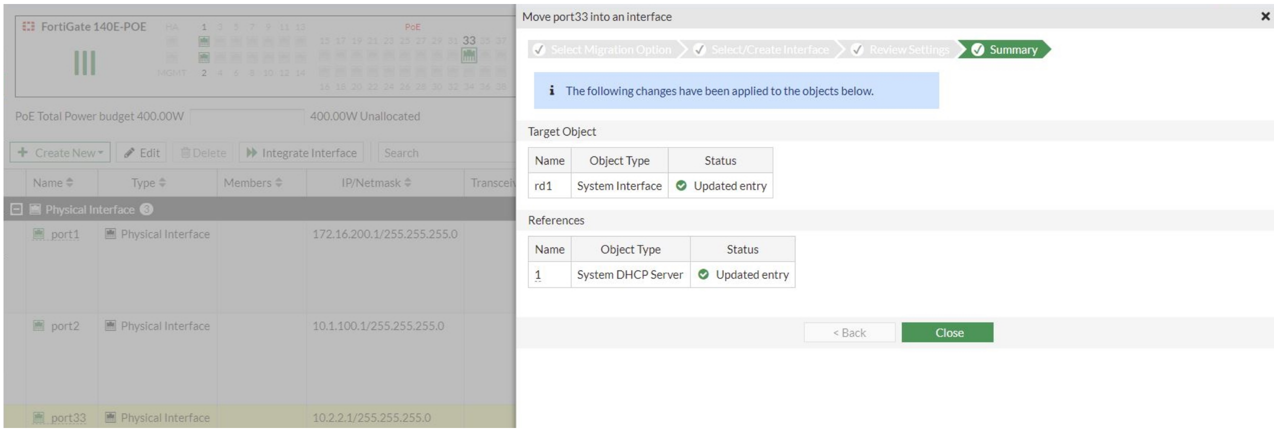
4. Select *Create an Interface*. Enter a name (*rd1*) and set the *Type* to *Redundant*.



5. Click *Next*. The *References* sections lists the associated services with options to *Replace Instance* or *Delete Entry*.
6. For the DHCP server *Action*, select *Replace Instance* and click *Create*.



7. The migration occurs automatically and the statuses for the object and reference change to *Updated entry*. Click *Close*.

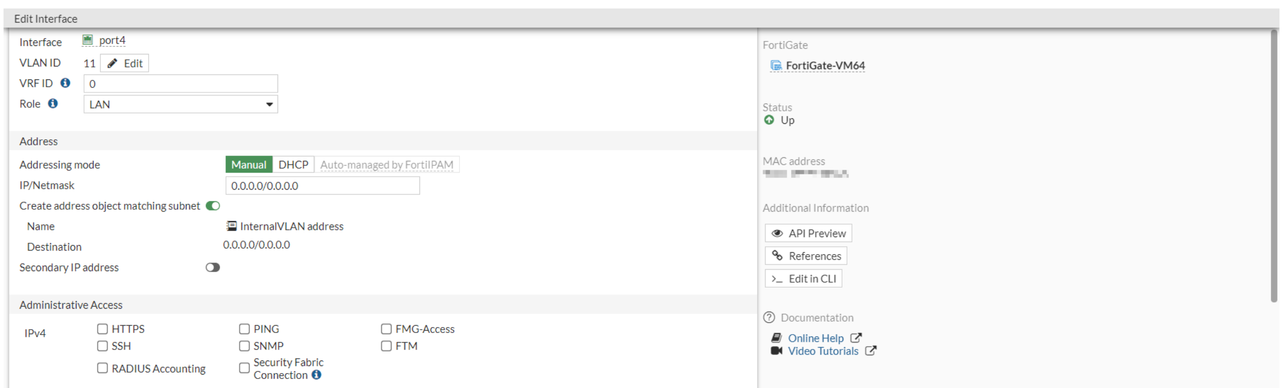


Changing the VLAN ID

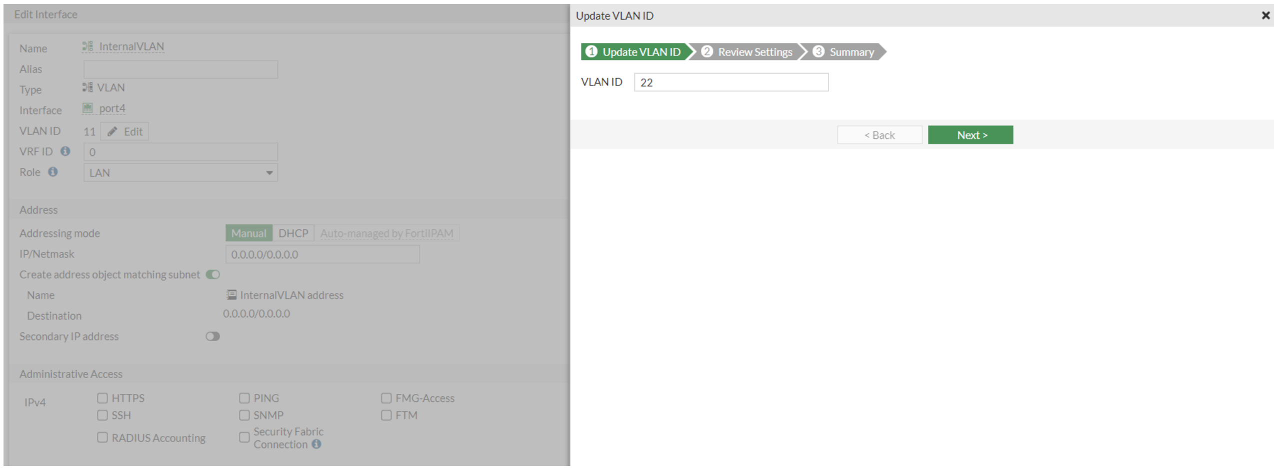
In this example, the VLAN ID of *InternalVLAN* is changed from 11 to 22.

To change the VLAN ID:

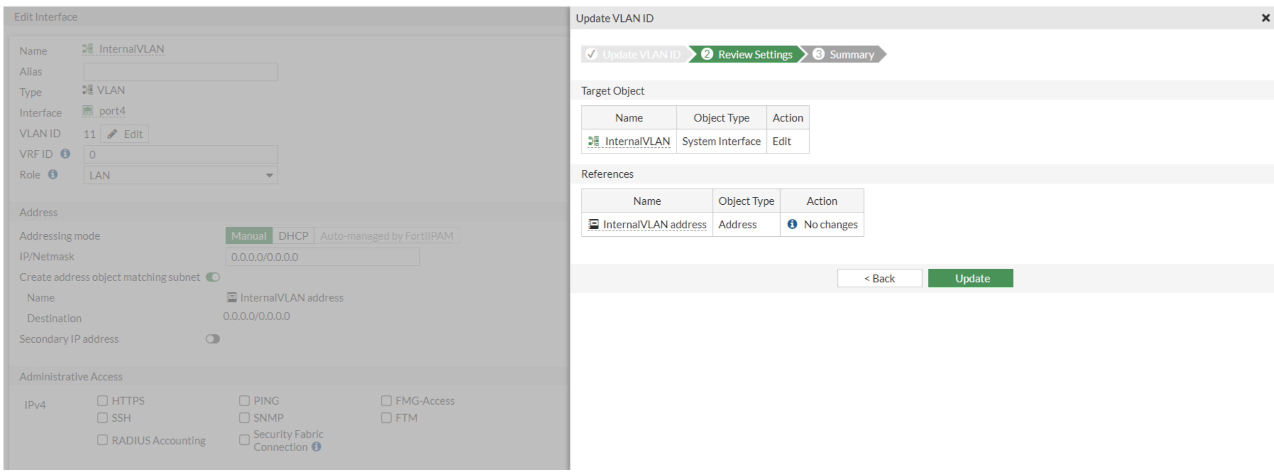
1. Go to *Network > Interfaces* and edit an existing interface.
2. Beside the *VLAN ID* field, click *Edit*. The *Update VLAN ID* window opens.



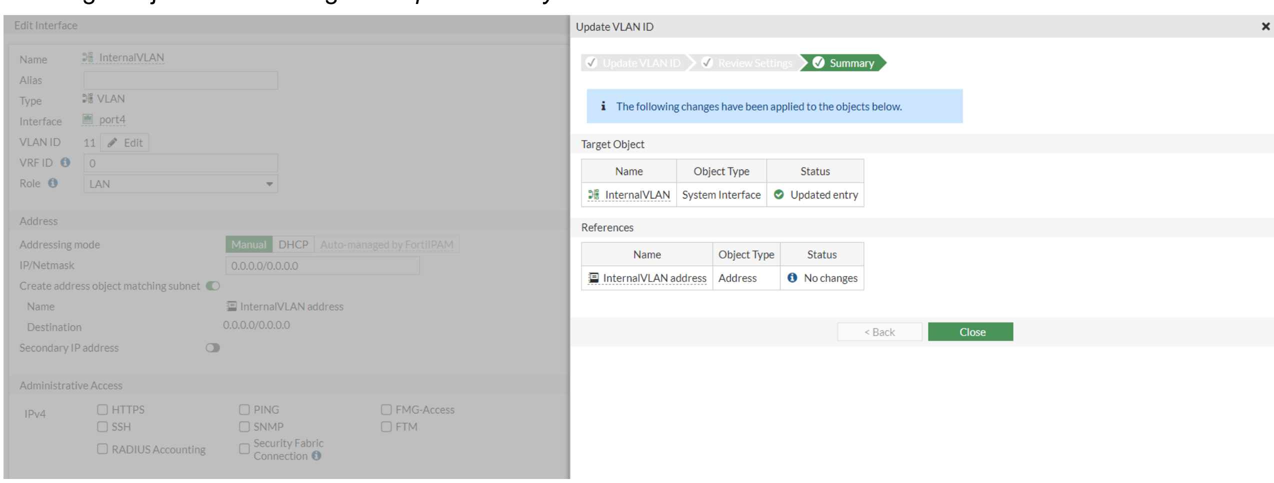
3. Enter the new ID (*22*) and click *Next*.



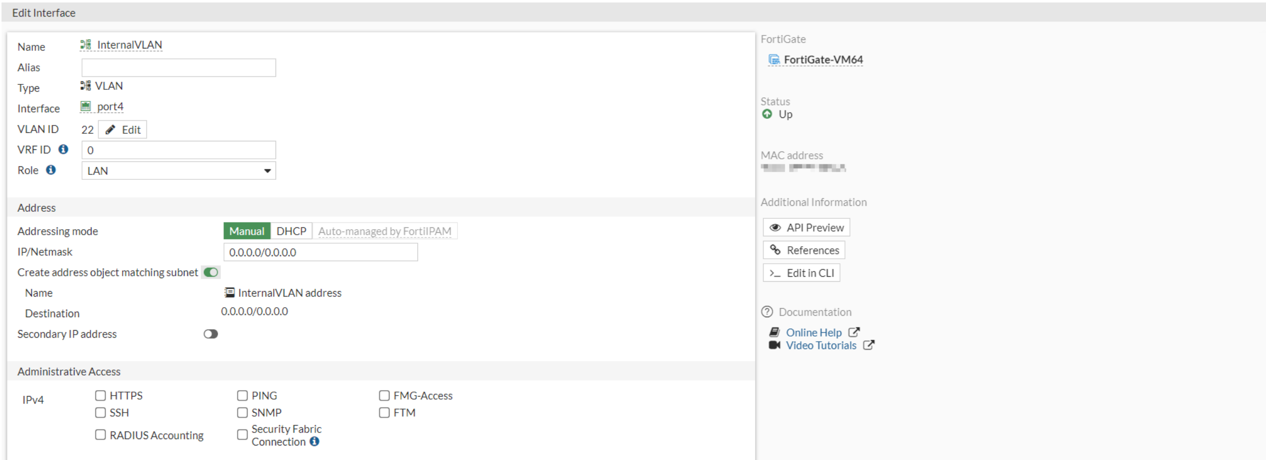
4. Verify the changes, then click *Update* and *OK*.



5. The target object status changes to *Updated entry*. Click *Close*.



In the interface settings, the ID displays as *22*.



Captive portals

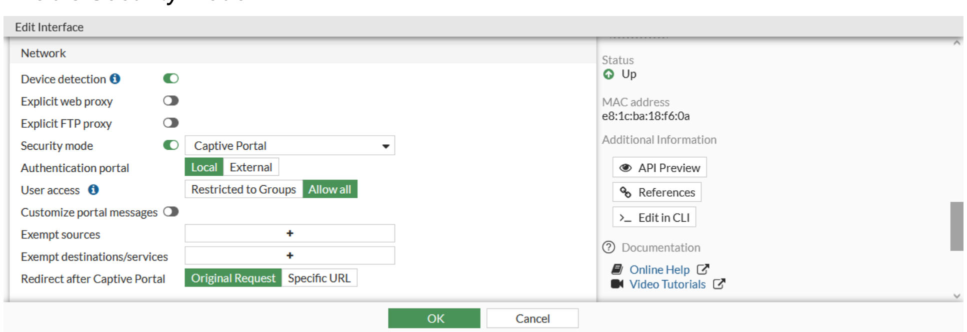
A captive portal is used to enforce authentication before web resources can be accessed. Until a user authenticates successfully, any HTTP request returns the authentication page. After successfully authenticating, a user can access the requested URL and other web resources, as permitted by policies. The captive portal can also be configured to only allow access to members of specific user groups.

Captive portals can be hosted on the FortiGate or an external authentication server. They can be configured on any network interface, including VLAN and WiFi interfaces. On a WiFi interface, the access point appears open, and the client can connect to access point with no security credentials, but then sees the captive portal authentication page. See Captive Portal Security, in the FortiWiFi and FortiAP Configuration Guide for more information.

All users on the interface are required to authenticate. Exemption lists can be created for devices that are unable to authenticate, such as a printer that requires access to the internet for firmware upgrades.

To configure a captive portal in the GUI:

1. Go to *Network > Interfaces* and edit the interface that the users connect to. The interface *Role* must be *LAN* or *Undefined*.
2. Enable *Security mode*.



3. Configure the following settings, then click *OK*.

Authentication Portal	Configure the location of the portal: • *Local*: the portal is hosted on the FortiGate unit.

	• *External*: enter the FQDN or IP address of external portal.
User access	Select if the portal applies to all users, or selected user groups: • *Restricted to Groups*: restrict access to the selected user groups. The *Login page* is shown when a user tries to log in to the captive portal. • *Allow all*: all users can log in, but access will be defined by relevant policies. The *Disclaimer page* is shown when a user tried to log in to the captive portal.
Customize portal messages	Enable to use custom portal pages, then select a replacement message group. See Custom captive portal pages on page 162.
Exempt sources	Select sources that are exempt from the captive portal. Each exemption is added as a rule in an automatically generated exemption list.
Exempt destinations/services	Select destinations and services that are exempt from the captive portal. Each exemption is added as a rule in an automatically generated exemption list.
Redirect after Captive Portal	Configure website redirection after successful captive portal authentication: • *Original Request*: redirect to the initially browsed to URL . • *Specific URL*: redirect to the specified URL.

To configure a captive portal in the CLI:

1. If required, create a security exemption list:

```
config user security-exempt-list
    edit <list>
        config rule
            edit 1
                set srcaddr <source(s)>
                set dstaddr <source(s)>
                set service <service(s)>
            next
            edit 2
                set srcaddr <source(s)>
                set dstaddr <source(s)>
                set service <service(s)>
            next
        end
    next
end
```

2. Configure captive portal authentication on the interface:

```
config system interface
    edit <interface>
        set security-mode {none | captive-portal}
        set security-external-web <string>
        set replacemsg-override-group <group>
        set security-redirect-url <string>
        set security-exempt-list <list>
        set security-groups <group(s)>
```

```
    next
end
```

Custom captive portal pages

Portal pages are HTML files that can be customized to meet user requirements.

Most of the text and some of the HTML in the message can be changed. Tags are enclosed by double percent signs (%%); most of them should not be changed because they might carry information that the FortiGate unit needs. For information about customizing replacement messages, see Modifying replacement messages on page 2134.

The images on the pages can be replaced. For example, your organization's logo can replace the Fortinet logo. For information about uploading and using new images in replacement messages, see Replacement message images on page 2136.

The following pages are used by captive portals:

Login Page	Requests user credentials. The *%%QUESTION%%* tag provides the *Please enter the required information to continue.* text. This page is shown to users that are trying to log in when *User access* is set to *Restricted to Groups*.
Login Failed Page	Reports that incorrect credentials were entered, and requests correct credentials. The *%%FAILED_MESSAGE%%* tag provides the *Firewall authentication failed. Please try again.* text.
Disclaimer Page	A statement of the legal responsibilities of the user and the host organization that the user must agree to before proceeding. This page is shown users that are trying to log in when *User access* is set to *Allow all*.
Declined Disclaimer Page	Shown if the user does not agree to the statement on the Disclaimer page. Access is denied until the user agrees to the disclaimer.

Physical interface

A FortiGate has several physical interfaces that can connect to Ethernet or optical cables. Depending on the FortiGate model, it can have a varying combination of Ethernet, small form-factor pluggable (SFP), and enhanced small form-factor pluggable (SFP+) interfaces.

The port names, as labeled on the FortiGate, appear in the interfaces list on the *Network > Interfaces* page. Hover the cursor over a port to view information, such as the name and the IP address.

Refer to Configuring an interface for basic GUI and CLI configuration steps.

Displaying transceiver status information for SFP and SFP+ interfaces

Transceiver status information for SFP and SFP+ interfaces installed on the FortiGate can be displayed in the GUI and CLI. For example, the type, vendor name, part number, serial number, and port name. The CLI output includes additional information that can be useful for diagnosing transmission problems, such as the temperature, voltage, and optical transmission power.

To view transceiver status information in the GUI:

1. Go to *Network > Interfaces*. The *Transceiver* column is visible in the table, which displays the transceiver vendor name and part number.
2. Hover the cursor over a transceiver to view more information.

To view transceiver status information in the CLI:

```
# get system interface transceiver
Interface port9 - SFP/SFP+
  Vendor Name  :            FINISAR CORP.
  Part No.     :            FCLF-8521-3
  Serial No.   :            PMS***
Interface port10 - Transceiver is not detected.
Interface port11 - SFP/SFP+
  Vendor Name  :            QNC
  Part No.     :            LCP-1250RJ3SRQN
  Serial No.   :            QNDT****
Interface port12 - SFP/SFP+
  Vendor Name  :            QNC
  Part No.     :            LCP-1250RJ3SRQN
  Serial No.   :            QNDT****
Interface s1 - SFP/SFP+
  Vendor Name  :            JDSU
  Part No.     :            PLRXPLSCS4322N
  Serial No.   :            CB26U****
Interface s2 - SFP/SFP+
  Vendor Name  :            JDSU
  Part No.     :            PLRXPLSCS4321N
  Serial No.   :            C825U****
Interface vw1 - Transceiver is not detected.
Interface vw2 - Transceiver is not detected.
Interface x1 - SFP/SFP+
  Vendor Name  :            Fortinet
  Part No.     :            LCP-10GRJ3SRFN
  Serial No.   :            19090910****
Interface x2 - Transceiver is not detected.
                                        Optical      Optical      Optical
SFP/SFP+     Temperature  Voltage      Tx Bias      Tx Power     Rx Power
Interface    (Celsius)    (Volts)      (mA)         (dBm)        (dBm)
------------ ------------ ------------ ------------ ------------ ------------
port9         N/A          N/A          N/A          N/A          N/A
port11        N/A          N/A          N/A          N/A          N/A
port12        N/A          N/A          N/A          N/A          N/A
s1            38.3         3.35         6.80         -2.3         -3.2
s2            42.1         3.34         7.21         -2.3         -3.0
x1            N/A          N/A          N/A          N/A          N/A
  ++ : high alarm, + : high warning, - : low warning, -- : low alarm, ? : suspect.
```

VLAN

Virtual local area networks (VLANs) multiply the capabilities of your FortiGate and can also provide added network security. VLANs use ID tags to logically separate devices on a network into smaller broadcast domains. These smaller domains forward packets only to devices that are part of that VLAN domain. This reduces traffic and increases network security.

VLANs in NAT mode

In NAT mode, the FortiGate unit functions as a layer-3 device. In this mode, the FortiGate unit controls the flow of packets between VLANs and can also remove VLAN tags from incoming VLAN packets. The FortiGate unit can also forward untagged packets to other networks such as the Internet.

In NAT mode, the FortiGate unit supports VLAN trunk links with IEEE 802.1Q-compliant switches or routers. The trunk link transports VLAN-tagged packets between physical subnets or networks. When you add VLAN subinterfaces to the FortiGate's physical interfaces, the VLANs have IDs that match the VLAN IDs of packets on the trunk link. The FortiGate unit directs packets with VLAN IDs to subinterfaces with matching IDs.

You can define VLAN subinterfaces on all FortiGate physical interfaces. However, if multiple virtual domains are configured on the FortiGate unit, you only have access to the physical interfaces on your virtual domain. The FortiGate unit can tag packets leaving on a VLAN subinterface. It can also remove VLAN tags from incoming packets and add a different VLAN tag to outgoing packets.

Normally in VLAN configurations, the FortiGate unit's internal interface is connected to a VLAN trunk, and the external interface connects to an Internet router that is not configured for VLANs. In this configuration, the FortiGate unit can apply different policies for traffic on each VLAN interface connected to the internal interface, which results in less network traffic and better security.

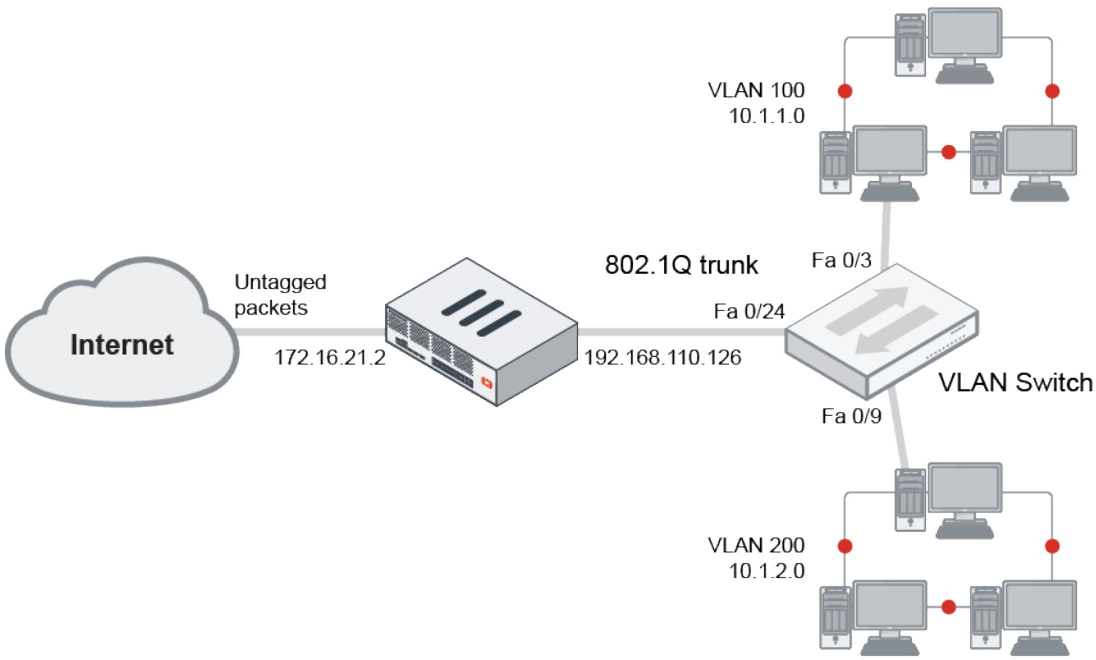
Sample topology

In this example, two different internal VLAN networks share one interface on the FortiGate unit and share the connection to the Internet. This example shows that two networks can have separate traffic streams while sharing a single interface. This configuration can apply to two departments in a single company or to different companies.

There are two different internal network VLANs in this example. VLAN_100 is on the 10.1.1.0/255.255.255.0 subnet, and VLAN_200 is on the 10.1.2.0/255.255.255.0 subnet. These VLANs are connected to the VLAN switch.

The FortiGate internal interface connects to the VLAN switch through an 802.1Q trunk. The internal interface has an IP address of 192.168.110.126 and is configured with two VLAN subinterfaces (VLAN_100 and VLAN_200). The external interface has an IP address of 172.16.21.2 and connects to the Internet. The external interface has no VLAN subinterfaces.

When the VLAN switch receives packets from VLAN_100 and VLAN_200, it applies VLAN ID tags and forwards the packets of each VLAN both to local ports and to the FortiGate unit across the trunk link. The FortiGate unit has policies that allow traffic to flow between the VLANs, and from the VLANs to the external network.



Sample configuration

In this example, both the FortiGate unit and the Cisco 2950 switch are installed and connected and basic configuration has been completed. On the switch, you need access to the CLI to enter commands. No VDOMs are enabled in this example.

General configuration steps include:

1. Configure the external interface.
2. Add two VLAN subinterfaces to the internal network interface.
3. Add firewall addresses and address ranges for the internal and external networks.
4. Add security policies to allow:
   - the VLAN networks to access each other.
   - the VLAN networks to access the external network.

To configure the external interface:

```
config system interface
    edit external
        set mode static
        set ip 172.16.21.2 255.255.255.0
    next
end
```

To add VLAN subinterfaces:

```
config system interface
    edit VLAN_100
        set vdom root
        set interface internal
        set type vlan
```

```
        set vlanid 100
        set mode static
        set ip 10.1.1.1 255.255.255.0
        set allowaccess https ping
    next
    edit VLAN_200
        set vdom root
        set interface internal
        set type vlan
        set vlanid 200
        set mode static
        set ip 10.1.2.1 255.255.255.0
        set allowaccess https ping
    next
end
```

To add the firewall addresses:

```
config firewall address
    edit VLAN_100_Net
        set type ipmask
        set subnet 10.1.1.0 255.255.255.0
    next
    edit VLAN_200_Net
        set type ipmask
        set subnet 10.1.2.0 255.255.255.0
    next
end
```

To add security policies:

Policies 1 and 2 do not need NAT enabled, but policies 3 and 4 do need NAT enabled.

```
config firewall policy
    edit 1
        set srcintf VLAN_100
        set srcaddr VLAN_100_Net
        set dstintf VLAN_200
        set dstaddr VLAN_200_Net
        set schedule always
        set service ALL
        set action accept
        set nat disable
        set status enable
    next
    edit 2
        set srcintf VLAN_200
        set srcaddr VLAN_200_Net
        set dstintf VLAN_100
        set dstaddr VLAN_100_Net
        set schedule always
        set service ALL
        set action accept
        set nat disable
        set status enable
    next
```

```
    edit 3
        set srcintf VLAN_100
        set srcaddr VLAN_100_Net
        set dstintf external
        set dstaddr all
        set schedule always
        set service ALL
        set action accept
        set nat enable
        set status enable
    next
    edit 4
        set srcintf VLAN_200
        set srcaddr VLAN_200_Net
        set dstintf external
        set dstaddr all
        set schedule always
        set service ALL
        set action accept
        set nat enable
        set status enable
    next
end
```

VLANs in transparent mode

In transparent mode, the FortiGate unit behaves like a layer-2 bridge but can still provide services such as antivirus scanning, web filtering, spam filtering, and intrusion protection to traffic. Some limitations of transparent mode is that you cannot use SSL VPN, PPTP/L2TP VPN, DHCP server, or easily perform NAT on traffic. The limits in transparent mode apply to IEEE 802.1Q VLAN trunks passing through the unit.

You can insert the FortiGate unit operating in transparent mode into the VLAN trunk without making changes to your network. In a typical configuration, the FortiGate unit internal interface accepts VLAN packets on a VLAN trunk from a VLAN switch or router connected to internal network VLANs. The FortiGate external interface forwards VLAN-tagged packets through another VLAN trunk to an external VLAN switch or router and on to external networks such as the Internet. You can configure the unit to apply different policies for traffic on each VLAN in the trunk.

To pass VLAN traffic through the FortiGate unit, you add two VLAN subinterfaces with the same VLAN ID, one to the internal interface and the other to the external interface. You then create a security policy to permit packets to flow from the internal VLAN interface to the external VLAN interface. If required, create another security policy to permit packets to flow from the external VLAN interface to the internal VLAN interface. Typically in transparent mode, you do not permit packets to move between different VLANs. Network protection features such as spam filtering, web filtering, and anti-virus scanning, are applied through the UTM profiles specified in each security policy, enabling very detailed control over traffic.

When the FortiGate unit receives a VLAN-tagged packet on a physical interface, it directs the packet to the VLAN subinterface with the matching VLAN ID. The VLAN tag is removed from the packet and the FortiGate unit then applies security policies using the same method it uses for non-VLAN packets. If the packet exits the FortiGate unit through a VLAN subinterface, the VLAN ID for that subinterface is added to the packet and the packet is sent to the corresponding physical interface.

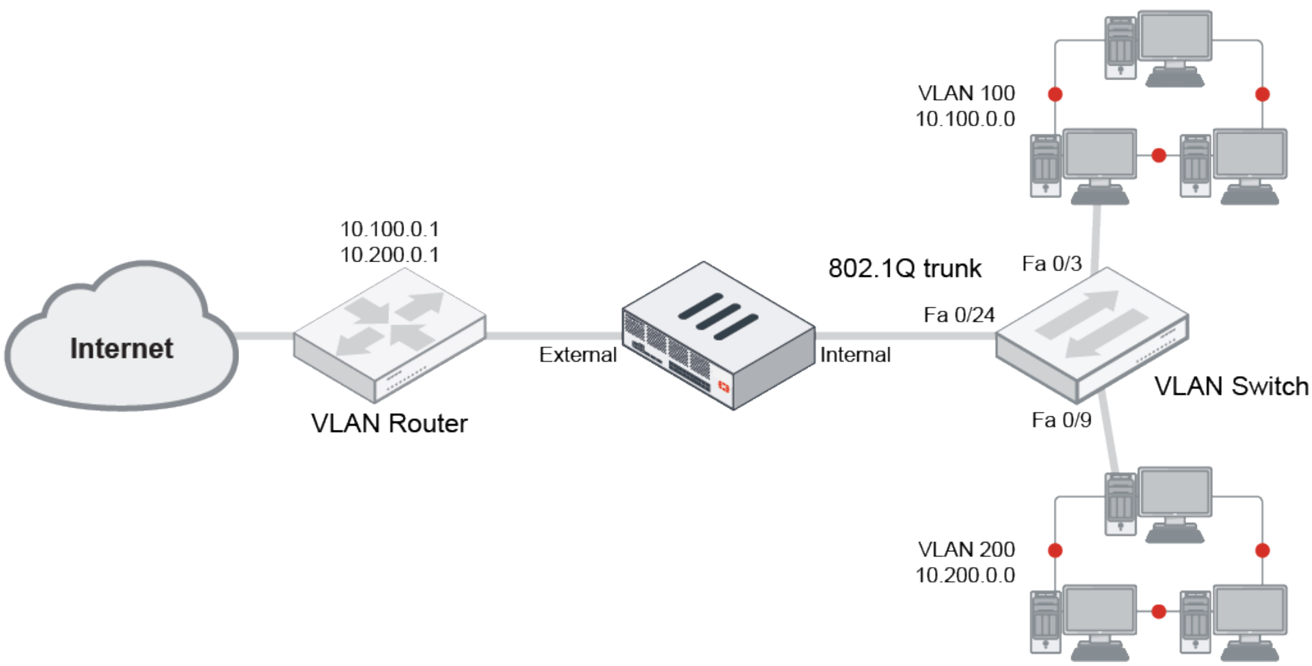
Sample topology

In this example, the FortiGate unit is operating in transparent mode and is configured with two VLANs: one with an ID of 100 and the other with ID 200. The internal and external physical interfaces each have two VLAN subinterfaces, one for VLAN_100 and one for VLAN_200.

The IP range for the internal VLAN_100 network is 10.100.0.0/255.255.0.0, and for the internal VLAN_200 network is 10.200.0.0/255.255.0.0.

The internal networks are connected to a Cisco 2950 VLAN switch which combines traffic from the two VLANs onto one in the FortiGate unit's internal interface. The VLAN traffic leaves the FortiGate unit on the external network interface, goes on to the VLAN switch, and on to the Internet. When the FortiGate units receives a tagged packet, it directs it from the incoming VLAN subinterface to the outgoing VLAN subinterface for that VLAN.

In this example, we create a VLAN subinterface on the internal interface and another one on the external interface, both with the same VLAN ID. Then we create security policies that allow packets to travel between the VLAN_100_int interface and the VLAN_100_ext interface. Two policies are required: one for each direction of traffic. The same is required between the VLAN_200_int interface and the VLAN_200_ext interface, for a total of four security policies.



Sample configuration

There are two main steps to configure your FortiGate unit to work with VLANs in transparent mode:

1. Add VLAN subinterfaces.
2. Add security policies.

You can also configure the protection profiles that manage antivirus scanning, web filtering, and spam filtering.

To add VLAN subinterfaces:

```
config system interface
    edit VLAN_100_int
        set type vlan
```

```
        set interface internal
        set vlanid 100
    next
    edit VLAN_100_ext
        set type vlan
        set interface external
        set vlanid 100
    next
    edit VLAN_200_int
        set type vlan
        set interface internal
        set vlanid 200
    next
    edit VLAN_200_ext
        set type vlan
        set interface external
        set vlanid 200
    next
end
```

To add security policies:

```
config firewall policy
    edit 1
        set srcintf VLAN_100_int
        set srcaddr all
        set dstintf VLAN_100_ext
        set dstaddr all
        set action accept
        set schedule always
        set service ALL
    next
    edit 2
        set srcintf VLAN_100_ext
        set srcaddr all
        set dstintf VLAN_100_int
        set dstaddr all
        set action accept
        set schedule always
        set service ALL
    next
    edit 3
        set srcintf VLAN_200_int
        set srcaddr all
        set dstintf VLAN_200_ext
        set dstaddr all
        set action accept
        set schedule always
        set service ALL
    next
    edit 4
        set srcintf VLAN_200_ext
        set srcaddr all
        set dstintf VLAN_200_int
        set dstaddr all
        set action accept
```

```
        set schedule always
        set service ALL
    next
end
```

Virtual VLAN switch

The hardware switch ports on FortiGate models that support virtual VLAN switches can be used as a layer 2 switch. Virtual VLAN switch mode allows 802.1Q VLANs to be assigned to ports, and the configuration of one interface as a trunk port.

The following FortiGate series are supported in FortiOS 7.2: 60F, 80F, 100E, 100F, 140E, 200F, 300E, 400E, and 1100E.

The `virtual-switch-vlan` option must be enabled in the CLI to configure VLAN switch mode from the GUI or CLI.

To enable VLAN switches:

```
config system global
    set virtual-switch-vlan enable
end
```

After this setting is enabled, any previously configured hardware switches will appear in the *Network > Interfaces* page under *VLAN Switch*.

To enable VLAN switch mode in the GUI:

1. Go to *System > Settings*.
2. In the *View Settings* section, enable *VLAN switch mode*.
3. Click *Apply*.

Basic configurations

Hardware switch ports can be configured as either a VLAN switch port or a trunk port. The available interfaces and allowable VLAN IDs that can be used depend on the FortiGate model. It is recommended to remove ports from the default VLAN switch before you begin configurations.

To create a new VLAN and assign ports in the GUI:

1. Go to *Network > Interfaces* and click *Create New > Interface*.
2. Enter a name and configure the following:
    a. Set the *Type* to *VLAN Switch*.
    b. Enter a *VLAN ID*.
    c. Click the + and add the *Interface Members*.
    d. Configure the *Address* and *Administrative Access* settings as needed.
3. Click *OK*.

To create a new VLAN and assign ports in the CLI:

1. Configure the VLAN:

```
config system virtual-switch
    edit "VLAN10"
        set physical-switch "sw0"
        set vlan 10
        config port
            edit "internal1"
            next
            edit "internal2"
            next
        end
    next
end
```

2. Configure the VLAN switch interface addressing:

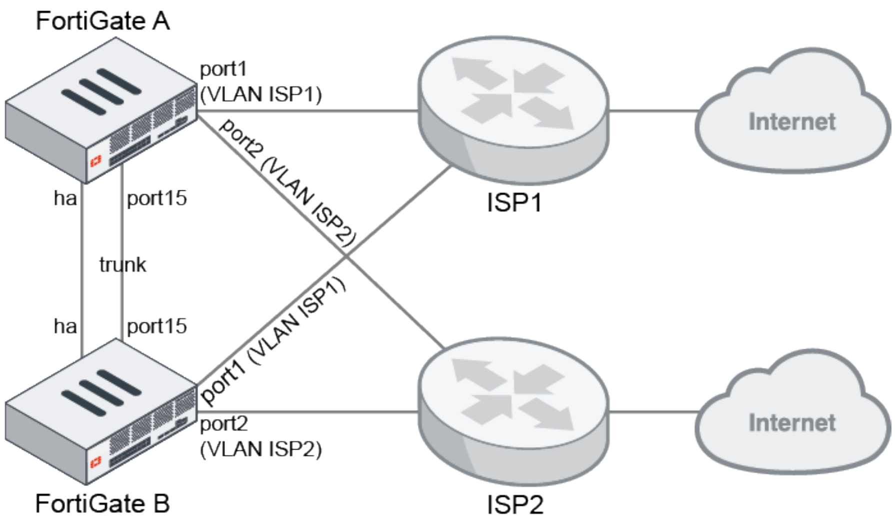
```
config system interface
    edit "VLAN10"
        set vdom "root"
        set ip 192.168.10.99 255.255.255.0
        set allowaccess ping https ssh snmp http fgfm
        set type hard-switch
    next
end
```

To designate an interface as a trunk port:

```
config system interface
    edit internal5
        set trunk enable
    next
end
```

Example 1: HA using a VLAN switch

In this example, two FortiGates in an HA cluster are connected to two ISP routers. Instead of connecting to external L2 switches, each FortiGate connects to each ISP router on the same hardware switch port on the same VLAN. A trunk port connects the two FortiGates to deliver the 802.1Q tagged traffic to the other. A full mesh between the FortiGate cluster and the ISP routers is achieved where no single point of failure will cause traffic disruptions.



This example assumes that the HA settings are already configured. The interface and VLAN switch settings are identical between cluster members and synchronized. See HA using a hardware switch to replace a physical switch on page 2059 for a similar example that does not use a VLAN switch.

To configure the VLAN switches:

1. Configure the ISP interfaces with the corresponding VLAN IDs:

```
config system virtual-switch
    edit "ISP1"
        set physical-switch "sw0"
        set vlan 2951
        config port
            edit "port1"
            next
        end
    next
    edit "ISP2"
        set physical-switch "sw0"
        set vlan 2952
        config port
            edit "port2"
            next
        end
    next
end
```

2. Configure the VLAN switch interface addressing:

```
config system interface
    edit "ISP1"
        set vdom "root"
        set ip 192.168.10.99 255.255.255.0
        set allowaccess ping
        set type hard-switch
    next
    edit "ISP2"
        set vdom "root"
```

```
        set ip 192.168.20.99 255.255.255.0
        set allowaccess ping
        set type hard-switch
    next
end
```

3. Designate port15 as the trunk port:

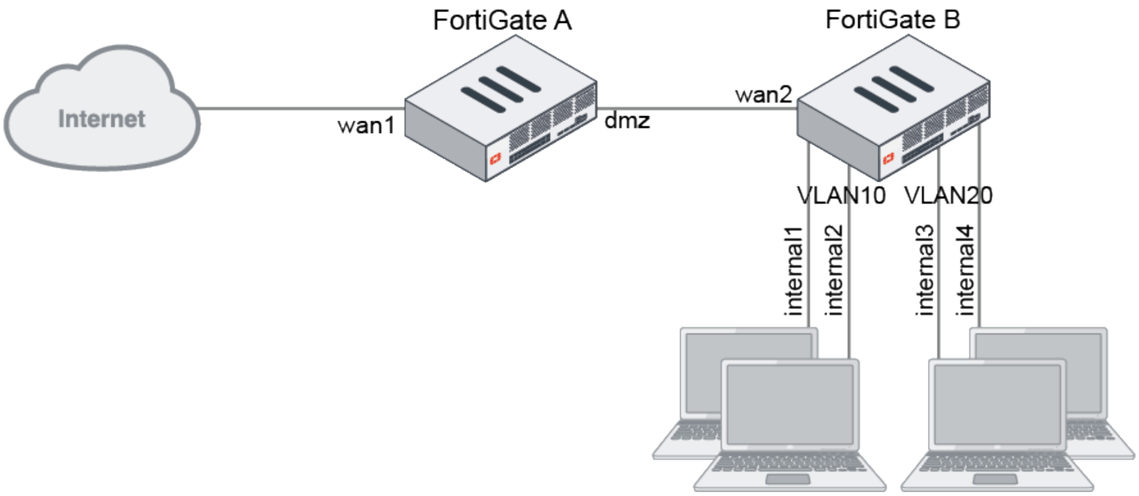
```
config system interface
    edit port15
        set trunk enable
    next
end
```

4. Configure firewall policies to allow outgoing traffic on the ISP1 and ISP2 interfaces:

```
config firewall policy
    edit 1
        set srcintf "port11"
        set dstintf "ISP1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set nat enable
    next
    edit 2
        set srcintf "port11"
        set dstintf "ISP2"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set nat enable
    next
end
```

Example 2: LAN extension

In this example, two hardware switch ports are assigned VLAN10, and two ports are assigned VLAN20 on FortiGate B. The wan2 interface is designated as the trunk port, and is connected to the upstream FortiGate A. The corresponding VLAN subinterfaces VLAN10 and VLAN20 on the upstream FortiGate allow further access to other networks.



The available interfaces and VLAN IDs varies between FortiGate models. The FortiGate B in this example is a 60F model.

To configure FortiGate B:

1. Configure the VLAN interfaces:

```
config system virtual-switch
    edit "VLAN10"
        set physical-switch "sw0"
        set vlan 10
        config port
            edit "internal1"
            next
            edit "internal2"
            next
        end
    next
    edit "VLAN20"
        set physical-switch "sw0"
        set vlan 20
        config port
            edit "internal3"
            next
            edit "internal4"
            next
        end
    next
end
```

2. Configure the VLAN switch interface addressing:

```
config system interface
    edit "VLAN10"
        set vdom "root"
        set ip 192.168.10.99 255.255.255.0
        set allowaccess ping https ssh snmp http fgfm
```

```
        set type hard-switch
    next
    edit "VLAN20"
        set vdom "root"
        set ip 192.168.20.99 255.255.255.0
        set allowaccess ping https ssh snmp http fgfm
        set type hard-switch
    next
end
```

3. Designate wan2 as the trunk port:

```
config system interface
    edit wan2
        set trunk enable
    next
end
```

To configure FortiGate A:

1. Configure the VLAN subinterfaces:

```
config system interface
    edit "VLAN10"
        set ip 192.168.10.98 255.255.255.0
        set allowaccess ping https ssh
        set role lan
        set interface "dmz"
        set vlanid 10
    next
    edit "VLAN20"
        set ip 192.168.20.98 255.255.255.0
        set allowaccess ping https ssh
        set role lan
        set interface "dmz"
        set vlanid 20
    next
end
```

2. Configure the DHCP server on VLAN10:

```
config system dhcp server
    edit 0
        set dns-service default
        set default-gateway 192.168.10.98
        set netmask 255.255.255.0
        set interface "VLAN10 "
        config ip-range
            edit 1
                set start-ip 192.168.10.100
                set end-ip 192.168.10.254
            next
        end
        set timezone-option default
    next
end
```

3. Configure firewall policies that allow traffic from the VLAN10 and VLAN20 interfaces to the internet:

```
config firewall policy
    edit 0
        set name "VLAN10-out"
        set srcintf "VLAN10"
        set dstintf "wan1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set nat enable
    next
    edit 0
        set name "VLAN20-out"
        set srcintf "VLAN20"
        set dstintf "wan1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set nat enable
    next
end
```

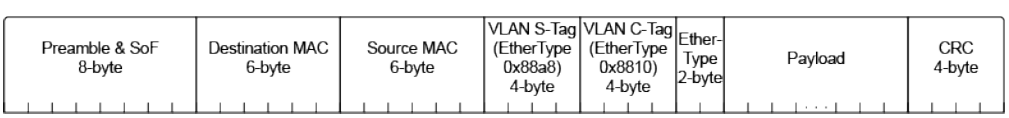
To test the connection:

1. Connect a PC to internal1 on FortiGate B.
2. Verify that it receives an IP address from FortiGate A’s DHCP server.
3. From the PC, ping FortiGate B on 192.168.10.99.
4. Ping FortiGate A on 192.168.10.98.
5. Connect to the internet. Traffic is allowed by the VLAN10-out policy.

QinQ 802.1Q in 802.1ad

QinQ (802.1ad) allows multiple VLAN tags to be inserted into a single frame, and can be configured on supported FortiGate devices.

In this example, the customer connects to a provider that uses 802.1ad double-tagging to separate their customer VLANs. The FortiGate connecting to the provider double-tags its frames with an outer provider-tag (S-Tag) and an inner customer-tag (C-Tag).



The customer identifies itself with the provider-tag (S-Tag) 232 and uses the customer-tag (C-Tag) 444 for traffic to its VLAN.

To configure the interfaces:

1. Configure the interface to the provider that uses the outer tag (S-Tag):

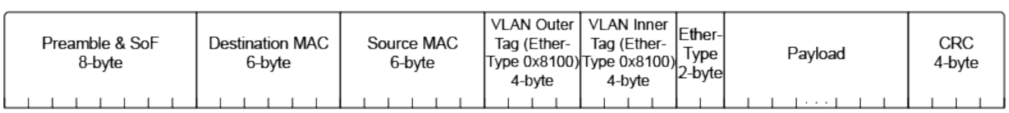
```
config system interface
    edit "vlan-8021ad"
        set vdom "root"
        set vlan-protocol 8021ad
        set device-identification enable
        set role lan
        set snmp-index 47
        set interface "PORT"
        set vlanid 232
    next
end
```

2. Configure a dynamic VLAN interface that uses the inner tag (C-Tag):

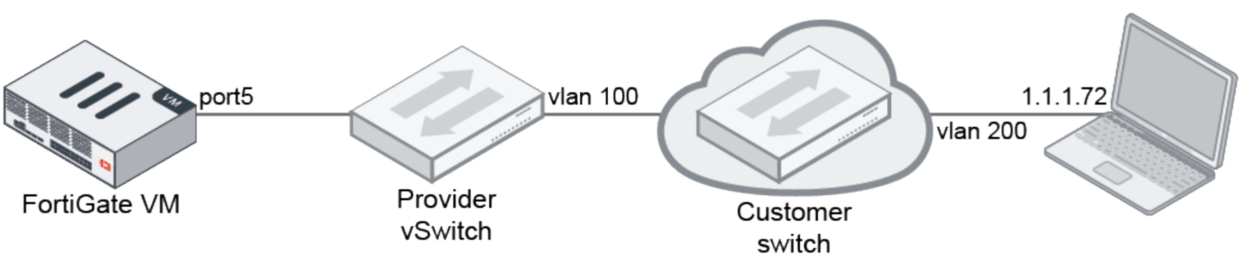
```
config system interface
    edit "DVLAN"
        set vdom "vdom1"
        set device-identification enable
        set role lan
        set snmp-index 48
        set interface "vlan-8021ad"
        set vlanid 444
    next
end
```

QinQ 802.1Q in 802.1Q

QinQ (802.1Q in 802.1Q) is supported for FortiGate VM models, where multiple VLAN tags can be inserted into a single frame.



In this example, the FortiGate VM is connected to a provider vSwitch and then a customer switch. The FortiGate encapsulates the frame with an outer 802.1Q tag of VLAN 100 and an inner 802.1Q tag of VLAN 200; port5 is used as the physical port. The provider vSwitch strips the outer tag and forwards traffic to the appropriate customer. Then the customer switch strips the inner tag and forwards the packet to the appropriate customer VLAN.



To configure the interfaces:

1. Configure the interface to the provider that uses the outer tag:

```
config system interface
    edit "vlan-8021q"
        set vdom "root"
        set device-identification enable
        set role lan
        set interface "port5"
        set vlan-protocol 8021q
        set vlanid 100
    next
end
```

2. Configure the interface to the provider that uses the inner tag:

```
config system interface
    edit "vlan-qinq8021q"
        set vdom "root"
        set ip 1.1.1.71 255.255.255.0
        set allowaccess ping https ssh snmp http
        set device-identification enable
        set role lan
        set interface "vlan-8021q"
        set vlanid 200
    next
end
```

To verify the traffic:

1. From the FortiGate, ping 1.1.1.72:

```
# execute ping 1.1.1.72
PING 1.1.1.72 (1.1.1.72): 56 data bytes
64 bytes from 1.1.1.72: icmp_seq=0 ttl=255 time=0.2 ms
64 bytes from 1.1.1.72: icmp_seq=1 ttl=255 time=0.1 ms
64 bytes from 1.1.1.72: icmp_seq=2 ttl=255 time=0.1 ms
64 bytes from 1.1.1.72: icmp_seq=3 ttl=255 time=0.1 ms
^C
--- 1.1.1.72 ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max = 0.1/0.1/0.2 ms
```

2. Verify the packet capture frame header output captured from the FortiGate's port5:

```
Frame 2: 106 bytes on wire (848 bits), 106 bytes captured (848 bits)
Ethernet II, Src: VMware_93:ae:8f (00:50:56:93:ae:8f), Dst: VMware_93:e3:72
(00:50:56:93:e3:72)
    Destination: VMware_93:e3:72 (00:50:56:93:e3:72)
    Source: VMware_93:ae:8f (00:50:56:93:ae:8f)
    Type: 802.1Q Virtual LAN (0x8100)
802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 100
    000. .... .... .... = Priority: Best Effort (default) (0)
    ...0 .... .... .... = DEI: Ineligible
    .... 0000 0110 0100 = ID: 100
    Type: 802.1Q Virtual LAN (0x8100)
802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 200
```

```
        000. .... .... .... = Priority: Best Effort (default) (0)
        ...0 .... .... .... = DEI: Ineligible
        .... 0000 1100 1000 = ID: 200
        Type: IPv4 (0x0800)
    Internet Protocol Version 4, Src: 1.1.1.71, Dst: 1.1.1.72
    Internet Control Message Protocol
```

The outer tag (first tag) is an 802.1Q tag with VLAN ID 100. The inner tag (second tag) is also an 802.1Q tag with VLAN ID 200.

Aggregation and redundancy

Link aggregation (IEEE 802.3ad) enables you to bind two or more physical interfaces together to form an aggregated (combined) link. This new link has the bandwidth of all the links combined. If a link in the group fails, traffic is transferred automatically to the remaining interfaces. The only noticeable effect is reduced bandwidth.

This feature is similar to redundant interfaces. The major difference is a redundant interface group only uses one link at a time, where an aggregate link group uses the total bandwidth of the functioning links in the group, up to eight (or more).

An interface is available to be an aggregate interface if:

- It is a physical interface and not a VLAN interface or subinterface.
- It is not already part of an aggregate or redundant interface.
- It is in the same VDOM as the aggregated interface. Aggregate ports cannot span multiple VDOMs.
- It does not have an IP address and is not configured for DHCP or PPPoE.
- It is not referenced in any security policy, VIP, IP Pool, or multicast policy.
- It is not an HA heartbeat interface.
- It is not one of the FortiGate-5000 series backplane interfaces.

When an interface is included in an aggregate interface, it is not listed on the *Network > Interfaces* page. Interfaces still appear in the CLI although configuration for those interfaces do not take affect. You cannot configure the interface individually and it is not available for inclusion in security policies, VIPs, IP pools, or routing.

Example configuration

This example creates an aggregate interface on a FortiGate-140D POE using ports 3-5 with an internal IP address of 10.1.1.123, as well as the administrative access to HTTPS and SSH.

To create an aggregate interface in the GUI:

1. Go to *Network > Interfaces* and select *Create New > Interface*.
2. Set *Name* to *aggregate*.
3. Set *Type* to *802.3ad Aggregate*.
4. Set *Interface members* to *port4*, *port5*, and *port6*.
5. Set *Addressing mode* to *Manual*.
6. Set *IP/Netmask* to *10.1.1.123/24*.
7. For *Administrative Access*, select *HTTPS* and *SSH*.
8. Click *OK*.

To create an aggregate interface in the CLI:

```
config system interface
    edit "aggregate"
        set vdom "root"
        set ip 10.1.1.123 255.255.255.0
        set allowaccess https ssh
        set type aggregate
        set member "port4" "port5" "port6"
        set snmp-index 45
    next
end
```

Redundancy

In a redundant interface, traffic only goes over one interface at any time. This differs from an aggregated interface where traffic goes over all interfaces for increased bandwidth. This difference means redundant interfaces can have more robust configurations with fewer possible points of failure. This is important in a fully-meshed HA configuration.

An interface is available to be in a redundant interface if:

- It is a physical interface and not a VLAN interface.
- It is not already part of an aggregated or redundant interface.
- It is in the same VDOM as the redundant interface.
- It does not have an IP address and is not configured for DHCP or PPPoE.
- It has no DHCP server or relay configured on it.
- It does not have any VLAN subinterfaces.
- It is not referenced in any security policy, VIP, or multicast policy.
- It is not monitored by HA.
- It is not one of the FortiGate-5000 series backplane interfaces.

When an interface is included in a redundant interface, it is not listed on the *Network > Interfaces* page. You cannot configure the interface individually and it is not available for inclusion in security policies, VIPs, or routing.

Example configuration

To create a redundant interface in the GUI:

1. Go to *Network > Interfaces* and select *Create New > Interface*.
2. Set *Name* to *redundant*.
3. Set *Type* to *Redundant Interface*.
4. Set *Interface members* to *port4*, *port5*, and *port6*.
5. Set *Addressing mode* to *Manual*.
6. Set *IP/Netmask* to *10.13.101.100/24*.
7. For *Administrative Access*, select *HTTPS* and *SSH*.
8. Click *OK*.

To create a redundant interface in the CLI:

```
config system interface
    edit "redundant"
```

```
        set vdom "root"
        set ip 10.13.101.100 255.255.255.0
        set allowaccess https http
        set type redundant
        set member "port4" "port5" "port6"
        set snmp-index 9
    next
end
```

Enhanced hashing for LAG member selection

FortiGate models that have an internal switch that supports modifying the distribution algorithm can use enhanced hashing to help distribute traffic evenly, or load balance, across links on the Link Aggregation (LAG) interface.

The enhanced hashing algorithm is based on a 5-tuple of the IP protocol, source IP address, destination IP address, source port, and destination port.

Different computation methods allow for more variation in the load balancing distribution, in case one algorithm does not distribute traffic evenly between links across different XAUIs. The available methods are:

xor16	Use the XOR operator to make a 16 bit hash.
xor8	Use the XOR operator to make an 8 bit hash.
xor4	Use the XOR operator to make a 4 bit hash.
crc16	Use the CRC-16-CCITT polynomial to make a 16 bit hash.

The following NP6 non-service FortiGate models support this feature: 1500D, 1500DT, 3000D, 3100D, 3200D, 3700D, and 5001D.

To configure the enhanced hashing:

```
config system npu
    set lag-out-port-select {enable | disable}
    config sw-eh-hash
        set computation {xor4 | xor8 | xor16 | crc16}
        set ip-protocol {include | exclude}
        set source-ip-upper-16 {include | exclude}
        set source-ip-lower-16 {include | exclude}
        set destination-ip-upper-16 {include | exclude}
        set destination-ip-lower-16 {include | exclude}
        set source-port {include | exclude}
        set destination-port {include | exclude}
        set netmask-length {0 - 32}
    end
end
```

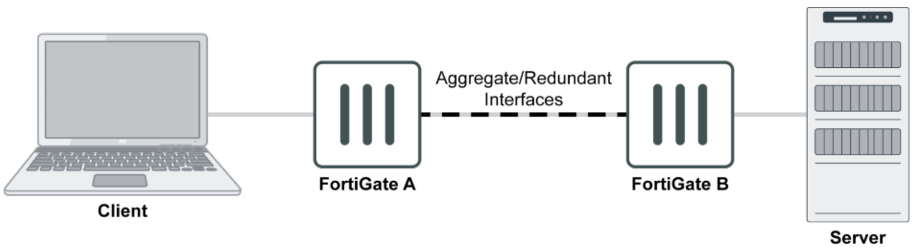
For example, to use XOR16 and include all of the fields in the 5-tuple to compute the link in the LAG interface that the packet is distributed to:

```
config system npu
    set lag-out-port-select enable
```

```
    config sw-eh-hash
        set computation xor16
        set ip-protocol include
        set source-ip-upper-16 include
        set source-ip-lower-16 include
        set destination-ip-upper-16 include
        set destination-ip-lower-16 include
        set source-port include
        set destination-port include
        set netmask-length 32
    end
end
```

Failure detection for aggregate and redundant interfaces

When an aggregate or redundant interface goes down, the corresponding fail-alert interface changes to down. When an aggregate or redundant interface comes up, the corresponding fail-alert interface changes to up.



Fail-detect for aggregate and redundant interfaces can be configured using the CLI.

To configure an aggregate interface so that port3 goes down with it:

```
config system interface
   edit "agg1"
      set vdom "root"
      set fail-detect enable
      set fail-alert-method link-down
      set fail-alert-interfaces "port3"
      set type aggregate
      set member "port1" "port2"
   next
end
```

To configure a redundant interface so that port4 goes down with it:

```
config system interface
   edit "red1"
      set vdom "root"
      set fail-detect enable
      set fail-alert-method link-down
      set fail-alert-interfaces "port4"
      set type redundant
      set member "port1" "port2"
   next
end
```

Loopback interface

A loopback interface is a logical interface that is always up. Its IP address does not depend on one specific physical port, and the attached subnet is always present in the routing table. Therefore, it can be accessed through several physical or VLAN interfaces.

Typically, a loopback interface can be used with management access, BGP peering, PIM rendezvous points, and SD-WAN.

Loopback interfaces require appropriate firewall policies to allow traffic to and from the interfaces. Multiple loopback interfaces can be configured in either non-VDOM mode or in each VDOM.

Dynamic routing protocols can be enabled on loopback interfaces. For example, loopback interfaces are a good practice for OSPF. To make it easier to troubleshoot OSPF, set the OSPF router ID to the same value as the loopback IP address to access a specific FortiGate using that IP address and SSH.

A loopback interface is configured using similar steps as a physical interface (see Configuring an interface).

Software switch

A software switch is a virtual switch that is implemented at the software or firmware level and not at the hardware level. A software switch can be used to simplify communication between devices connected to different FortiGate interfaces. For example, using a software switch, you can place the FortiGate interface connected to an internal network on the same subnet as your wireless interfaces. Then devices on the internal network can communicate with devices on the wireless network without any additional configuration on the FortiGate unit, such as additional security policies.

A software switch can also be useful if you require more hardware ports for the switch on a FortiGate unit. For example, if your FortiGate unit has a 4-port switch, WAN1, WAN2, and DMZ interfaces, and you need one more port, you can create a soft switch that can include the four-port switch and the DMZ interface, all on the same subnet. These types of applications also apply to wireless interfaces, virtual wireless interfaces, and physical interfaces such as those in FortiWiFi and FortiAP units.

Similar to a hardware switch, a software switch functions like a single interface. It has one IP address and all the interfaces in the software switch are on the same subnet. Traffic between devices connected to each interface are not regulated by security policies, and traffic passing in and out of the switch are controlled by the same policy.

When setting up a software switch, consider the following:

- Ensure that you have a back up of the configuration.
- Ensure that you have at least one port or connection, such as the console port, to connect to the FortiGate unit. If you accidentally combine too many ports, you need a way to undo errors.
- The ports that you include must not have any link or relation to any other aspect of the FortiGate unit, such as DHCP servers, security policies, and so on.
- For increased security, you can create a captive portal for the switch to allow only specific user groups access to the resources connected to the switch.

Some of the difference between software and hardware switches are:

Feature	Software switch	Hardware switch
Processing	Packets are processed in software by the CPU.	Packets are processed in hardware by the hardware switch controller, or SPU where applicable.
STP	Not Supported	Supported
Wireless SSIDs	Supported	Not Supported
Intra-switch traffic	Allowed by default. Can be explicitly set to require a policy.	Allowed by default.

To create a software switch in the GUI:

1. Go to *Network > Interfaces*.
2. Click *Create New > Interface*.
3. Set *Type* to *Software Switch*.
4. Configure the *Name*, *Interface members*, and other fields as required.
   To add an interface to a software switch, it cannot be referenced by an existing configuration and its IP address must be set to 0.0.0.0/0.0.0.0.
5. Click *OK*.

To create a software switch in the CLI:

```
config system switch-interface
    edit <interface>
        set vdom <vdom>
        set member <interface_list>
        set type switch
    next
end
config system interface
    edit <interface>
        set vdom <vdom>
        set type switch
        set ip <ip_address>
        set allowaccess https ssh ping
    next
end
```

To add an interface to a software switch, it cannot be referenced by an existing configuration and its IP address must be set to 0.0.0.0/0.0.0.0.

Example

For this example, the wireless interface (WiFi) needs to be on the same subnet as the DMZ1 interface to facilitate wireless synchronizing from an iPhone and a local computer. Because synchronizing between two subnets is problematic, putting both interfaces on the same subnet allows the synchronizing will work. The software switch will accomplish this.

1. Clear the interfaces and back up the configuration:
   a. Ensure the interfaces are not used for other security policy or for other use on the FortiGate unit.
   b. Check the WiFi and DMZ1 ports to ensure that DHCP is not enabled and that there are no other dependencies on these interfaces.
   c. Save the current configuration so that it can be recovered if something foes wrong.
2. Merge the WiFi port and DMZ1 port to create a software switch named `synchro` with an IP address of 10.10.21.12 and administrative access for HTTPS, SSH and PING:

```
config system switch-interface
    edit synchro
        set vdom "root"
        set type switch
        set member dmz1 wifi
    next
end
config system interface
    edit synchro
        set ip 10.10.21.12 255.255.255.0
        set allowaccess https ssh ping
    next
end
```

   After the switch is set up, you add security policies, DHCP servers, and any other settings that are required.

Hardware switch

A hardware switch is a virtual switch interface that groups different ports together so that the FortiGate can use the group as a single interface. Supported FortiGate models have a default hardware switch called either *internal* or *lan*. The hardware switch is supported by the chipset at the hardware level.

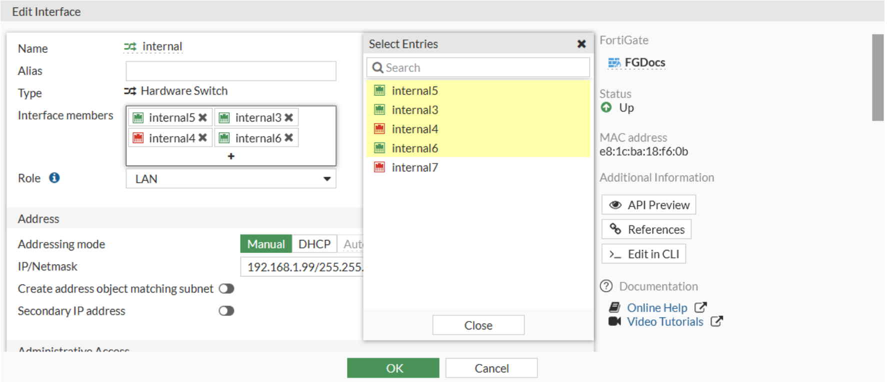
Ports that are connected to the same hardware switch behave like they are on the same physical switch in the same broadcast domain. Ports can be removed from a hardware switch and assigned to another switch or used as standalone interfaces.

Some of the difference between hardware and software switches are:

Feature	Hardware switch	Software switch
Processing	Packets are processed in hardware by the hardware switch controller, or SPU where applicable.	Packets are processed in software by the CPU.
STP	Supported	Not Supported
Wireless SSIDs	Not Supported	Supported
Intra-switch traffic	Allowed by default.	Allowed by default. Can be explicitly set to require a policy.

To change the ports in a hardware switch in the GUI:

1. Go to *Network > Interface* and edit the hardware switch.
2. Click inside the *Interface members* field.



3. Select interfaces to add or remove them from the hardware switch, then click *Close*.
   To add an interface to a hardware switch, it cannot be referenced by an existing configuration and its IP address must be set to 0.0.0.0/0.0.0.0.
4. Click *OK*.
   Removed interfaces will now be listed as standalone interfaces in the *Physical Interface* section.

To remove ports from a hardware switch in the CLI:

```
config system virtual-switch
    edit "internal"
        config port
            delete internal2
            delete internal7
            ...
        end
    next
end
```

To add ports to a hardware switch in the CLI:

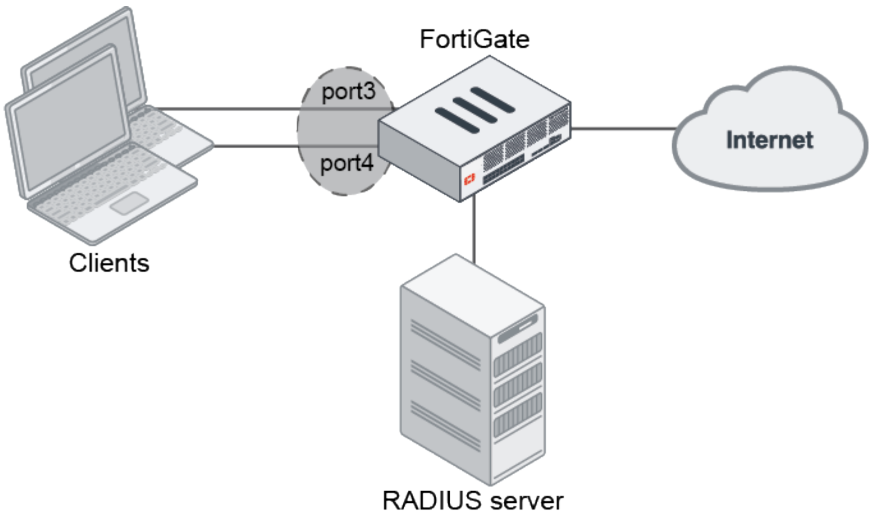
```
config system virtual-switch
    edit "internal"
        set physical-switch "sw0"
        config port
            edit "internal3"
            next
            edit "internal5"
            next
            edit "internal4"
            next
            edit "internal6"
            next
        end
    next
end
```

To add an interface to a hardware switch, it cannot be referenced by an existing configuration and its IP address must be set to 0.0.0.0/0.0.0.0.

Using 802.1X on virtual switches for certain NP6 platforms

802.1X is supported under the hardware switch interface on the following NP6 platforms: FG-30xE, FG-40xE, and FG-110xE.

In this example, port3 and port4 are part of a hardware switch interface. The hardware switch acts as a virtual switch so that devices can connect directly to these ports and perform 802.1X authentication on the port.



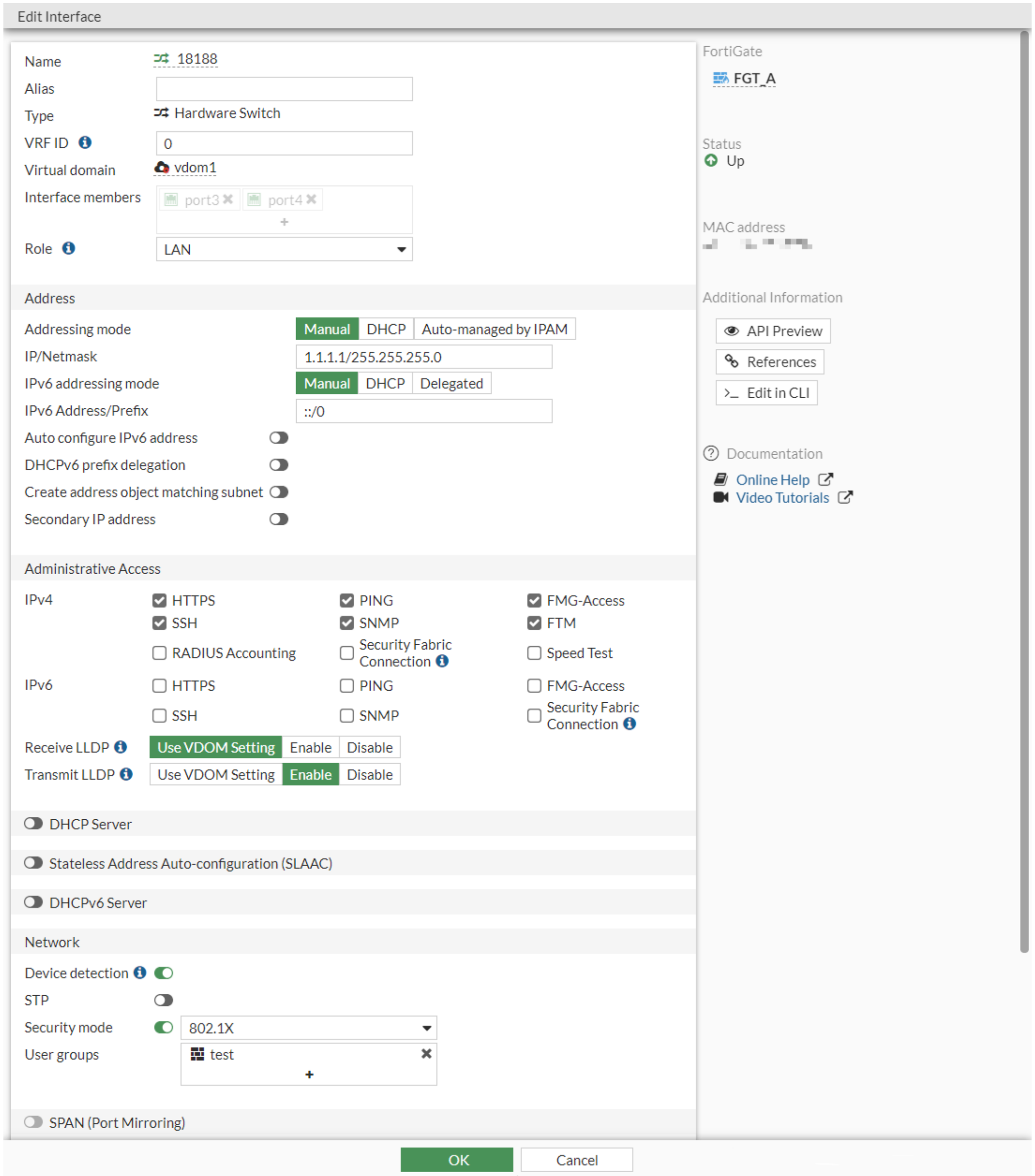
Prerequisites:

1. Configure a RADIUS server (see RADIUS servers on page 1818).
2. Define a user group named test to use the remote RADIUS server and for 802.1X authentication (see User definition and groups on page 1790).
3. Configure a hardware switch (named 18188) with port3 and port4 as the members.
4. Configure a firewall policy that allows traffic from the 18188 hardware switch to go to the internet.
5. Enable 802.1X authentication on the client devices.

To configure 802.1X authentication on a hardware switch in the GUI:

1. Go to *Network > Interfaces* and edit the hardware switch.
2. In the *Network* section, enable *Security mode* and select *802.1X*.

3. Click the + to add the *User group*.



4. Click *OK*.

To configure 802.1X authentication on a hardware switch in the CLI:

1. Configure the virtual hardware switch interfaces:

```
config system virtual-switch
    edit "18188"
        set physical-switch "sw0"
        config port
            edit "port3"
            next
            edit "port4"
            next
```

```
        end
    next
end
```

2. Configure 802.1X authentication:

```
config system interface
    edit "18188"
        set vdom "vdom1"
        set ip 1.1.1.1 255.255.255.0
        set allowaccess ping https ssh snmp fgfm ftm
        set type hard-switch
        set security-mode 802.1X
        set security-groups "test"
        set device-identification enable
        set lldp-transmission enable
        set role lan
        set snmp-index 52
    next
end
```

To verify the that the 802.1X authentication was successful:

1. Get a client connected to port3 to authenticate to access the internet.
2. In FortiOS, verify the 802.1X authentication port status:

```
# diagnose sys 802-1x status

Virtual switch '18188' (default mode) 802.1x member status:
  port3: Link up, 802.1X state: authorized
  port4: Link up, 802.1X state: unauthorized
```

Zone

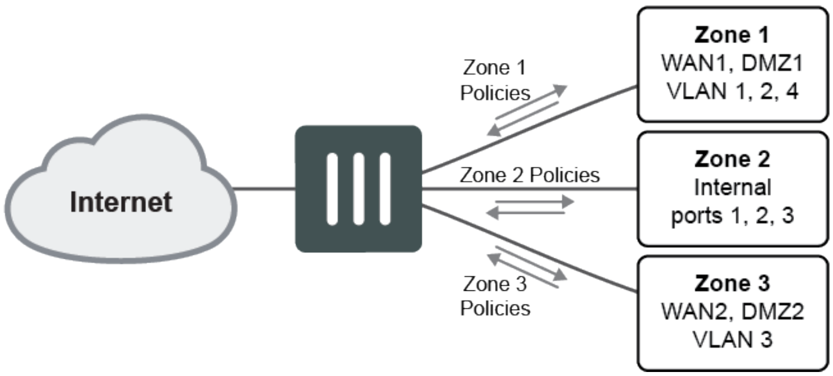
Zones are a group of one or more physical or virtual FortiGate interfaces that you can apply firewall policies to for controlling inbound and outbound traffic. Grouping interfaces and VLAN subinterfaces into zones simplifies creating firewall policies where a number of network segments can use the same policy settings and protection profiles.

When you add a zone, you select the names of the interfaces and VLAN subinterfaces to add to the zone. Each interface still has its own address. Routing is still done between interfaces, that is, routing is not affected by zones. You can use firewall policies to control the flow of intra-zone traffic.

For example, in the sample configuration below, the network includes three separate groups of users representing different entities on the company network. While each group has its own set of ports and VLANs in each area, they can all use the same firewall policy and protection profiles to access the Internet. Rather than the administrator making nine separate firewall policies, he can make administration simpler by adding the required interfaces to a zone and creating three policies.

Example configuration

You can configure policies for connections to and from a zone but not between interfaces in a zone. For this example, you can create a firewall policy to go between zone 1 and zone 3, but not between WAN2 and WAN1, or WAN1 and DMZ1.



To create a zone in the GUI:

1. Go to *Network > Interfaces*.

> If VDOMs are enabled, go to the VDOM to create a zone.

2. Click *Create New > Zone*.
3. Configure the *Name* and add the *Interface Members*.
4. Enable or disable *Block intra-zone traffic* as required.
5. Click *OK*.

To configure a zone to include the internal interface and a VLAN using the CLI:

```
config system zone
    edit zone_1
        set interface internal VLAN_1
        set intrazone {deny | allow}
    next
end
```

Using zone in a firewall policy

To configure a firewall policy to allow any interface to access the Internet using the CLI:

```
config firewall policy
    edit 2
        set name "2"
        set srcintf "Zone_1"
        set dstintf "port15"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set nat enable
    next
end
```

Intra-zone traffic

In the zone configuration you can `set intrazone deny` to prohibit the different interfaces in the same zone to talk to each other.

For example, if you have ten interfaces in your zone and the `intrazone` setting is `deny`. You now want to allow traffic between a very small number of networks on different interfaces that are part of the zone but you do not want to disable the intra-zone blocking.

In this example, the zone VLANs are defined as: `192.168.1.0/24`, `192.168.2.0/24`, ... `192.168.10.0/24`.

This policy allows traffic from 192.168.1.x to 192.168.2.x even though they are in the same zone and intra-zone blocking is enabled. The intra-zone blocking acts as a default deny rule and you have to specifically override it by creating a policy within the zone.

To enable intra-zone traffic, create the following policy:

Source Interface	Zone-name, e.g., `Vlans`
Source Address	192.168.1.0/24
Destination	Zone-name (same as Source Interface, i.e., `Vlans`)
Destination Address	192.168.2.0/24

Virtual wire pair

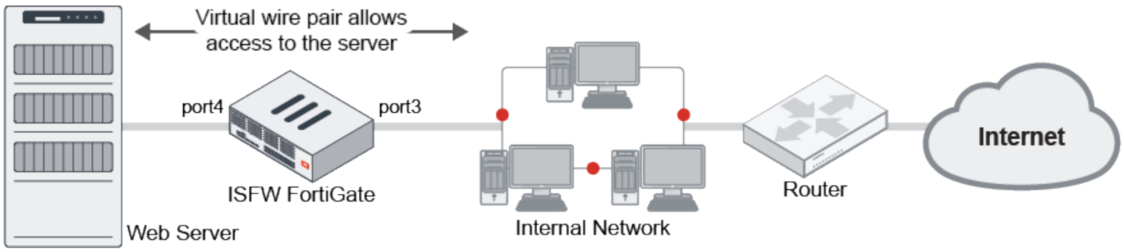
A virtual wire pair consists of two interfaces that do not have IP addressing and are treated like a transparent mode VDOM. All traffic received by one interface in the virtual wire pair can only be forwarded to the other interface, provided a virtual wire pair firewall policy allows this traffic. Traffic from other interfaces cannot be routed to the interfaces in a virtual wire pair. Redundant and 802.3ad aggregate (LACP) interfaces can be included in a virtual wire pair.

Virtual wire pairs are useful for a typical topology where MAC addresses do not behave normally. For example, port pairing can be used in a Direct Server Return (DSR) topology where the response MAC address pair may not match the request's MAC address pair.

Example

In this example, a virtual wire pair (port3 and port4) makes it easier to protect a web server that is behind a FortiGate operating as an Internal Segmentation Firewall (ISFW). Users on the internal network access the web server through the ISFW over the virtual wire pair.

Interfaces used in a virtual wire pair cannot be used to access the ISFW FortiGate. Before creating a virtual wire pair, make sure you have a different port configured to allow admin access using your preferred protocol.



To add a virtual wire pair using the GUI:

1. Go to *Network > Interfaces*.
2. Click *Create New > Virtual Wire Pair*.
3. Enter a name for the virtual wire pair.
4. Select the *Interface Members* to add to the virtual wire pair (*port3* and *port 4*).
   These interfaces cannot be part of a switch, such as the default LAN/internal interface.
5. If required, enable *Wildcard VLAN* and set the *VLAN Filter*.
6. Click *OK*.

To add a virtual wire pair using the CLI:

```
config system virtual-wire-pair
    edit "VWP-name"
        set member "port3" "port4"
        set wildcard-vlan disable
    next
end
```

To create a virtual wire pair policy using the GUI:

1. Go to *Policy & Objects > Firewall Virtual Wire Pair Policy*.
2. Click *Create New*.
3. In the *Virtual Wire Pair* field, click the + to add the virtual wire pair.
4. Select the direction (arrows) that traffic is allowed to flow.
5. Configure the other settings as needed.
6. Click *OK*.

To create a virtual wire pair policy using the CLI:

```
config firewall policy
    edit 1
        set name "VWP-Policy"
        set srcintf "port3" "port4"
        set dstintf "port3" "port4"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set fsso disable
```

```
    next
end
```

Configuring multiple virtual wire pairs in a virtual wire pair policy

You can create a virtual wire pair policy that includes different virtual wire pairs in NGFW profile and policy mode. This reduces overhead to create multiple similar policies for each VWP. In NGFW policy mode, multiple virtual wire pairs can be configured in a *Security Virtual Wire Pair Policy* and *Virtual Wire Pair SSL Inspection & Authentication* policy.

The virtual wire pair settings must have wildcard VLAN enabled. When configuring a policy in the CLI, the virtual wire pair members must be entered in `srcintf` and `dstintf` as pairs.

To configure multiple virtual wire pairs in a policy in the GUI:

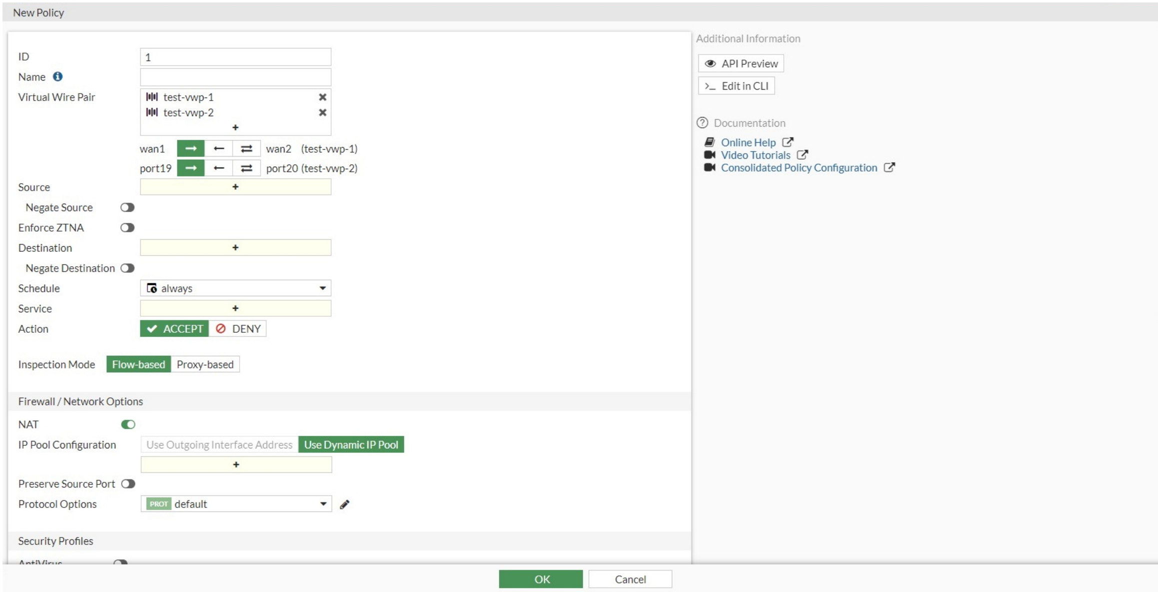
1. Configure the virtual wire pairs:
   a. Go to *Network > Interfaces* and click *Create New > Virtual Wire Pair*.
   b. Create a pair with the following settings:

Name	test-vwp-1
Interface members	wan1, wan2
Wildcard VLAN	Enable

   c. Click *OK*.
   d. Click *Create New > Virtual Wire Pair* and create another pair with the following settings:

Name	test-vwp-2
Interface members	port19, port20
Wildcard VLAN	Enable

   e. Click *OK*.
2. Configure the policy:
   a. Go to *Policy & Objects > Firewall Virtual Wire Pair Policy* and click *Create New*.
   b. In the *Virtual Wire Pair* field, click the + to add *test-vwp-1* and *test-vwp-2*. Select the direction for each of the selected virtual wire pairs.



c. Configure the other settings as needed.

d. Click *OK*.

To configure multiple virtual wire pairs in a policy in the CLI:

1. Configure the virtual wire pairs:

```
config system virtual-wire-pair
    edit "test-vwp-1"
        set member "wan1" "wan2"
        set wildcard-vlan enable
    next
    edit "test-vwp-2"
        set member "port19" "port20"
        set wildcard-vlan enable
    next
end
```

2. Configure the policy:

```
config firewall policy
    edit 1
        set name "vwp1&2-policy"
        set srcintf "port19" "wan1"
        set dstintf "port20" "wan2"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set logtraffic all
    next
end
```

PRP handling in NAT mode with virtual wire pair

PRP (Parallel Redundancy Protocol) is supported in NAT mode for a virtual wire pair. This preserves the PRP RCT (redundancy control trailer) while the packet is processed by the FortiGate.

To configure PRP handling on a device in NAT mode:

1. Enable PRP in the VDOM settings:

```
(root) # config system settings
    set prp-trailer-action enable
end
```

2. Enable PRP in the NPU attributes:

```
(global) # config system npu
    set prp-port-in "port15"
    set prp-port-out "port16"
end
```

3. Configure the virtual wire pair:

```
(root) # config system virtual-wire-pair
    edit "test-vwp-1"
        set member "port15" "port16"
    next
end
```

Enhanced MAC VLAN

The Media Access Control (MAC) Virtual Local Area Network (VLAN) feature in Linux allows you to configure multiple virtual interfaces with different MAC addresses (and therefore different IP addresses) on a physical interface.

FortiGate implements an enhanced MAC VLAN consisting of a MAC VLAN with bridge functionality. Because each MAC VLAN has a unique MAC address, virtual IP addresses (VIPs) and IP pools are supported, and you can disable Source Network Address Translation (SNAT) in policies.

MAC VLAN cannot be used in a transparent mode virtual domain (VDOM). In a transparent mode VDOM, a packet leaves an interface with the MAC address of the original source instead of the interface's MAC address. FortiGate implements an enhanced version of MAC VLAN where it adds a MAC table in the MAC VLAN which learns the MAC addresses when traffic passes through.

If you configure a VLAN ID for an enhanced MAC VLAN, it won't join the switch of the underlying interface. When a packet is sent to this interface, a VLAN tag is inserted in the packet and the packet is sent to the driver of the underlying interface. When the underlying interface receives a packet, if the VLAN ID doesn't match, it won't deliver the packet to this enhanced MAC VLAN interface.

When using a VLAN ID, the ID and the underlying interface must be a unique pair, even if the belong to different VDOMs. This is because the underlying, physical interface uses the VLAN ID as the identifier to dispatch traffic among the VLAN and enhanced MAC VLAN interfaces.

If you use an interface in an enhanced MAC VLAN, do not use it for other purposes such as a management interface, HA heartbeat interface, or in Transparent VDOMs.

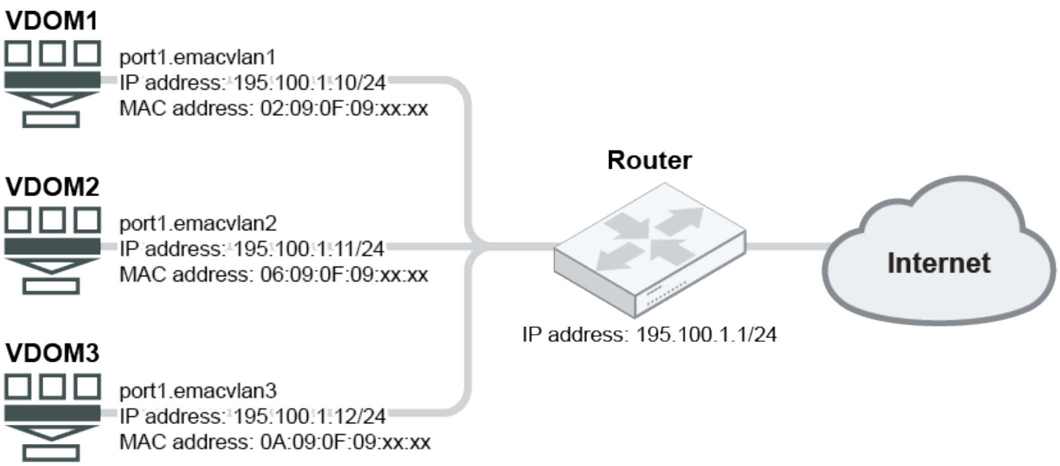
If a physical interface is used by an EMAC VLAN interface, you cannot use it in a Virtual Wire Pair.

In high availability (HA) configurations, enhanced MAC VLAN is treated as a physical interface. It's assigned a unique physical interface ID and the MAC table is synchronized with the secondary devices in the same HA cluster.

Example 1: Enhanced MAC VLAN configuration for multiple VDOMs that use the same interface or VLAN

In this example, a FortiGate is connected, through port 1 to a router that's connected to the Internet. Three VDOMs share the same interface (port 1) which connects to the same router that's connected to the Internet. Three enhanced MAC VLAN interfaces are configured on port 1 for the three VDOMs. The enhanced MAC VLAN interfaces are in the same IP subnet segment and each have unique MAC addresses.

The underlying interface (port 1) can be a physical interface, an aggregate interface, or a VLAN interface on a physical or aggregate interface.



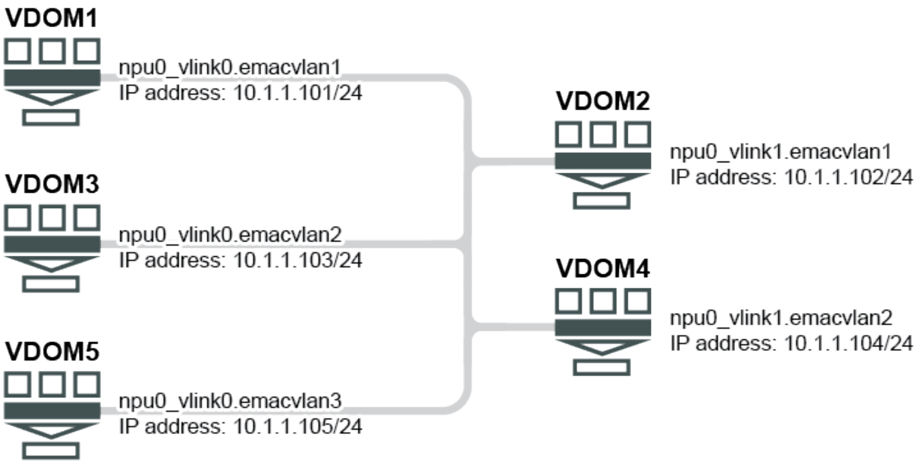
To configure enhanced MAC VLAN for this example in the CLI:

```
config system interface
    edit port1.emacvlan1
        set vdom VDOM1
        set type emac-vlan
        set interface port1
    next
    edit port 1.emacvlan2
        set vdom VDOM2
        set type emac-vlan
        set interface port1
    next
    edit port1.emacvlan3
        set vdom VDOM3
        set type emac-vlan
        set interface port1
    next
end
```

Example 2: Enhanced MAC VLAN configuration for shared VDOM links among multiple VDOMs

In this example, multiple VDOMs can connect to each other using enhanced MAC VLAN on network processing unit (NPU) virtual link (Vlink) interfaces.

FortiGate VDOM links (NPU-Vlink) are designed to be peer-to-peer connections and VLAN interfaces on NPU Vlink ports use the same MAC address. Connecting more than two VDOMs using NPU Vlinks and VLAN interfaces is not recommended.



To configure enhanced MAC VLAN for this example in the CLI:

```
config system interface
    edit npu0_vlink0.emacvlan1
        set vdom VDOM1
        set type emac-vlan
        set interface npu0_vlink0
    next
    edit npu0_vlink0.emacvlan2
        set vdom VDOM3
        set type emac-vlan
        set interface npu0_vlink0
    next
    edit npu0_vlink1.emacvlan1
        set vdom VDOM2
        set type emac-vlan
        set interface npu0_vlink1
    next
end
```

Example 3: Enhanced MAC VLAN configuration for unique MAC addresses for each VLAN interface on the same physical port

Some networks require a unique MAC address for each VLAN interface when the VLAN interfaces share the same physical port. In this case, the enhanced MAC VLAN interface is used the same way as normal VLAN interfaces.

To configure this, use the `set vlanid` command for the VLAN tag. The VLAN ID and interface must be a unique pair, even if they belong to different VDOMs.

To configure enhanced MAC VLAN:

```
config system interface
    edit <interface-name>
        set type emac-vlan
        set vlanid <VLAN-ID>
        set interface <physical-interface>
    next
end
```

VXLAN

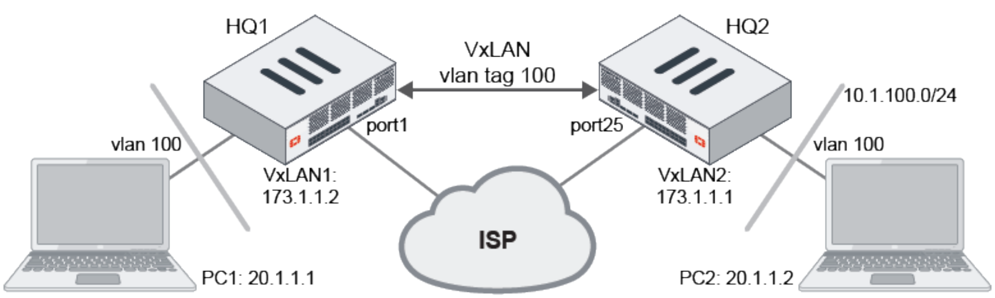
Virtual Extensible LAN (VXLAN) is a network virtualization technology used in large cloud computing deployments. It encapsulates layer 2 Ethernet frames within layer 3 IP packets using the standard destination port 4789. VXLAN endpoints that terminate VXLAN tunnels can be virtual or physical switch ports, and are known as VXLAN tunnel endpoints (VTEPs). For more information about VXLAN, see RFC 7348.

The following topics provide information about VXLAN:

- VLAN inside VXLAN on page 198
- Virtual wire pair with VXLAN on page 200

VLAN inside VXLAN

VLANs can be assigned to VXLAN interfaces. In a data center network where VXLAN is used to create an L2 overlay network and for multitenant environments, a customer VLAN tag can be assigned to VXLAN interface. This allows the VLAN tag from VLAN traffic to be encapsulated within the VXLAN packet.



To configure VLAN inside VXLAN on HQ1:

1. Configure VXLAN:

```
config system vxlan
   edit "vxlan1"
      set interface port1
      set vni 1000
      set remote-ip 173.1.1.1

   next
end
```

2. Configure system interface:

```
config system interface
   edit vlan100
     set vdom root
     set vlanid 100
     set interface dmz
   next
   edit vxlan100
     set type vlan
     set vlanid 100
     set vdom root
     set interface vxlan1
   next
end
```

3. Configure software-switch:

```
config system switch-interface
   edit sw1
      set vdom root
      set member vlan100 vxlan100
      set intra-switch-policy implicit
   next
end
```

The default `intra-switch-policy implicit` behavior allows traffic between member interfaces within the switch. Therefore, it is not necessary to create firewall policies to allow this traffic.

Instead of creating a software-switch, it is possible to use a virtual-wire-pair as well. See Virtual wire pair with VXLAN on page 200.

To configure VLAN inside VXLAN on HQ2:

1. Configure VXLAN:

```
config system vxlan
   edit "vxlan2"
      set interface port25
      set vni 1000
      set remote-ip 173.1.1.2
   next
end
```

2. Configure system interface:

```
config system interface
      edit vlan100
      set vdom root
      set vlanid 100
      set interface port20
   next
   edit vxlan100
      set type vlan
      set vlanid 100
      set vdom root
```

```
        set interface vxlan2
    next
end
```

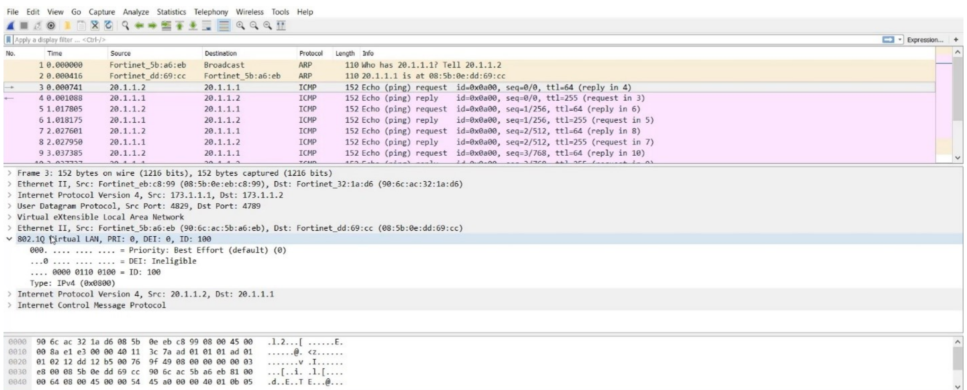
3. Configure software-switch:

```
config system switch-interface
    edit sw1
        set vdom root
        set member vlan100 vxlan100
    next
end
```

To verify the configuration:

Ping PC1 from PC2.

The following is captured on HQ2:

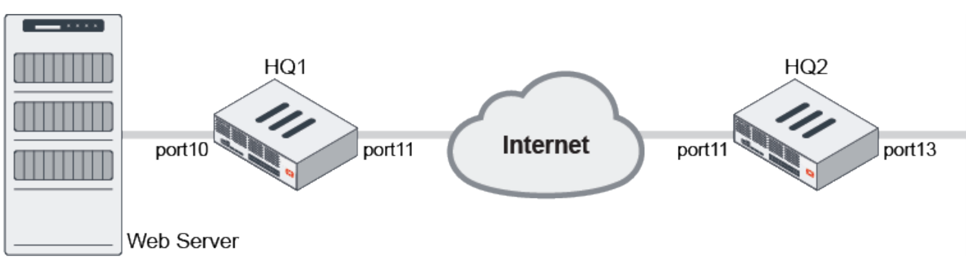


This captures the VXLAN traffic between 172.1.1.1 and 172.1.1.2 with the VLAN 100 tag inside.

Virtual wire pair with VXLAN

Virtual wire pairs can be used with VXLAN interfaces.

In this examples, VXLAN interfaces are added between FortiGate HQ1 and FortiGate HQ2, a virtual wire pair is added in HQ1, and firewall policies are created on both HQ1 and HQ2.



To create VXLAN interface on HQ1:

```
config system interface
    edit "port11"
        set vdom "root"
        set ip 10.2.2.1 255.255.255.0
```

```
        set allowaccess ping https ssh snmp telnet
    next
end
config system vxlan
    edit "vxlan1"
        set interface "port11"
        set vni 1000
        set remote-ip "10.2.2.2"
    next
end
```

To create VXLAN interface on HQ2:

```
config system interface
    edit "port11"
        set vdom "root"
        set ip 10.2.2.2 255.255.255.0
        set allowaccess ping https ssh snmp http
    next
end
config system vxlan
    edit "vxlan1"
        set interface "port11"
        set vni 1000
        set remote-ip "10.2.2.1"
    next
end
config system interface
    edit "vxlan1"
        set vdom "root"
        set ip 10.1.100.2 255.255.255.0
        set allowaccess ping https ssh snmp
    next
end
```

To create a virtual wire pair on HQ1:

```
config system virtual-wire-pair
    edit "vwp1"
        set member "port10" "vxlan1"
    next
end
```

To create a firewall policy on HQ1:

```
config firewall policy
    edit 5
        set name "vxlan-policy"
        set srcintf "port10" "vxlan1"
        set dstintf "port10" "vxlan1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
```

```
        set ssl-ssh-profile "certificate-inspection"
        set av-profile "default"
        set webfilter-profile "default"
        set dnsfilter-profile "default"
        set ips-sensor "default"
        set application-list "default"
        set fsso disable
    next
end
```

To create a firewall policy on HQ2:

```
config firewall policy
    edit 5
        set name "1"
        set srcintf "port13"
        set dstintf "vxlan1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set fsso disable
        set nat enable
    next
end
```

DNS

Domain name system (DNS) is used by devices to locate websites by mapping a domain name to a website's IP address.

A FortiGate can serve different roles based on user requirements:

- A FortiGate can control what DNS server a network uses.
- A FortiGate can function as a DNS server.

FortiGuard Dynamic DNS (DDNS) allows a remote administrator to access a FortiGate's Internet-facing interface using a domain name that remains constant even when its IP address changes.

FortiOS supports DNS configuration for both IPv4 and IPv6 addressing. When a user requests a website, the FortiGate looks to the configured DNS servers to provide the IP address of the website in order to know which server to contact to complete the transaction.

The FortiGate queries the DNS servers whenever it needs to resolve a domain name into an IP address, such as for NTP or web servers defined by their domain names.

The following topics provide information about DNS:

- Important DNS CLI commands on page 203
- DNS domain list on page 204
- FortiGate DNS server on page 206
- DDNS on page 208
- DNS latency information on page 212

- DNS over TLS and HTTPS on page 214
- DNS troubleshooting on page 218

Important DNS CLI commands

DNS settings can be configured with the following CLI command:

```
config system dns
    set primary <ip_address>
    set secondary <ip_address>
    set protocol {cleartext dot doh}
    set ssl-certificate <string>
    set server-hostname <hostname>
    set domain <domains>
    set ip6-primary <ip6_address>
    set ip6-secondary <ip6_address>
    set timeout <integer>
    set retry <integer>
    set dns-cache-limit <integer>
    set dns-cache-ttl <integer>
    set cache-notfound-responses {enable | disable}
    set interface-select-method {auto | sdwan | specify}
    set interface <interface>
    set source-ip <class_ip>
end
```

For a FortiGate with multiple logical CPUs, you can set the DNS process number from 1 to the number of logical CPUs. The default DNS process number is 1.

```
config system global
    set dnsproxy-worker-count <integer>
end
```

DNS protocols

The following DNS protocols can be enabled:

- `cleartext`: Enable clear text DNS over port 53 (default).
- `dot`: Enable DNS over TLS.
- `doh`: Enable DNS over HTTPS.

For more information, see DNS over TLS and HTTPS on page 214.

cache-notfound-responses

When enabled, any DNS requests that are returned with `NOT FOUND` can be stored in the cache. The DNS server is not asked to resolve the host name for `NOT FOUND` entries. By default, this option is disabled.

dns-cache-limit

Set the number of DNS entries that are stored in the cache (0 to 4294967295, default = 5000). Entries that remain in the cache provide a quicker response to requests than going out to the Internet to get the same information.

dns-cache-ttl

The duration that the DNS cache retains information, in seconds (60 to 86400 (1 day), default = 1800).

VDOM DNS

When the FortiGate is in multi-vdom mode, DNS is handled by the management VDOM. However in some cases, administrators may want to configure custom DNS settings on a non-management VDOM. For example, in a multi-tenant scenario, each VDOM might be occupied by a different tenant, and each tenant might require its own DNS server.

To configure a custom VDOM within a non-management VDOM:

```
config vdom
edit <vdom>
config system vdom-dns
    set vdom-dns enable
    set primary <primary_DNS>
    set secondary <secondary_DNS>
    set protocol {cleartext dot doh}
    set ip6-primary <primary_IPv6_DNS>
    set ip6-secondary <secondary_IPv6_DNS>
    set source-ip <IP_address>
    set interface-select-method {auto | sdwan | specify}
end
```

DNS domain list

You can configure up to eight domains in the DNS settings using the GUI or the CLI.

When a client requests a URL that does not include an FQDN, FortiOS resolves the URL by traversing through the DNS domain list and performing a query for each domain until the first match is found.

By default, FortiGate uses FortiGuard's DNS servers:

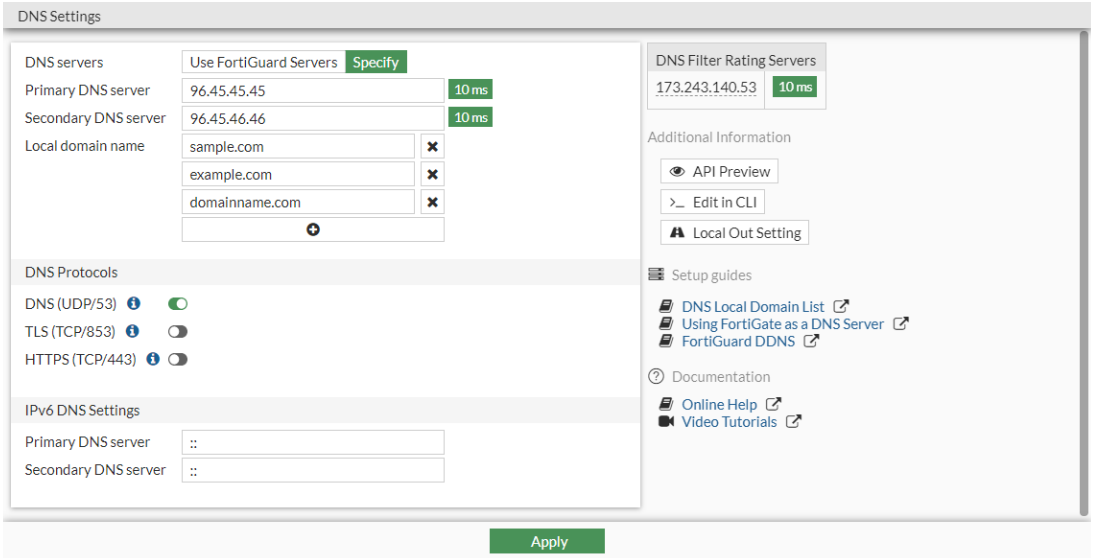
- Primary: 96.45.45.45
- Secondary: 96.45.46.46

You can also customize the DNS timeout time and the number of retry attempts.

To configure a DNS domain list in the GUI:

1. Go to *Network* > *DNS*.
2. Set *DNS Servers* to *Specify*.
3. Configure the primary and secondary DNS servers as needed.

4. In the *Local Domain Name* field, enter the first domain (`sample.com` in this example).
5. Click the + to add more domains (`example.com` and `domainname.com` in this example). You can enter up to eight domains.
6. Configure additional DNS protocol and IPv6 settings as needed.



7. Click *Apply*.

To configure a DNS domain list in the CLI:

```
config system dns
    set primary 96.45.45.45
    set secondary 96.45.46.46
    set domain "sample.com" "example.com" "domainname.com"
end
```

Verify the DNS configuration

In the following example, the local DNS server has the entry for *host1* mapped to the FQDN of *host1.sample.com*, and the entry for *host2* is mapped to the FQDN of *host2.example.com*.

To verify that the DNS domain list is configured:

1. Open Command Prompt.
2. Enter `ping host1`.
   The system returns the following response:
   ```
   PING host1.sample.com (1.1.1.1): 56 data bytes
   ```
   As the request does not include an FQDN, FortiOS traverses the configured DNS domain list to find a match. Because *host1* is mapped to the *host1.sample.com*, FortiOS resolves *host1* to *sample.com*, the first entry in the domain list.
3. Enter `ping host2`.
   The system returns the following response:
   ```
   PING host2.example.com (2.2.2.2): 56 data bytes
   ```
   FortiOS traverses the domain list to find a match. It first queries sample.com, the first entry in the domain list, but does not find a match. It then queries the second entry in the domain list, example.com. Because *host2* is mapped

to the FQDN of *host2.example.com*, FortiOS resolves *host2* to *example.com*.

DNS timeout and retry settings

The DNS timeout and retry settings can be customized using the CLI.

```
config system dns
    set timeout <integer>
    set retry <integer>
end
```

`timeout <integer>`	The DNS query timeout interval, in seconds (1 - 10, default = 5).
`retry <integer>`	The number of times to retry the DNS query (0 - 5, default - 2).

FortiGate DNS server

You can create local DNS servers for your network. Depending on your requirements, you can either manually maintain your entries (primary DNS server), or use it to refer to an outside source (secondary DNS server).

A local, primary DNS server requires that you to manually add all URL and IP address combinations. Using a primary DNS server for local services can minimize inbound and outbound traffic, and access time. Making it authoritative is not recommended, because IP addresses can change, and maintaining the list can become labor intensive.

A secondary DNS server refers to an alternate source to obtain URL and IP address combinations. This is useful when there is a primary DNS server where the entry list is maintained.

FortiGate as a DNS server also supports TLS and HTTPS connections to a DNS client. See DNS over TLS and HTTPS on page 214 for details.

By default, DNS server options are not available in the FortiGate GUI.

To enable DNS server options in the GUI:

1. Go to *System > Feature Visibility*.
2. Enable *DNS Database* in the *Additional Features* section.
3. Click *Apply*.

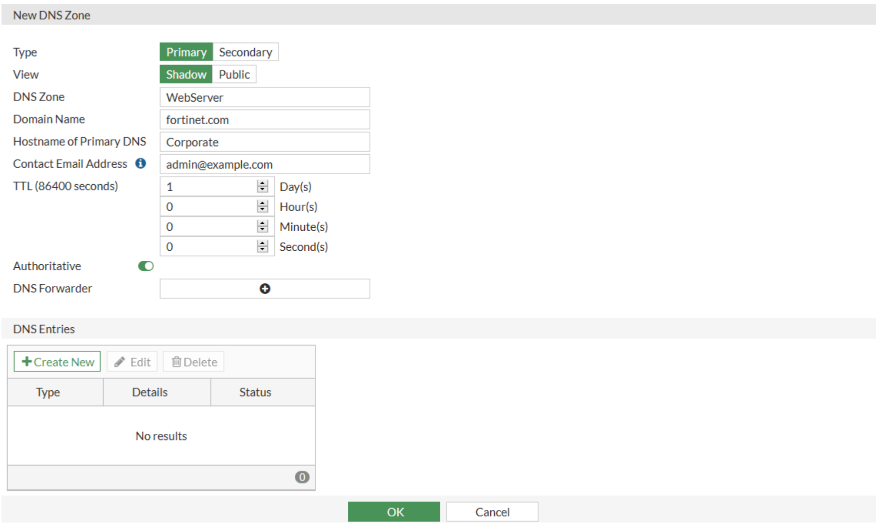
Example configuration

This section describes how to create an unauthoritative primary DNS server. The interface mode is recursive so that, if the request cannot be fulfilled, the external DNS servers will be queried.

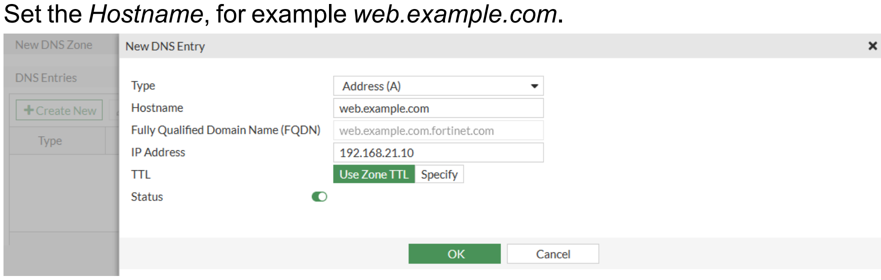
To configure FortiGate as a primary DNS server in the GUI:

1. Go to *Network > DNS Servers*.
2. In the *DNS Database* table, click *Create New*.
3. Set *Type* to *Primary*.

4. Set *View* to *Shadow*.
   The *View* setting controls the accessibility of the DNS server. If you select *Public*, external users can access or use the DNS server. If you select *Shadow*, only internal users can use it.
5. Enter a *DNS Zone*, for example, *WebServer*.
6. Enter the *Domain Name* of the zone, for example, *fortinet.com*.
7. Enter the *Hostname* of the DNS server, for example, *Corporate*.
8. Enter the *Contact Email Address* for the administrator, for example, *admin@example.com*.
9. Disable *Authoritative*.

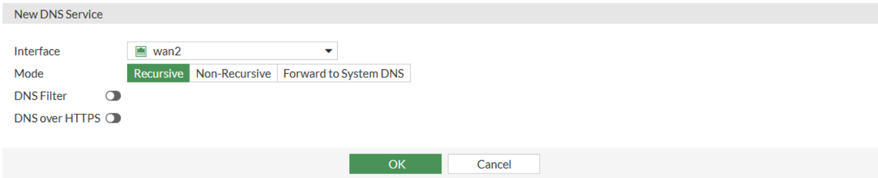


10. Add DNS entries:
    a. In the *DNS Entries* table, click *Create New*.
    b. Select a *Type*, for example *Address (A)*.
    c. Set the *Hostname*, for example *web.example.com*.



    d. Configure the remaining settings as needed. The options vary depending on the selected *Type*.
    e. Click *OK*.
11. Add more DNS entries as needed.
12. Click *OK*.
13. Enable DNS services on an interface:
    a. Go to *Network > DNS Servers*.
    b. In the *DNS Service on Interface* table, click *Create New*.
    c. Select the *Interface* for the DNS server, such as *wan2*.

d. Set the *Mode* to *Recursive*.



e. Click *OK*.

To configure FortiGate as a primary DNS server in the CLI:

```
config system dns-database
    edit WebServer
        set domain example.com
        set type master
        set view shadow
        set ttl 86400
        set primary-name corporate
        set contact admin@example.com
        set authoritative disable
        config dns-entry
            edit 1
                set status enable
                set hostname web.example.com
                set type A
                set ip 192.168.21.12
            next
        end
    next
end
```

```
config system dns-server
    edit wan2
        set mode recursive
    next
end
```

DDNS

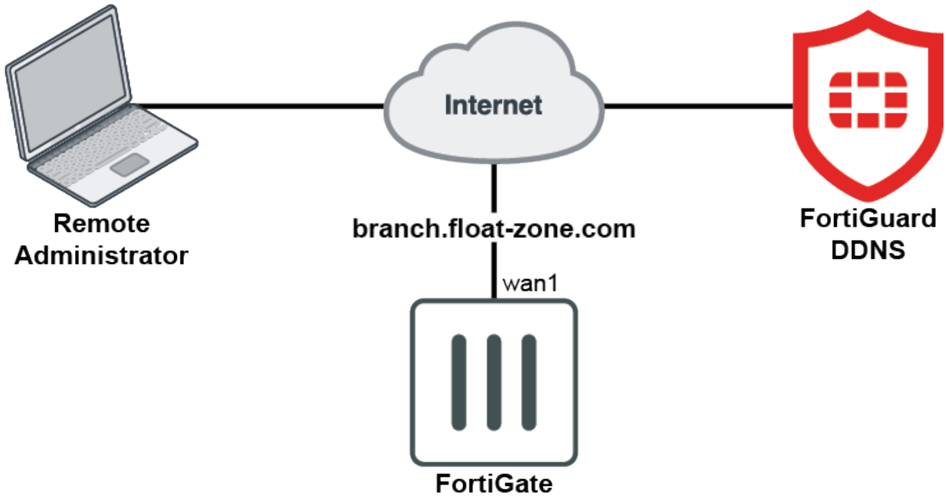
If your external IP address changes regularly and you want a static domain name, you can configure the external interface to use a dynamic DNS (DDNS) service. This ensures that external users and customers can always connect to your company firewall. You can configure FortiGuard as the DDNS server using the GUI or CLI.

A license or subscription is not required to use the DDNS service, but configuring DDNS in the GUI is not supported if:

- The FortiGate model is a 1000-series or higher.
- The FortiGate is a VM.
- The DNS server is not using FortiGuard as the DNS.

FortiGate does not support DDNS when in transparent mode.

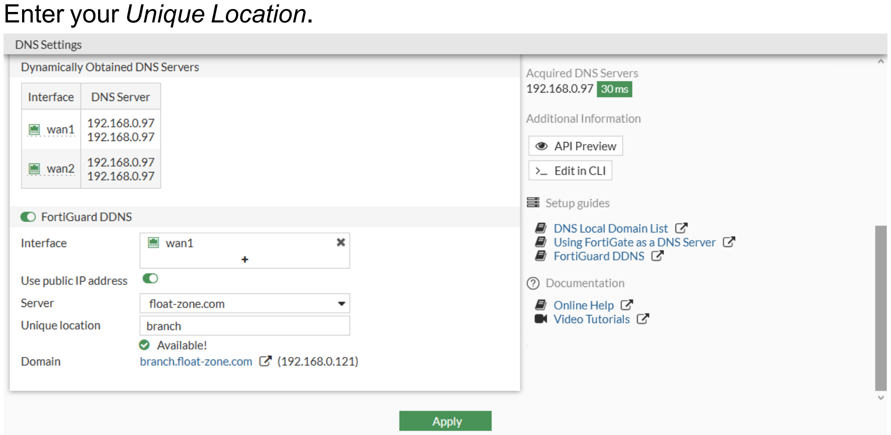
Sample topology



In this example, FortiGuard DDNS is enabled and the DDNS server is set to *float-zone.com*. Other DDNS server options include *fortiddns.com* and *fortidyndns.com*.

To configure FortiGuard DDNS service as a DDNS server in the GUI:

1. Go to *Network > DNS*
2. Enable *FortiGuard DDNS*.
3. Select the *Interface* with the dynamic connection.
4. Select the *Server* that you have an account with.
5. Enter your *Unique Location*.



6. Click *Apply*.

To configure the FortiGuard DDNS service as an IPv4 DDNS server in the CLI:

```
config system ddns
    edit 1
        set ddns-server FortiGuardDDNS
        set server-type ipv4
        set ddns-domain "branch.float-zone.com"
        set addr-type ipv4
```

```
        set use-public-ip enable
        set monitor-interface "wan1"
    next
end
```

To configure the FortiGuard DDNS service as an IPv6 DDNS server in the CLI:

```
config system ddns
    edit 1
        set ddns-server FortiGuardDDNS
        set server-type ipv6
        set ddns-domain "fgtatest001.float-zone.com"
        set addr-type ipv6
        set monitor-interface "wan1"
    next
end
```

DDNS servers other than FortiGuard

If you do not have a FortiGuard subscription, or want to use a different DDNS server, you can configure a DDNS server for each interface. Only the first configure port appears in the GUI.

The available commands vary depending on the selected DDNS server.

To configure DDNS servers other than FortiGuard in the CLI:

```
config system ddns
    edit <DDNS_ID>
        set monitor-interface <external_interface>
        set ddns-server <ddns_server_selection>
        set server-type {ipv4 | ipv6}
        set ddns-server-addr <address>
        set addr-type ipv6 {ipv4 | ipv6}
    next
end
```

To configure an IPv6 DDNS client with generic DDNS on port 3 in the CLI:

```
config system ddns
    edit 1
        set ddns-server genericDDNS
        set server-type ipv6
        set ddns-server-addr "2004:16:16:16::2" "16.16.16.2" "ddns.genericddns.com"
        set ddns-domain "test.com"
        set addr-type ipv6
        set monitor-interface "port3"
    next
end
```

Refresh DDNS IP addresses

When using a public IP that is not assigned to the FortiGate, the FortiGate cannot trigger an update when the IP address changes. The FortiGate can be configured to refresh DDNS IP addresses by periodically checking the DDNS server at an update interval.

To configure FortiGate to refresh DDNS IP addresses in the CLI:

```
config system ddns
    edit 1
        set use-public-ip enable
        set update-interval <seconds>
    next
end
```

When `update-interval` is set to `0`:

- For FortiGuard DDNS, the interval is 300 seconds.
- For third part DDNS servers, the interval is assigned by the DDNS server.

Disable cleartext

When `clear-text` is disabled, FortiGate uses the SSL connection to send and receive DDNS updates.

To disable cleartext and set the SSL certificate in the CLI:

```
config system ddns
    edit 2
        set clear-text disable
        set ssl-certificate <cert_name>
    next
end
```

DDNS update override

A DHCP server has an override command option that allows DHCP server communications to go through DDNS to perform updates for the DHCP client. This enforces a DDNS update of the A field every time even if the DHCP client does not request it. This allows support for the `allow`, `ignore`, and `deny client-updates` options.

To enable DDNS update override in the CLI:

```
config system dhcp server
    edit 1
        set ddns-update enable
        set ddns-update-override enable
        set ddns-server-ip <ddns_server_ip>
        set ddns-zone <ddns_zone>
    next
end
```

Troubleshooting

To debug DDNS:

```
# diagnose debug application ddnscd -1
# diagnose debug enable
```

To check if a DDNS server is available:

```
# diagnose test application ddnscd 3
```

Not available:

```
FortiDDNS status:
ddns_ip=0.0.0.0, ddns_ip6=::, ddns_port=443 svr_num=0 domain_num=0
```

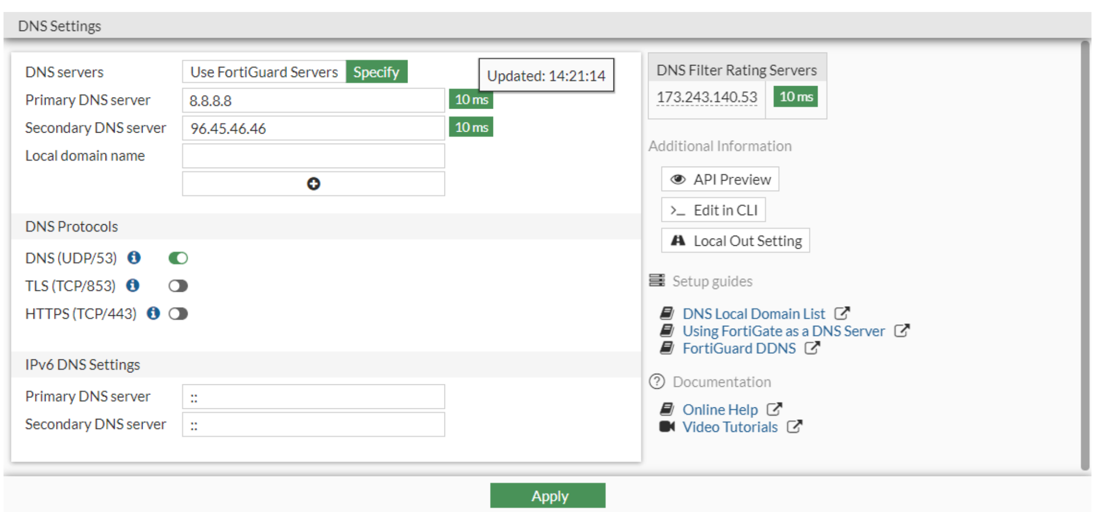
Available:

```
FortiDDNS status:
ddns_ip=208.91.113.230, ddns_ip6=::, ddns_port=443 svr_num=1 domain_num=3
svr[0]= 208.91.113.230
domain[0]= fortiddns.com
domain[1]= fortidyndns.com
domain[2]= float-zone.com
```

DNS latency information

High latency in DNS traffic can result in an overall sluggish experience for end-users. In the *DNS Settings* pane, you can quickly identify DNS latency issues in your configuration.

Go to *Network > DNS* to view DNS latency information in the right side bar. If you use FortiGuard DNS, latency information for DNS, DNS filter, web filter, and outbreak prevention servers is also visible. Hover your pointer over a latency value to see when it was last updated.



To view DNS latency information using the CLI:

```
# diagnose test application dnsproxy 2
worker idx: 0
worker: count=1 idx=0
retry_interval=500 query_timeout=1495
DNS latency info:
vfid=0 server=2001::1 latency=1494 updated=73311
vfid=0 server=96.45.46.46 latency=1405 updated=2547
vfid=0 server=8.8.8.8 latency=19 updated=91
SDNS latency info:
vfid=0 server=173.243.140.53 latency=1 updated=707681
DNS_CACHE: alloc=35, hit=26
RATING_CACHE: alloc=1, hit=49
DNS UDP: req=66769 res=63438 fwd=83526 alloc=0 cmp=0 retrans=16855 to=3233
          cur=111 switched=8823467 num_switched=294 v6_cur=80 v6_switched=7689041 num_v6_
switched=6
         ftg_res=8 ftg_fwd=8 ftg_retrans=0
DNS TCP: req=0, res=0, fwd=0, retrans=0 alloc=0, to=0
FQDN: alloc=45 nl_write_cnt=9498 nl_send_cnt=21606 nl_cur_cnt=0
Botnet: searched=57 hit=0 filtered=57 false_positive=0
```

To view the latency from web filter and outbreak protection servers using the CLI:

```
# diagnose debug rating
Locale   : english

Service  : Web-filter
Status   : Enable
License  : Contract

Service  : Antispam
Status   : Disable

Service  : Virus Outbreak Prevention
Status   : Disable

-=- Server List (Tue Jan 22 08:03:14 2019) -=-

IP               Weight RTT Flags  TZ  Packets  Curr Lost Total Lost  Updated Time
173.243.138.194  10     0 DI       -8  700      0         2           Tue Jan 22 08:02:44
2019
173.243.138.195  10     0          -8  698      0         4           Tue Jan 22 08:02:44
2019
173.243.138.198  10     0          -8  698      0         4           Tue Jan 22 08:02:44
2019
173.243.138.196  10     0          -8  697      0         3           Tue Jan 22 08:02:44
2019
173.243.138.197  10     1          -8  694      0         0           Tue Jan 22 08:02:44
2019
96.45.33.64      10     22 D       -8  701      0         6           Tue Jan 22 08:02:44
2019
64.26.151.36     40     62         -5  704      0         10          Tue Jan 22 08:02:44
2019
64.26.151.35     40     62         -5  703      0         9           Tue Jan 22 08:02:44
2019
```

```
209.222.147.43   40     70 D       -5   696      0         1           Tue Jan 22 08:02:44
2019
66.117.56.42     40     70         -5   697      0         3           Tue Jan 22 08:02:44
2019
66.117.56.37     40     71         -5   702      0         9           Tue Jan 22 08:02:44
2019
65.210.95.239    40     74         -5   695      0         1           Tue Jan 22 08:02:44
2019
65.210.95.240    40     74         -5   695      0         1           Tue Jan 22 08:02:44
2019
45.75.200.88     90     142        0    706      0         12          Tue Jan 22 08:02:44
2019
45.75.200.87     90     155        0    714      0         20          Tue Jan 22 08:02:44
2019
45.75.200.85     90     156        0    711      0         17          Tue Jan 22 08:02:44
2019
45.75.200.86     90     159        0    704      0         10          Tue Jan 22 08:02:44
2019
62.209.40.72     100    157        1    701      0         7           Tue Jan 22 08:02:44
2019
62.209.40.74     100    173        1    705      0         11          Tue Jan 22 08:02:44
2019
62.209.40.73     100    173        1    699      0         5           Tue Jan 22 08:02:44
2019
121.111.236.179  180    138        9    706      0         12          Tue Jan 22 08:02:44
2019
121.111.236.180  180    138        9    704      0         10          Tue Jan 22 08:02:44
2019
```

DNS over TLS and HTTPS

DNS over TLS (DoT) is a security protocol for encrypting and encapsulating DNS queries and responses over the TLS protocol. DoT increases user privacy and security by preventing eavesdropping and manipulation of DNS data via man-in-the-middle attacks. Similarly, DNS over HTTPS (DoH) provides a method of performing DNS resolution over a secure HTTPS connection. DoT and DoH are supported in explicit mode where the FortiGate acts as an explicit DNS server that listens for DoT and DoH requests. Local-out DNS traffic over TLS and HTTPS is also supported.

Basic configurations for enabling DoT and DoH for local-out DNS queries

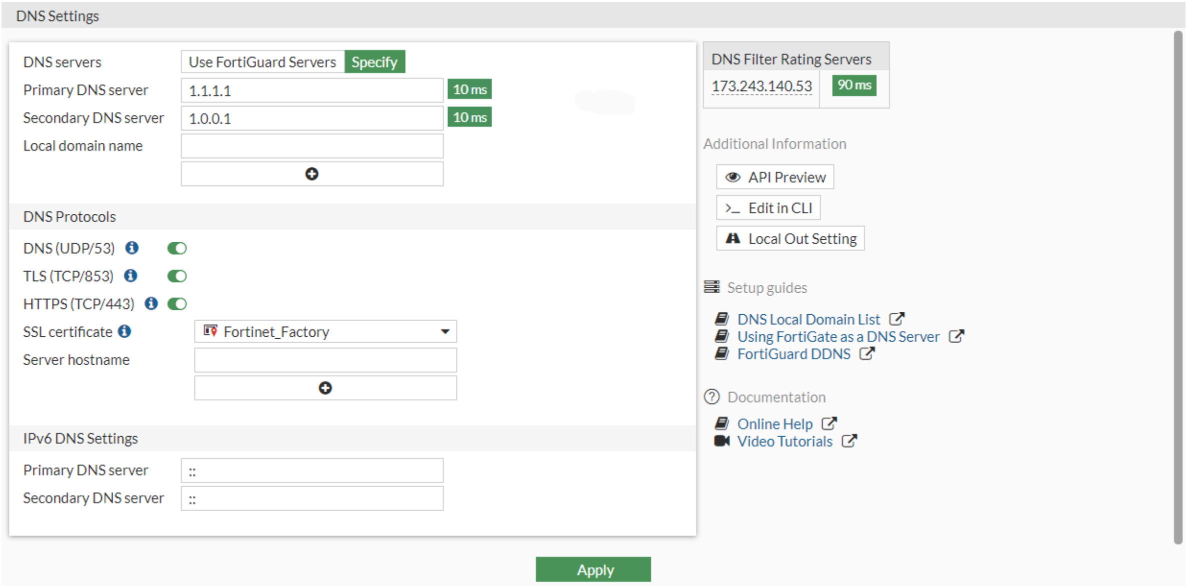
Before enabling DoT or DoH, ensure that they are supported by the DNS servers. The legacy FortiGuard DNS servers (208.91.112.53 and 208.91.112.52) do not support DoT or DoH queries, and will drop these packets. At times, the latency status of the DNS servers might also appear high or unreachable.

Disabling DoT and DoH is recommended when they are not supported by the DNS servers.

To enable DoT and DoH DNS in the GUI:

1. Go to *Network > DNS*.
2. Enter the primary and secondary DNS server addresses.

3. In the *DNS Protocols* section, enable *TLS (TCP/853)* and *HTTPS (TCP/443)*.



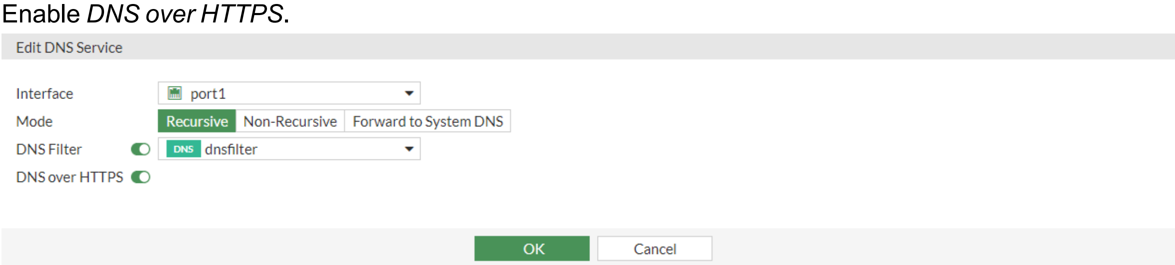
4. Configure the other settings as needed.
5. Click *Apply*.

To enable DoT and DoH DNS in the CLI:

```
config system dns
    set primary 1.1.1.1
    set secondary 1.0.0.1
    set protocol {cleartext dot doh}
end
```

To enable DoH on the DNS server in the GUI:

1. Go to *Network > DNS Servers*.
2. In the *DNS Service on Interface* section, edit an existing interface, or create a new one.
3. Select a *Mode*, and *DNS Filter* profile.
4. Enable *DNS over HTTPS*.



5. Click *OK*.

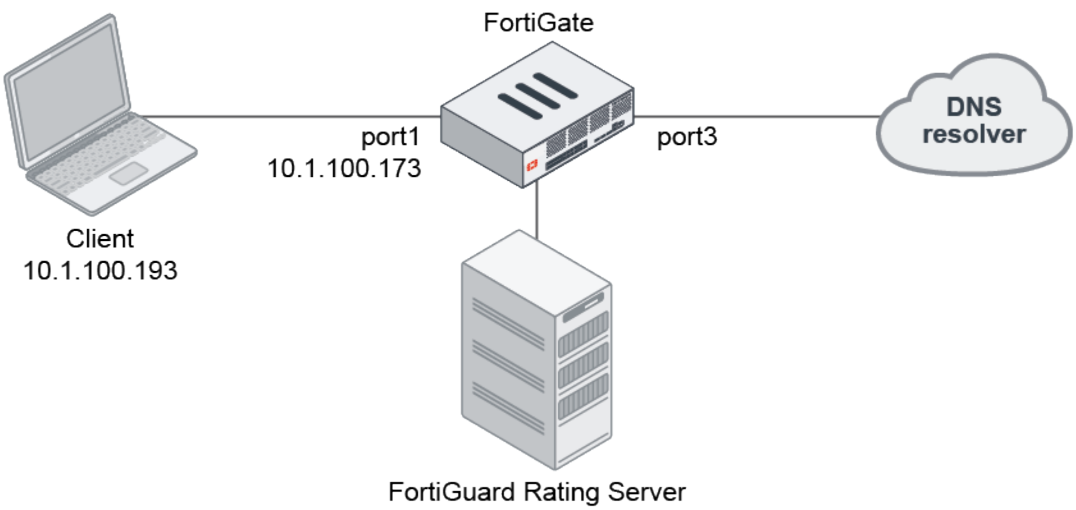
To enable DoH on the DNS server in the CLI:

```
config system dns-server
    edit "port1"
        set dnsfilter-profile "dnsfilter"
```

```
        set doh enable
    next
end
```

Examples

The following examples demonstrate how configure DNS settings to support DoT and DoH queries made to the FortiGate.



DoT

The following example uses a DNS filter profile where the education category is blocked.

To enable scanning DoT traffic in explicit mode with a DNS filter:

1. Configure the DNS settings:

```
config system dns
    set primary 1.1.1.1
    set secondary 1.0.0.1
    set protocol dot
end
```

2. Configure the DNS filter profile:

```
config dnsfilter profile
    edit "dnsfilter"
        config ftgd-dns
            config filters
                edit 1
                    set category 30
                    set action block
                next
            end
        end
    next
end
```

3. Configure the DNS server settings:

```
config system dns-server
    edit "port1"
        set dnsfilter-profile "dnsfilter"
    next
end
```

4. Send a DNS query over TLS (this example uses kdig on an Ubuntu client) using the FortiGate as the DNS server. The www.ubc.ca domain belongs to the education category:

```
root@client:/tmp# kdig -d @10.1.100.173 +tls +header +all www.ubc.ca
;; DEBUG: Querying for owner(www.ubc.ca.), class(1), type(1), server(10.1.100.173), port
(853), protocol(TCP)
;; DEBUG: TLS, received certificate hierarchy:
;; DEBUG:  #1,
C=US,ST=California,L=Sunnyvale,O=Fortinet,OU=FortiGate,CN=FG3H1E5818903681,EMAIL=support
@fortinet.com
;; DEBUG:      SHA-256 PIN: Xhkpv9ABEhxDLtWG+lGEndNrBR7B1xjRYlGn2ltlkb8=
;; DEBUG:  #2, C=US,ST=California,L=Sunnyvale,O=Fortinet,OU=Certificate
Authority,CN=fortinet-subca2001,EMAIL=support@fortinet.com
;; DEBUG:      SHA-256 PIN: 3T8EqFBjpRSkxQNPFagjUNeEUghXOEYp904ROlJM8yo=
;; DEBUG:  #3, C=US,ST=California,L=Sunnyvale,O=Fortinet,OU=Certificate
Authority,CN=fortinet-ca2,EMAIL=support@fortinet.com
;; DEBUG:      SHA-256 PIN: /QfV4N3k5oxQR5RHtW/rbn/HrHgKpMLN0DEaeXY5yPg=
;; DEBUG: TLS, skipping certificate PIN check
;; DEBUG: TLS, skipping certificate verification
;; TLS session (TLS1.2)-(ECDHE-RSA-SECP256R1)-(AES-256-GCM)
;; ->>HEADER<<- opcode: QUERY; status: NOERROR; id: 56719
;; Flags: qr rd; QUERY: 1; ANSWER: 1; AUTHORITY: 0; ADDITIONAL: 0

;; QUESTION SECTION:
;; www.ubc.ca.                  IN      A

;; ANSWER SECTION:
www.ubc.ca.             60      IN      A       208.91.112.55

;; Received 44 B
;; Time 2021-03-12 23:11:27 PST
;; From 10.1.100.173@853(TCP) in 0.2 ms
root@client:/tmp#
```

The IP returned by the FortiGate for ubc.ca belongs to the FortiGuard block page, so the query was blocked successfully.

DoH

The following example uses a DNS filter profile where the education category is blocked.

To configure scanning DoH traffic in explicit mode with a DNS filter:

1. Configure the DNS settings:

```
config system dns
    set primary 1.1.1.1
    set secondary 1.0.0.1
```

```
        set protocol doh
    end
```

2. Configure the DNS filter profile:

```
config dnsfilter profile
    edit "dnsfilter"
        config ftgd-dns
            config filters
                edit 1
                    set category 30
                    set action block
                next
            end
        end
    next
end
```

3. Configure the DNS server settings:

```
config system dns-server
    edit "port1"
        set dnsfilter-profile "dnsfilter"
        set doh enable
    next
end
```

4. In your browser, enable DNS over HTTPS.
5. On your computer, edit the TCP/IP settings to use the FortiGate interface address as the DNS server.
6. In your browser, go to a website in the education category (www.ubc.ca). The website is redirected to the block page.

DNS troubleshooting

The following diagnose command can be used to collect DNS debug information. If you do not specify worker ID, the default worker ID is `0`.

```
# diagnose test application dnsproxy
worker idx: 0
1. Clear DNS cache
2. Show stats
3. Dump DNS setting
4. Reload FQDN
5. Requery FQDN
6. Dump FQDN
7. Dump DNS cache
8. Dump DNS DB
9. Reload DNS DB
10. Dump secure DNS policy/profile
11. Dump Botnet domain
12. Reload Secure DNS setting
```

```
13. Show Hostname cache
14. Clear Hostname cache
15. Show SDNS rating cache
16. Clear SDNS rating cache
17. DNS debug bit mask
18. DNS debug obj mem
99. Restart dnsproxy worker
```

To view useful information about the ongoing DNS connection:

```
# diagnose test application dnsproxy 3
worker idx: 0
vdom: root, index=0, is primary, vdom dns is disabled, mip-169.254.0.1 dns_log=1 tls=0 cert=
dns64 is disabled
vdom: vdom1, index=1, is primary, vdom dns is enabled, mip-169.254.0.1 dns_log=1 tls=0 cert=
dns64 is disabled
dns-server:96.45.45.220:45 tz=-480 tls=0 req=0 to=0 res=0 rt=0 rating=1 ready=0 timer=37
probe=9 failure=0 last_failed=0
dns-server:8.8.8.8:53 tz=0 tls=0 req=73 to=0 res=73 rt=5 rating=0 ready=1 timer=0 probe=0
failure=0 last_failed=0
dns-server:65.39.139.63:53 tz=0 tls=0 req=39 to=0 res=39 rt=1 rating=0 ready=1 timer=0
probe=0 failure=0 last_failed=0
dns-server:62.209.40.75:53 tz=60 tls=0 req=0 to=0 res=0 rt=0 rating=1 ready=0 timer=37
probe=9 failure=0 last_failed=0
dns-server:209.222.147.38:53 tz=-300 tls=0 req=0 to=0 res=0 rt=0 rating=1 ready=0 timer=37
probe=9 failure=0 last_failed=0
dns-server:173.243.138.221:53 tz=-480 tls=0 req=0 to=0 res=0 rt=0 rating=1 ready=0 timer=37
probe=9 failure=0 last_failed=0
dns-server:45.75.200.89:53 tz=0 tls=0 req=0 to=0 res=0 rt=0 rating=1 ready=0 timer=37
probe=9 failure=0 last_failed=0
DNS_CACHE: hash-size=2048, ttl=1800, min-ttl=60, max-num=-1
DNS FD: udp_s=12 udp_c=17:18 ha_c=22 unix_s=23, unix_nb_s=24, unix_nc_s=25
        v6_udp_s=11, v6_udp_c=20:21, snmp=26, redir=13, v6_redir=14
DNS FD: tcp_s=29, tcp_s6=27, redir=31 v6_redir=32
FQDN: hash_size=1024, current_query=1024
DNS_DB: response_buf_sz=131072
LICENSE: expiry=2015-04-08, expired=1, type=2
FDG_SERVER:96.45.45.220:45
FGD_CATEGORY_VERSION:8
SERVER_LDB: gid=eb19, tz=-480, error_allow=0
FGD_REDIR_V4:208.91.112.55 FGD_REDIR_V6:
```

Important fields include:

`tls`	1 if the connection is TLS, 0 if the connection is not TLS.
`rt`	The round trip time of the DNS latency.
`probe`	The number of probes sent.

To dump the second DNS worker's cache:

```
diagnose test application dnsproxy 7 1
```

To enable debug on the second worker:

```
diagnose debug application dnsproxy -1 1
```

To enable debug on all workers by specifying -1 as worker ID:

```
diagnose debug application dnsproxy -1 -1
```

Explicit and transparent proxies

This section contains instructions for configuring explicit and transparent proxies.

Explicit web proxy

Explicit web proxy can be configured on FortiGate for proxying HTTP and HTTPS traffic.

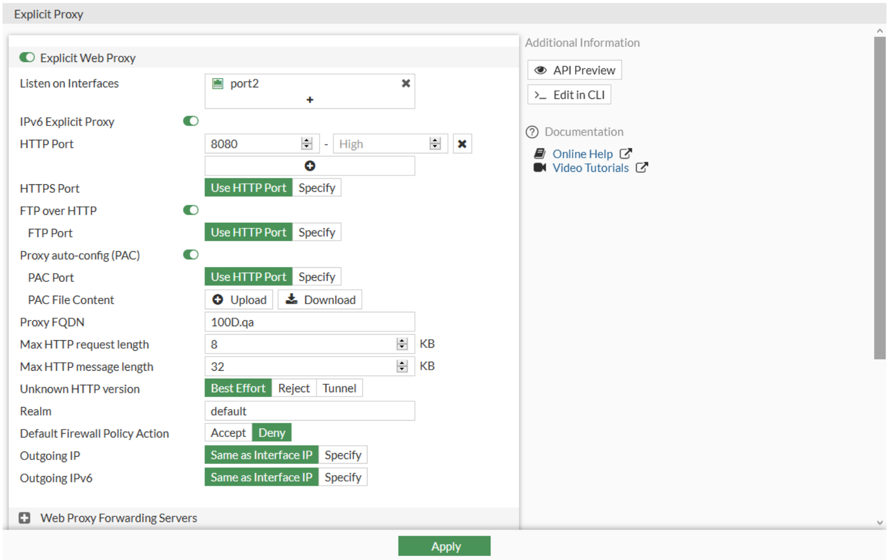
To deploy explicit proxy, individual client browsers can be manually configured to send requests directly to the proxy, or they can be configured to download proxy configuration instructions from a Proxy Auto-Configuration (PAC) file.

When explicit proxy is configured on an interface, the interface IP address can be used by client browsers to forward requests directly to the FortiGate. FortiGate also supports PAC file configuration.

To configure explicit web proxy in the GUI:

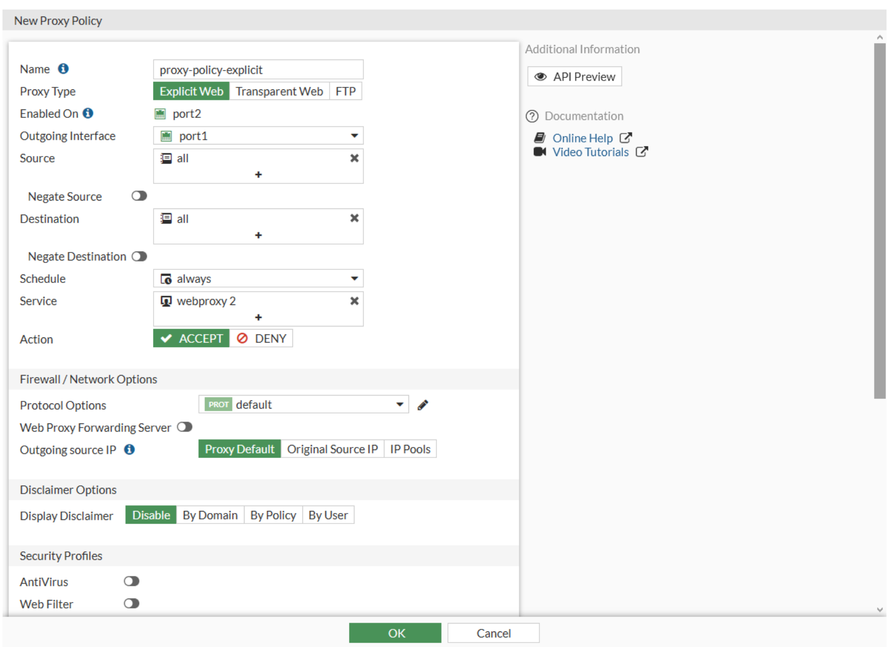
1. Enable and configure explicit web proxy:
   a. Go to *Network > Explicit Proxy*.
   b. Enable *Explicit Web Proxy*.

c. Select *port2* as the *Listen on Interfaces* and set the *HTTP Port* to *8080*.
d. Configure the remaining settings as needed.



e. Click *Apply*.

2. Create an explicit web proxy policy:
   a. Go to *Policy & Objects > Proxy Policy*.
   b. Click *Create New*.
   c. Set *Proxy Type* to *Explicit Web* and *Outgoing Interface* to *port1*.
   d. Also set *Source* and *Destination* to *all*, *Schedule* to *always*, *Service* to *webproxy*, and *Action* to *ACCEPT*.



   e. Click *OK* to create the policy.

This example creates a basic policy. If required, security profiles can be enabled, and deep SSL inspection can be selected to inspect HTTPS traffic.

3. Configure a client to use the FortiGate explicit proxy:

   Set the FortiGate IP address as the proxy IP address in the browser, or use an automatic configuration script for the PAC file.

To configure explicit web proxy in the CLI:

1. Enable and configure explicit web proxy:

```
config web-proxy explicit
    set status enable
    set ftp-over-http enable
    set socks enable
    set http-incoming-port 8080
    set ipv6-status enable
    set unknown-http-version best-effort
end
config system interface
    edit "port2"
        set vdom "vdom1"
        set ip 10.1.100.1 255.255.255.0
        set allowaccess ping https ssh snmp http telnet
        set type physical
        set explicit-web-proxy enable
        set snmp-index 12
        end
    next
end
```

2. Create an explicit web proxy policy:

```
config firewall proxy-policy
    edit 1
        set name "proxy-policy-explicit"
        set proxy explicit-web
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set service "webproxy"
        set action accept
        set schedule "always"
        set logtraffic all
    next
end
```

This example creates a basic policy. If required, security profiles can be enabled, and deep SSL inspection can be selected to inspect HTTPS traffic.

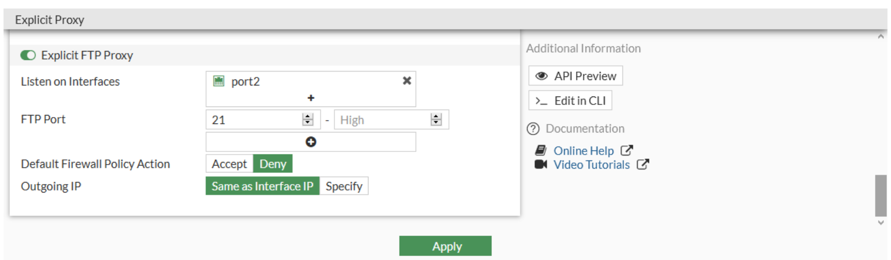
3. Configure a client to use the FortiGate explicit web proxy:
   Set the FortiGate IP address as the proxy IP address in the browser, or use an automatic configuration script for the PAC file.

FTP proxy

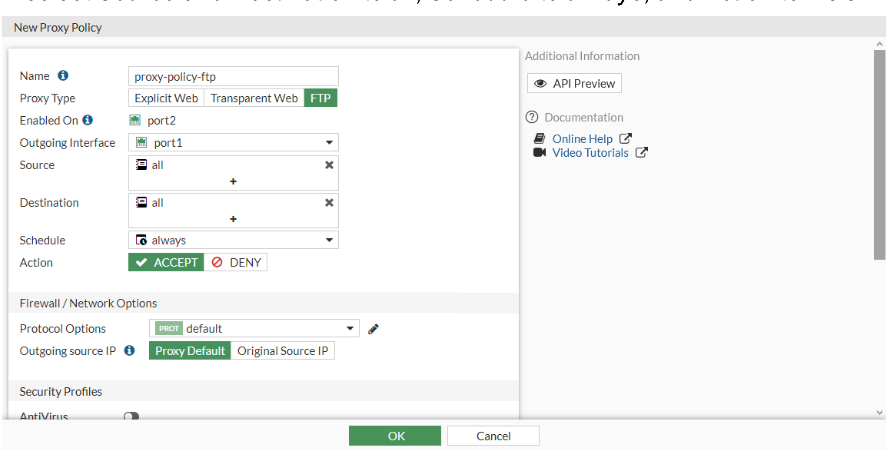
FTP proxies can be configured on the FortiGate so that FTP traffic can be proxied. When the FortiGate is configured as an FTP proxy, FTP client applications should be configured to send FTP requests to the FortiGate.

To configure explicit FTP proxy in the GUI:

1. Enable and configure explicit FTP proxy:
   a. Go to *Network > Explicit Proxy*.
   b. Enable *Explicit FTP Proxy*.
   c. Select *port2* as the *Listen on Interfaces* and set the *HTTP Port* to *21*.
   d. Configure the *Default Firewall Policy Action* as needed.



   e. Click *Apply*.
2. Create an explicit FTP proxy policy:
   a. Go to *Policy & Objects > Proxy Policy*.
   b. Click *Create New*.
   c. Set *Proxy Type* to *FTP* and *Outgoing Interface* to *port1*.
   d. Also set *Source* and *Destination* to *all*, *Schedule* to *always*, and *Action* to *ACCEPT*.



e. Click *OK* to create the policy.

This example creates a basic policy. If required, security profiles can be enabled.

3. Configure the FTP client application to use the FortiGate IP address.

To configure explicit FTP proxy in the CLI:

1. Enable and configure explicit FTP proxy:

```
config ftp-proxy explicit
    set status enable
    set incoming-port 21
end
config system interface
    edit "port2"
        set vdom "vdom1"
        set ip 10.1.100.1 255.255.255.0
        set allowaccess ping https ssh snmp http telnet
        set type physical
        set explicit-ftp-proxy enable
        set snmp-index 12
    next
end
```

2. Create an explicit FTP proxy policy:

```
config firewall proxy-policy
    edit 4
        set name "proxy-policy-ftp"
        set proxy ftp
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
    next
end
```

This example creates a basic policy. If required, security profiles can be enabled.

3. Configure the FTP client application to use the FortiGate IP address.

Transparent proxy

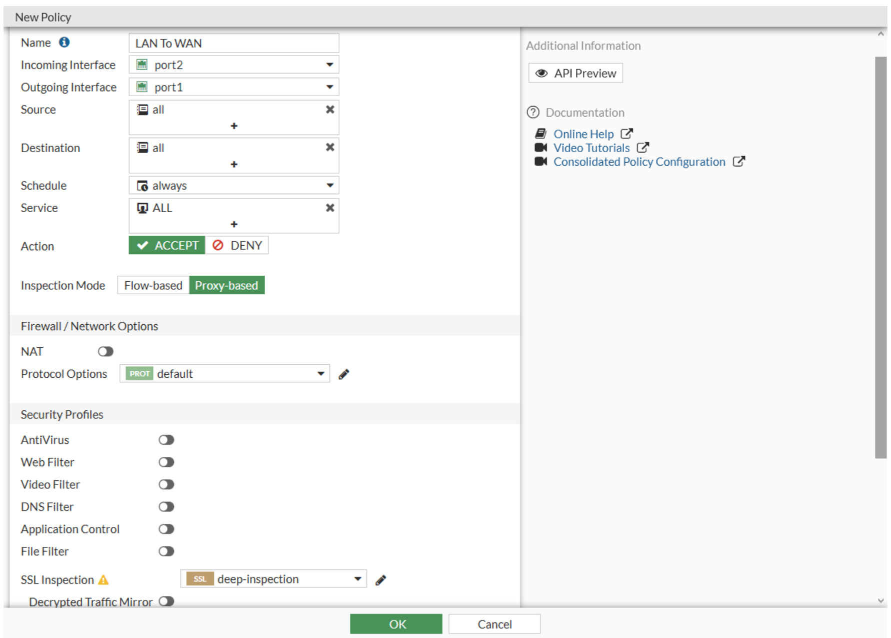
In a transparent proxy deployment, the user's client software, such as a browser, is unaware that it is communicating with a proxy.

Users request internet content as usual, without any special client configuration, and the proxy serves their requests. FortiGate also allows users to configure in transparent proxy mode.

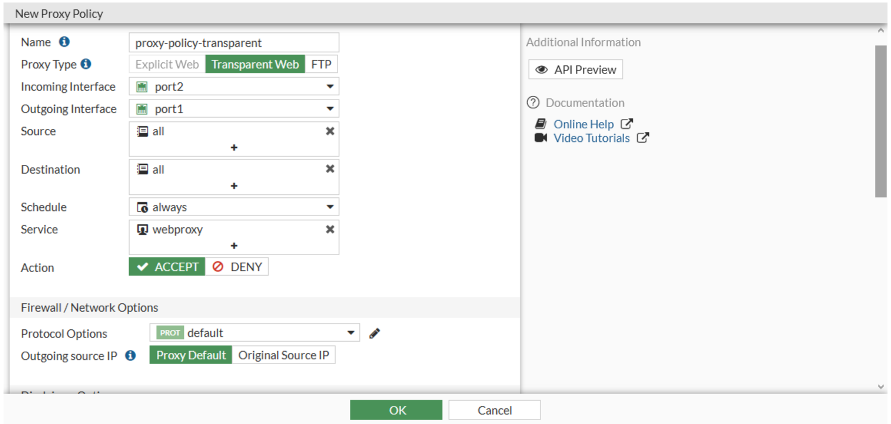
To redirect HTTPS traffic, SSL inspection is required.

To configure transparent proxy in the GUI:

1. Configure a regular firewall policy with HTTP redirect:
   a. Go to *Policy & Objects* > *Firewall Policy*.
   b. Click *Create New*.
   c. Name the policy appropriately, set the *Incoming Interface* to *port2*, and set the *Outgoing Interface* to *port1*.
   d. Also set *Source* and *Destination* to *all*, *Schedule* to *always*, *Service* to *ALL*, and *Action* to *ACCEPT*.
   e. Set *Inspection Mode* to *Proxy-based* and *SSL Inspection* to *deep-inspection*.



   f. Configure the remaining settings as needed.
   g. Click *OK*.
2. Configure a transparent proxy policy:
   a. Go to *Policy & Objects* > *Proxy Policy*.
   b. Click *Create New*.
   c. Set *Proxy Type* to *Transparent Web*, set the *Incoming Interface* to *port2*, and set the *Outgoing Interface* to *port1*.
   d. Also set *Source* and *Destination* to *all*, *Schedule*to *always*, *Service* to *webproxy*, and *Action* to *ACCEPT*.



e. Configure the remaining settings as needed.

f. Click *OK* to create the policy.

3. No special configuration is required on the client to use FortiGate transparent proxy. As the client is using the FortiGate as its default gateway, requests will first hit the regular firewall policy, and then be redirected to the transparent proxy policy.

To configure transparent proxy in the CLI:

1. Configure a regular firewall policy with HTTP redirect:

```
config firewall policy
    edit 1
        set name "LAN To WAN"
        set srcintf "port2"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set inspection-mode proxy
        set http-policy-redirect enable
        set fsso disable
        set ssl-ssh-profile "deep-inspection"
        set nat enable
    next
end
```

2. Configure a transparent proxy policy:

```
config firewall proxy-policy
    edit 5
        set name "proxy-policy-transparent"
        set proxy transparent-web
        set srcintf "port2"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set service "webproxy"
        set action accept
        set schedule "always"
```

```
    next
end
```

This example creates a basic policy. If required, security profiles can be enabled, and deep SSL inspection can be selected to inspect HTTPS traffic.

3. No special configuration is required on the client to use FortiGate transparent proxy. As the client is using the FortiGate as its default gateway, requests will first hit the regular firewall policy, and then be redirected to the transparent proxy policy.

Proxy policy addresses

Proxy addresses are designed to be used only by proxy policies. The following address types are available:

- Host regex match on page 227
- URL pattern on page 228
- URL category on page 229
- HTTP method on page 230
- HTTP header on page 230
- User agent on page 231
- Advanced (source) on page 232
- Advanced (destination) on page 233

Fast policy match

The fast policy match function improves the performance of IPv4 explicit and transparent web proxies on FortiGate devices.

When enabled, after the proxy policies are configured, the FortiGate builds a fast searching table based on the different proxy policy matching criteria. When fast policy matching is disabled, web proxy traffic is compared to the policies one at a time from the beginning of the policy list.

Fast policy matching is enabled by default, and can be configured with the following CLI command:

```
config web-proxy global
    set fast-policy-match {enable | disable}
end
```

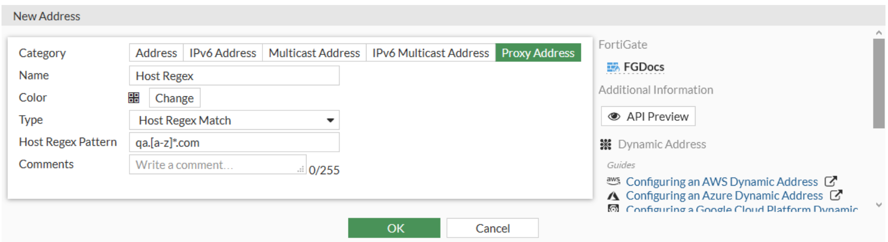
Host regex match

In this address type, a user can create a hostname as a regular expression. Once created, the hostname address can be selected as a destination of a proxy policy. This means that a policy will only allow or block requests that match the regular expression.

This example creates a host regex match address with the pattern *qa.[a-z]*.com*.

To create a host regex match address in the GUI:

1. Go to *Policy & Objects > Addresses*.
2. Click *Create New > Address*.
3. Set the following:
   - *Category* to *Proxy Address*,
   - *Name* to *Host Regex*,
   - *Type* to *Host Regex Match*, and
   - *Host Regex Pattern* to *qa.[a-z]*.com*.



4. Click *OK*.

To create a host regex match address in the CLI:

```
config firewall proxy-address
    edit "Host Regex"
        set type host-regex
        set host-regex "qa.[a-z]*.com"
    next
end
```

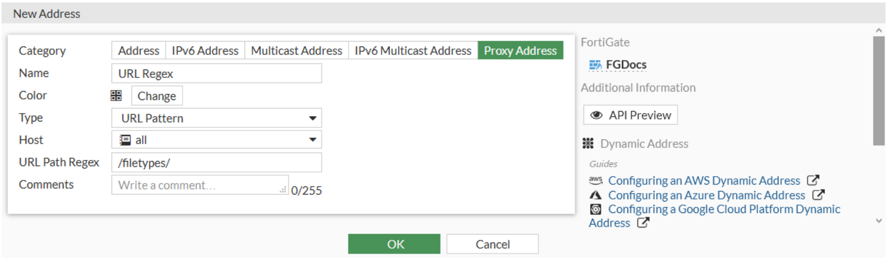
URL pattern

In this address type, a user can create a URL path as a regular expression. Once created, the path address can be selected as a destination of a proxy policy. This means that a policy will only allow or block requests that match the regular expression.

This example creates a URL pattern address with the pattern */filetypes/*.

To create a URL pattern address in the GUI:

1. Go to *Policy & Objects > Addresses*.
2. Click *Create New > Address*.
3. Set the following:
   - *Category* to *Proxy Address*,
   - *Name* to *URL Regex*,
   - *Type* to *URL Pattern*,
   - *Host* to *all*, and
   - *URL Path Regex* to */filetypes/*.



4. Click *OK*.

To create a URL pattern address in the CLI:

```
config firewall proxy-address
    edit "URL Regex"
        set type url
        set host "all"
        set path "/filetypes/"
    next
end
```

URL category

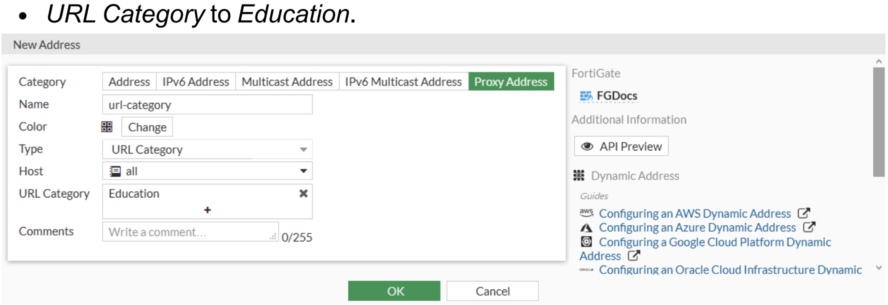
In this address type, a user can create a URL category based on a FortiGuard URL ID. Once created, the address can be selected as a destination of a proxy policy. This means that a policy will only allow or block requests that match the URL category.

The example creates a URL category address for URLs in the *Education* category. For more information about categories, see https://fortiguard.com/webfilter/categories.

For information about creating and using custom local and remote categories, see Web rating override on page 1318 and Threat feeds on page 2504.

To create a URL category address in the GUI:

1. Go to *Policy & Objects > Addresses*.
2. Click *Create New > Address*.
3. Set the following:
   - *Category* to *Proxy Address*,
   - *Name* to *url-category*,
   - *Type* to *URL Category*,
   - *Host* to *all*, and
   - *URL Category* to *Education*.



4. Click *OK*.

To create a URL category address in the CLI:

```
config firewall proxy-address
    edit "url-category"
        set type category
        set host "all"
        set category 30
    next
end
```

To see a list of all the categories and their numbers, when editing the address, enter `set category ?`.

HTTP method

In this address type, a user can create an address based on the HTTP request methods that are used. Multiple method options are supported, including: *CONNECT*, *DELETE*, *GET*, *HEAD*, *OPTIONS*, *POST*, *PUT*, and *TRACE*. Once created, the address can be selected as a source of a proxy policy. This means that a policy will only allow or block requests that match the selected HTTP method.

The example creates a HTTP method address that uses the GET method.

To create a HTTP method address in the GUI:

1. Go to *Policy & Objects > Addresses*.
2. Click *Create New > Address*.
3. Set the following:
   - *Category* to *Proxy Address*,
   - *Name* to *method_get*,
   - *Type* to *HTTP Method*,
   - *Host* to *all*, and
   - *Request Method* to *GET*.
4. Click *OK*.

To create a HTTP method address in the CLI:

```
config firewall proxy-address
    edit "method_get"
        set type method
        set host "all"
        set method get
    next
end
```

HTTP header

In this address type, a user can create a HTTP header as a regular expression. Once created, the header address can be selected as a source of a proxy policy. This means that a policy will only allow or block requests where the HTTP header matches the regular expression.

This example creates a HTTP header address with the pattern *Q[A-B]*.

To create a HTTP header address in the GUI:

1. Go to *Policy & Objects > Addresses*.
2. Click *Create New > Address*.
3. Set the following:
   - *Category* to *Proxy Address*,
   - *Name* to *HTTP-header*,
   - *Type* to *HTTP Header*,
   - *Host* to *all*,
   - *Header Name* to *Header_Test*, and
   - *Header Regex* to *Q[A-B]*.
4. Click *OK*.

To create a HTTP header address in the CLI:

```
config firewall proxy-address
    edit "method_get"
        set type header
        set host "all"
        set header-name "Header_Test"
        set header "Q[A-B]"
    next
end
```

User agent

In this address type, a user can create an address based on the names of the browsers that are used as user agents. Multiple browsers are supported, such as Chrome, Firefox, Internet Explorer, and others. Once created, the address can be selected as a source of a proxy policy. This means that a policy will only allow or block requests from the specified user agent.

This example creates a user agent address for Google Chrome.

To create a user agent address in the GUI:

1. Go to *Policy & Objects > Addresses*.
2. Click *Create New > Address*.
3. Set the following:
   - *Category* to *Proxy Address*,
   - *Name* to *UA-Chrome*,
   - *Type* to *User Agent*,
   - *Host* to *all*, and
   - *User Agent* to *Google Chrome*.
4. Click *OK*.

To create a user agent address in the CLI:

```
config firewall proxy-address
    edit "UA-Chrome"
        set type ua
        set host "all"
        set ua chrome
    next
end
```

Advanced (source)

In this address type, a user can create an address based on multiple parameters, including HTTP method, User Agent, and HTTP header. Once created, the address can be selected as a source of a proxy policy. This means that a policy will only allow or block requests that match the selected address.

This example creates an address that uses the get method, a user agent for Google Chrome, and an HTTP header with the pattern *Q[A-B]*.

To create an advanced (source) address in the GUI:

1. Go to *Policy & Objects > Addresses*.
2. Click *Create New > Address*.
3. Set the following:
    - *Category* to *Proxy Address*,
    - *Name* to *advanced_src*,
    - *Type* to *Advanced (Source)*,
    - *Host* to *all*,
    - *Request Method* to *GET*,
    - *User Agent* to *Google Chrome*, and
    - *HTTP header* to *Header_Test* : *Q[A-B]*.
4. Click *OK*.

To create an advanced (source) address in the CLI:

```
config firewall proxy-address
    edit "advance_src"
        set type src-advanced
        set host "all"
        set method get
        set ua chrome
        config header-group
            edit 1
                set header-name "Header_Test"
                set header "Q[A-B]"
            next
        end
    next
end
```

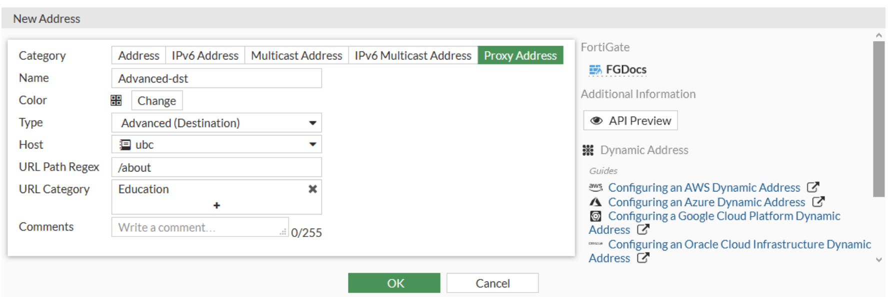
Advanced (destination)

In this address type, a user can create an address based on URL pattern and URL category parameters. Once created, the address can be selected as a destination of a proxy policy. This means that a policy will only allow or block requests that match the selected address.

This example creates an address with the URL pattern */about* that are in the *Education* category. For more information about categories, see https://fortiguard.com/webfilter/categories.

To create an advanced (destination) address in the GUI:

1. Go to *Policy & Objects > Addresses*.
2. Click *Create New > Address*.
3. Set the following:
   - *Category* to *Proxy Address*,
   - *Name* to *Advanced-dst*,
   - *Type* to *Advanced (Destination)*,
   - *Host* to *all*,
   - *URL Path Regex* to */about*, and
   - *URL Category* to *Education*.



4. Click *OK*.

To create an advanced (destination) address in the CLI:

```
config firewall proxy-address
    edit "Advanced-dst"
        set type dst-advanced
        set host "ubc"
        set path "/about"
        set category 30
    next
end
```

Proxy policy security profiles

Web proxy policies support most security profile types.

Security profiles must be created before they can be used in a policy, see Security Profiles on page 1084 for information.

Explicit web proxy policy

The security profiles supported by explicit web proxy policies are:

- *AntiVirus*
- *Web Filter*
- *Video Filter*
- *Application Control*
- *IPS*
- *DLP Profile*
- *ICAP*
- *Web Application Firewall*
- *File Filter*
- *SSL Inspection*

To configure security profiles on an explicit web proxy policy in the GUI:

1. Go to *Policy & Objects > Proxy Policy*.
2. Click *Create New*.
3. Set the following:

Proxy Type	Explicit Web
Outgoing Interface	port1
Source	all
Destination	all
Schedule	always
Service	webproxy
Action	ACCEPT

4. In the *Firewall / Network Options* section, set *Protocol Options* to *default*.
5. In the *Security Profiles* section, make the following selections (for this example, these profiles have all already been created):

AntiVirus	av
Web Filter	urlfiler
Application Control	app
IPS	Sensor-1
DLP Profile	dlp
ICAP	default
Web Application Firewall	default
SSL Inspection	deep-inspection

6. Click *OK* to create the policy.

To configure security profiles on an explicit web proxy policy in the CLI:

```
config firewall proxy-policy
    edit 1
        set proxy explicit-web
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set service "web"
        set action accept
        set schedule "always"
        set utm-status enable
        set av-profile "av"
        set webfilter-profile "urlfilter"
        set dlp-profile "dlp"
        set ips-sensor "sensor-1"
        set application-list "app"
        set icap-profile "default"
        set waf-profile "default"
        set ssl-ssh-profile "deep-inspection"
    next
end
```

Transparent proxy

The security profiles supported by transparent proxy policies are:

- *AntiVirus*
- *Web Filter*
- *Video Filter*
- *Application Control*
- *IPS*
- *DLP Profile*
- *ICAP*
- *Web Application Firewall*
- *File Filter*
- *SSL Inspection*

To configure security profiles on a transparent proxy policy in the GUI:

1. Go to *Policy & Objects > Proxy Policy*.
2. Click *Create New*.
3. Set the following:

Proxy Type	Transparent Web
Incoming Interfae	port2
Outgoing Interface	port1
Source	all

Destination	all
Schedule	always
Service	webproxy
Action	ACCEPT

4. In the *Firewall / Network Options* section, set *Protocol Options* to *default*.
5. In the *Security Profiles* section, make the following selections (for this example, these profiles have all already been created):

AntiVirus	av
Web Filter	urlfiler
Application Control	app
IPS	Sensor-1
DLP Profile	dlp
ICAP	default
Web Application Firewall	default
SSL Inspection	deep-inspection

6. Click *OK* to create the policy.

To configure security profiles on a transparent proxy policy in the CLI:

```
config firewall proxy-policy
    edit 2
        set proxy transparent-web
        set srcintf "port2"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set service "webproxy"
        set action accept
        set schedule "always"
        set utm-status enable
        set av-profile "av"
        set webfilter-profile "urlfilter"
        set dlp-profile "dlp"
        set ips-sensor "sensor-1"
        set application-list "app"
        set icap-profile "default"
        set waf-profile "default"
        set ssl-ssh-profile "certificate-inspection"
    next
end
```

FTP proxy

The security profiles supported by FTP proxy policies are:

- *AntiVirus*
- *Application Control*
- *IPS*
- *File Filter*
- *DLP Profile*

To configure security profiles on an FTP proxy policy in the GUI:

1. Go to *Policy & Objects > Proxy Policy*.
2. Click *Create New*.
3. Set the following:

Proxy Type	FTP
Outgoing Interface	port1
Source	all
Destination	all
Schedule	always
Action	ACCEPT

4. In the *Firewall / Network Options* section, set *Protocol Options* to *default*.
5. In the *Security Profiles* section, make the following selections (for this example, these profiles have all already been created):

AntiVirus	av
Application Control	app
IPS	Sensor-1
DLP Profile	dlp

6. Click *OK* to create the policy.

To configure security profiles on an FTP proxy policy in the CLI:

```
config firewall proxy-policy
    edit 3
        set proxy ftp
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set utm-status enable
        set av-profile "av"
        set dlp-profile "dlp"
```

```
        set ips-sensor "sensor-1"
        set application-list "app"
    next
end
```

Explicit proxy authentication

FortiGate supports multiple authentication methods. This topic explains using an external authentication server with Kerberos as the primary and NTLM as the fallback.

To configure Explicit Proxy with authentication:

1. Enable and configure the explicit proxy on page 238.
2. Configure the authentication server and create user groups on page 239.
3. Create an authentication scheme and rules on page 241.
4. Create an explicit proxy policy and assign a user group to the policy on page 242.
5. Verify the configuration on page 242.

Enable and configure the explicit proxy

To enable and configure explicit web proxy in the GUI:

1. Go to *Network > Explicit Proxy*.
2. Enable *Explicit Web Proxy*.
3. Select *port2* as the *Listen on Interfaces* and set the *HTTP Port* to *8080*.
4. Configure the remaining settings as needed.
5. Click *Apply*.

To enable and configure explicit web proxy in the CLI:

```
config web-proxy explicit
    set status enable
    set ftp-over-http enable
    set socks enable
    set http-incoming-port 8080
    set ipv6-status enable
    set unknown-http-version best-effort
end
config system interface
    edit "port2"
        set vdom "vdom1"
        set ip 10.1.100.1 255.255.255.0
        set allowaccess ping https ssh snmp http telnet
        set type physical
        set explicit-web-proxy enable
        set snmp-index 12
        end
    next
end
```

Configure the authentication server and create user groups

Since we are using an external authentication server with Kerberos authentication as the primary and NTLM as the fallback, Kerberos authentication is configured first and then FSSO NTLM authentication is configured.

For successful authorization, the FortiGate checks if user belongs to one of the groups that is permitted in the security policy.

To configure an authentication server and create user groups in the GUI:

1. Configure Kerberos authentication:
   a. Go to *User & Authentication > LDAP Servers*.
   b. Click *Create New*.
   c. Set the following:

Name	ldap-kerberos
Server IP	172.18.62.220
Server Port	389
Common Name Identifier	cn
Distinguished Name	dc=fortinetqa,dc=local

   d. Click *OK*
2. Define Kerberos as an authentication service. This option is only available in the CLI. For information on generating a keytab, see Generating a keytab on a Windows server on page 243.
3. Configure FSSO NTLM authentication:
   FSSO NTLM authentication is supported in a Windows AD network. FSSO can also provide NTLM authentication service to the FortiGate unit. When a user makes a request that requires authentication, the FortiGate initiates NTLM negotiation with the client browser, but does not process the NTLM packets itself. Instead, it forwards all the NTLM packets to the FSSO service for processing.
   a. Go to *Security Fabric > External Connectors*.
   b. Click *Create New* and select *FSSO Agent on Windows AD* from the *Endpoint/Identity* category.
   c. Set the *Name* to *FSSO*, *Primary FSSO Agent* to *172.16.200.220*, and enter a password.
   d. Click *OK*.
4. Create a user group for Kerberos authentication:
   a. Go to *User & Authentication > User Groups*.
   b. Click *Create New*.
   c. Set the *Name* to *Ldap-Group*, and *Type* to *Firewall*.
   d. In the *Remote Groups* table, click *Add*, and set the *Remote Server* to the previously created *ldap-kerberos* server.
   e. Click *OK*.
5. Create a user group for NTLM authentication:
   a. Go to *User & Authentication > User Groups*.
   b. Click *Create New*.
   c. Set the *Name* to *NTLM-FSSO-Group*, *Type* to *Fortinet Single Sign-On (FSSO)*, and add *FORTINETQA/FSSO*

as a member.

d. Click *OK*.

To configure an authentication server and create user groups in the CLI:

1. Configure Kerberos authentication:

```
config user ldap
    edit "ldap-kerberos"
        set server "172.18.62.220"
        set cnid "cn"
        set dn "dc=fortinetqa,dc=local"
        set type regular
        set username "CN=root,CN=Users,DC=fortinetqa,DC=local"
        set password *********
    next
end
```

2. Define Kerberos as an authentication service:

```
config user krb-keytab
    edit "http_service"
        set pac-data disable
        set principal "HTTP/FGT.FORTINETQA.LOCAL@FORTINETQA.LOCAL"
        set ldap-server "ldap-kerberos"
        set keytab
"BQIAAABFAAIAEEZPUlRJTkVUUUEuTE9DQUwABEhUVFAAFEZHVC5GT1JUSU5FVFFBLkxPQ0FMAAAAAQAAAAAEAAE
ACKLCMonpitnVAAAARQACABBGT1JUSU5FVFFBLkxPQ0FMAARIVFRQABRGR1QuRk9SVElORVRRQS5MT0NBTAAAAAE
AAAAABAADAAiiwjKJ6YrZ1QAAAE0AAgAQRk9SVElORVRRQS5MT0NBTAAESFRUUAAURkdULkZPUlRJTkVUUUEuTE9
DQUwAAAABAAAAAAQAFwAQUHo9uqR9cSkzyxdzKCEXdwAAAF0AAgAQRk9SVElORVRRQS5MT0NBTAAESFRUUAAURkd
ULkZPUlRJTkVUUUEuTE9DQUwAAAABAAAAAAQAEgAgzee854Aq1HhQiKJZvV4tL2Poy7hMIARQpK8MCB//BIAAAAB
NAAIAEEZPUlRJTkVUUUEuTE9DQUwABEhUVFAAFEZHVC5GT1JUSU5FVFFBLkxPQ0FMAAAAAQAAAAAEABEAEG49vHE
iiBghr63Z/lnwYrU="
    next
end
```

For information on generating a keytab, see Generating a keytab on a Windows server on page 243.

3. Configure FSSO NTLM authentication:

```
config user fsso
    edit "1"
        set server "172.18.62.220"
        set password *********
    next
end
```

4. Create a user group for Kerberos authentication:

```
config user group
    edit "Ldap-Group"
        set member "ldap" "ldap-kerberos"
    next
end
```

5. Create a user group for NTLM authentication:

```
config user group
    edit "NTLM-FSSO-Group"
```

```
        set group-type fsso-service
        set member "FORTINETQA/FSSO"
    next
end
```

Create an authentication scheme and rules

Explicit proxy authentication is managed by authentication schemes and rules. An authentication scheme must be created first, and then the authentication rule.

To create an authentication scheme and rules in the GUI:

1. Create an authentication scheme:
   a. Go to *Policy & Objects* > *Authentication Rules*.
   b. Click *Create New* > *Authentication Schemes*.
   c. Set the *Name* to *Auth-scheme-Negotiate* and select *Negotiate* as the *Method*.
   d. Click *OK*.
2. Create an authentication rule:
   a. Go to *Policy & Objects* > *Authentication Rules*.
   b. Click *Create New* > *Authentication Rules*.
   c. Set the *Name* to *Auth-Rule*, *Source Address* to *all*, and *Protocol* to *HTTP*.
   d. Enable *Authentication Scheme*, and select the just created *Auth-scheme-Negotiate* scheme.
   e. Click *OK*.

To create an authentication scheme and rules in the CLI:

1. Create an authentication scheme:

```
config authentication scheme
    edit "Auth-scheme-Negotiate"
        set method negotiate       <<< Accepts both Kerberos and NTLM as fallback
    next
end
```

2. Create an authentication rule:

```
config authentication rule
    edit "Auth-Rule"
        set status enable
        set protocol http
        set srcaddr "all"
        set ip-based enable
        set active-auth-method "Auth-scheme-Negotiate"
        set comments "Testing"
    next
end
```

Create an explicit proxy policy and assign a user group to the policy

To create an explicit proxy policy and assign a user group to it in the GUI:

1. Go to *Policy & Objects* > *Proxy Policy*.
2. Click *Create New*.
3. Set *Proxy Type* to *Explicit Web* and *Outgoing Interface* to *port1*.
4. Set *Source* to *all*, and the just created user groups *NTLM-FSSO-Group* and *Ldap-Group*.
5. Also set *Destination* to *all*, *Schedule* to *always*, *Service* to *webproxy*, and *Action* to *ACCEPT*.
6. Click *OK*.

To create an explicit proxy policy and assign a user group to it in the CLI:

```
config firewall proxy-policy
    edit 1
        set proxy explicit-web
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set service "web"
        set action accept
        set schedule "always"
        set logtraffic all
        set groups "NTLM-FSSO-Group" "Ldap-Group"
        set av-profile "av"
        set ssl-ssh-profile "deep-custom"
    next
end
```

Verify the configuration

Log in using a domain and system that would be authenticated using the Kerberos server, then enter the `diagnose wad user list` CLI command to verify:

```
# diagnose wad user list
ID: 8, IP: 10.1.100.71, VDOM: vdom1
  user name   : test1@FORTINETQA.LOCAL
  duration    : 389
  auth_type   : IP
  auth_method : Negotiate
  pol_id      : 1
  g_id        : 1
  user_based  : 0
  expire      : no
  LAN:
    bytes_in=4862 bytes_out=11893
  WAN:
    bytes_in=7844 bytes_out=1023
```

Log in using a system that is not part of the domain. The NTLM fallback server should be used:

```
# diagnose wad user list
ID: 2, IP: 10.1.100.202, VDOM: vdom1
```

```
user name   : TEST31@FORTINETQA
duration    : 7
auth_type   : IP
auth_method : NTLM
pol_id      : 1
g_id        : 5
user_based  : 0
expire      : no
LAN:
  bytes_in=6156 bytes_out=16149
WAN:
  bytes_in=7618 bytes_out=1917
```

Generating a keytab on a Windows server

A keytab is used to allow services that are not running Windows to be configured with service instance accounts in the Active Directory Domain Service (AD DS). This allows Kerberos clients to authenticate to the service through Windows Key Distribution Centers (KDCs).

For an explanation of the process, see https://docs.microsoft.com/en-us/windows-server/administration/windows-commands/ktpass.

To generate a keytab on a Windows server:

1. On the server, create a user for the FortiGate:
   - The service name is the FQDN for the explicit proxy interface, such as the hostname in the client browser proxy configuration. In this example, the service name is *FGT*.
   - The account only requires *domain users* membership.
   - The password must be very strong.
   - The password is set to never expire.
2. Add the FortiGate FQDN in to the Windows DNS domain, as well as in-addr.arpa.
3. Generate the Kerberos keytab using the `ktpass` command on Windows servers and many domain workstations:

   ```
   # ktpass -princ HTTP/<domain name of test fgt>@realm -mapuser <user> -pass <password> -
   crypto all -ptype KRB5_NT_PRINCIPAL -out fgt.keytab
   ```

   For example:

   ```
   ktpass -princ HTTP/FGT.FORTINETQA.LOCAL@FORTINETQA.LOCAL -mapuser FGT -pass ***********
   -crypto all -ptype KRB5_NT_PRINCIPAL -out fgt.keytab
   ```

   If the FortiGate is handling multiple keytabs in Kerberos authentication, use different passwords when generating each keytab.

4. Encode the keytab to base64 in a text file:
   - On Windows: `certutil -encode fgt.keytab tmp.b64 && findstr /v /c:- tmp.b64 > fgt.txt`
   - On Linux: `base64 fgt.keytab > fgt.txt`
   - On MacOS: `base64 -i fgt.keytab -o fgt.txt`
5. Use the code in `fgt.txt` as the keytab parameter when configuring the FortiGate.

Transparent web proxy forwarding

In FortiOS, there is an option to enable proxy forwarding for transparent web proxy policies and regular firewall policies for HTTP and HTTPS.

In previous versions of FortiOS, you could forward proxy traffic to another proxy server (proxy chaining) with explicit proxy. Now, you can forward web traffic to the upstream proxy without having to reconfigure your browsers or publish a proxy auto-reconfiguration (PAC) file.

Once configured, the FortiGate forwards traffic generated by a client to the upstream proxy. The upstream proxy then forwards it to the server.

To configure proxy forwarding:

1. Configure the web proxy forwarding server:

```
config web-proxy forward-server
    edit "upStream_proxy_1"
        set ip 172.16.200.20
        set healthcheck enable
        set monitor "http://www.google.ca"
    next
end
```

2. Append the web proxy forwarding server to a firewall policy:

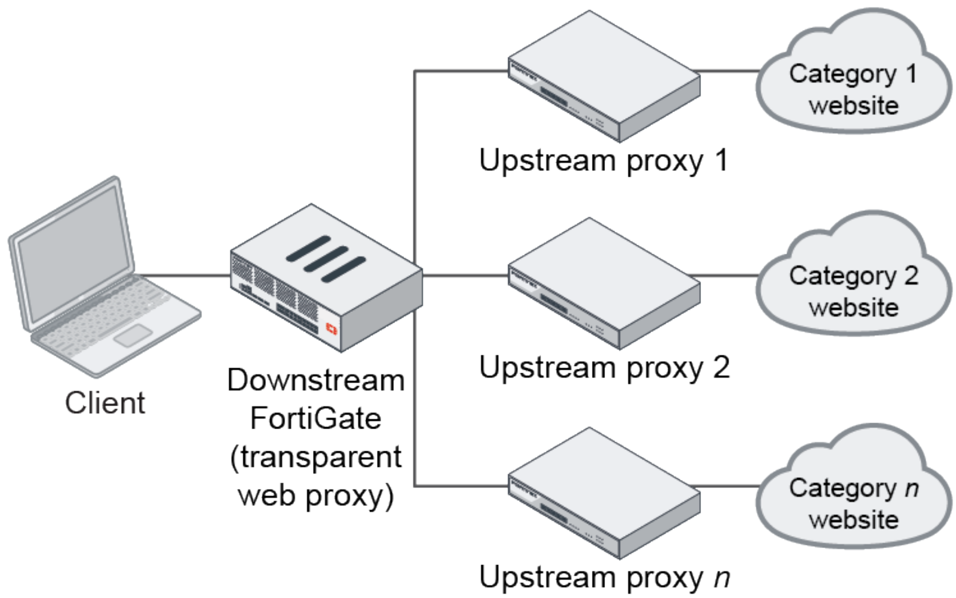
```
config firewall policy
    edit 1
        set name "LAN To WAN"
        set srcintf "port10"
        set dstintf "port9"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set logtraffic all
        set webproxy-forward-server "upStream_proxy_1"
        set fsso disable
        set av-profile "av"
        set ssl-ssh-profile "deep-custom"
        set nat enable
    next
end
```

Selectively forward web requests to a transparent web proxy

Web traffic over HTTP/HTTPS can be forwarded selectively by the FortiGate's transparent web proxy to an upstream web proxy to avoid overwhelming the proxy server. Traffic can be selected by specifying the proxy address, which can be based on a FortiGuard URL category.

The FortiGuard web filter service must be enabled on the downstream FortiGate.

Topology



Forwarding behavior

The forward server will be ignored if the proxy policy matching for a particular session needs the FortiGate to see authentication information inside the HTTP (plain text) message. For example, assume that user authentication is required and a forward server is configured in the transparent web proxy, and the authentication method is an active method (such as basic). When the user or client sends the HTTP request over SSL with authentication information to the FortiGate, the request cannot be forwarded to the upstream proxy. Instead, it will be forwarded directly to the original web server (assuming deep inspection and `http-policy-redirect` are enabled in the firewall policy).

The FortiGate will close the session before the client request can be forwarded if all of the following conditions are met:

- The certificate inspection is configured in the firewall policy that has the `http-policy-redirect` option enabled.
- A previously authenticated IP-based user record cannot be found by the FortiGate's memory during the SSL handshake.
- Proxy policy matching needs the FortiGate to see the HTTP request authentication information.

This means that in order to enable user authentication and use `webproxy-forward-server` in the transparent web proxy policy at the same time, the following best practices should be followed:

- In the firewall policy that has the `http-policy-redirect` option enabled, set `ssl-ssh-profile` to use the `deep-inspection` profile.
- Use IP-based authentication rules; otherwise, the `webproxy-forward-server` setting in the transparent web proxy policy will be ignored.
- Use a passive authentication method such as FSSO. With FSSO, once the user is authenticated as a domain user by a successful login, the web traffic from the user's client will always be forwarded to the upstream proxy as long as the authenticated user remains unexpired. If the authentication method is an active authentication method (such as basic, digest, NTLM, negotiate, form, and so on), the first session containing authentication information will bypass the forward server, but the following sessions will be connected through the upstream proxy.

Sample configuration

On the downstream FortiGate proxy, there are two category proxy addresses used in two separate transparent web proxy policies as the destination address:

- In the policy with `upstream_proxy_1` as the forward server, the proxy address `category_infotech` is used to match URLs in the information technology category.
- In the policy with `upstream_proxy_2` as the forward server, the proxy address `category_social` is used to match URLs in the social media category.

To configure forwarding requests to transparent web proxies:

1. Configure the proxy forward servers:

```
config web-proxy forward-server
    edit "upStream_proxy_1"
        set ip 172.16.200.20
    next
    edit "upStream_proxy_2"
        set ip 172.16.200.46
    next
end
```

2. Configure the web proxy addresses:

```
config firewall proxy-address
    edit "category_infotech"
        set type category
        set host "all"
        set category 52
    next
    edit "category_social"
        set type category
        set host "all"
        set category 37
    next
end
```

3. Configure the firewall policy:

```
config firewall policy
    edit 1
        set srcintf "port10"
        set dstintf "port9"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set inspection-mode proxy
        set http-policy-redirect enable
        set ssl-ssh-profile "deep-inspection"
        set av-profile "av"
        set nat enable
    next
end
```

4. Configure the proxy policies:

```
config firewall proxy-policy
    edit 1
        set proxy transparent-web
        set srcintf "port10"
        set dstintf "port9"
        set srcaddr "all"
        set dstaddr "category_infotech"
        set service "webproxy"
        set action accept
        set schedule "always"
        set logtraffic all
        set webproxy-forward-server "upStream_proxy_1"
        set utm-status enable
        set ssl-ssh-profile "deep-inspection"
        set av-profile "av"
    next
    edit 2
        set proxy transparent-web
        set srcintf "port10"
        set dstintf "port9"
        set srcaddr "all"
        set dstaddr "category_social"
        set service "webproxy"
        set action accept
        set schedule "always"
        set logtraffic all
        set webproxy-forward-server "upStream_proxy_2"
        set utm-status enable
        set ssl-ssh-profile "deep-inspection"
        set av-profile "av"
    next
end
```

Upstream proxy authentication in transparent proxy mode

A downstream proxy FortiGate that needs to be authenticated by the upstream web proxy can use the basic authentication method to send its username and password, in the base64 format, to the upstream web proxy for authentication. If the authentication succeeds, web traffic that is forwarded from the downstream proxy FortiGate to the upstream proxy can be accepted and forwarded to its destinations.

In this example, a school has a FortiGate acting as a downstream proxy that is configured with firewall policies for each user group (students and staff). In each policy, a forwarding server is configured to forward the web traffic to the upstream web proxy.

The username and password that the upstream web proxy uses to authenticate the downstream proxy are configured on the forwarding server, and are sent to the upstream web proxy with the forwarded HTTP requests.

	Username	Password
student.proxy.local:8080	students	ABC123
staff.proxy.local:8081	staff	123456

On the downstream FortiGate, configure forwarding servers with the usernames and passwords for authentication on the upstream web proxy, then apply those servers to firewall policies for transparent proxy. For explicit web proxy, the forwarding servers can be applied to proxy policies.

When the transparent proxy is configured, clients can access websites without configuring a web proxy in their browser. The downstream proxy sends the username and password to the upstream proxy with forwarded HTTP requests to be authenticated.

To configure the forwarding server on the downstream FortiGate:

```
config web-proxy forward-server
    edit "Student_Upstream_WebProxy"
        set addr-type fqdn
        set fqdn "student.proxy.local"
        set port 8080
        set username "student"
        set password ABC123
    next
    edit "Staff_Upstream_WebProxy"
        set addr-type fqdn
        set fqdn "staff.proxy.local"
        set port 8081
        set username "staff"
        set password 123456
    next
end
```

To configure firewall policies for transparent proxy:

```
config firewall policy
    edit 1
        set srcintf "Vlan_Student"
        set dstintf "port9"
        set srcaddr "Student_Subnet"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set inspection-mode proxy
        set ssl-ssh-profile "deep-inspection"
        set av-profile "av"
        set webproxy-forward-server "Student_Upstream_WebProxy"
        set nat enable
    next
    edit 2
        set srcintf "Vlan_Staff"
        set dstintf "port9"
        set srcaddr "Staff_Subnet"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set inspection-mode proxy
        set ssl-ssh-profile "deep-inspection"
```

```
        set av-profile "av"
        set webproxy-forward-server "Staff_Upstream_WebProxy"
        set nat enable
    next
end
```

Multiple dynamic header count

Multiple dynamic headers are supported for web proxy profiles, as well as Base64 encoding and the append/new options.

Administrators only have to select the dynamic header in the profile. The FortiGate will automatically display the corresponding static value. For example, if the administrator selects the `$client-ip` header, the FortiGate will display the actual client IP address.

The supported headers are:

`$client-ip`	Client IP address
`$user`	Authentication user name
`$domain`	User domain name
`$local_grp`	Firewall group name
`$remote_grp`	Group name from authentication server
`$proxy_name`	Proxy realm name

To configure dynamic headers using the CLI:

Since authentication is required, FSSO NTLM authentication is configured in this example.

1. Configure LDAP:

```
config user ldap
    edit "ldap-kerberos"
        set server "172.18.62.220"
        set cnid "cn"a
        set dn "dc=fortinetqa,dc=local"
        set type regular
        set username "CN=root,CN=Users,DC=fortinetqa,DC=local"
        set password *********
    next
end
```

2. Configure FSSO:

```
config user fsso
    edit "1"
        set server "172.18.62.220"
        set password *********
    next
end
```

3. Configure a user group:

```
config user group
    edit "NTLM-FSSO"
        set group-type fsso-service
        set member "FORTINETQA/FSSO"
    next
end
```

4. Configure an authentication scheme:

```
config authentication scheme
    edit "au-sch-ntlm"
        set method ntlm
    next
end
```

5. Configure an authentication rule:

```
config authentication rule
    edit "au-rule-fsso"
        set srcaddr "all"
        set active-auth-method "au-sch-ntlm"
    next
end
```

6. Create a web proxy profile that adds a new dynamic and custom `Via` header:

```
config web-proxy profile
    edit "test"
        set log-header-change enable
        config headers
            edit 1
                set name "client-ip"
                set content "$client-ip"
            next
            edit 2
                set name "Proxy-Name"
                set content "$proxy_name"
            next
            edit 3
                set name "user"
                set content "$user"
            next
            edit 4
                set name "domain"
                set content "$domain"
            next
            edit 5
                set name "local_grp"
                set content "$local_grp"
            next
            edit 6
                set name "remote_grp"
                set content "$remote_grp"
            next
            edit 7
                set name "Via"
                set content "Fortigate-Proxy"
            next
```

```
        end
    next
end
```

7. In the proxy policy, append the web proxy profile created in the previous step:

```
config firewall proxy-policy
    edit 1
        set proxy explicit-web
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set service "web"
        set action accept
        set schedule "always"
        set logtraffic all
        set groups "NTLM-FSSO"
        set webproxy-profile "test"
        set utm-status enable
        set av-profile "av"
        set webfilter-profile "content"
        set ssl-ssh-profile "deep-custom"
    next
end
```

8. Once traffic is being generated from the client, look at the web filter logs to verify that it is working.

   The corresponding values for all the added header fields are shown in the *Web Filter* card at *Log & Report > Security Events*, in the *Change headers* section at the bottom of the *Log Details* pane.

```
1: date=2019-02-07 time=13:57:24 logid="0344013632" type="utm" subtype="webfilter"
eventtype="http_header_change" level="notice" vd="vdom1" eventtime=1549576642 policyid=1
transid=50331689 sessionid=1712788383 user="TEST21@FORTINETQA" group="NTLM-FSSO"
profile="test" srcip=10.1.100.116 srcport=53278 dstip=172.16.200.46 dstport=80
srcintf="port2" srcintfrole="undefined" dstintf="port1" dstintfrole="undefined" proto=6
service="HTTP" url="http://172.16.200.46/" agent="curl/7.22.0" chgheaders="Added=client-
ip: 10.1.100.116|Proxy-Name: 1.1 100D.qa|user: TEST21|domain: FORTINETQA|local_grp:
NTLM-FSSO|remote_grp: FORTINETQA/FSSO|Via: Fortigate-Proxy"
```

Restricted SaaS access

Large organizations may want to restrict SaaS access to resources like Microsoft Office 365, Google Workspace, and Dropbox by tenant to block non-company login attempts and secure the users from accessing non-approved cloud resources. Many cloud vendors enable this by applying tenant restrictions for access control. For example, users accessing Microsoft 365 applications with tenant restrictions through the corporate proxy will only be allowed to log in as the company’s tenant and access the organization’s applications.

To implement this, access requests from the clients pass through the company’s web proxy, which inserts headers to notify the SaaS service to apply tenant restrictions with the permitted tenant list. Users are redirected the SaaS service login page, and are only allowed to log in if they belong to the permitted tenant list.

For more information, refer to the vendor-specific documentation:

- Office 365: Restrict access to a tenant
- Google Workspace: Block access to consumer accounts
- Dropbox: Network control

Basic configuration

A web proxy profile can specify access permissions for Microsoft Office 365, Google Workspace, and Dropbox by inserting vendor-defined headers that restrict access to the specific accounts. Custom headers can also be inserted for any destination. The web proxy profile can then be applied to a firewall policy to control the header's insertion.

To implement Office 365 tenant restriction, Google Workspace account access control, and Dropbox network access control:

1. Configure a web proxy profile according to the vendors' specifications:
   a. Set the header name (defined by the service provider).
   b. Set the traffic destination (the service provider).
   c. Set the HTTP header content to be inserted into the traffic (defined by your settings).

```
config web-proxy profile
    edit <name>
        config headers
            edit <id>
                set name <string>
                set dstaddr <address>
                set action add-to-request
                set base64-encoding disable
                set add-option new
                set protocol https http
                set content <string>
            next
        end
    next
end
```

2. Apply the web proxy profile to a policy. SSL deep inspection must be used in the firewall policy:

The following table lists the vendor-specific `config headers` settings that must be configured in the web proxy profile (`config web-proxy profile`):

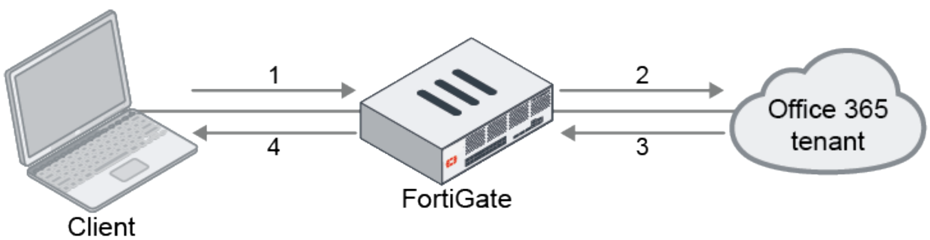
Setting	Vendor specification		
	Microsoft Office 365	Google Workspace	Dropbox
`name <string>`	• `Restrict-Access-To-Tenants` • `Restrict-Access-Context`	• `X-GoogApps-Allowed-Domains`	• `X-Dropbox-allowed-Team-Ids`
`dstaddr <address>`	• Use the built-in `Microsoft Office 365` address.	• Use the built-in `G Suite` address.	• Use the built-in `wildcard.dropbox.com` address.
`content <string>`	• Enter the domain for `Restrict-Access-To-`	• Enter the domain.	• Enter the Dropbox team ID.

Setting	Vendor specification		
	Microsoft Office 365	Google Workspace	Dropbox
	`Tenants`. • Enter the directory ID for `Restrict-Access-Context`.		

Due to vendors' changing requirements, these settings may no longer comply with the vendors' official guidelines. See the vendor documentation for more details.

Microsoft Office 365 example

In this example, a web proxy profile is created to control permissions for Microsoft Office 365 to allow corporate domains and deny personal accounts, such as Hotmail and Outlook that are accessed through login.live.com.



1. When a user attempts to access login.microsoftonline.com, login.microsoft.com, or login.windows.net, the traffic will match a proxy inspection mode firewall policy with the assigned web proxy profile.
2. The web proxy profile adds new headers to the customer tenant, indicating the allowed domain and restricted access for personal accounts. Next, the FortiGate starts a new connection with the Microsoft Office 365 domain controller including the new headers.
3. The Microsoft Office 365 domain controller assesses this data and will allow or deny this access, then sends a reply to the FortiGate.
4. The FortiGate sends a reply to the client.

The FortiGate will only indicate the correct domains to be allowed or denied through the headers to Microsoft. The custom sign-in portal in the browser is generated by Microsoft.

Configuration summary

The following must be configured in FortiOS:

- An FQDN address for login.live.com
- An SSL inspection profile that uses deep inspection with an exemption for login.live.com

Ensure that the firewall certificate is installed on the client machines. A company certificate signed by an internal CA is recommended.

- A web filter profile in proxy mode with static URL filters for the SNI URLs
- A web proxy profile that adds new headers to the customer tenant

- A firewall policy using proxy mode inspection that applies the configured SSL SSL inspection, web filter, and web proxy profiles

The `Restrict-Access-To-Tenants` and `Restrict-Access-Context` headers are inserted for incoming requests to: login.microsoftonline.com, login.microsoft.com, and login.windows.net, which are part of the `Microsoft Office 365` address group.

To restrict access to personal accounts using the login.live.com domain, the `sec-Restrict-Tenant-Access-Policy` header is inserted and uses `restrict-msa` as the header content.

Before configuring the FortiGate, collect the information related to the company domain in the Office 365 contract.

- `Restrict-Access-To-Tenants`: your <domain.com>
- `Restrict-Access-Context`: Directory ID

To find the Directory ID related to the domain, locate it in the Azure portal, or use the whatismytenantid.com open tool.

To configure the FortiGate:

1. Add the FQDN address for login.live.com:

```
config firewall address
    edit "login.live.com"
        set type fqdn
        set fqdn "login.live.com"
    next
end
```

2. Configure the SSL inspection profile. In this example, the `deep-inspection` profile is cloned, and the `live.com` FQDN is removed from the exemption list.
   a. Clone the `deep-inspection` profile:

   ```
   config firewall ssl-ssh-profile
       clone "deep-inspection" to "Tenant"
   end
   ```

   b. Edit the `Tenant` profile and remove `live.com` from the `config ssl-exempt` list.
3. Configure the URL filter list:

```
config webfilter urlfilter
    edit 1
        set name "Auto-webfilter-urlfilter"
        config entries
            edit 1
                set url "login.microsoftonline.com"
                set action allow
            next
            edit 2
                set url "login.microsoft.com"
                set action allow
            next
            edit 3
                set url "login.windows.net"
                set action allow
```

```
            next
            edit 4
                set url "login.live.com"
                set action allow
            next
        end
    next
end
```

4. Configure the web filter profile:

```
config webfilter profile
    edit "Tenant"
        set comment "Office 365"
        set feature-set proxy
        config web
            set urlfilter-table 1
        end
    next
end
```

5. Configure the web proxy profile (enter the header names exactly as shown):

```
config web-proxy profile
    edit "SaaS-Tenant-Restriction"
        set header-client-ip pass
        set header-via-request pass
        set header-via-response pass
        set header-x-forwarded-for pass
        set header-x-forwarded-client-cert pass
        set header-front-end-https pass
        set header-x-authenticated-user pass
        set header-x-authenticated-groups pass
        set strip-encoding disable
        set log-header-change disable
        config headers
            edit 1
                set name "Restrict-Access-To-Tenants"
                set dstaddr "login.microsoft.com" "login.microsoftonline.com"
"login.windows.net"
                set action add-to-request
                set base64-encoding disable
                set add-option new
                set protocol https http
                set content <domain>
            next
            edit 2
                set name "Restrict-Access-Context"
                set dstaddr "login.microsoftonline.com" "login.microsoft.com"
"login.windows.net"
                set action add-to-request
                set base64-encoding disable
                set add-option new
                set protocol https http
                set content <directory_ID>
            next
            edit 3
```

```
                set name "sec-Restrict-Tenant-Access-Policy"
                set dstaddr "login.live.com"
                set action add-to-request
                set base64-encoding disable
                set add-option new
                set protocol https http
                set content "restrict-msa"
            next
        end
    next
end
```

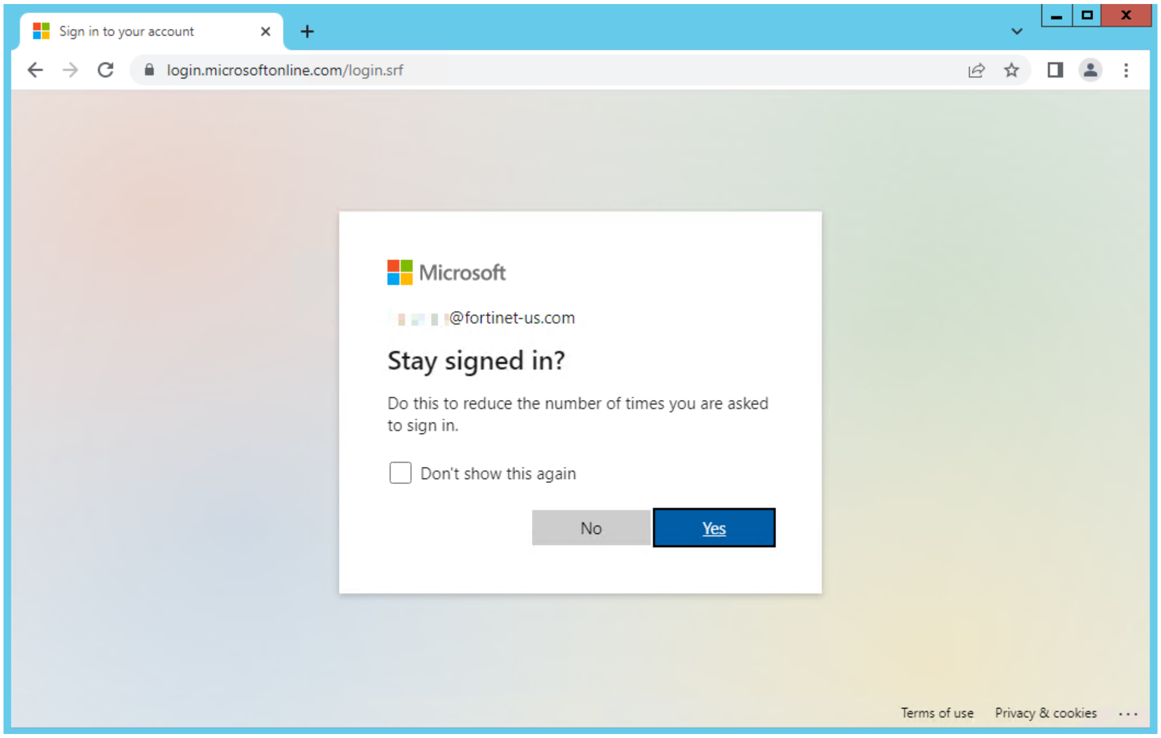
6. Configure the firewall policy:

```
config firewall policy
    edit 10
        set name "Tenant"
        set srcintf "port2"
        set dstintf "port1"
        set action accept
        set srcaddr "users-lan"
        set dstaddr "login.microsoft.com" "login.microsoftonline.com"
"login.windows.net" "login.live.com"
        set schedule "always"
        set service "HTTP" "HTTPS"
        set utm-status enable
        set inspection-mode proxy
        set webproxy-profile "SaaS-Tenant-Restriction"
        set ssl-ssh-profile "Tenant"
        set webfilter-profile "Tenant"
        set logtraffic all
        set nat enable
    next
end
```

Testing the access

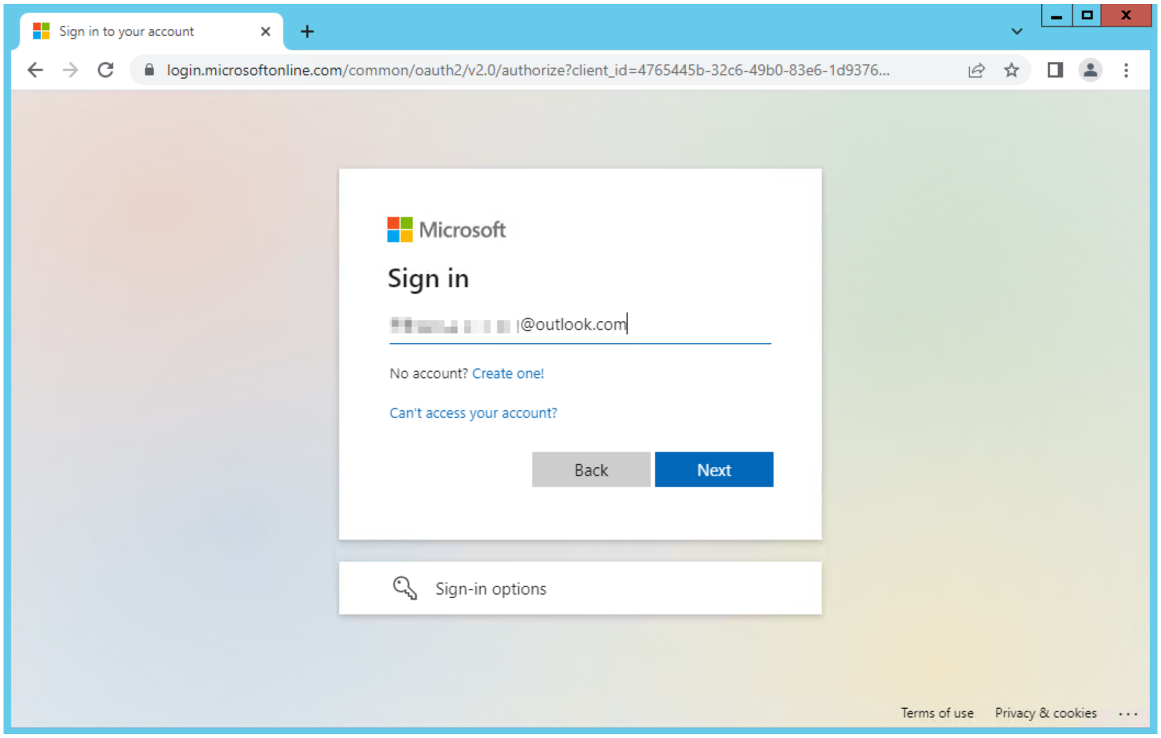
To test the access to corporate domains and personal accounts:

1. Get a client to log in with their corporate email using the login.microsoftonline.com domain.

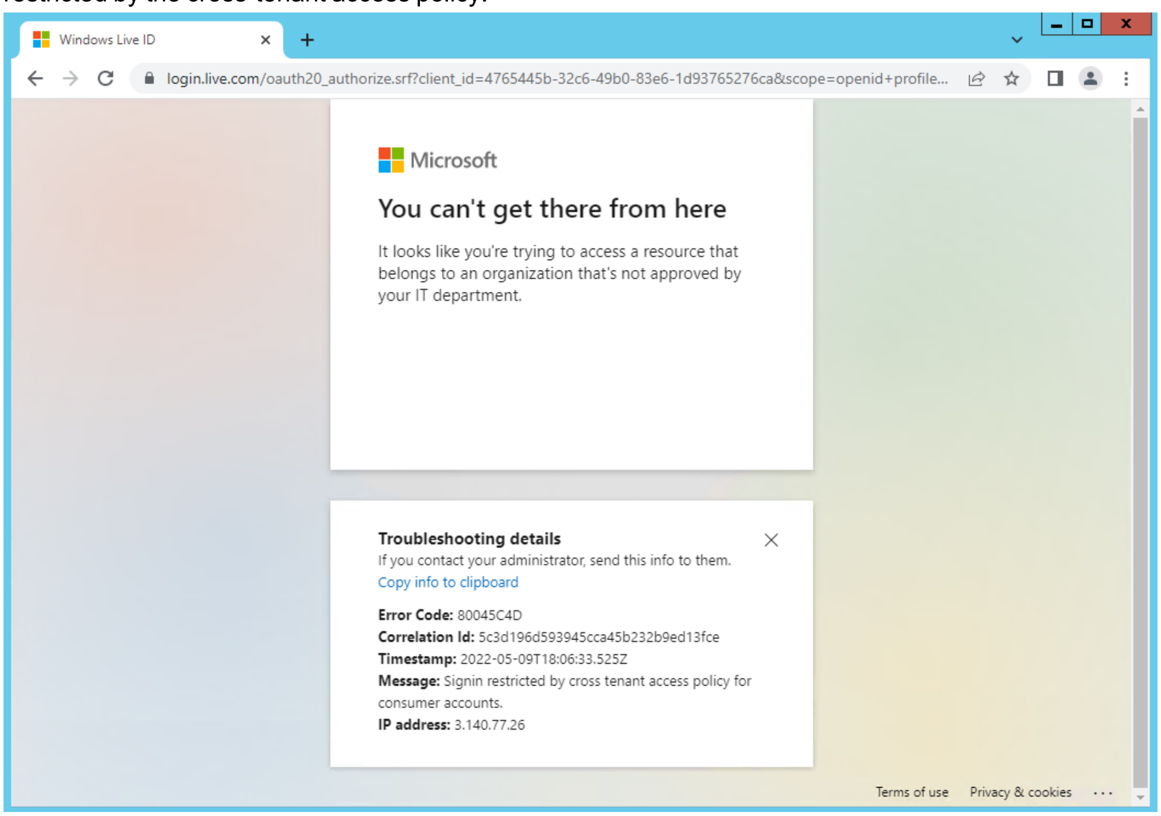


2. The client is able to enter their credentials and log in successfully.

3. Get a client to log in to their personal Outlook account.



4. After the client enters their credentials, a message appears that they cannot access this resource because it is restricted by the cross-tenant access policy.



Verifying the header insertion

To verify the header insertion for corporate domains and personal accounts:

1. On the FortiGate, start running the WAD debugs:

```
# diagnose wad debug enable category http
# diagnose wad debug enable level info
# diagnose debug enable
```

2. After a client attempts to access corporate domains, verify that the header information is sent to the Microsoft Active Directory:

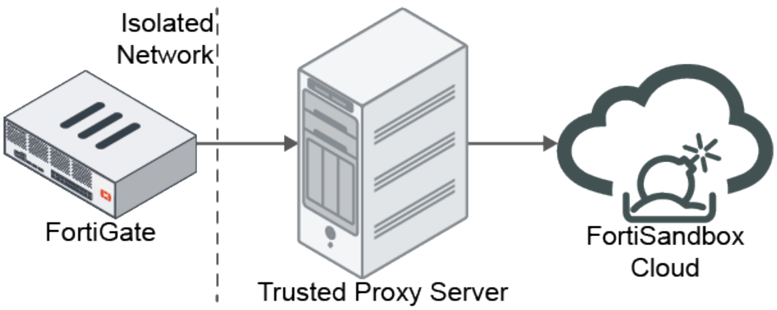
```
[I][p:234][s:2481][r:33] wad_dump_fwd_http_req             :2567  hreq=0x7fc75f0cd468
Forward request to server:
POST /common/GetCredentialType?mkt=en-US HTTP/1.1
Host: login.microsoftonline.com
Connection: keep-alive
Content-Length: 1961
sec-ch-ua: " Not A;Brand";v="99", "Chromium";v="101", "Google Chrome";v="101"
hpgrequestid: d7f706a8-1143-4cdd-ad52-1cc69dc7bb00
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 6.3; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/101.0.4951.54 Safari/537.36
client-request-id: 5c3d196d-5939-45cc-a45b-232b9ed13fce
...
Restrict-Access-To-Tenants: fortinet-us.com
Restrict-Access-Context: ********-****-452f-8535-************
```

3. After a client attempts to access a personal account, verify that the header information is sent to the Microsoft Active Directory:

```
[I][p:234][s:2519][r:34] wad_dump_fwd_http_req             :2567  hreq=0x7fc75f0ce6a8
Forward request to server:
GET /oauth20_authorize.srf?client_id=4765445b-32c6-49b0-83e6-
1d93765276ca&scope=openid+profile+https%3a%2f%2fwww.office.com%2fv2%2fOfficeHome.All&red
irect_uri=https%3a%2f%2fwww.office.com%2flandingv2&response_type=code+id_
token&state=7tAtndYhcA3132S--UOTyLVEtyIZs8FgndTpeYM9mJ1EeA-
X5nfqrSalnnPH41cHxfHGug6N5cbliK676v6xZgszgH_
JARVKrptZwBvjI2cbnZ4mttYNNdK1FTlbEtu5VBjgtBOX2u6v3F_
9g7UikCpGTnBRGhvO2pyTndT3EEIyAHvhg9LsKRtY3kxce8dQkfk1iDjLcc3q-01r4rpxSx2xZSbwg_
KkAN3kCRQ9uLfE0ziHAcpvunuKmzGBWKnBhC4sJJkXrMEfXwCg4nsOjg&response_mode=form_
post&nonce=637877163655610380.MjNjZmM4NzQtOTU5My00OGZlLTk0NTItZTE5NDU2YjVlODdjNjViOTQwYm
UtOTZlMS00M2Y5LTkyN2MtN2QyMjgwNjcxY2Uz&x-client-SKU=ID_NETSTANDARD2_0&x-client-
Ver=6.12.1.0&uaid=5c3d196d593945cca45b232b9ed13fce&msproxy=1&issuer=mso&tenant=common&u
i_locales=en-US&epct=AQABAAAAAAD--DLA3VO7QrddgJg7WevrfA6SLaDsJUcjb1Bg9OKonF3d_
lfNJsdDAIH5hlJdUSGejEBIqsko-A7JX67PzaGdEJgOIGa37VhJzGTYBZ-KgATe9FHssnNmLjM_
dojr0dAT83xDhiqQTN2-UcYdcP2s3vPainF7Nqes5ecXRaEoE9Vw9-
sN7jfASOkPRWW03aI6buz0niABvA860YOWDb98vdJWPGkWE-euDr6n8_
zI5iAA&jshs=0&username=***************%40outlook.com&login_
hint=**************%40outlook.com HTTP/1.1
Host: login.live.com
Connection: keep-alive
...
Referer: https://login.microsoftonline.com/
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9
sec-Restrict-Tenant-Access-Policy: restrict-msa
```

Explicit proxy and FortiGate Cloud Sandbox

Explicit proxy connections can leverage FortiGate Cloud Sandbox for advanced threat scanning and updates. This allows FortiGates behind isolated networks to connect to FortiCloud services.



To configure FortiGuard services to communicate with an explicit proxy server:

```
config system fortiguard
    set proxy-server-ip 172.16.200.44
    set proxy-server-port 3128
    set proxy-username "test1"
    set proxy-password *********
end
```

To verify the explicit proxy connection to FortiGate Cloud Sandbox:

```
# diagnose debug application forticldd -1
Debug messages will be on for 30 minutes.
# diagnose debug enable
[2942] fds_handle_request: Received cmd 23 from pid-2526, len 0
[40] fds_queue_task: req-23 is added to Cloud-sandbox-controller
[178] fds_svr_default_task_xmit: try to get IPs for Cloud-sandbox-controller
[239] fds_resolv_addr: resolve aptctrl1.fortinet.com
[169] fds_get_addr: name=aptctrl1.fortinet.com, id=32, cb=0x2bc089
[101] dns_parse_resp: DNS aptctrl1.fortinet.com -&gt; 172.16.102.21
[227] fds_resolv_cb: IP-1: 172.16.102.21
[665] fds_ctx_set_addr: server: 172.16.102.21:443
[129] fds_svr_default_pickup_server: Cloud-sandbox-controller: 172.16.102.21:443
[587] fds_https_start_server: server: 172.16.102.21:443
[579] ssl_new: SSL object is created
[117] https_create: proxy server 172.16.200.44 port:3128
[519] fds_https_connect: https_connect(172.16.102.21) is established.
[261] fds_svr_default_on_established: Cloud-sandbox-controller has connected to
ip=172.16.102.21
[268] fds_svr_default_on_established: server-Cloud-sandbox-controller handles cmd-23
[102] fds_pack_objects: number of objects: 1
[75] fds_print_msg: FCPC: len=109
[81] fds_print_msg: Protocol=2.0
[81] fds_print_msg: Command=RegionList
[81] fds_print_msg: Firmware=FG101E-FW-6.02-0917
[81] fds_print_msg: SerialNumber=FG101E4Q17002429
[81] fds_print_msg: TimeZone=-7
[75] fds_print_msg: http req: len=248
[81] fds_print_msg: POST https://172.16.102.21:443/FCPService HTTP/1.1
```

```
[81] fds_print_msg: User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)
[81] fds_print_msg: Host: 172.16.102.21:443
[81] fds_print_msg: Cache-Control: no-cache
[81] fds_print_msg: Connection: close
[81] fds_print_msg: Content-Type: application/octet-stream
[81] fds_print_msg: Content-Length: 301
[524] fds_https_connect: http request to 172.16.102.21: header=248, ext=301.
[257] fds_https_send: sent 248 bytes: pos=0, len=248
[265] fds_https_send: 172.16.102.21: sent 248 byte header, now send 301-byte body
[257] fds_https_send: sent 301 bytes: pos=0, len=301
[273] fds_https_send: sent the entire request to server: 172.16.102.21:443
[309] fds_https_recv: read 413 bytes: pos=413, buf_len=2048
[332] fds_https_recv: received the header from server: 172.16.102.21:443, [HTTP/1.1 200
Content-Type: application/octet-stream
Content-Length: 279
Date: Thu, 20 Jun 2019 16:41:11 GMT
Connection: close]
[396] fds_https_recv: Do memmove buf_len=279, pos=279
[406] fds_https_recv: server: 172.16.102.21:443, buf_len=279, pos=279
[453] fds_https_recv: received a packet from server-172.16.102.21:443: sz=279, objs=1
[194] __ssl_data_ctx_free: Done
[839] ssl_free: Done
[830] ssl_disconnect: Shutdown
[481] fds_https_recv: obj-0: type=FCPR, len=87
[294] fds_svr_default_on_response: server-Cloud-sandbox-controller handles cmd-23
[75] fds_print_msg: fcpr:  len=83
[81] fds_print_msg: Protocol=2.0
[81] fds_print_msg: Response=202
[81] fds_print_msg: ResponseItem=Region:Europe,Global,Japan,US
[81] fds_print_msg: existing:Japan
[3220] aptctrl_region_res: Got rsp: Region:Europe,Global,Japan,US
[3222] aptctrl_region_res: Got rsp: Region existing:Japan
[439] fds_send_reply: Sending 28 bytes data.
[395] fds_free_tsk: cmd=23; req.noreply=1
# [136] fds_on_sys_fds_change: trace
[2942] fds_handle_request: Received cmd 22 from pid-170, len 0
[40] fds_queue_task: req-22 is added to Cloud-sandbox-controller
[587] fds_https_start_server: server: 172.16.102.21:443
[579] ssl_new: SSL object is created
[117] https_create: proxy server 172.16.200.44 port:3128
[519] fds_https_connect: https_connect(172.16.102.21) is established.
[261] fds_svr_default_on_established: Cloud-sandbox-controller has connected to
ip=172.16.102.21
[268] fds_svr_default_on_established: server-Cloud-sandbox-controller handles cmd-22
[102] fds_pack_objects: number of objects: 1
[75] fds_print_msg: FCPC: len=146
[81] fds_print_msg: Protocol=2.0
[81] fds_print_msg: Command=UpdateAPT
[81] fds_print_msg: Firmware=FG101E-FW-6.02-0917
[81] fds_print_msg: SerialNumber=FG101E4Q17002429
[81] fds_print_msg: TimeZone=-7
[81] fds_print_msg: TimeZoneInMin=-420
[81] fds_print_msg: DataItem=Region:US
[75] fds_print_msg: http req: len=248
[81] fds_print_msg: POST https://172.16.102.21:443/FCPService HTTP/1.1
[81] fds_print_msg: User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)
```

```
[81] fds_print_msg: Host: 172.16.102.21:443
[81] fds_print_msg: Cache-Control: no-cache
[81] fds_print_msg: Connection: close
[81] fds_print_msg: Content-Type: application/octet-stream
[81] fds_print_msg: Content-Length: 338
[524] fds_https_connect: http request to 172.16.102.21: header=248, ext=338.
[257] fds_https_send: sent 248 bytes: pos=0, len=248
[265] fds_https_send: 172.16.102.21: sent 248 byte header, now send 338-byte body
[257] fds_https_send: sent 338 bytes: pos=0, len=338
[273] fds_https_send: sent the entire request to server: 172.16.102.21:443
[309] fds_https_recv: read 456 bytes: pos=456, buf_len=2048
[332] fds_https_recv: received the header from server: 172.16.102.21:443, [HTTP/1.1 200
Content-Type: application/octet-stream
Content-Length: 322
Date: Thu, 20 Jun 2019 16:41:16 GMT
Connection: close]
[396] fds_https_recv: Do memmove buf_len=322, pos=322
[406] fds_https_recv: server: 172.16.102.21:443, buf_len=322, pos=322
[453] fds_https_recv: received a packet from server-172.16.102.21:443: sz=322, objs=1
[194] __ssl_data_ctx_free: Done
[839] ssl_free: Done
[830] ssl_disconnect: Shutdown
[481] fds_https_recv: obj-0: type=FCPR, len=130
[294] fds_svr_default_on_response: server-Cloud-sandbox-controller handles cmd-22
[75] fds_print_msg: fcpr:  len=126
[81] fds_print_msg: Protocol=2.0
[81] fds_print_msg: Response=202
[81] fds_print_msg: ResponseItem=Server1:172.16.102.51:514
[81] fds_print_msg: Server2:172.16.102.52:514
[81] fds_print_msg: Contract:20210215
[81] fds_print_msg: NextRequest:86400
[615] parse_apt_contract_time_str: The APTContract is valid to Mon Feb 15 23:59:59 2021
[616] parse_apt_contract_time_str: FGT current local time is Thu Jun 20 09:41:16 2019
[3289] aptctrl_update_res: Got rsp: APT=172.16.102.51:514 APTAlter=172.16.102.52:514 next-
upd=86400
[395] fds_free_tsk: cmd=22; req.noreply=1
```

Proxy chaining

For the explicit web proxy you can configure web proxy forwarding servers to use proxy chaining to redirect web proxy sessions to other proxy servers. Proxy chaining can be used to forward web proxy sessions from the FortiGate unit to one or more other proxy servers on your network or on a remote network. You can use proxy chaining to integrate the FortiGate explicit web proxy with a web proxy solution that you already have in place.

A FortiGate unit can forward sessions to most web proxy servers including a remote FortiGate unit with the explicit web proxy enabled. No special configuration of the explicit web proxy on the remote FortiGate unit is required.

You can deploy the explicit web proxy with proxy chaining in an enterprise environment consisting of small satellite offices and a main office. If each office has a FortiGate unit, users at each of the satellite offices can use their local FortiGate unit as an explicit web proxy server. The satellite office FortiGate units can forward explicit web proxy sessions to an explicit web proxy server at the central office. From here the sessions can connect to web servers on the Internet.

FortiGate proxy chaining does not support web proxies in the proxy chain authenticating each other.

The following examples assume explicit web proxy has been enabled.

To enable explicit web proxy in the GUI:

1. Go to *System > Feature Visibility*.
2. In the *Security Features* column, enable *Explicit Proxy*.
3. Configure the explicit web proxy settings. See Explicit web proxy on page 220.

To add a web proxy forwarding server in the GUI:

1. Go to *Network > Explicit Proxy*. The *Explicit Proxy* page opens.
2. In the *Web Proxy Forwarding Servers* section, click *Create New*.
3. Configure the server settings:

Name	Enter the name of the forwarding server.
Proxy Address Type	Select the type of IP address of the forwarding server. A forwarding server can have an *FQDN* or *IP* address.
Proxy Address	Enter the IP address of the forwarding server.
Port	Enter the port number on which the proxy receives connections. Traffic leaving the FortiGate explicit web proxy for this server has its destination port number changed to this number.
Server Down Action	Select the action the explicit web proxy will take if the forwarding server is down. • *Block*: Blocks the traffic if the remote server is down. • *Use Original Server*: Forwards the traffic from the FortiGate to its destination as if no forwarding server is configured.
Health Monitor	Select to enable health check monitoring.
Health Check Monitor Site	Enter the address of a remote site.

4. Click *OK*.

Example

The following example adds a web proxy forwarding server named `fwd-srv` at address `proxy.example.com` and port `8080`.

To add a web proxy forwarding server in the CLI:

```
config web-proxy forward-server
    edit fwd-srv
        set addr-type fqdn
        set fqdn proxy.example.com
        set port 8080
    next
end
```

Web proxy forwarding server monitoring and health checking

By default, a FortiGate unit monitors a web proxy forwarding server by forwarding a connection to the remote server every 10 seconds. The remote server is assumed to be down if it does not respond to the connection. FortiGate continues checking the server. The server is assumed to be back up when the server sends a response. If you enable health checking, the FortiGate unit attempts to get a response from a web server every 10 seconds by connecting through the remote forwarding server.

You can configure health checking for each remote server and specify a different website to check for each one.

If the remote server is found to be down you can configure the FortiGate unit to block sessions until the server comes back up or to allow sessions to connect to their destination, bypassing the remote forwarding server. You cannot configure the FortiGate unit to fail over to another remote forwarding server.

To configure proxy server monitor and health checking in the GUI:

1. Go to *Network > Explicit Proxy*. The *Explicit Proxy* page opens.
2. In the *Web Proxy Forwarding Servers* section, edit a server.
3. Configure the *Server Down Action* and *Health Monitor* settings.

Server Down Action	Select the action the explicit web proxy will take if the forwarding server is down. • *Block*: Blocks the traffic if the remote server is down. • *Use Original Server*: Forwards the traffic from the FortiGate to its destination as if no forwarding server configured.
Health Monitor	Select to enable health check monitoring.
Health Check Monitor Site	Enter the address of a remote site.

4. Click *OK*.

Example

The following example enables health checking for a web proxy forwarding server and sets the server down option to bypass the forwarding server if it is down.

To configure proxy server monitor and health checking in the CLI:

```
config web-proxy forward-server
    edit fwd-srv
        set healthcheck enable
        set monitor http://example.com
        set server-down-option pass
    next
end
```

Grouping forwarding servers and load balancing traffic to the servers

You can add multiple web proxy forwarding servers to a forwarding server group and then add the server group to an explicit web proxy policy instead of adding a single server. Forwarding server groups are created from the FortiGate CLI but can be added to policies from the web-based manager (or from the CLI).

When you create a forwarding server group you can select a load balancing method to control how sessions are load balanced to the forwarding servers in the server group. Two load balancing methods are available:

- *Weighted* load balancing sends more sessions to the servers with higher weights. You can configure the weight for each server when you add it to the group.
- *Least-session* load balancing sends new sessions to the forwarding server that is processing the fewest sessions.

When you create a forwarding server group you can also enable *affinity*. Enable affinity to have requests from the same client processed by the same server. This can reduce delays caused by using multiple servers for a single multi-step client operation. Affinity takes precedence over load balancing.

You can also configure the behavior of the group if all of the servers in the group are down. You can select to block traffic or you can select to have the traffic pass through the FortiGate explicit proxy directly to its destination instead of being sent to one of the forwarding servers.

Example

The following example adds a forwarding server group that uses weighted load balancing to load balance traffic to three forwarding servers. Server weights are configured to send most traffic to `server2`. The group has `affinity` enabled and blocks traffic if all of the forward servers are down.

To configure load balancing in the CLI:

```
config web-proxy forward-server
    edit server_1
        set ip 172.20.120.12
        set port 8080
    next
    edit server_2
        set ip 172.20.120.13
        set port 8000
    next
    edit server_3
        set ip 172.20.120.14
        set port 8090
    next
end
```

```
config web-proxy forward-server-group
    edit New-fwd-group
        set affinity enable
        set ldb-method weighted
        set group-down-option block
        config server-list
            edit server_1
                set weight 10
            next
            edit server_2
                set weight 40
            next
            edit server_3
                set weight 10
            next
        end
    next
end
```

Adding proxy chaining to an explicit web proxy policy

You can enable proxy chaining for web proxy sessions by adding a web proxy forwarding server or server group to an explicit web proxy policy. In a policy you can select one web proxy forwarding server or server group. All explicit web proxy traffic accepted by this security policy is forwarded to the specified web proxy forwarding server or server group.

To add an explicit web proxy forwarding server in the GUI:

1. Go to *Policy & Objects* > *Proxy Policy* and click *Create New*.
2. Configure the policy settings:

Proxy Type	Explicit Web
Outgoing Interface	wan1
Source	Internal_subnet
Destination	all
Schedule	always
Service	webproxy
Action	Accept

3. Enable *Web Proxy Forwarding Server* and select the forwarding server, (for example,*fwd-srv*).
4. Click *OK*.

Example

The following example adds a security policy that allows all users on the `10.31.101.0` subnet to use the explicit web proxy for connections through the `wan1` interface to the Internet. The policy forwards web proxy sessions to a remote forwarding server named *fwd-srv*.

To add an explicit web proxy forwarding server in the CLI:

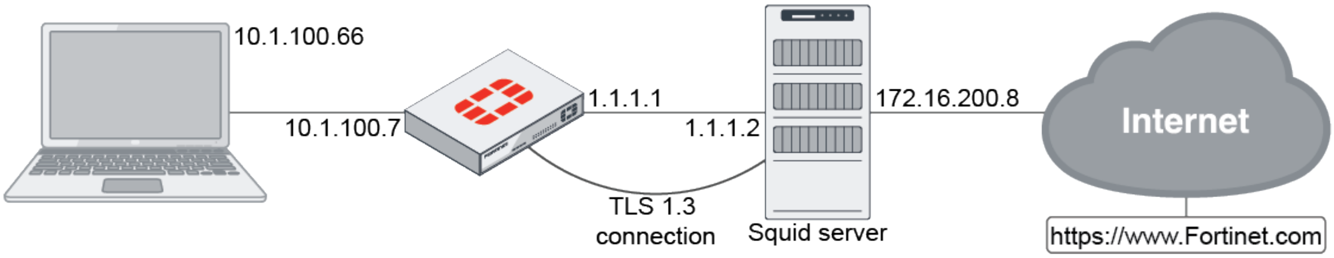
```
config firewall proxy-policy
    edit 0
        set proxy explicit-web
        set dstintf "wan1"
        set srcaddr "Internal_subnet"
        set dstaddr "all"
        set service "webproxy"
        set action accept
        set schedule "always"
        set webproxy-forward-server "fwd-srv"
    next
end
```

Using TLS 1.3 with web proxy forward servers

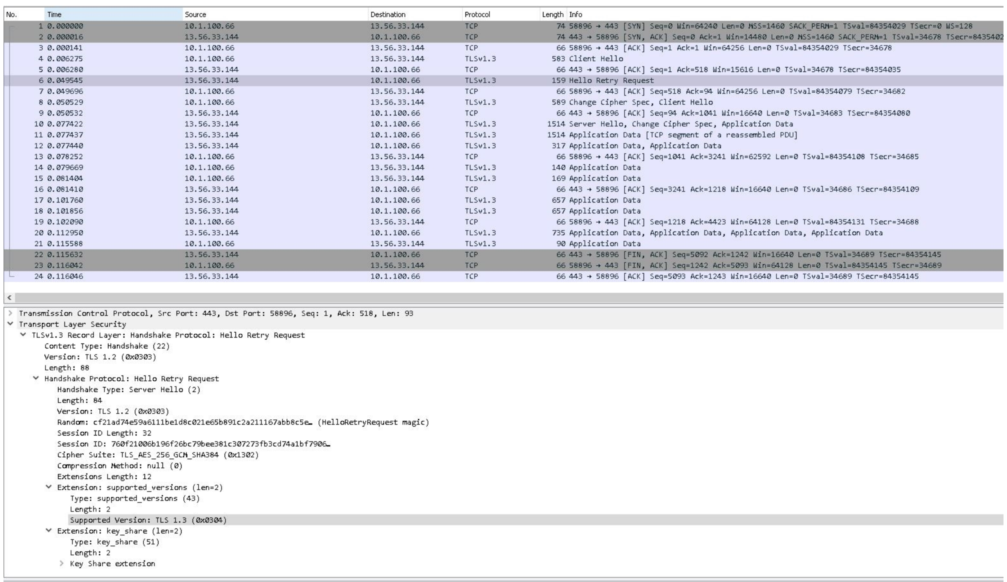
A FortiGate can handle TLS 1.3 traffic in both deep and certificate inspection modes.

Example

The following example demonstrates that the Squid server and the FortiGate can handle TLS 1.3 traffic.



The following output from the Squid server demonstrates that the FortiGate supports TLS 1.3 traffic and forwards the hello retry request back to the client PC. The client PC then sends the client hello again, and the connection is successfully established.



WAN optimization SSL proxy chaining

An SSL server does not need to be defined for WAN optimization (WANOpt) SSL traffic offloading (traffic acceleration). The server side FortiGate uses an SSL profile to resign the HTTP server's certificate, both with and without an external proxy, without an SSL server configured. GCM and ChaCha ciphers can also be used in the SSL connection.

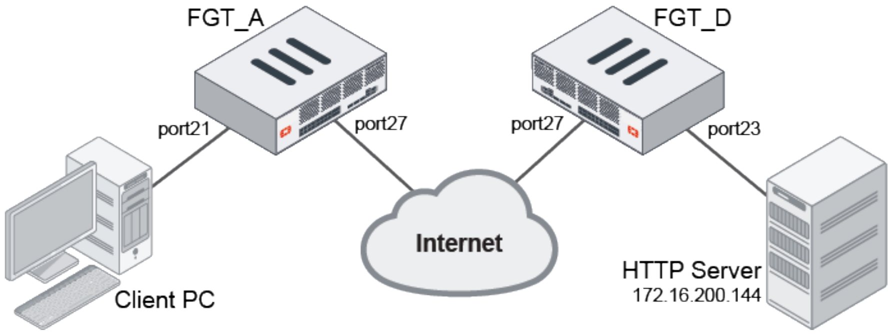
Examples

In these examples, HTTPS traffic is accelerated without configuring an SSL server, including with a proxy in between, and when the GCM or ChaCha ciphers are used.

Example 1

In this example, the server certificate is resigned by the server side FortiGate, and HTTPS traffic is accelerated without configuring an SSL server.

HTTPS traffic with the GCM or ChaCha cipher can pass though WANOpt tunnel.



To configure FGT_A:

1. Configure the hard disk to perform WANOpt:

```
config system storage
    edit "HDD2"
        set status enable
        set usage wanopt
        set wanopt-mode mix
    next
end
```

2. Configure the WANOpt peer and profile:

```
config wanopt peer
    edit "FGT-D"
        set ip 120.120.120.172
    next
end
```

```
config wanopt profile
    edit "test"
        config http
            set status enable
            set ssl enable
        end
    next
end
```

3. Create an SSL profile with deep inspection on HTTPS port 443:

```
config firewall ssl-ssh-profile
    edit "ssl"
        config https
            set ports 443
            set status deep-inspection
        end
    next
end
```

4. Configure a firewall policy in proxy mode with WANOpt enabled and the WANOpt profile selected:

```
config firewall policy
    edit 1
        set name "WANOPT-A"
```

```
            set srcintf "port21"
            set dstintf "port27"
            set action accept
            set srcaddr "all"
            set dstaddr "all"
            set schedule "always"
            set service "ALL"
            set utm-status enable
            set inspection-mode proxy
            set profile-protocol-options "protocol"
            set ssl-ssh-profile "ssl"
            set wanopt enable
            set wanopt-profile "test"
            set nat enable
        next
    end
```

To configure FGT_D:

1. Configure the hard disk to perform WANOpt:

```
config system storage
    edit "HDD2"
        set status enable
        set usage wanopt
        set wanopt-mode mix
    next
end
```

2. Configure the WANOpt peer:

```
config wanopt peer
    edit "FGT-A"
        set ip 110.110.110.171
    next
end
```

3. Create an SSL profile with deep inspection on HTTPS port 443. The default *Fortinet_CA_SSL* certificate is used to resign the server certificate:

```
config firewall ssl-ssh-profile
    edit "ssl"
        config https
            set ports 443
            set status deep-inspection
        end
    next
end
```

4. Configure a firewall policy in proxy mode with WANOpt enabled and passive WANOpt detection:

```
config firewall policy
    edit 1
        set name "WANOPT-B"
        set srcintf "port27"
        set dstintf "port23"
        set action accept
        set srcaddr "all"
```

```
        set dstaddr "all"
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set inspection-mode proxy
        set wanopt enable
        set wanopt-detection passive
        set nat enable
    next
end
```

5. Configure a proxy policy to apply the SSL profile:

```
config firewall proxy-policy
    edit 100
        set proxy wanopt
        set dstintf "port23"
        set srcaddr "all"
        set dstaddr "all"
        set service "ALL"
        set action accept
        set schedule "always"
        set utm-status enable
        set profile-protocol-options "protocol"
        set ssl-ssh-profile "ssl"
    next
end
```

To confirm that traffic is accelerated:

1. On the client PC, curl a 10MB test sample for the first time:

```
root@client:/tmp# curl -k https://172.16.200.144/test_10M.pdf -O
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 9865k  100 9865k    0     0   663k      0  0:00:14  0:00:15 --:--:-- 1526k
```

   It takes 15 seconds to finish the download.

2. On FGT_A, check the WAD statistics:

```
# diagnose wad stats worker.tunnel
comp.n_in_raw_bytes                  10155840
comp.n_in_comp_bytes                 4548728
comp.n_out_raw_bytes                 29624
comp.n_out_comp_bytes                31623
```

```
# diagnose wad stats worker.protos.http
wan.bytes_in                        0
wan.bytes_out                       0
lan.bytes_in                        760
lan.bytes_out                       10140606
tunnel.bytes_in                     4548728
tunnel.bytes_out                    31623
```

3. Curl the same test sample a second time:

```
root@client:/tmp# curl -k https://172.16.200.144/test_10M.pdf -O
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
```

```
                                 Dload  Upload   Total   Spent    Left  Speed
100 9865k  100 9865k    0     0   663k      0  0:00:01  0:00:01 --:--:-- 1526k
```

It now takes less than one second to finish the download.

4. On FGT_A, check the WAD statistics again:

```
# diagnose wad stats worker.tunnel
comp.n_in_raw_bytes                  10181157
comp.n_in_comp_bytes                 4570331
comp.n_out_raw_bytes                 31627
comp.n_out_comp_bytes                34702
```

```
# diagnose wad stats worker.protos.http
wan.bytes_in                        0
wan.bytes_out                       0
lan.bytes_in                        1607
lan.bytes_out                       20286841
tunnel.bytes_in                     4570331
tunnel.bytes_out                    34702
```

The tunnel bytes are mostly unchanged, but the LAN bytes are doubled. This means that the bytes of the second curl come from the cache, showing that the traffic is accelerated.

To confirm that a curl using the GCM cipher is accepted and accelerated:

1. On the client PC, curl a 10MB test sample with the GCM cipher:

```
root@client:/tmp# curl -v -k --ciphers DHE-RSA-AES128-GCM-SHA256
https://172.16.200.144/test_10M.pdf -O
*   Trying 172.16.200.144...
* TCP_NODELAY set
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--     0*
Connected to 172.16.200.144 (172.16.200.144) port 443 (#0)
* ALPN, offering h2
* ALPN, offering http/1.1
* Cipher selection: DHE-RSA-AES128-GCM-SHA256
* successfully set certificate verify locations:
*   CAfile: /etc/ssl/certs/ca-certificates.crt
  CApath: none
} [5 bytes data]
* TLSv1.3 (OUT), TLS handshake, Client hello (1):
} [512 bytes data]
* TLSv1.3 (IN), TLS handshake, Server hello (2):
{ [100 bytes data]
* TLSv1.2 (IN), TLS handshake, Certificate (11):
{ [1920 bytes data]
* TLSv1.2 (IN), TLS handshake, Server key exchange (12):
{ [783 bytes data]
* TLSv1.2 (IN), TLS handshake, Server finished (14):
{ [4 bytes data]
* TLSv1.2 (OUT), TLS handshake, Client key exchange (16):
} [262 bytes data]
* TLSv1.2 (OUT), TLS change cipher, Change cipher spec (1):
} [1 bytes data]
* TLSv1.2 (OUT), TLS handshake, Finished (20):
```

```
} [16 bytes data]
* TLSv1.2 (IN), TLS handshake, Finished (20):
{ [16 bytes data]
* SSL connection using TLSv1.2 / DHE-RSA-AES128-GCM-SHA256
* ALPN, server accepted to use http/1.1
* Server certificate:
*  subject: CN=ubuntu
*  start date: Sep 20 21:38:01 2018 GMT
*  expire date: Sep 17 21:38:01 2028 GMT
*  issuer: C=US; ST=California; L=Sunnyvale; O=Fortinet; OU=Certificate Authority;
CN=Fortinet Untrusted CA; emailAddress=support@fortinet.com
*  SSL certificate verify result: self signed certificate in certificate chain (19),
continuing anyway.
} [5 bytes data]
> GET /test_10M.pdf HTTP/1.1
> Host: 172.16.200.144
> User-Agent: curl/7.64.1
> Accept: */*
>
{ [5 bytes data]
< HTTP/1.1 200 OK
< Date: Sat, 12 Jun 2021 00:31:08 GMT
< Server: Apache/2.4.37 (Ubuntu)
< Upgrade: h2,h2c
< Connection: Upgrade
< Last-Modified: Fri, 29 Jan 2021 20:10:25 GMT
< ETag: "9a2572-5ba0f98404aa5"
< Accept-Ranges: bytes
< Content-Length: 10102130
< Content-Type: application/pdf
<
{ [5 bytes data]
100 9865k  100 9865k    0     0  16.7M      0 --:--:-- --:--:-- --:--:-- 16.8M
* Connection #0 to host 172.16.200.144 left intact
* Closing connection 0
```

To confirm that a curl using the ChaCha cipher is accepted and accelerated:

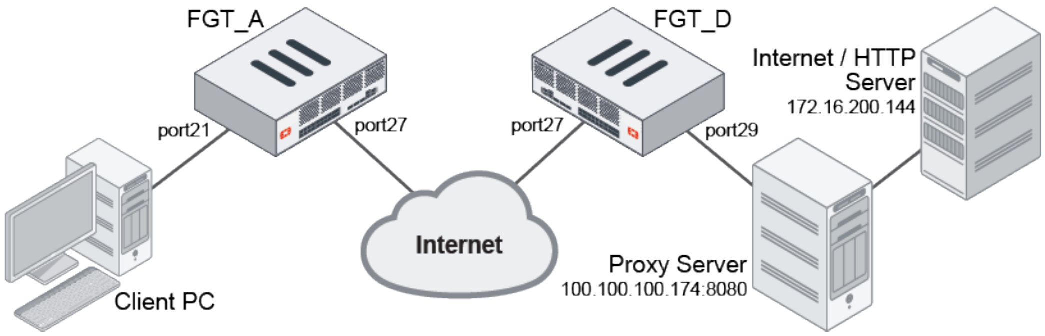
1. On the client PC, curl a 10MB test sample with the ChaCha cipher:

```
root@client:/tmp# curl -v -k --ciphers ECDHE-RSA-CHACHA20-POLY1305
https://172.16.200.144/test.doc -O
*   Trying 172.16.200.144...
* TCP_NODELAY set
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--     0*
Connected to 172.16.200.144 (172.16.200.144) port 443 (#0)
* ALPN, offering h2
* ALPN, offering http/1.1
* Cipher selection: ECDHE-RSA-CHACHA20-POLY1305
* successfully set certificate verify locations:
*   CAfile: /etc/ssl/certs/ca-certificates.crt
  CApath: none
} [5 bytes data]
* TLSv1.3 (OUT), TLS handshake, Client hello (1):
```

```
} [512 bytes data]
* TLSv1.3 (IN), TLS handshake, Server hello (2):
{ [100 bytes data]
* TLSv1.2 (IN), TLS handshake, Certificate (11):
{ [1920 bytes data]
* TLSv1.2 (IN), TLS handshake, Server key exchange (12):
{ [300 bytes data]
* TLSv1.2 (IN), TLS handshake, Server finished (14):
{ [4 bytes data]
* TLSv1.2 (OUT), TLS handshake, Client key exchange (16):
} [37 bytes data]
* TLSv1.2 (OUT), TLS change cipher, Change cipher spec (1):
} [1 bytes data]
* TLSv1.2 (OUT), TLS handshake, Finished (20):
} [16 bytes data]
* TLSv1.2 (IN), TLS handshake, Finished (20):
{ [16 bytes data]
* SSL connection using TLSv1.2 / ECDHE-RSA-CHACHA20-POLY1305
* ALPN, server accepted to use http/1.1
* Server certificate:
*  subject: CN=ubuntu
*  start date: Sep 20 21:38:01 2018 GMT
*  expire date: Sep 17 21:38:01 2028 GMT
*  issuer: C=US; ST=California; L=Sunnyvale; O=Fortinet; OU=Certificate Authority;
CN=Fortinet Untrusted CA; emailAddress=support@fortinet.com
*  SSL certificate verify result: self signed certificate in certificate chain (19),
continuing anyway.
} [5 bytes data]
> GET /test.doc HTTP/1.1
> Host: 172.16.200.144
> User-Agent: curl/7.64.1
> Accept: */*
>
{ [5 bytes data]
< HTTP/1.1 200 OK
< Date: Sat, 12 Jun 2021 00:32:11 GMT
< Server: Apache/2.4.37 (Ubuntu)
< Upgrade: h2,h2c
< Connection: Upgrade
< Last-Modified: Wed, 05 May 2021 21:59:49 GMT
< ETag: "4c00-5c19c504b63f4"
< Accept-Ranges: bytes
< Content-Length: 19456
< Content-Type: application/msword
<
{ [5 bytes data]
100 19456  100 19456    0     0   137k      0 --:--:-- --:--:-- --:--:--  138k
* Connection #0 to host 172.16.200.144 left intact
* Closing connection 0
```

Example 2

In this example, an external proxy is added to the configuration in Example 1.



To reconfigure FGT_A:

```
config firewall profile-protocol-options
    edit "protocol"
        config http
            set ports 80 8080
            unset options
            unset post-lang
        end
    next
end
```

To reconfigure FGT_D:

1. Configure a new firewall policy for traffic passing from port27 to port29:

```
config firewall policy
    edit 1
        set name "WANOPT-B"
        set srcintf "port27"
        set dstintf "port29"
        set action accept
        set srcaddr "all"
        set dstaddr "all"
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set inspection-mode proxy
        set wanopt enable
        set wanopt-detection passive
        set nat enable
    next
end
```

2. Configure a proxy policy for traffic on destination interface port29:

```
config firewall proxy-policy
    edit 100
        set proxy wanopt
        set dstintf "port29"
        set srcaddr "all"
        set dstaddr "all"
        set service "ALL"
        set action accept
```

```
        set schedule "always"
        set profile-protocol-options "protocol"
        set ssl-ssh-profile "ssl"
    next
end
```

To confirm that HTTPS traffic is still being accelerated:

1. On the client PC, curl the same 10MB test sample through the explicit proxy:

```
root@client:/tmp# curl -x 100.100.100.174:8080 -v -k https://172.16.200.144/test_10M.pdf
-O
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 9865k  100 9865k    0     0   663k      0  0:00:01  0:00:01 --:--:-- 1526k
```

   It takes less than a second to finish the download.

Agentless NTLM authentication for web proxy

Agentless Windows NT LAN Manager (NTLM) authentication includes support for the following items:

- Multiple servers
- Individual users

You can use multiple domain controller servers for the agentless NTLM. They can be used for load balancing and high service stability.

You can also use user-based matching in groups for Kerberos and agentless NTLM. In these scenarios, FortiOS matches the user's group information from an LDAP server.

To support multiple domain controllers for agentless NTLM using the CLI:

1. Configure an LDAP server:

```
config user ldap
    edit "ldap-kerberos"
        set server "172.18.62.177"
        set cnid "cn"
        set dn "dc=fortinetqa,dc=local"
        set type regular
        set username "CN=root,CN=Users,DC=fortinetqa,DC=local"
        set password *********
    next
end
```

2. Configure multiple domain controllers:

```
config user domain-controller
    edit "dc1"
        set ip-address 172.18.62.177
        config extra-server
            edit 1
                set ip-address 172.18.62.220
            next
        end
```

```
        set ldap-server "ldap-kerberos"
    next
end
```

3. Create an authentication scheme and rule:

```
config authentication scheme
    edit "au-ntlm"
        set method ntlm
        set domain-controller "dc1"
    next
end
```

```
config authentication rule
    edit "ru-ntlm"
        set srcaddr "all"
        set ip-based disable
        set active-auth-method "au-ntlm"
    next
end
```

4. In the proxy policy, append the user group for authorization:

```
config firewall proxy-policy
    edit 1
        set proxy explicit-web
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set service "web"
        set action accept
        set schedule "always"
        set groups "ldap-group"
        set utm-status enable
        set av-profile "av"
        set ssl-ssh-profile "deep-custom"
    next
end
```

   This configuration uses a round-robin method. When the first user logs in, the FortiGate sends the authentication request to the first domain controller. Later when another user logs in, the FortiGate sends the authentication request to another domain controller.

5. Verify the behavior after the user successfully logs in:

```
# diagnose wad user list
ID: 1825, IP: 10.1.100.71, VDOM: vdom1
    user name   : test1
    duration    : 497
    auth_type   : Session
    auth_method : NTLM
    pol_id      : 1   g_id        : 5
    user_based  : 0   e
    xpire      : 103
    LAN:
        bytes_in=2167 bytes_out=7657
    WAN:
        bytes_in=3718 bytes_out=270
```

To support individual users for agentless NTLM using the CLI:

1. Configure an LDAP server:

```
config user ldap
    edit "ldap-kerberos"
        set server "172.18.62.177"
        set cnid "cn"
        set dn "dc=fortinetqa,dc=local"
        set type regular
        set username "CN=root,CN=Users,DC=fortinetqa,DC=local"
        set password *********
    next
end
```

2. Configure the user group and allow user-based matching:

```
config user group
    edit "ldap-group"
        set member "ldap" "ldap-kerberos"
        config match
            edit 1
                set server-name "ldap-kerberos"
                set group-name "test1"
            next
        end
    next
end
```

3. Create an authentication scheme and rule:

```
config authentication scheme
    edit "au-ntlm"
        set method ntlm
        set domain-controller "dc1"
    next
end
```

```
config authentication rule
    edit "ru-ntlm"
        set srcaddr "all"
        set ip-based disable
        set active-auth-method "au-ntlm"
    next
end
```

4. In the proxy policy, append the user group for authorization:

```
config firewall proxy-policy
    edit 1
        set proxy explicit-web
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set service "web"
        set action accept
        set schedule "always"
        set groups "ldap-group"
        set utm-status enable
```

```
        set av-profile "av"
        set ssl-ssh-profile "deep-custom"
    next
end
```

This implementation lets you configure a single user instead of a whole group. The FortiGate will now allow the user named `test1`.

To verify the configuration using the CLI:

```
diagnose wad user list
    ID: 1827, IP: 10.1.15.25, VDOM: vdom1
    user name   : test1
    duration    : 161
    auth_type   : Session
    auth_method : NTLM
    pol_id      : 1
    g_id        : 5
    user_based  : 0
    expire      : 439
    LAN:
            bytes_in=1309 bytes_out=4410
      WAN:
            bytes_in=2145 bytes_out=544
```

Multiple LDAP servers in Kerberos keytabs and agentless NTLM domain controllers

Multiple LDAP servers can be configured in Kerberos keytabs and agentless NTLM domain controllers for multi-forest deployments.

To use multiple LDAP servers in Kerberos keytabs and agentless NTLM domain controllers:

1. Add multiple LDAP servers:

```
config user ldap
    edit "ldap-kerberos"
        set server "172.16.200.98"
        set cnid "cn"
        set dn "dc=fortinetqa,dc=local"
        set type regular
        set username "CN=root,CN=Users,DC=fortinetqa,DC=local"
        set password xxxxxxxxx
    next
    edit "ldap-two"
        set server "172.16.106.128"
        set cnid "cn"
        set dn "OU=Testing,DC=ad864r2,DC=com"
        set type regular
        set username "cn=Testadmin,cn=users,dc=AD864R2,dc=com"
        set password xxxxxxxxx
    next
end
```

2. Configure a Kerberos keytab entry that uses both LDAP servers:

```
config user krb-keytab
    edit "http_service"
        set pac-data disable
        set principal "HTTP/FGT.FORTINETQA.LOCAL@FORTINETQA.LOCAL"
        set ldap-server "ldap-kerberos" "ldap-two"
        set keytab xxxxxxxxx
    next
end
```

3. Configure a domain controller that uses both LDAP servers:

```
config user domain-controller
    edit "dc1"
        set ip-address 172.16.200.98
        set ldap-server "ldap-two" "ldap-kerberos"
    next
end
```

Learn client IP addresses

Learning the actual client IP addresses is imperative for authorization. This function identifies the real client IP address when there is a NATing device between the FortiGate and the client.

```
config web-proxy global
    set learn-client-ip {enable | disable}
    set learn-client-ip-from-header {true-client-ip | x-real-ip | x-forwarded-for}
    set learn-client-ip-srcaddr <address> ... <address>
end
```

`learn-client-ip {enable | disable}`	Enable/disable learning the client's IP address from headers.
`learn-client-ip-from-header {true-client-ip | x-real-ip | x-forwarded-for}`	Learn client IP addresses from the specified headers.
`learn-client-ip-srcaddr <address> ... <address>`	The source address names.

Example

In this example, the real client IP address is used to match a policy for FSSO authentication.

To enable learning the client IP address:

```
config web-proxy global
    set proxy-fqdn "default.fqdn"
    set webproxy-profile "default"
    set learn-client-ip enable
       set learn-client-ip-from-header x-forwarded-for
    set learn-client-ip-srcaddr "all"
end
```

To configure the proxy policy:

```
config firewall proxy-policy
    edit 1
        set proxy explicit-web
        set dstintf "mgmt1"
        set srcaddr "all"
        set dstaddr "all"
        set service "w"
        set action accept
        set schedule "always"
        set groups "fsso1"
        set utm-status enable
        set av-profile "default"
        set dlp-profile "default"
        set profile-protocol-options "default"
        set ssl-ssh-profile "deep-inspection"
    next
end
```

To configure the authentication scheme and rule:

```
config authentication scheme
    edit "scheme1"
        set method fsso
    next
end
```

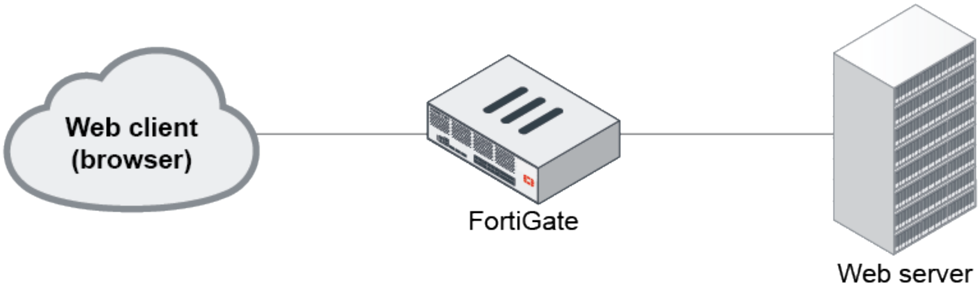
```
config authentication rule
    edit "rule1"
        set srcaddr "all"
        set sso-auth-method "scheme1"
    next
end
```

Explicit proxy authentication over HTTPS

When a HTTP request requires authentication in an explicit proxy, the authentication can be redirected to a secure HTTPS captive portal. Once authentication is complete, the client can be redirected back to the original destination over HTTP.

Example

A user visits a website via HTTP through the explicit web proxy on a FortiGate. The user is required to authenticate by either basic or form IP-based authentication for the explicit web proxy service. The user credentials need to be transmitted over the networks in a secured method over HTTPS rather than in plain text. The user credentials are protected by redirecting the client to a captive portal of the FortiGate over HTTPS for authentication where the user credentials are encrypted and transmitted over HTTPS.



In this example, explicit proxy authentication over HTTPS is configured with form IP-based authentication. Once configured, you can enable authorization for an explicit web proxy by configuring users or groups in the firewall proxy policy.

To configure explicit proxy authentication over HTTPS:

1. Configure the authentication settings:

```
config authentication setting
    set captive-portal-type fqdn
    set captive-portal "fgt-cp"
    set auth-https enable
end
```

2. Configure the authentication scheme:

```
config authentication scheme
    edit "form"
        set method form
        set user-database "local-user-db"
    next
end
```

3. Configure the authentication rule:

```
config authentication rule
    edit "form"
        set srcaddr "all"
        set active-auth-method "form"
    next
end
```

If a session-based basic authentication method is used, enable `web-auth-cookie`.

4. Configure the firewall address:

```
config firewall address
    edit "fgt-cp"
        set type fqdn
        set fqdn "fgt.fortinetqa.local"
    next
end
```

5. Configure the interface:

```
config system interface
    edit "port10"
        set ip 10.1.100.1 255.255.255.0
        set explicit-web-proxy enable
        set proxy-captive-portal enable
    next
end
```

6. Configure a firewall proxy policy with users or groups (see Explicit web proxy on page 220).

Verification

When a client visits a HTTP website, the client will be redirected to the captive portal for authentication by HTTPS. For example, the client could be redirected to a URL by a HTTP 303 message similar to the following:

HTTP/1.1 303 See Other

Connection: close

Content-Type: text/html

Cache-Control: no-cache

Location: https://fgt.fortinetqa.local:7831/XX/YY/ZZ/cpauth?scheme=http&4Tmthd=0&host=172.16.200.46&port=80&rule=75&uri=Lw==&

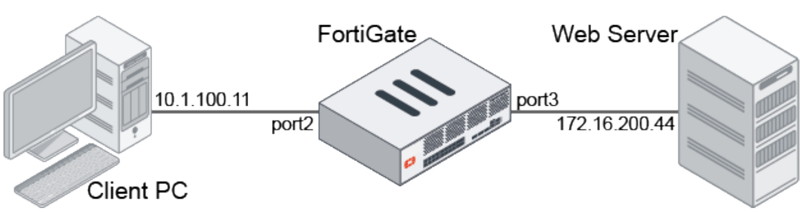
Content-Length: 0

The captive portal URL used for authentication is *https://fgt.fortinetqa.local:7831/...*. Once the authentication is complete with all user credentials protected by HTTPS, the client is redirected to the original HTTP website they intended to visit.

mTLS client certificate authentication

FortiGate supports client certificate authentication used in mutual Transport Layer Security (mTLS) communication between a client and server. Clients are issued certificates by the CA, and an access proxy configured on the FortiGate uses the new certificate method in the authentication scheme to identify and approve the certificate provided by the client when they try to connect to the access proxy. The FortiGate can also add the HTTP header X-Forwarded-Client-Cert to forward the certificate information to the server.

Examples



In these examples, the access proxy VIP IP address is 10.1.100.200.

Example 1

In this example, clients are issued unique client certificates from your CA. The FortiGate authenticates the clients by their user certificate before allowing them to connect to the access proxy. The access server acts as a reverse proxy for the web server that is behind the FortiGate.

This example assumes that you have already obtained the public CA certificate from your CA, the root CA of the client certificate has been imported (`CA_Cert_1`), and the client certificate has been distributed to the endpoints.

To configure the FortiGate:

1. Configure user authentication. Both an authentication scheme and rule must be configured, as the authentication is applied on the access proxy:

```
config authentication scheme
    edit "mtls"
        set method cert
        set user-cert enable
    next
end
```

```
config authentication rule
    edit "mtls"
        set srcintf "port2"
        set srcaddr "all"
        set dstaddr "all"
        set active-auth-method "mtls"
    next
end
```

2. Select the CA or CAs used to verify the client certificate:

```
config authentication setting
    set user-cert-ca "CA_Cert_1"
end
```

3. Configure the users. Users can be matched based on either the common-name on the certificate or the trusted issuer.
   - Verify the user based on the common name on the certificate:

```
config user certificate
    edit "single-certificate"
        set type single-certificate
        set common-name "client.fortinet.com"
    next
end
```

   - Verify the user based on the CA issuer:

```
config user certificate
    edit "trusted-issuer"
        set type trusted-issuer
        set issuer "CA_Cert_1"
    next
end
```

4. Configure the access proxy VIP. The SSL certificate is the server certificate that is presented to the user as they connect:

```
config firewall vip
    edit "mTLS"
        set type access-proxy
        set extip 10.1.100.200
        set extintf "port2"
        set server-type https
        set extport 443
        set ssl-certificate "Fortinet_CA_SSL"
    next
end
```

5. Configure the access proxy policy, including the real server to be mapped. To request the client certificate for authentication, `client-cert` is enabled:

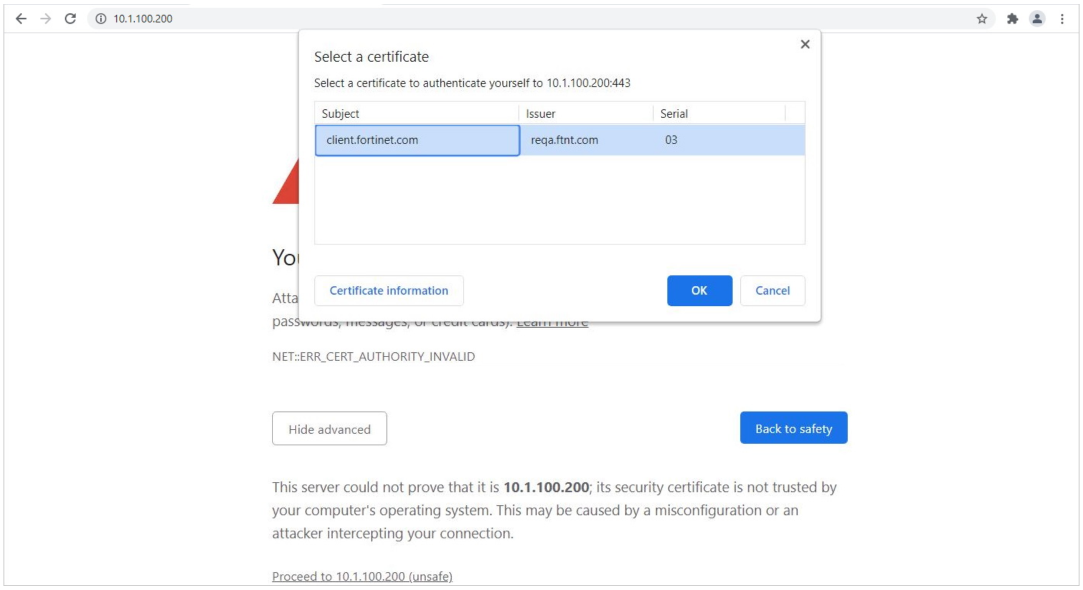
```
config firewall access-proxy
    edit "mTLS-access-proxy"
        set vip "mTLS"
        set client-cert enable
        set empty-cert-action accept
        config api-gateway
            edit 1
                config realservers
                    edit 1
                        set ip 172.16.200.44
                    next
                end
            next
        end
    next
end
```

6. Configure the proxy policy to apply authentication and the security profile, selecting the appropriate user object depending on the user type:

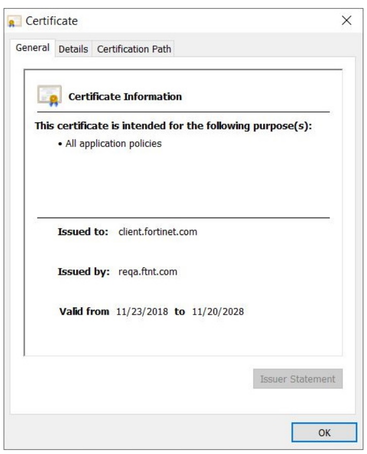
```
config firewall proxy-policy
    edit 3
        set proxy access-proxy
        set access-proxy "mTLS-access-proxy"
        set srcintf "port2"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set users {"single-certificate" | "trusted-issuer"}
        set utm-status enable
        set ssl-ssh-profile "deep-inspection-clone"
        set av-profile "av"
    next
end
```

To verify the results:

1. In a web browser, access the VIP address. This example uses Chrome.
2. When prompted, select the client certificate, then click *OK*.



3. Click *Certificate information* to view details about the certificate.



4. On the FortiGate, check the traffic logs.
   - If client certificate authentication passes:

     ```
     1: date=2021-06-03 time=15:48:36 eventtime=1622760516866635697 tz="-0700"
     logid="0000000010" type="traffic" subtype="forward" level="notice" vd="vdom1"
     srcip=10.1.100.11 srcport=45532 srcintf="port2" srcintfrole="undefined"
     dstcountry="Reserved" srccountry="Reserved" dstip=172.16.200.44 dstport=443
     dstintf="vdom1" dstintfrole="undefined" sessionid=154900 service="HTTPS"
     wanoptapptype="web-proxy" proto=6 action="accept" policyid=3 policytype="proxy-
     policy" poluuid="af5e2df2-c321-51eb-7d5d-42fa58868dcb" duration=0 user="single-
     certificate" wanin=2550 rcvdbyte=2550 wanout=627 lanin=4113 sentbyte=4113 lanout=2310
     appcat="unscanned"
     ```

   - If the CA issuer is used to verify the client:

     ```
     1: date=2021-06-03 time=15:43:02 eventtime=1622760182384776037 tz="-0700"
     logid="0000000010" type="traffic" subtype="forward" level="notice" vd="vdom1"
     srcip=10.1.100.11 srcport=45514 srcintf="port2" srcintfrole="undefined"
     dstcountry="Reserved" srccountry="Reserved" dstip=10.1.100.200 dstport=443
     dstintf="vdom1" dstintfrole="undefined" sessionid=153884 service="HTTPS"
     ```

```
wanoptapptype="web-proxy" proto=6 action="accept" policyid=3 policytype="proxy-
policy" poluuid="af5e2df2-c321-51eb-7d5d-42fa58868dcb" duration=0 user="trusted-
issuer" wanin=0 rcvdbyte=0 wanout=0 lanin=4089 sentbyte=4089 lanout=7517
appcat="unscanned" utmaction="block" countweb=1 crscore=30 craction=8 utmref=65535-0
```

- If the client certificate authentication fails, and the traffic is blocked:

```
1: date=2021-06-03 time=15:45:53 eventtime=1622760353789703671 tz="-0700"
logid="0000000013" type="traffic" subtype="forward" level="notice" vd="vdom1"
srcip=10.1.100.11 srcport=45518 srcintf="port2" srcintfrole="undefined"
dstip=172.16.200.44 dstport=443 dstintf="vdom1" dstintfrole="undefined"
srccountry="Reserved" dstcountry="Reserved" sessionid=154431 proto=6 action="deny"
policyid=0 policytype="proxy-policy" user="single-certificate" service="HTTPS"
trandisp="noop" url="https://10.1.100.200/" agent="curl/7.68.0" duration=0 sentbyte=0
rcvdbyte=0 sentpkt=0 rcvdpkt=0 appcat="unscanned" crscore=30 craction=131072
crlevel="high" msg="Traffic denied because of explicit proxy policy"
```

Example 2

In this example, the same configuration as in Example 1 is used, with a web proxy profile added to enable adding the client certificate to the HTTP header X-Forwarded-Client-Cert. The header is then forwarded to the server.

To configure the FortiGate:

1. Repeat steps 1 to 6 of Example 1, using the common name on the certificate to verify the user.
2. Configure a web proxy profile that adds the HTTP x-forwarded-client-cert header in forwarded requests:

```
config web-proxy profile
    edit "mtls"
        set header-x-forwarded-client-cert add
    next
end
```

3. Configure the proxy policy to apply authentication, the security profile, and web proxy profile:

```
config firewall proxy-policy
    edit 3
        set uuid af5e2df2-c321-51eb-7d5d-42fa58868dcb
        set proxy access-proxy
        set access-proxy "mTLS-access-proxy"
        set srcintf "port2"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set logtraffic all
        set users "single-certificate"
        set webproxy-profile "mtls"
        set utm-status enable
        set ssl-ssh-profile "deep-inspection-clone"
        set av-profile "av"
    next
end
```

To verify the results:

The WAD debug shows that the FortiGate adds the client certificate information to the HTTP header. The added header cannot be checked using the sniffer, because the FortiGate encrypts the HTTP header to forward it to the server.

1. Enable WAD debug on all categories:
```
# diagnose wad debug enable category all
```
2. Set the WAD debug level to verbose:
```
# diagnose wad debug enable level verbose
```
3. Enable debug output:
```
# diagnose debug enable
```
4. Check the debug output.
   - When the FortiGate receives the client HTTP request:
```
[0x7fc8d4bc4910] Received request from client: 10.1.100.11:45544

GET / HTTP/1.1
Host: 10.1.100.200
User-Agent: curl/7.68.0
Accept: */*
```
   - When the FortiGate adds the client certificate in to the HTTP header and forwards the client HTTP request:
```
[0x7fc8d4bc4910] Forward request to server:
GET / HTTP/1.1
Host: 172.16.200.44
User-Agent: curl/7.68.0
Accept: */*
X-Forwarded-Client-Cert: -----BEGIN CERTIFICATE-----
MIIFXzCCA0egAwI...aCFHDHlR+wb39s=
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
MIIFpTCCA42gAwI...OtDtetkNoFLbvb
-----END CERTIFICATE-----
```

DHCP server

A DHCP server leases IP addresses from a defined address range to clients on the network that request dynamically assigned addresses.

A DHCP server can be in server or relay mode. In server mode, you can define one or more address ranges it assigns addresses from, and options such as the default gateway, DNS server, lease time, and other advanced options. In relay mode, the interface forwards DHCP requests from DHCP clients to an external DHCP server and returns the responses to the DHCP clients. The DHCP server must have appropriate routing so that its response packets to the DHCP clients arrive at the unit.

- DHCP options on page 290
- IP address assignment with relay agent information option on page 291
- DHCP client options on page 293

Configure a DHCP server on an interface

To configure a DHCP server in the GUI:

1. Go to *Network > Interfaces*.
2. Edit an interface.
3. Enable the *DHCP Server* option and configure the settings.
4. Click *OK*.

To configure a DHCP server in the CLI:

```
config system dhcp server
    edit 1
        set dns-service default
        set default-gateway 192.168.1.2
        set netmask 255.255.255.0
        set interface "port1"
        config ip-range
            edit 1
                set start-ip 192.168.1.1
                set end-ip 192.168.1.1
            next
            edit 2
                set start-ip 192.168.1.3
                set end-ip 192.168.1.254
            next
        end
        set timezone-option default
        set tftp-server "172.16.1.2"
    next
end
```

Configure a DHCP relay on an interface

To configure a DHCP relay in the GUI:

1. Go to *Network > Interfaces*.
2. Edit an interface.
3. Enable the *DHCP Server* option and set *DHCP status* to *Disabled*.
4. Expand the *Advanced* section and set *Mode* to *Relay*.
5. Enter the *DHCP Server IP*.
6. Click *OK*.

To configure a DHCP relay in the CLI:

1. Configure the interface:

```
config system interface
    edit "port2"
        set vdom "root"
        set dhcp-relay-service enable
```

```
        set ip 10.1.1.5 255.255.255.0
        set allowaccess ping https ssh fabric
        set type physical
        set snmp-index 4
        set dhcp-relay-ip "192.168.20.10"
    next
end
```

2. On the DHCP server settings for the interface, set the status to disable:

```
config system dhcp server
    edit 17
        set status disable
        set dns-service default
        set default-gateway 10.1.1.5
        set netmask 255.255.255.0
        set interface "port2"
    next
end
```

Configure a DHCP server and relay on an interface

A FortiGate interface can be configured to work in DHCP server mode to lease out addresses, and at the same time relay the DHCP packets to another device, such as a FortiNAC to perform device profiling.

The DHCP message to be forwarded to the relay server under the following conditions:

- `dhcp-relay-request-all-server` is enabled
- Message type is either DHCPDISCOVER or DHCPINFORM
- Client IP address in client message is 0
- Server ID is NULL in the client message
- Server address is a broadcast address (255.255.255.255)
- Server address is 0

To configure a DHCP server and relay in the GUI:

1. Go to *Network > Interfaces*.
2. Edit an interface.
3. Enable the *DHCP Server* option and set *DHCP status* to *Enabled*.
4. Edit the address range as required.
5. Expand the *Advanced* section and set *Mode* to *Relay*.
6. Enter the *DHCP Server IP*.
7. Click *OK*.
8. In the CLI, enable `dhcp-relay-request-all-server`.

To configure a DHCP server and relay in the CLI:

1. Configure the interface:

```
config system interface
    edit "port2"
        set vdom "root"
```

```
        set dhcp-relay-service enable
        set ip 10.1.1.5 255.255.255.0
        set allowaccess ping https ssh fabric
        set type physical
        set snmp-index 4
        set dhcp-relay-ip "192.168.20.10"
        set dhcp-relay-request-all-server enable
    next
end
```

2. Configure the DHCP server settings:

```
config system dhcp server
    edit 17
        set status enable
        set dns-service default
        set default-gateway 10.1.1.5
        set netmask 255.255.255.0
        set interface "port2"
    next
end
```

DHCP options

When adding a DHCP server, you can include DHCP codes and options. The DHCP options are BOOTP vendor information fields that provide additional vendor-independent configuration parameters to manage the DHCP server. For example, you might need to configure a FortiGate DHCP server that gives out a separate option as well as an IP address, such as an environment that needs to support PXE boot with Windows images.

The option numbers and codes are specific to the application. The documentation for the application indicates the values to use. The option is a value between 1 and 255.

For detailed information about DHCP options, see RFC 2132, DHCP Options and BOOTP Vendor Extensions.

Option 82

The DHCP relay agent information option (option 82 in RFC 3046) helps protect the FortiGate against attacks such as spoofing (forging) of IP addresses and MAC addresses, and DHCP IP address starvation.

This option is disabled by default. However, when `dhcp-relay-service` is enabled, `dhcp-relay-agent-option` becomes enabled.

To configure the DHCP relay agent option using the CLI:

```
config system interface
    edit <interface>
        set vdom root
        set dhcp-relay-service enable
        set dhcp-relay-ip <ip>
        set dhcp-relay-agent-option enable
        set vlanid <id>
    next
end
```

See IP address assignment with relay agent information option on page 291 for an example.

Option 42

This option specifies a list of the NTP servers available to the client by IP address.

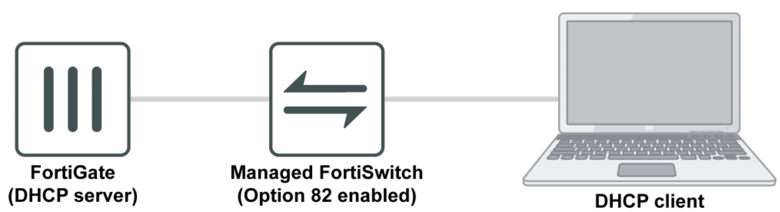
```
config system dhcp server
    edit 2
        set ntp-service {local | default | specify}
        set ntp-server1 <class_ip>
        set ntp-server2 <class_ip>
        set ntp-server3 <class_ip>
    next
end
```

The NTP service options include:

- `local`: The IP address of the interface that the DHCP server is added to becomes the client's NTP server IP address.
- `default`: Clients are assigned the FortiGate's configured NTP servers.
- `specify`: Specify up to three NTP servers in the DHCP server configuration.

IP address assignment with relay agent information option

Option 82 (DHCP relay information option) helps protect the FortiGate against attacks such as spoofing (or forging) of IP and MAC addresses, and DHCP IP address starvation.



The following CLI variables are included in the `config system dhcp server` > `config reserved-address` command:

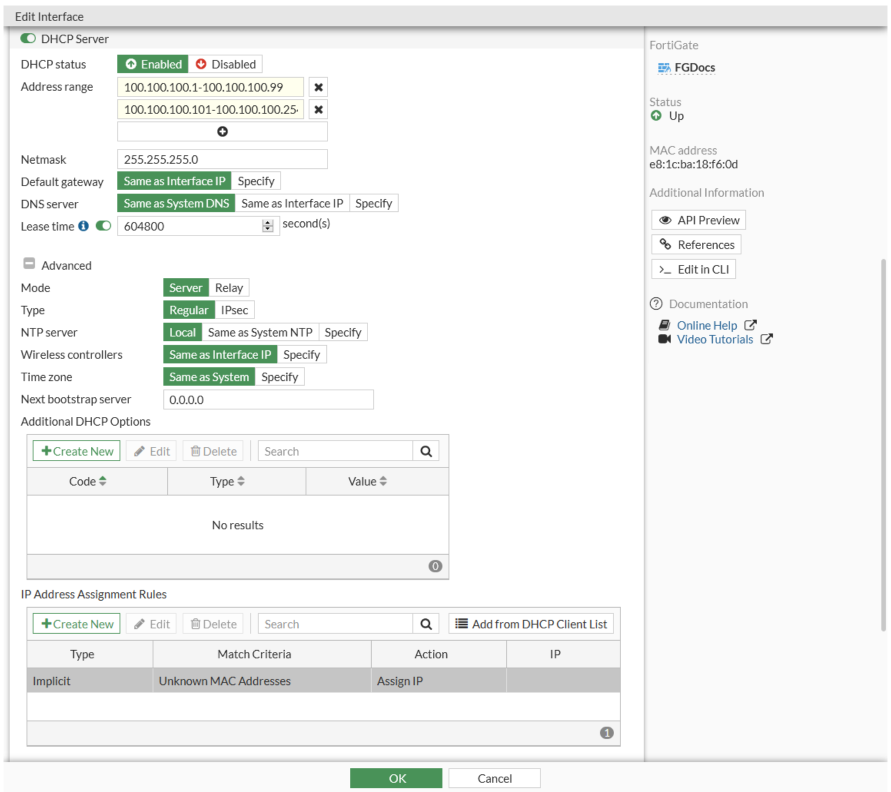
circuit-id-type {hex | string}	DHCP option type; hex or string (default).
circuit-id <value>	Option 82 circuit ID of the client that will get the reserved IP address. Format: *vlan-mod-port* • vlan: VLAN ID (2 bytes) • mod: 1 = snoop, 0 = relay (1 byte) • port: port number (1 byte)
remote-id-type {hex | string}	DHCP option type; hex or string (default).
remote-id <value>	Option 82 remote ID of the client that will get the reserved IP address. Format: the MAC address of the client.
type {mac | option82}	The DHCP reserved address type; mac (default) or option82.

To create an IP address assignment rule using option 82 in the GUI:

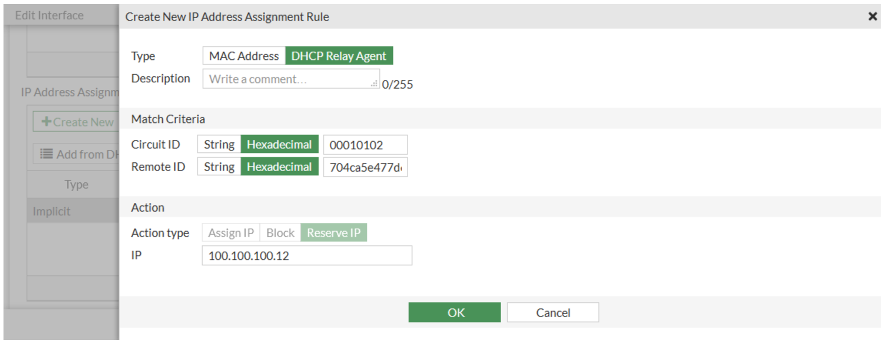
1. Go to *Network > Interfaces*.
2. Edit an existing port, or create a new one.

> The port *Role* must be *LAN* or *Undefined*.

3. Enable *DHCP Server*.
4. Configure the address ranges and other settings as needed.
5. Click + to expand the *Advanced* options.



6. In the *IP Address Assignment Rules* table, click *Create New*.
   The *Create New IP Address Assignment Rule* pane opens.
7. Configure the new rule:
   a. For the *Type*, select *DHCP Relay Agent*.
   b. Enter the *Circuit ID* and *Remote ID*.
   c. Enter the *IP* address that will be reserved.



8. Click *OK*.

To create an IP address assignment rule using option 82 with the CLI:

```
config system dhcp server
    edit 1
        set netmask 255.255.255.0
        set interface "port4"
        config ip-range
            edit 1
                set start-ip 100.100.100.1
                set end-ip 100.100.100.99
            next
            edit 2
                set start-ip 100.100.100.101
                set end-ip 100.100.100.254
            next
        end
        config reserved-address
            edit 1
                set type option82
                set ip 100.100.100.12
                set circuit-id-type hex
                set circuit-id "00010102"
                set remote-id-type hex
                set remote-id "704ca5e477d6"
            next
        end
    next
end
```

DHCP client options

When an interface is in DHCP addressing mode, DHCP client options can be configured in the CLI. For example, a vendor class identifier (usually DCHP client option 60) can be specified so that a request can be matched by a specific DHCP offer.

Multiple options can be configured, but any options not recognized by the DHCP server are discarded.

To configure client option 60 - vendor class identifier:

```
config system interface
    edit port1
        set vdom vdom1
        set mode dhcp
        config client-options
            edit 1
                set code 60
                set type hex
                set value aabbccdd
            next
        end
        set type physical
        set snmp-index 4
    next
end
```

Variable	Description
`code <integer>`	DHCP client option code (0 - 255, default = 0). See Dynamic Host Configuration Protocol (DHCP) and Bootstrap Protocol (BOOTP) Parameters for a list of possible options.
`type {hex | string | ip | fqdn}`	DHCP client option type (default = hex).
`value <string>`	DHCP client option value.
`ip <ip>`	DHCP client option IP address. This option is only available when `type` is `ip`.

Static routing

Static routing is one of the foundations of firewall configuration. It is a form of routing in which a device uses manually-configured routes. In the most basic setup, a firewall will have a default route to its gateway to provide network access. In a more complex setup with dynamic routing, ADVPN, or SD-WAN involved, you would still likely find static routes being deployed.

This section explores concepts in using static routing and provides examples in common use cases:

- Routing concepts on page 295
- Policy routes on page 306
- Equal cost multi-path on page 308
- Dual internet connections on page 312

The following topics include additional information about static routes:

- Deploying the Security Fabric on page 2288
- Security Fabric over IPsec VPN on page 2308
- Adding a static route on page 489
- IPsec VPN in an HA environment on page 1466
- IPsec VPN to Azure with virtual network gateway on page 1392
- FortiGate as dialup client on page 1412

- ADVPN with BGP as the routing protocol on page 1534
- ADVPN with OSPF as the routing protocol on page 1543
- ADVPN with RIP as the routing protocol on page 1552
- Basic site-to-site VPN with pre-shared key on page 1358
- Site-to-site VPN with digital certificate on page 1363
- Site-to-site VPN with overlapping subnets on page 1370
- Tunneled Internet browsing on page 1440
- Multiple concurrent SDN connectors on page 2482
- Packet distribution and redundancy for aggregate IPsec tunnels on page 1472
- Use MAC addresses in SD-WAN rules and policy routes on page 563
- Using BGP tags with SD-WAN rules on page 608

Routing concepts

This section contains the following topics:

Default route

The default route has a destination of `0.0.0.0/0.0.0.0`, representing the least specific route in the routing table. It is a catch all route in the routing table when traffic cannot match a more specific route. Typically this is configured with a static route with an administrative distance of `10`. In most instances, you will configure the next hop interface and the gateway address pointing to your next hop. If your FortiGate is sitting at the edge of the network, your next hop will be your ISP gateway. This provides internet access for your network.

Sometimes the default route is configured through DHCP. On some desktop models, the WAN interface is preconfigured in DHCP mode. Once the WAN interface is plugged into the network modem, it will receive an IP address, default gateway, and DNS server. FortiGate will add this default route to the routing table with a distance of `5`, by default. This will take precedence over any default static route with a distance of `10`. Therefore, take caution when you are configuring an interface in DHCP mode, where *Retrieve default gateway from server* is enabled. You may disable it and/or change the distance from the *Network* > *Interfaces* page when you edit an interface.

Adding or editing a static route

To add a static route using the GUI:

1. Go to *Network > Static Routes* and click *Create New*.
2. Enter the following information:

Dynamic Gateway	When enabled, a selected DHCP/PPPoE interface will automatically retrieve its dynamic gateway.
Destination	• Subnet Enter the destination IP address and netmask. A value of `0.0.0.0/0.0.0.0` creates a default route. • Named Address Select an address or address group object. Only addresses with static route configuration enabled will appear on the list. This means a geography type address cannot be used. • Internet Service Select an Internet Service. These are known IP addresses of popular services across the Internet.
Interface	Select the name of the interface that the static route will connect through.
Gateway Address	Enter the gateway IP address. When selecting an IPsec VPN interface or SD-WAN creating a blackhole route, the gateway cannot be specified.
Administrative Distance	Enter the distance value, which will affect which routes are selected first by different protocols for route management or load balancing. The default is 10.
Advanced Options	Optionally, expand *Advanced Options* and enter a *Priority*. When two routes have an equal distance, the route with a lower priority number will take precedence. The default is 1.

3. Click *OK*.

Configuring FQDNs as a destination address in static routes

You can configure FQDN firewall addresses as destination addresses in a static route, using either the GUI or the CLI.

In the GUI, to add an FQDN firewall address to a static route in the firewall address configuration, enable the *Static Route Configuration* option. Then, when you configure the static route, set *Destination* to *Named Address*.

To configure an FQDN as a destination address in a static route using the CLI:

```
config firewall address
    edit 'Fortinet-Documentation-Website'
        set type fqdn
        set fqdn docs.fortinet.com
        set allow-routing enable
    next
end
```

```
config router static
    edit 0
        set dstaddr Fortinet-Documentation-Website
        ...
    next
end
```

Routing table

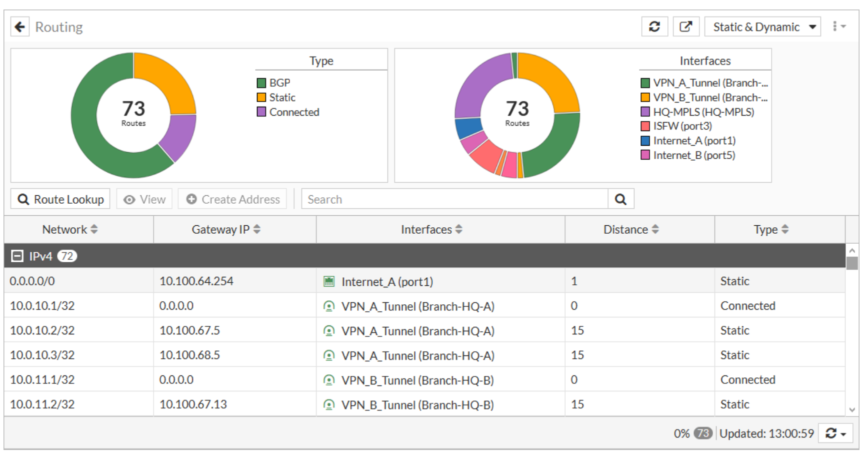
A routing table consists of only the best routes learned from the different routing protocols. The most specific route always takes precedence. If there is a tie, then the route with a lower administrative distance will be injected into the routing table. If administrative distances are also equal, then all the routes are injected into the routing table, and *Cost* and *Priority* become the deciding factors on which a route is preferred. If these are also equal, then FortiGate will use Equal cost multi-path on page 308 to distribute traffic between these routes.

Viewing the routing table in the GUI

You can view routing tables in the FortiGate GUI under *Dashboard > Network > Static & Dynamic Routing* by default. Expand the widget to see the full page. Additionally, if you want to convert the widget into a dashboard, click on the *Save as Monitor* icon on the top right of the page.

You can also monitor policy routes by toggling from *Static & Dynamic* to *Policy* on the top right corner of the page. The active policy routes include policy routes that you created, SD-WAN rules, and Internet Service static routes. It also supports downstream devices in the Security Fabric.

The following figure show an example of the static and dynamic routes in the Routing Monitor:



To view more columns, right-click on the column header to select the columns to be displayed:

Field	Description
IP Version	Shows whether the route is IPv4 or IPv6.
Network	The IP addresses and network masks of destination networks that the FortiGate can reach.
Gateway IP	The IP addresses of gateways to the destination networks.
Interfaces	The interface through which packets are forwarded to the gateway of the destination network.

Field	Description
Distance	The administrative distance associated with the route. A lower value means the route is preferable compared to other routes to the same destination.
Type	The type values assigned to FortiGate routes (Static, Connected, RIP, OSPF, or BGP): • *Connected*: All routes associated with direct connections to FortiGate interfaces • *Static*: The static routes that have been added to the routing table manually • *RIP*: All routes learned through RIP • *RIPNG*: All routes learned through RIP version 6 (which enables the sharing of routes through IPv6 networks) • *BGP*: All routes learned through BGP • *OSPF*: All routes learned through OSPF • *OSPF6*: All routes learned through OSPF version 6 (which enables the sharing of routes through IPv6 networks) • *IS-IS*: All routes learned through IS-IS • *HA*: RIP, OSPF, and BGP routes synchronized between the primary unit and the subordinate units of a high availability (HA) cluster. HA routes are maintained on subordinate units and are visible only if you're viewing the router monitor from a virtual domain that is configured as a subordinate virtual domain in a virtual cluster.
Metric	The metric associated with the route type. The metric of a route influences how the FortiGate dynamically adds it to the routing table. The following are types of metrics and the protocols they are applied to: • *Hop count*: Routes learned through RIP • *Relative cost*: Routes learned through OSPF • *Multi-Exit Discriminator (MED)*: Routes learned through BGP. By default, the MED value associated with a BGP route is zero. However, the MED value can be modified dynamically. If the value was changed from the default, the Metric column displays a non-zero value.
Priority	In static routes, priorities are 0 by default. When two routes have an equal distance, the route with the lower priority number will take precedence.
VRF	Virtual routing and forwarding (VRF) allows multiple routing table instances to co-exist. VRF can be assigned to an Interface. Packets are only forwarded between interfaces with the same VRF.
Up Since	The total accumulated amount of time that a route learned through RIP, OSPF, or BGP has been reachable.

Viewing the routing table in the CLI

Viewing the routing table using the CLI displays the same routes as you would see in the GUI.

If VDOMs are enabled on the FortiGate, all routing-related CLI commands must be run within a VDOM and not in the global context.

To view the routing table using the CLI:

```
# get router info routing-table all
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
```

```
       O - OSPF, IA - OSPF inter area

N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2

E1 - OSPF external type 1, E2 - OSPF external type 2
 i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
 * - candidate default
Routing table for VRF=0
S*      0.0.0.0/0 [1/0] via 172.31.0.1, MPLS [1/0]via 192.168.2.1, port1 [1/0] via
192.168.122.1, port2
S       1.2.3.4/32 [10/0] via 172.16.100.81, VLAN100
C       10.10.2.0/24 is directly connected, hub
C       10.10.2.1/32 is directly connected, hub
O       10.10.10.0/24 [110/101] via 192.168.2.1, port1, 01:54:18
C       10.253.240.0/20 is directly connected, wqt.root
S       110.2.2.122/32 [22/0] via 2.2.2.2, port2, [3/3]
C       172.16.50.0/24 is directly connected, WAN1-VLAN50
C       172.16.60.0/24 is directly connected, WAN2-VLAN60
C       172.16.100.0/24 is directly connected, VLAN100
C       172.31.0.0/30 is directly connected, MPLS
C       172.31.0.2/32 is directly connected, MPLS
B       192.168.0.0/24 [20/0] via 172.31.0.1, MPLS, 00:31:43
C       192.168.2.0/24 is directly connected, port1
C       192.168.20.0/24 is directly connected, port3
C       192.168.99.0/24 is directly connected, Port1-VLAN99
C       192.168.122.0/24 is directly connected, port2
Routing table for VRF=10
C       172.16.101.0/24 is directly connected, VLAN101
```

Examining an entry:

```
B       192.168.0.0/24 [20/0] via 172.31.0.1, MPLS, 00:31:43
```

Value	Description
`B`	BGP. The routing protocol used.
`192.168.0.0/24`	The destination of this route, including netmask.
`[20/0]`	`20` indicates an administrative distance of `20` out of a range of `0` to `255`. `0` is an additional metric associated with this route, such as in OSPF.
`172.31.0.1`	The gateway or next hop.
`MPLS`	The interface that the route uses.
`00:31:43`	The age of the route in `HH:MM:SS`.

Viewing the routing database

The routing database consists of all learned routes from all routing protocols before they are injected into the routing table. This likely lists more routes than the routing table as it consists of routes to the same destinations with different distances. Only the best routes are injected into the routing table. However, it is useful to see all learned routes for troubleshooting purposes.

To view the routing database using the CLI:

```
# get router info routing-table database
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area

N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
     E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       > - selected route, * - FIB route, p - stale info
Routing table for VRF=0
S    *> 0.0.0.0/0 [1/0] via 172.31.0.1, MPLS
     *>           [1/0] via 192.168.2.1, port1
     *>           [1/0] via 192.168.122.1, port2
S    *> 1.2.3.4/32 [10/0] via 172.16.100.81, VLAN100
C    *> 10.10.2.0/24 is directly connected, hub
C    *> 10.10.2.1/32 is directly connected, hub
O    *> 10.10.10.0/24 [110/101] via 192.168.2.1, port1, 02:10:17
C    *> 10.253.240.0/20 is directly connected, wqt.root
S    *> 110.2.2.122/32 [22/0] via 2.2.2.2, port2, [3/3]
C    *> 172.16.50.0/24 is directly connected, WAN1-VLAN50
C    *> 172.16.60.0/24 is directly connected, WAN2-VLAN60
C    *> 172.16.100.0/24 is directly connected, VLAN100
O       172.31.0.0/30 [110/201] via 192.168.2.1, port1, 00:47:36
C    *> 172.31.0.0/30 is directly connected, MPLS
```

Selected routes are marked by the > symbol. In the above example, the OSPF route to destination `172.31.0.0/30` is not selected.

Kernel routing table

The kernel routing table makes up the actual Forwarding Information Base (FIB) that used to make forwarding decisions for each packet. The routes here are often referred to as kernel routes. Parts of this table are derived from the routing table that is generated by the routing daemon.

To view the kernel routing table using the CLI:

```
# get router info kernel
tab=254 vf=0 scope=0 type=1 proto=11 prio=0 0.0.0.0/0.0.0.0/0->0.0.0.0/0 pref=0.0.0.0
        gwy=172.31.0.1 flag=04 hops=0 oif=31(MPLS)
        gwy=192.168.2.1 flag=04 hops=0 oif=3(port1)
        gwy=192.168.122.1 flag=04 hops=0 oif=4(port2)
tab=254 vf=0 scope=0 type=1 proto=17 prio=0 192.168.122.98/255.255.255.255/0->1.1.1.1/32
pref=0.0.0.0 gwy=192.168.122.1 dev=4(port2)
tab=254 vf=0 scope=0 type=1 proto=17 prio=0 172.31.0.2/255.255.255.255/0->1.1.1.1/32
pref=0.0.0.0 gwy=172.31.0.1 dev=31(MPLS)
tab=254 vf=0 scope=0 type=1 proto=17 prio=0 192.168.2.5/255.255.255.255/0->1.1.1.1/32
pref=0.0.0.0 gwy=192.168.2.1 dev=3(port1)
tab=254 vf=0 scope=0 type=1 proto=11 prio=0 0.0.0.0/0.0.0.0/0->1.2.3.4/32 pref=0.0.0.0
gwy=172.16.100.81 dev=20(VLAN100)
tab=254 vf=0 scope=0 type=1 proto=17 prio=0 192.168.122.98/255.255.255.255/0->8.8.8.8/32
pref=0.0.0.0 gwy=192.168.122.1 dev=4(port2)
```

The kernel routing table entries are:

Value	Description
`tab`	Table number: It will either be `254` (unicast) or `255` (multicast).
`vf`	Virtual domain of the firewall: It is the VDOM index number. If VDOMs are not enabled, this number is `0`.
`type`	Type of routing connection. Valid values include: • `0` - unspecific • `1` - unicast • `2` - local • `3` - broadcast • `4` - anycast • `5` - multicast • `6` - blackhole • `7` - unreachable • `8` - prohibited
`proto`	Type of installation that indicates where the route came from. Valid values include: • `0` - unspecific • `2` - kernel • `11` - ZebOS routing module • `14` - FortiOS • `15` - HA • `16` - authentication based • `17` - HA1
`prio`	Priority of the route. Lower priorities are preferred.
`->0.0.0.0/0` `(->x.x.x.x/mask)`	The IP address and subnet mask of the destination.
`pref`	Preferred next hop along this route.
`gwy`	Gateway: The address of the gateway this route will use.
`dev`	Outgoing interface index: This number is associated with the interface for this route. If VDOMs are enabled, the VDOM is also included here. If an interface alias is set for this interface, it is also displayed here.

Route look-up

Route look-up typically occurs twice in the life of a session. Once when the first packet is sent by the originator and once more when the first reply packet is sent from the responder. When a route look-up occurs, the routing information is written to the session table. If routing changes occur during the life of a session, additional routing look-ups may occur.

FortiGate performs a route look-up in the following order:

1. Policy-based routes: If a match occurs and the action is to forward, traffic is forwarded based on the policy route.
2. Forwarding Information Base, otherwise known as the kernel routing table.
3. If no match occurs, the packet is dropped.

Searching the routing table

When there are many routes in your routing table, you can perform a quick search by using the search bar to specify your criteria, or apply filters on the column header to display only certain routes. For example, if you want to only display static routes, you may use "static" as the search term, or filter by the *Type* field with value *Static*.

Route look-up on the other hand provides a utility for you to enter criteria such as *Destination*, *Destination Port*, *Source*, *Protocol* and/or *Source Interface*, in order to determine the route that a packet will take. Once you click *Search*, the corresponding route will be highlighted.

You can also use the CLI for a route look-up. The CLI provides a basic route look-up tool.

To look-up a route in the CLI:

```
# get router info routing-table details 4.4.4.4
Routing table for VRF=0
Routing entry for 0.0.0.0/0
      Known via "static", distance 1, metric 0, best
      * 172.31.0.1, via MPLS distance 0
      * 192.168.2.1, via port1 distance 0
      * 192.168.122.1, via port2 distance 0
```

Blackhole routes

Sometimes upon routing table changes, it is not desirable for traffic to be routed to a different gateway. For example, you may have traffic destined for a remote office routed through your IPsec VPN interface. When the VPN is down, traffic will try to re-route to another interface. However, this may not be viable and traffic will instead be routed to your default route through your WAN, which is not desirable. Traffic may also be routed to another VPN, which you do not want. For such scenarios, it is good to define a blackhole route so that traffic is dropped when your desired route is down. Upon reconnection, your desired route is once again added to the routing table and your traffic will resume routing to your desired interface. For this reason, blackhole routes are created when you configure an IPsec VPN using the IPsec wizard.

To create a blackhole route in the GUI:

1. Go to *Network > Static Routes*.
2. Click *Create New*. The *New Static Route* screen appears.
3. Specify a *Destination* type.
4. Select *Blackhole* from the *Interface* field.
5. Type the desired *Administrative Distance*.
6. Click *OK*.

Route priority for a *Blackhole* route can only be configured from the CLI.

Reverse path look-up

Whenever a packet arrives at one of the interfaces on a FortiGate, the FortiGate determines whether the packet was received on a legitimate interface by doing a reverse look-up using the source IP address in the packet header. This protects against IP spoofing attacks. If the FortiGate does not have a route to the source IP address through the interface on which the packet was received, the FortiGate drops the packet as per Reverse Path Forwarding (RPF) check. There are two modes of RPF – feasible path and strict. The default feasible RPF mode checks only for the existence of at least one active route back to the source using the incoming interface. The strict RPF check ensures the best route back to the source is used as the incoming interface.

To configure a strict Reverse Path Forwarding check in the CLI:

```
config system settings
    set strict-src-check enable
end
```

You can remove RPF state checks without needing to enable asymmetric routing by disabling state checks for traffic received on specific interfaces. Disabling state checks makes a FortiGate less secure and should only be done with caution for troubleshooting purposes.

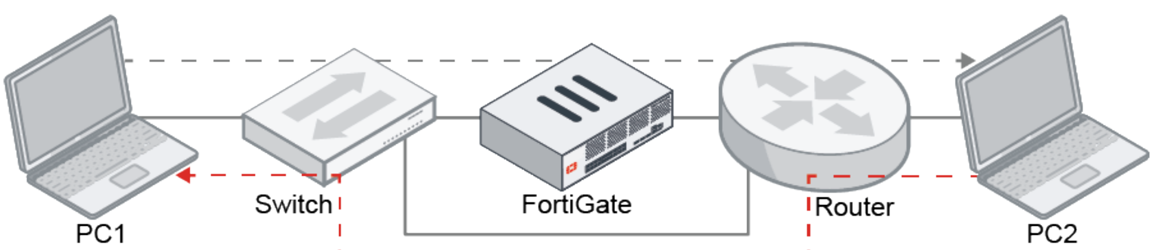
To remove Reverse Path Forwarding checks from the state evaluation process in the CLI:

```
config system interface
    edit <interface_name>
        set src-check disable
    next
end
```

Asymmetric routing

Asymmetric routing occurs when request and response packets follow different paths that do not cross the same firewall.

In the following topology, traffic between PC1 and PC2 takes two different paths.



Traffic from PC1 to PC2 goes through the FortiGate, while traffic from PC2 to PC1 does not.

In TCP, if the packets in the request and response directions follow different paths, the FortiGate will block the packets, since the TCP three-way handshake is not established through the FortiGate.

Scenario 1: PC1 starts a TCP connection with PC2

1. The TCP SYN is allowed by the FortiGate.
2. The TCP SYN/ACK bypasses the FortiGate.

3. The TCP ACK is blocked by the FortiGate.
4. Subsequent TCP packets are blocked by the FortiGate.

Scenario 2: PC2 starts a TCP connection with PC1

1. The TCP SYN bypasses the FortiGate.
2. The TCP SYN/ACK is blocked by the FortiGate.
3. Subsequent TCP packets are blocked by the FortiGate.

In ICMP, consider the following scenarios.

Scenario 1: PC1 pings PC2

1. The ICMP request passes through the FortiGate. A session is created.
2. The ICMP reply bypasses the FortiGate, but reaches PC1. The ping is successful.
3. The ICMP request passes through the FortiGate, and it matches the previous session.
4. The ICMP reply bypasses the FortiGate, but it reaches PC1. The ping is successful.
5. Subsequent ICMP requests are allowed by the FortiGate.

Scenario 2: PC2 pings PC1

1. The ICMP request bypasses the FortiGate, but it reaches PC1.
2. The ICMP reply passes through the FortiGate. No session is matched, and the packet is dropped.
3. Subsequent ICMP replies are blocked by the FortiGate.

If an ICMP request does not pass through the FortiGate, but the response passes through the FortiGate, then by default it blocks the packet as invalid.

Permitting asymmetric routing

If required, the FortiGate can be configured to permit asymmetric routing.

To permit asymmetric routing:

```
config system settings
    set asymroute enable
end
```

This setting should be used only when the asymmetric routing issue cannot be resolved by ensuring both directions of traffic pass through the FortiGate.

When asymmetric routing is enabled and occurs, the FortiGate cannot inspect all traffic. Potentially malicious traffic may pass through and compromise the security of the network.

Asymmetric routing behaves as follows when it is permitted by the FortiGate:

TCP packets

Scenario 1: PC1 starts a TCP connection with PC2

1. The TCP SYN is allowed by the FortiGate. The FortiGate creates a session, checks the firewall policies, and applies the configuration from the matching policy (UTM inspection, NAT, traffic shaping, and so on).

2. The TCP SYN/ACK bypasses the FortiGate.
3. The TCP ACK is allowed by the FortiGate. The packet matches the previously created session.
4. Subsequent TCP packets are allowed by the FortiGate. The packets in the session can also be offloaded where applicable.

Scenario 2: PC2 starts a TCP connection with PC1

1. The TCP SYN bypasses the FortiGate.
2. The TCP SYN/ACK is allowed by the FortiGate. No session is matched. The packet passes to the CPU and is forwarded based on the routing table.
3. The TCP ACK bypasses the FortiGate.
4. Subsequent TCP packets are allowed by the FortiGate. The FortiGate acts as a router that only makes routing decisions. No security inspection is performed.

ICMP packets

Scenario 1: PC1 pings PC2

1. There is no difference from when asymmetric routing is disabled.

Scenario 2: PC2 pings PC1

1. The ICMP request bypasses the FortiGate, but it reaches PC1.
2. The ICMP reply passes through the FortiGate. No session is matched. The packet passes to the CPU and is forwarded based on the routing table.
3. Subsequent ICMP replies are allowed by the FortiGate. The FortiGate acts as a router that only makes routing decisions. No security inspection is performed.

UDP packets

Asymmetric routing does not affect UDP packets. UDP packets are checked by the session table regardless of asymmetric routing. A policy is required to allow UDP.

Routing changes

When routing changes occur, routing look-up may occur on an existing session depending on certain configurations.

Routing changes without SNAT

When a routing change occurs, FortiGate flushes all routing information from the session table and performs new routing look-up for all new packets on arrival by default. You can modify the default behavior using the following commands:

```
config system interface
    edit <interface>
        set preserve-session-route enable
    next
end
```

By enabling `preserve-session-route`, the FortiGate marks existing session routing information as persistent. Therefore, routing look-up only occurs on new sessions.

Routing changes with SNAT

When SNAT is enabled, the default behavior is opposite to that of when SNAT is not enabled. After a routing change occurs, sessions with SNAT keep using the same outbound interface as long as the old route is still active. This may be the case if the priority of the static route was changed. You can modify this default behavior using the following commands:

```
config system global
    set snat-route-change enable
end
```

By enabling `snat-route-change`, sessions with SNAT will require new route look-up when a routing change occurs. This will apply a new SNAT to the session.

Policy routes

Policy routing allows you to specify an interface to route traffic. This is useful when you need to route certain types of network traffic differently than you would if you were using the routing table. You can use the incoming traffic's protocol, source or destination address, source interface, or port number to determine where to send the traffic.

When a packet arrives, the FortiGate starts at the top of the policy route list and attempts to match the packet with a policy. For a match to be found, the policy must contain enough information to route the packet. At a minimum, this requires the outgoing interface to forward the traffic, and the gateway to route the traffic to. If one or both of these are not specified in the policy route, then the FortiGate searches the routing table to find the best active route that corresponds to the policy route. If no routes are found in the routing table, then the policy route does not match the packet. The FortiGate continues down the policy route list until it reaches the end. If no matches are found, then the FortiGate does a route lookup using the routing table.

Policy routes are sometimes referred to as Policy-based routes (PBR).

Configuring a policy route

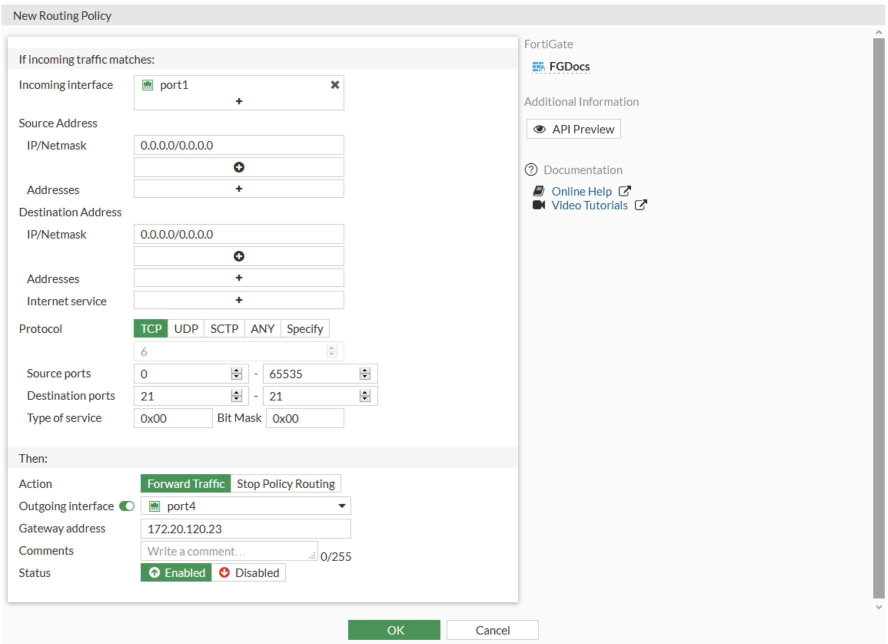
In this example, a policy route is configured to send all FTP traffic received at port1 out through port4 and to a next hop router at 172.20.120.23. To route FTP traffic, the protocol is set to TCP (6) and the destination ports are set to 21 (the FTP port).

To configure a policy route in the GUI:

1. Go to *Network > Policy Routes*.
2. Click *Create New > Policy Route*.
3. Configure the following fields:

Incoming interface	port1
Source Address	0.0.0.0/0.0.0.0
Destination Address	0.0.0.0/0.0.0.0

Protocol	TCP
Destination ports	21 - 21
Type of service	0x00
Bit Mask	0x00
Outgoing interface	Enable and select port4
Gateway address	172.20.120.23



4. Click *OK*.

To configure a policy route in the CLI:

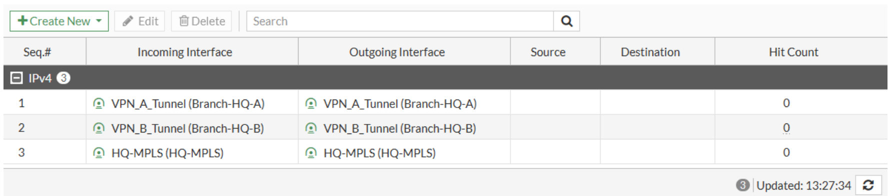
```
config router policy
    edit 1
        set input-device "port1"
        set src "0.0.0.0/0.0.0.0"
        set dst "0.0.0.0/0.0.0.0"
        set protocol 6
        set start-port 21
        set end-port 21
        set gateway 172.20.120.23
        set output-device "port4"
        set tos 0x00
        set tos-mask 0x00
    next
end
```

Moving a policy route

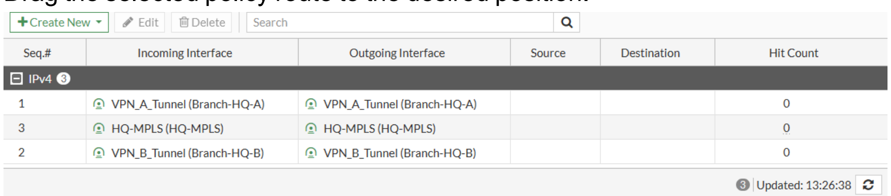
A routing policy is added to the bottom of the table when it is created. Routing policies can be moved to a different location in the table to change the order of preference. In this example, routing policy 3 will be moved before routing policy 2.

To move a policy route in the GUI:

1. Go to *Network > Policy Routes*.
2. In the table, select the policy route.



3. Drag the selected policy route to the desired position.



To move a policy route in the CLI:

```
config router policy
    move 3 after 1
end
```

Equal cost multi-path

Equal cost multi-path (ECMP) is a mechanism that allows a FortiGate to load-balance routed traffic over multiple gateways. Just like routes in a routing table, ECMP is considered after policy routing, so any matching policy routes will take precedence over ECMP.

ECMP pre-requisites are as follows:

- Routes must have the same destination and costs. In the case of static routes, costs include distance and priority
- Routes are sourced from the same routing protocol. Supported protocols include static routing, OSPF, and BGP

ECMP and SD-WAN implicit rule

ECMP and SD-WAN implicit rule are essentially similar in the sense that an SD-WAN implicit rule is processed after SD-WAN service rules are processed. See Implicit rule on page 546 to learn more.

The following table summarizes the different load-balancing algorithms supported by each:

ECMP	SD-WAN		Description
	GUI	CLI	
`source-ip-based`	Source IP	`source-ip-based`	Traffic is divided equally between the interfaces. Sessions that start at the same source IP address use the same path. This is the default selection.
`weight-based`	Sessions	`weight-based`	The workload is distributed based on the number of sessions that are connected through the interface. The weight that you assign to each interface is used to calculate the percentage of the total sessions allowed to connect through an interface, and the sessions are distributed to the interfaces accordingly.
`usage-based`	Spillover	`usage-based`	The interface is used until the traffic bandwidth exceeds the ingress and egress thresholds that you set for that interface. Additional traffic is then sent through the next interface member.
`source-dest-ip-based`	Source-Destination IP	`source-dest-ip-based`	Traffic is divided equally between the interfaces. Sessions that start at the same source IP address and go to the same destination IP address use the same path.
Not supported	Volume	`measured-volume-based`	This mode is supported in SD-WAN only. The workload is distributed based on the number of packets that are going through the interface.

To configure the ECMP algorithm from the CLI:

- At the VDOM level:

```
config system settings
    set v4-ecmp-mode {source-ip-based* | weight-based | usage-based | source-dest-ip-
based}
end
```

- If SD-WAN is enabled, the above option is not available and ECMP is configured under the SD-WAN settings:

```
config system sdwan
    set status enable
    set load-balance-mode {source-ip-based* | weight-based | usage-based | source-dest-
ip-based | measured-volume-based}
end
```

For ECMP in IPv6, the mode must also be configured under SD-WAN:

```
# diagnose sys vd list
system fib version=63
```

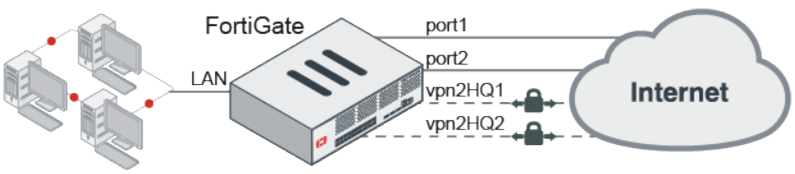
```
list virtual firewall info:
name=root/root index=0 enabled fib_ver=40 use=168 rt_num=46 asym_rt=0 sip_helper=0, sip_nat_
trace=1, mc_fwd=0, mc_ttl_nc=0, tpmc_sk_pl=0
ecmp=source-ip-based, ecmp6=source-ip-based asym_rt6=0 rt6_num=55 strict_src_check=0 dns_
log=1 ses_num=20 ses6_num=0 pkt_num=19154477
```

To change the number of paths allowed by ECMP:

```
config system settings
    set ecmp-max-paths <number of paths>
end
```

Setting `ecmp-max-paths` to the lowest value of `1` is equivalent to disabling ECMP.

ECMP configuration examples



The following examples demonstrate the behavior of ECMP in different scenarios:

- Example 1: Default ECMP on page 310
- Example 2: Same distance, different priority on page 311
- Example 3: Weight-based ECMP on page 311
- Example 4: Load-balancing BGP routes on page 312

Example 1: Default ECMP

```
config router static
    edit 1
        set gateway 172.16.151.1
        set device "port1"
    next
    edit 2
        set gateway 192.168.2.1
        set device "port2"
    next
end
```

```
# get router info routing-table all
Routing table for VRF=0
S*    0.0.0.0/0 [10/0] via 172.16.151.1, port1
                [10/0] via 192.168.2.1, port2
C    172.16.151.0/24 is directly connected, port1
C    192.168.2.0/24 is directly connected, port2
```

Result:

Both routes are added to the routing table and load-balanced based on the source IP.

Example 2: Same distance, different priority

```
config router static
    edit 1
        set gateway 172.16.151.1
        set priority 5
        set device "port1"
    next
    edit 2
        set gateway 192.168.2.1
        set device "port2"
    next
end
```

```
# get router info routing-table all
Routing table for VRF=0
S*    0.0.0.0/0 [10/0] via 192.168.2.1, port2
                [10/0] via 172.16.151.1, port1, [5/0]
C     172.16.151.0/24 is directly connected, port1
C     192.168.2.0/24 is directly connected, port2
```

Result:

Both routes are added to the routing table, but traffic is routed to `port2` which has a lower priority value with a default of `0`.

Example 3: Weight-based ECMP

```
config router static
    edit 3
        set dst 10.10.30.0 255.255.255.0
        set weight 80
        set device "vpn2HQ1"
    next
    edit 5
        set dst 10.10.30.0 255.255.255.0
        set weight 20
        set device "vpn2HQ2"
    next
end
```

```
# get router info routing-table all
Routing table for VRF=0
...
S     10.10.30.0/24 [10/0] is directly connected, vpn2HQ1, [0/80]
                    [10/0] is directly connected, vpn2HQ2, [0/20]
C     172.16.151.0/24 is directly connected, port1
C     192.168.0.0/24 is directly connected, port3
C     192.168.2.0/24 is directly connected, port2
```

Result:

Both routes are added to the routing table, but 80% of the sessions to `10.10.30.0/24` are routed to `vpn2HQ1`, and 20% are routed to `vpn2HQ2`.

Example 4: Load-balancing BGP routes

```
config router bgp
    set as 64511
    set router-id 192.168.2.86
    set ebgp-multipath enable
    config neighbor
        edit "192.168.2.84"
            set remote-as 64512
        next
        edit "192.168.2.87"
            set remote-as 64512
        next
   end
end
# get router info routing-table all
Routing table for VRF=0
...
C    172.16.151.0/24 is directly connected, port1
C    192.168.0.0/24 is directly connected, port3
C    192.168.2.0/24 is directly connected, port2
B    192.168.80.0/24 [20/0] via 192.168.2.84, port2, 00:00:33
                     [20/0] via 192.168.2.87, port2, 00:00:33
```

Result:

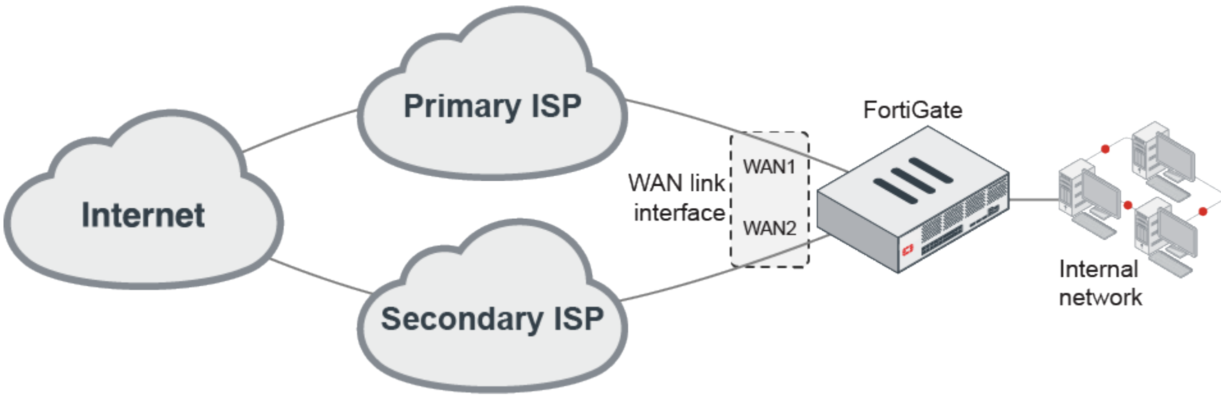
The network `192.168.80.0/24` is advertised by two BGP neighbors. Both routes are added to the routing table, and traffic is load-balanced based on Source IP.

For multiple BGP paths to be added to the routing table, you must enable `ebgp-multipath` for eBGP or `ibgp-multipath` for iBGP. These settings are disabled by default.

Dual internet connections

Dual internet connections, also referred to as dual WAN or redundant internet connections, refers to using two FortiGate interfaces to connect to the Internet. This is generally accomplished with SD-WAN, but this legacy solution provides the means to configure dual WAN without using SD-WAN. You can use dual internet connections in several ways:

- Link redundancy: If one interface goes down, the second interface automatically becomes the main connection.
- Load sharing: This ensures better throughput.
- Use a combination of link redundancy and load sharing.



This section describes the following dual internet connection scenarios:

- Scenario 1: Link redundancy and no load-sharing on page 313
- Scenario 2: Load-sharing and no link redundancy on page 315
- Scenario 3: Link redundancy and load-sharing on page 316

Scenario 1: Link redundancy and no load-sharing

Link redundancy ensures that if your Internet access is no longer available through a certain port, the FortiGate uses an alternate port to connect to the Internet.

In this scenario, two interfaces, WAN1 and WAN2, are connected to the Internet using two different ISPs. WAN1 is the primary connection. In the event of a failure of WAN1, WAN2 automatically becomes the connection to the Internet. For this configuration to function correctly, you must configure the following settings:

- Link health monitor on page 313: To determine when the primary interface (WAN1) is down and when the connection returns.
- Routing on page 314: Configure a default route for each interface.
- Security policies on page 315: Configure security policies to allow traffic through each interface to the internal network.

Link health monitor

Adding a link health monitor is required for routing failover traffic. A link health monitor confirms the device interface connectivity by probing a gateway or server at regular intervals to ensure it is online and working. When the server is not accessible, that interface is marked as down.

Set the `interval` (how often to send a ping) and `failtime` (how many lost pings are considered a failure). A smaller interval value and smaller number of lost pings results in faster detection, but creates more traffic on your network.

The link health monitor supports both IPv4 and IPv6, and various other protocols including ping, tcp-echo, udp-echo, http, and twamp.

To add a link health monitor (IPv4) using the CLI:

```
config system link-monitor
    edit <link-monitor-name>
        set addr-mode ipv4
        set srcintf <interface-name>
        set server <server-IP-address>
```

```
        set protocol {ping tcp-echo udp-echo http twamp}
        set gateway-ip <gateway-IP-address>
        set interval <seconds>
        set failtime <retry-attempts>
        set recoverytime <number-of-successful-responses>
        set status enable
    next
end
```

Option	Description
`set update-cascade-interface {enable	disable}`	This option is used in conjunction with fail-detect and fail-alert options in interface settings to cascade the link failure down to another interface. See the Bring other interfaces down when link monitor fails KB article for details.
`set update-static-route {enable	disable}`	When the link fails, all static routes associated with the interface will be removed.

Routing

You must configure a default route for each interface and indicate your preferred route as follows:

- Specify different distances for the two routes. The lower of the two distance values is declared active and placed in the routing table.

  Or

- Specify the same distance for the two routes, but give a higher priority to the route you prefer by defining a lower value. Both routes will be added to the routing table, but the route with a higher priority will be chosen as the best route

In the following example, we will use the first method to configure different distances for the two routes. You might not be able to connect to the backup WAN interface because the FortiGate does not route traffic out of the backup interface. The FortiGate performs a reverse path look-up to prevent spoofed traffic. If an entry cannot be found in the routing table that sends the return traffic out through the same interface, the incoming traffic is dropped.

To configure the routing of the two interfaces using the GUI:

1. Go to *Network > Static Routes*, and click *Create New*.
2. Enter the following information:

Destination	For an IPv4 route, enter a subnet of `0.0.0.0/0.0.0.0`. For an IPv6 route, enter a subnet of `::/0`.
Interface	Select the primary connection. For example, `wan1`.
Gateway Address	Enter the gateway address.
Administrative Distance	Leave as the default of `10`.

3. Click *OK*.
4. Repeat the above steps to set *Interface* to `wan2` and *Administrative Distance* to `20`.

To configure the routing of the two interfaces using the CLI:

```
config router {static | static6}
    edit 1
        set dst 0.0.0.0 0.0.0.0
        set device wan1
        set gateway <gateway_address>
        set distance 10
    next
    edit 2
        set dst 0.0.0.0 0.0.0.0
        set device wan2
        set gateway <gateway_address>
        set distance 20
    next
end
```

Security policies

When you create security policies, you need to configure duplicate policies to ensure that after traffic fails over WAN1, regular traffic is allowed to pass through WAN2, as it did with WAN1. This ensures that failover occurs with minimal effect to users.

Scenario 2: Load-sharing and no link redundancy

Load sharing may be accomplished in a few of the following ways of the many possible ways:

- By defining a preferred route with a lower distance, and specifying policy routes to route certain traffic to the secondary interface.
- By defining routes with same distance values but different priorities, and specifying policy routes to route certain traffic to the secondary interface.
- By defining routes with same distance values and priorities, and use equal-cost multi-path (ECMP) routing to equally distribute traffic between the WAN interfaces.

In our example, we will use the first option for our configuration. In this scenario, because link redundancy is not required, you do not have to configure a link monitor.

Traffic behaviour without a link monitor is as follows:

- If the remote gateway is down but the primary WAN interface of a FortiGate is still up, the FortiGate will continue to route traffic to the primary WAN. This results in traffic interruptions.
- If the primary WAN interface of a FortiGate is down due to physical link issues, the FortiGate will remove routes to it and the secondary WAN routes will become active. Traffic will failover to the secondary WAN.

Routing

Configure routing as you did in Scenario 1: Link redundancy and no load-sharing on page 313 above.

Policy routes

By configuring policy routes, you can redirect specific traffic to the secondary WAN interface. This works in this case because policy routes are checked before static routes. Therefore, even though the static route for the secondary WAN is not in the routing table, traffic can still be routed using the policy route.

In this example, we will create a policy route to route traffic from one address group to the secondary WAN interface.

To configure a policy route from the GUI:

1. Go to *Network > Policy Routes*, and click *Create New*.
2. Enter the following information:

Incoming interface	Define the source of the traffic. For example, `internal`.
Source Address	If we prefer to route traffic only from a group of addresses, define an address or address group, and add here.
Destination Address	Because we want to route all traffic from the address group here, we do not specify a destination address.
Protocol	Specify any protocol.
Action	Forward traffic.
Outgoing interface	Select the secondary WAN as the outbound interface. For example, `wan2`.
Gateway address	Input the gateway address for your secondary WAN. Because its default route has a higher distance value and is not added to the routing table, the gateway address must be added here.

3. Click OK.

To configure a policy route from the CLI:

```
config router policy
    edit 1
        set input-device "internal"
        set srcaddr "Laptops"
        set gateway <gateway_address>
        set output-device "wan2"
    next
end
```

Security policies

Your security policies should allow all traffic from `internal` to WAN1. Because link redundancy is not needed, you do not need to duplicate all WAN1 policies to WAN2. You will only need to define policies used in your policy route.

Scenario 3: Link redundancy and load-sharing

In this scenario, both the links are available to distribute Internet traffic with the primary WAN being preferred more. Should one of the interfaces fail, the FortiGate will continue to send traffic over the other active interface. The configuration is a combination of both the link redundancy and the load-sharing scenarios. The main difference is that the configured routes have equal distance values, with the route with a higher priority being preferred more. This ensures both routes are active in the routing table, but the route with a higher priority will be the best route.

Link health monitor

Link monitor must be configured for both the primary and the secondary WAN interfaces. This ensures that if the primary or the secondary WAN fails, the corresponding route is removed from the routing table and traffic re-routed to the other WAN interface.

For configuration details, see sample configurations in Scenario 1: Link redundancy and no load-sharing on page 313.

Routing

Both WAN interfaces must have default routes with the same distance. However, preference is given to the primary WAN by giving it a higher priority.

To configure the routing of the two interfaces using the CLI:

```
config router {static | static6}
    edit 1
        set dst 0.0.0.0 0.0.0.0
        set device wan1
        set gateway <gateway_address>
        set distance 10
        set priority 1
    next
    edit 2
        set dst 0.0.0.0 0.0.0.0
        set device wan2
        set gateway <gateway_address>
        set distance 10
        set priority 10
    next
end
```

Policy routes

The policy routes configuration is very similar to that of the policy routes in Scenario 2: Load-sharing and no link redundancy on page 315, except that the gateway address should not be specified. When a policy route is matched and the gateway address is not specified, the FortiGate looks at the routing table to obtain the gateway. In case the secondary WAN fails, traffic may hit the policy route. Because there is no gateway specified and the route to the secondary WAN is removed by the link monitor, the policy route will by bypassed and traffic will continue through the primary WAN. This ensures that the policy route is not active when the link is down.

Security policies

When you create security policies, you need to configure duplicate policies to ensure that after traffic fails over WAN1, regular traffic is allowed to pass through WAN2, as it was with WAN1. This ensures that failover occurs with minimal effect to users.

Dynamic routing

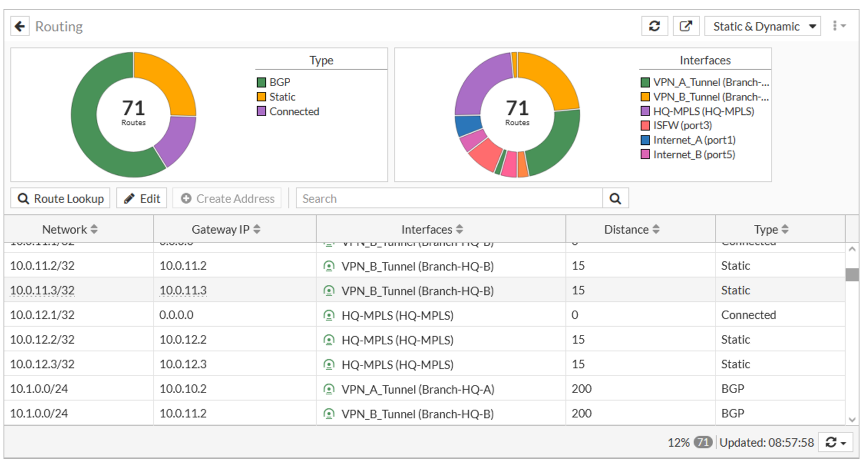
Dynamic routing protocols attempt to build a map of the network topology to identify the best routes to reach different destinations. Instead of manually defining static routes, which is not scalable, dynamic routing typically involves defining neighbors and peer routers that share their network topology and routing updates with each other. Protocols like distance vector, link state, and path vector are used by popular routing protocols. FortiGate supports RIP, OSPF, BGP, and IS-IS, which are interoperable with other vendors. When different dynamic routing protocols are used, the administrative distance of each protocol helps the FortiGate decide which route to pick.

Go to *System* > *Feature Visibility* and enable *Advanced Routing* to configure dynamic routing options in the GUI. See Feature visibility on page 2163 for more information.

This section includes:

- RIP on page 319
- OSPF on page 338
- BGP on page 355
- BFD on page 388

To view the routing table and perform route look-ups in the GUI, go to *Dashboard* > *Network* and expand the *Routing* widget.



To view the routing table in the CLI:

```
# get router info routing-table all
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default
Routing table for VRF=0
S*   0.0.0.0/0 [5/0] via 192.168.0.1, wan1
C    10.10.10.0/24 is directly connected, internal
C    169.254.2.1/32 is directly connected, Dialup-test
C    172.31.0.0/30 is directly connected, toKVM-MPLS
```

```
C     172.31.0.1/32 is directly connected, toKVM-MPLS
C     192.168.0.0/24 is directly connected, wan1
O     192.168.2.0/24 [110/101] via 10.10.10.11, internal, 00:00:26
S     192.168.20.0/24 [10/0] via 172.31.0.2, toKVM-MPLS
       [10/0] via 10.10.10.11, internal
```

RIP

Routing Information Protocol (RIP) is a distance-vector routing protocol that is intended for small and relatively homogeneous networks. It works well when there are minimal redundant paths and limited hop counts. FortiGate supports RIP version 1 (RFC 1058), RIP version 2 (RFC 2453), and RIPng (RFC 2080).

Basic configuration

To configure the FortiGate to participate in RIP using the most basic configurations in the GUI:

1. Go to *Network > RIP*.
2. Set the *Version*.
3. Add the networks that the FortiGate will advertise in and that will participate in RIP.
4. If the interface settings, such as passive interface, authentication, or enabling send/receive updates, must be edited, add the interfaces to the *Interface* table.
5. Click *Apply*.

To configure the FortiGate to participate in RIP using the most basic configurations in the CLI:

```
config router rip
    config network
        edit 1
            set prefix <subnet> <netmask>
        next
    end
    config interface
        edit <interface>
            set receive-version 2
            set send-version 2
        next
    end
end
```

Default route injection

Enabling *Inject default route* (`default-information-originate`) advertises a default route into the FortiGate's RIP network.

To enable/disable default route injection in the GUI:

1. Go to *Network > RIP*.
2. Expand the *Advanced Options*.

3. Enable/disable *Inject Default Route*.
4. Click *OK*.

To enable/disable default route injection in the CLI:

```
config router rip
    set default-information-originate {enable | disable}
end
```

Default metric

The default metric setting sets the default metric for all redistributed routes. If the default metric is set to five, and static routes are redistributed, then static routes have a metric of five. This value can be overridden by setting a specific metric value for a protocol. For example, the static route metric can be set to two, overriding the default metric.

```
config router rip
    set default-metric 5
    config redistribute "static"
        set status enable
        set metric 2
    end
end
```

The default metric is five, but redistributed static routes have a metric of two. So, the default metric is overridden and the metric for redistributed static routes is two.

Timers

RIP uses the update, timeout, and garbage timers to regulate its performance. The default timer settings are effective in most configurations. When customizing the settings, you must ensure that the new settings are compatible with your local routers and access servers.

Go to *Network > RIP* and expand the *Advanced Options* to configure the timers in the GUI, or use the CLI:

```
config router rip
    set timeout-timer <seconds>
    set update-timer <seconds>
    set garbage-timer <seconds>
end
```

Update timer

The update timer sets the interval between routing updates. The default value is 30 seconds. Randomness is added to help prevent network congestion due to multiple routers trying to update their neighbors simultaneously. The update timer must be at least three times shorter than the timeout timer.

If there is significant RIP traffic on the network, you can increase the update timer to send fewer updates. You must apply the same increase to all routers on the network to avoid timeouts that degrade your network speed.

Timeout timer

The timeout timer is the maximum amount of time that a reachable route is kept in the routing table since its last update. The default value is 180 seconds. If an update for the route is received before the timeout period elapses, then the timer is reset. The timeout timer should be at least three times longer than the update timer.

If routers are not responding to updates in time, increasing the timeout timer can help. A longer timeout timer results in longer update periods, and the FortiGate could wait a considerable amount of time for all of the timers to expire on an unresponsive route.

Garbage timer

The garbage timer is the amount of time that the FortiGate advertises a route as unreachable before deleting the route from the routing table. The default value is 120 seconds.

If the timer is short, older routes are removed from the routing table more quickly, resulting in a smaller routing table. This can be useful for large networks, or if the network changes frequently.

Authentication and key chain

RIP version 1 (RIPv1) has no authentication. RIP version 2 (RIPv2) uses text passwords or authentication keys to ensure that the routing information exchanged between routers is reliable. For authentication to work, both the sending and receiving routers must be set to use authentication and must be configured with the same password or keys. An authentication key that uses authentication key chains is more secure than a text password because the intervals when the key is valid can be configured.

A key chain is a list of one or more authentication keys that each have send and receive lifetimes. Keys are used to authenticate routing packets only during the keys specified lifetimes. The FortiGate migrates from one key to the next according to the scheduled lifetimes. The sending and receiving routers should have synchronized system dates and times to ensure that both ends are using the same keys at the same times. You can overlap the key lifetimes to make sure that a key is always available, even if there is some difference in the system times.

To configure a text password in the GUI:

1. Go to *Network > RIP*.
2. In the *Interfaces* table, click *Create New*, or edit an existing interface.
3. Enable *Authentication* and select *Text* or *MD5*.
4. Click *Change*, and enter the password.
5. Configure the remaining settings as needed.
6. Click *OK*.
7. Click *Apply*.

To configure a text password in the CLI:

```
config router rip
    config interface
        edit <interface>
            set auth-mode {text | md5}
            set auth-string **********
        next
    end
end
```

To configure a key chain with two sequentially valid keys and use it in a RIP interface:

```
config router key-chain
    edit rip_key
        config key
            edit 1
                set accept-lifetime 09:00:00 23 02 2020 09:00:00 17 03 2020
                set send-lifetime 09:00:00 23 02 2020 09:00:00 17 03 2020
                set key-string **********
            next
            edit 2
                set accept-lifetime 09:01:00 17 03 2020 09:00:00 1 04 2020
                set send-lifetime 09:01:00 17 03 2020 09:00:00 1 04 2020
                set key-string **********
            next
        end
    next
end
```

```
config router rip
    config interface
        edit port1
            set auth-keychain "rip_key"
        next
    end
end
```

Passive RIP interfaces

By default, an active RIP interface keeps the FortiGate routing table current by periodically asking neighbors for routes and sending out route updates. This can generate a significant amount of extra traffic in a large network.

A passive RIP interface listens to updates from other routers, but does not send out route updates. This can reduce network traffic when there are redundant routers in the network that would always send out essentially the same updates.

This example shows how to configure a passive RIPv2 interface on port1 using MD5 authentication.

To configure a passive RIP interface in the GUI:

1. Go to *Network > RIP*.
2. In the *Interfaces* table, click *Create New*.
3. Set *Interface* to the required interface.
4. Enable *Passive*.
5. Enable *Authentication* and set it to *MD5*.
6. Click *Change* and enter a password.
7. Set *Receive Version* to *2*.
8. Click *OK*.

To configure a passive RIP interface in the CLI:

```
config router rip
    set passive-interface "port1"
```

```
    config interface
        edit "port1"
            set auth-mode md5
            set auth-string **********
            set receive-version 2
            set send-version 2
        next
    end
end
```

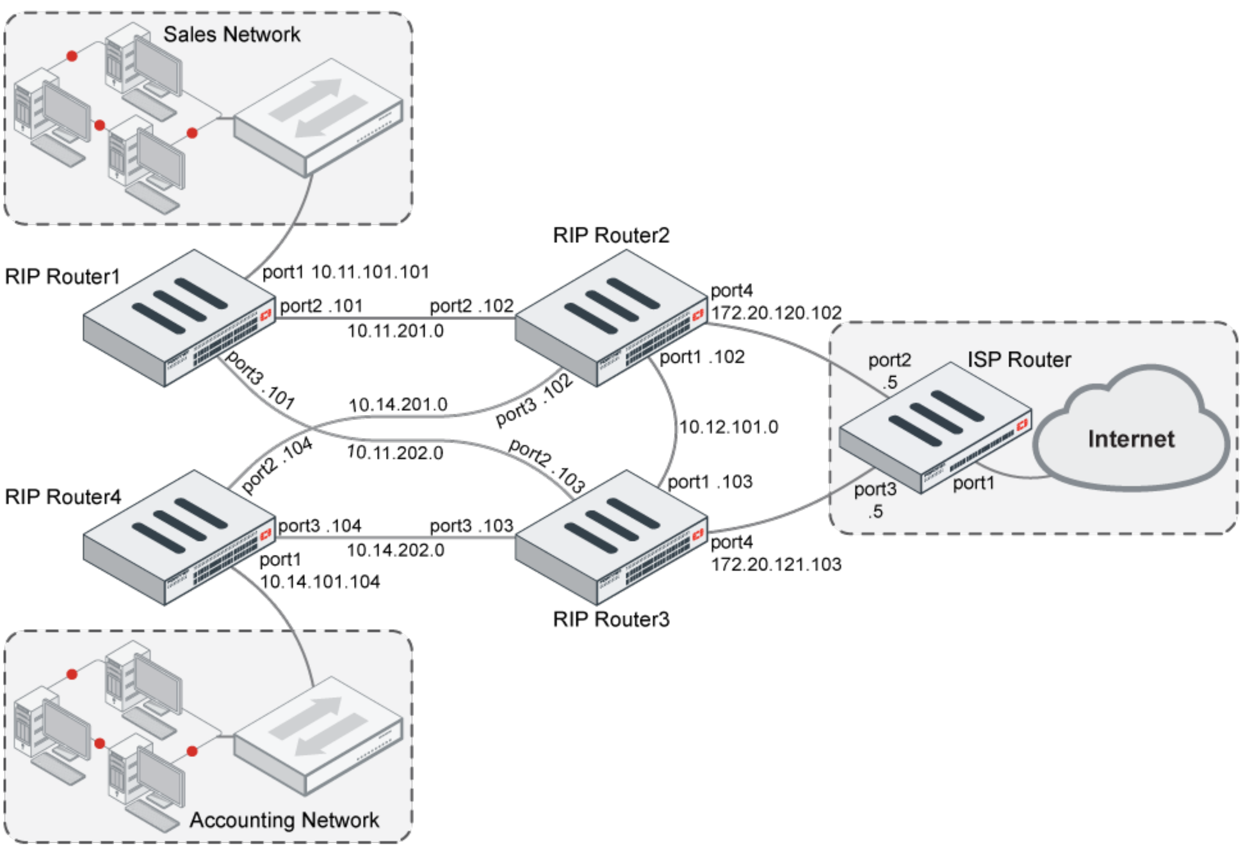
RIP and IPv6

RIP next generation (RIPng) is an extension of RIPv2 that includes support for IPv6. See Basic RIPng example on page 336 and IPv6 tunneling on page 465 for more information.

Basic RIP example

In this example, a medium-sized network is configured using RIPv2.

- Two core routers, RIP Router2 and RIP Router3, connect to the ISP router for two redundant paths to the internet.
- Two other routers, RIP Router1 and RIP Router4, connect to the two core routers and to different local networks.
- The ISP router is using RIP for its connections to the core routers, and redistributes its default route to the network - that is, default route injection is enabled.
- The ISP router uses NAT and has a static route to the internet. None of the other routers use NAT or static routes.



All of the FortiGate routers are configured as shown, using netmask 255.255.255.0. Firewall policies have been configured to allow the required traffic to flow across the interfaces.

Router	Interface	Interface name	IP address
Router1	port1	LoSales	10.11.101.101
	port2	vd12link0	10.11.201.101
	port3	vd13link0	10.11.202.101
Router2	port1	vd23link0	10.12.101.102
	port2	vd12link1	10.11.201.102
	port3	vd42link1	10.14.201.102
	port4	vdr2link1	172.20.120.102
Router3	port1	vd23link1	10.12.101.103
	port2	vd13link1	10.11.202.103
	port3	vd43link1	10.14.202.103
	port4	vdr3link1	172.20.121.103
Router4	port1	LoAccounting	10.14.101.104
	port2	vd42link0	10.14.201.104
	port3	vd43link0	10.14.202.104
ISP Router	port1	port1	To internet
	port2	vdr2link0	172.20.120.5
	port3	vdr3link0	172.20.121.5

After configuring each router, you can check the status of the connections by viewing the RIP database, RIP interfaces, and routing table. See Verifying the configuration on page 329.

After the network is configured, you can test it to ensure that when network events occur, such as a downed link, routing updates are triggered and converge as expected. See Testing the configuration and routing changes on page 333.

ISP router

To configure the ISP Router in the GUI:

1. Go to *Network > RIP*.
2. Set the *Version* to *2*.
3. Under *Networks*, add two networks:
   - *172.20.120.0/255.255.255.0*
   - *172.20.121.0/255.255.255.0*
4. Add the interfaces:

a. In the *Interfaces* table, click *Create New*.
b. Set *Interface* to *port2*.
c. Leave the remaining settings as their default values.
d. Click *OK*.
e. Repeat these steps for *port3*.

5. Under *Advanced Options*, enable *Inject Default Route*.
   This setting allows the ISP router to share its default 0.0.0.0 routes with other routers in the RIP network.
6. Click *Apply*.

To configure the ISP Router in the CLI:

```
config router rip
    set default-information-originate enable
    config network
        edit 1
            set prefix 172.20.121.0 255.255.255.0
        next
        edit 2
            set prefix 172.20.120.0 255.255.255.0
        next
    end
    config interface
        edit "port2"
            set receive-version 2
            set send-version 2
        next
        edit "port3"
            set receive-version 2
            set send-version 2
        next
    end
end
```

Router2 and Router3

Router2 and Router3 RIP configurations have different IP addresses, but are otherwise the same.

To configure Router2 and Router3 in the GUI:

1. Go to *Network > RIP*.
2. Set the *Version* to *2*.
3. Under *Networks*, add the IP addresses for each port:

Router2	10.12.101.0/255.255.255.0
	10.11.201.0/255.255.255.0
	10.14.201.0/255.255.255.0
	172.20.120.0/255.255.255.0

Router3	10.12.101.0/255.255.255.0
	10.11.202.0/255.255.255.0
	10.14.202.0/255.255.255.0
	172.20.121.0/255.255.255.0

4. Add the interfaces:
   a. In the *Interfaces* table, click *Create New*.
   b. Set *Interface* to *port1*.
   c. Leave the remaining settings as their default values.
   d. Click *OK*.
   e. Repeat these steps for *port2*, *port3*, and *port4*.
5. Click *Apply*.

To configure Router2 in the CLI:

```
config router rip
    config network
        edit 1
            set prefix 10.12.101.0 255.255.255.0
        next
        edit 2
            set prefix 10.11.201.0 255.255.255.0
        next
        edit 3
            set prefix 10.14.201.0 255.255.255.0
        next
        edit 4
            set prefix 172.20.120.0 255.255.255.0
        next
    end
    config interface
        edit "port1"
            set receive-version 2
            set send-version 2
        next
        edit "port2"
            set receive-version 2
            set send-version 2
        next
        edit "port3"
            set receive-version 2
            set send-version 2
        next
        edit "port4"
            set receive-version 2
            set send-version 2
        next
    end
end
```

To configure Router3 in the CLI:

```
config router rip
    config network
        edit 1
            set prefix 10.12.101.0 255.255.255.0
        next
        edit 2
            set prefix 10.11.202.0 255.255.255.0
        next
        edit 3
            set prefix 10.14.202.0 255.255.255.0
        next
        edit 4
            set prefix 172.20.121.0 255.255.255.0
        next
    end
    config interface
        edit "port1"
            set receive-version 2
            set send-version 2
        next
        edit "port2"
            set receive-version 2
            set send-version 2
        next
        edit "port3"
            set receive-version 2
            set send-version 2
        next
        edit "port4"
            set receive-version 2
            set send-version 2
        next
    end
end
```

Router1 and Router4

Router1 and Router4 RIP configurations have different IP addresses, but are otherwise the same.

To configure Router1 and Router4 in the GUI:

1. Go to *Network* > *RIP*.
2. Set the *Version* to *2*.
3. Under *Networks*, add the IP addresses for each port:

Router1	10.11.101.0/255.255.255.0
	10.11.201.0/255.255.255.0
	10.11.202.0/255.255.255.0

Router4	10.14.101.0/255.255.255.0
	10.14.201.0/255.255.255.0
	10.14.202.0/255.255.255.0

4. Add the interfaces:
   a. In the *Interfaces* table, click *Create New*.
   b. Set *Interface* to *port1*.
   c. For port1 only, enable *Passive*.
   d. Leave the remaining settings as their default values.
   e. Click *OK*.
   f. Repeat these steps for *port2* and *port3*, making sure that *Passive* is disabled.
5. Click *Apply*.

To configure Router1 in the CLI:

```
config router rip
    config network
        edit 1
            set prefix 10.11.101.0 255.255.255.0
        next
        edit 2
            set prefix 10.11.201.0 255.255.255.0
        next
        edit 3
            set prefix 10.11.202.0 255.255.255.0
        next
    end
    set passive-interface "port1"
    config interface
        edit "port1"
            set receive-version 2
            set send-version 2
        next
        edit "port2"
            set receive-version 2
            set send-version 2
        next
        edit "port3"
            set receive-version 2
            set send-version 2
        next
    end
end
```

To configure Router4 in the CLI:

```
config router rip
    config network
        edit 1
            set prefix 10.14.101.0 255.255.255.0
        next
```

```
        edit 2
            set prefix 10.14.201.0 255.255.255.0
        next
        edit 3
            set prefix 10.14.202.0 255.255.255.0
        next
    end
    set passive-interface "port1"
    config interface
        edit "port1"
            set receive-version 2
            set send-version 2
        next
        edit "port2"
            set receive-version 2
            set send-version 2
        next
        edit "port3"
            set receive-version 2
            set send-version 2
        next
    end
end
```

Verifying the configuration

The interface's names are shown in the debugs. The same commands should also be run on the other routers.

To verify the configuration after the ISP router, Router2, and Route3 have been configured:

This verification can be done after the ISP router, Router2, and Router3 have been configured. Only Router2's debugs are shown.

1. Check the RIP interface information:

```
# get router info rip interface
Router2 is up, line protocol is up
  RIP is not enabled on this interface
ssl.Router2 is up, line protocol is up
  RIP is not enabled on this interface
vdr2link1 is up, line protocol is up
  Routing Protocol: RIP
    Receive RIPv2 packets only
    Send RIPv2 packets only
    Passive interface: Disabled
    Split horizon: Enabled with Poisoned Reversed
    IP interface address:
      172.20.120.102/24
vd12link1 is up, line protocol is up
  Routing Protocol: RIP
    Receive RIPv2 packets only
    Send RIPv2 packets only
    Passive interface: Disabled
    Split horizon: Enabled with Poisoned Reversed
    IP interface address:
      10.11.201.102/24
```

```
vd42link1 is up, line protocol is up
  Routing Protocol: RIP
    Receive RIPv2 packets only
    Send RIPv2 packets only
    Passive interface: Disabled
    Split horizon: Enabled with Poisoned Reversed
    IP interface address:
      10.14.201.102/24
vd23link0 is up, line protocol is up
  Routing Protocol: RIP
    Receive RIPv2 packets only
    Send RIPv2 packets only
    Passive interface: Disabled
    Split horizon: Enabled with Poisoned Reversed
    IP interface address:
      10.12.101.102/24
```

RIP starts exchanging routes as soon as the networks are added to the Router2 and Router3 configurations because the RIP interfaces are active by default, and start sending and receiving RIP updates when a matching interface on the subnet is found. The interface configuration allows the interface settings to be fine tuned, in this case to specify only RIPv2 support.

2. Check the RIP database:

```
# get router info rip database
Codes: R - RIP, Rc - RIP connected, Rs - RIP static, K - Kernel,
       C - Connected, S - Static, O - OSPF, I - IS-IS, B - BGP
   Network            Next Hop         Metric From            If     Time
R  0.0.0.0/0          172.20.120.5          2 172.20.120.5    vdr2link1 02:55
Rc 10.11.201.0/24                           1                 vd12link1
R  10.11.202.0/24     10.12.101.103         2 10.12.101.103   vd23link0 02:33
Rc 10.12.101.0/24                           1                 vd23link0
Rc 10.14.201.0/24                           1                 vd42link1
R  10.14.202.0/24     10.12.101.103         2 10.12.101.103   vd23link0 02:33
Rc 172.20.120.0/24                          1                 vdr2link1
R  172.20.121.0/24    10.12.101.103         2 10.12.101.103   vd23link0 02:33
```

3. Check the routing table:

```
# get router info routing-table all
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default
Routing table for VRF=0
R*      0.0.0.0/0 [120/2] via 172.20.120.5, vdr2link1, 13:37:23
C       10.11.201.0/24 is directly connected, vd12link1
R       10.11.202.0/24 [120/2] via 10.12.101.103, vd23link0, 14:10:01
C       10.12.101.0/24 is directly connected, vd23link0
C       10.14.201.0/24 is directly connected, vd42link1
R       10.14.202.0/24 [120/2] via 10.12.101.103, vd23link0, 14:10:01
C       172.20.120.0/24 is directly connected, vdr2link1
R       172.20.121.0/24 [120/2] via 10.12.101.103, vd23link0, 13:20:36
```

Router2 has learned the default gateway from the ISP router, and has learned of other networks from Router3.

4. If firewall policies are correctly configured, the outside network can be reached:

```
# execute ping-options source 10.11.201.102
# execute ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8): 56 data bytes
64 bytes from 8.8.8.8: icmp_seq=0 ttl=115 time=4.5 ms
64 bytes from 8.8.8.8: icmp_seq=1 ttl=115 time=4.2 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=115 time=4.2 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=115 time=4.2 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=115 time=4.1 ms
--- 8.8.8.8 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 4.1/4.2/4.5 ms
```

```
# execute traceroute 8.8.8.8
traceroute to 8.8.8.8 (8.8.8.8), 32 hops max, 3 probe packets per hop, 84 byte packets
 1  172.20.120.5  0.101 ms  0.030 ms  0.014 ms
 2  172.16.151.1  0.169 ms  0.144 ms  0.131 ms
 3  * * *
```

To verify the configuration after Router1 and Router4 have also been configured:

This verification can be done after Router1 and Router4 have been configured. Only Router1's debugs are shown.

1. Check the RIP interface information:

```
# get router info rip interface
Router1 is up, line protocol is up
  RIP is not enabled on this interface
ssl.Router1 is up, line protocol is up
  RIP is not enabled on this interface
vd12link0 is up, line protocol is up
  Routing Protocol: RIP
    Receive RIPv2 packets only
    Send RIPv2 packets only
    Passive interface: Disabled
    Split horizon: Enabled with Poisoned Reversed
    IP interface address:
      10.11.201.101/24
vd13link0 is up, line protocol is up
  Routing Protocol: RIP
    Receive RIPv2 packets only
    Send RIPv2 packets only
    Passive interface: Disabled
    Split horizon: Enabled with Poisoned Reversed
    IP interface address:
      10.11.202.101/24
LoSales is up, line protocol is up
  Routing Protocol: RIP
    Receive RIPv2 packets only
    Send RIPv2 packets only
    Passive interface: Enabled
    Split horizon: Enabled with Poisoned Reversed
    IP interface address:
      10.11.101.101/24
      127.0.0.1/8
```

2. Check the RIP database:

```
# get router info rip database
Codes: R - RIP, Rc - RIP connected, Rs - RIP static, K - Kernel,
       C - Connected, S - Static, O - OSPF, I - IS-IS, B - BGP
   Network            Next Hop         Metric From            If     Time
R  0.0.0.0/0          10.11.202.103         3 10.11.202.103   vd13link0 02:35
Rc 10.11.101.0/24                           1                 LoSales
Rc 10.11.201.0/24                           1                 vd12link0
Rc 10.11.202.0/24                           1                 vd13link0
R  10.12.101.0/24     10.11.202.103         2 10.11.202.103   vd13link0 02:35
R  10.14.101.0/24     10.11.202.103         3 10.11.202.103   vd13link0 02:35
R  10.14.201.0/24     10.11.201.102         2 10.11.201.102   vd12link0 02:30
R  10.14.202.0/24     10.11.202.103         2 10.11.202.103   vd13link0 02:35
R  172.20.120.0/24    10.11.201.102         2 10.11.201.102   vd12link0 02:30
R  172.20.121.0/24    10.11.202.103         2 10.11.202.103   vd13link0 02:35
```

3. Check the routing table:

```
# get router info routing-table all
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default
Routing table for VRF=0
R*      0.0.0.0/0 [120/3] via 10.11.202.103, vd13link0, 00:09:42
C       10.11.101.0/24 is directly connected, LoSales
C       10.11.201.0/24 is directly connected, vd12link0
C       10.11.202.0/24 is directly connected, vd13link0
R       10.12.101.0/24 [120/2] via 10.11.202.103, vd13link0, 00:09:42
R       10.14.101.0/24 [120/3] via 10.11.202.103, vd13link0, 00:09:42
R       10.14.201.0/24 [120/2] via 10.11.201.102, vd12link0, 00:09:42
R       10.14.202.0/24 [120/2] via 10.11.202.103, vd13link0, 00:09:42
R       172.20.120.0/24 [120/2] via 10.11.201.102, vd12link0, 00:09:42
R       172.20.121.0/24 [120/2] via 10.11.202.103, vd13link0, 00:09:42
```

4. If firewall policies are correctly configured, the accounting network and the internet are reachable from the sales network:

```
# execute ping-options source 10.11.101.101
# execute ping 10.14.101.104
PING 10.14.101.104 (10.14.101.104): 56 data bytes
64 bytes from 10.14.101.104: icmp_seq=0 ttl=254 time=0.1 ms
64 bytes from 10.14.101.104: icmp_seq=1 ttl=254 time=0.0 ms
64 bytes from 10.14.101.104: icmp_seq=2 ttl=254 time=0.0 ms
64 bytes from 10.14.101.104: icmp_seq=3 ttl=254 time=0.0 ms
64 bytes from 10.14.101.104: icmp_seq=4 ttl=254 time=0.0 ms
--- 10.14.101.104 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 0.0/0.0/0.1 ms
```

```
# execute traceroute 10.14.101.104
traceroute to 10.14.101.104 (10.14.101.104), 32 hops max, 3 probe packets per hop, 84
byte packets
 1  10.11.202.103  0.079 ms  0.029 ms  0.013 ms
 2  10.14.101.104  0.043 ms  0.020 ms  0.010 ms
```

```
# execute ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8): 56 data bytes
64 bytes from 8.8.8.8: icmp_seq=0 ttl=114 time=4.3 ms
64 bytes from 8.8.8.8: icmp_seq=1 ttl=114 time=4.1 ms
--- 8.8.8.8 ping statistics ---
2 packets transmitted, 2 packets received, 0% packet loss
round-trip min/avg/max = 4.1/4.2/4.3 ms
```

```
# execute traceroute 8.8.8.8
traceroute to 8.8.8.8 (8.8.8.8), 32 hops max, 3 probe packets per hop, 84 byte packets
 1  10.11.202.103  0.094 ms  0.036 ms  0.030 ms
 2  172.20.121.5  0.216 ms  0.045 ms  0.038 ms
```

Testing the configuration and routing changes

After the network is configured, test it to ensure that when network events occur, such as a downed link, routing updates are triggered and converge as expected.

In the following examples, we disable certain links to simulate network outages, then verify that routing and connectivity is restored after the updates have converged.

Example 1 - ISP router port3 interface goes down

In this example, a link outage occurs on port3 of the ISP router. Consequently, all routers must use Router2, and not Router3, to reach the internet. Note the RIP database before and after the link failure, and the time taken for the route updates to propagate and return to a functioning state.

Router4's debugs are shown.

Before:

```
# get router info rip database
Codes: R - RIP, Rc - RIP connected, Rs - RIP static, K - Kernel,
       C - Connected, S - Static, O - OSPF, I - IS-IS, B - BGP
   Network            Next Hop         Metric From            If      Time
R  0.0.0.0/0          10.14.202.103         3 10.14.202.103   vd43link0 02:31
R  10.11.101.0/24     10.14.202.103         3 10.14.202.103   vd43link0 02:31
R  10.11.201.0/24     10.14.201.102         2 10.14.201.102   vd42link0 02:47
R  10.11.202.0/24     10.14.202.103         2 10.14.202.103   vd43link0 02:31
R  10.12.101.0/24     10.14.202.103         2 10.14.202.103   vd43link0 02:31
Rc 10.14.101.0/24                           1                 LoAccounting
Rc 10.14.201.0/24                           1                 vd42link0
Rc 10.14.202.0/24                           1                 vd43link0
R  172.20.120.0/24    10.14.201.102         2 10.14.201.102   vd42link0 02:47
R  172.20.121.0/24    10.14.202.103         2 10.14.202.103   vd43link0 02:31
```

```
# get router info routing-table all
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default
Routing table for VRF=0
R*      0.0.0.0/0 [120/3] via 10.14.202.103, vd43link0, 02:45:15
R       10.11.101.0/24 [120/3] via 10.14.202.103, vd43link0, 02:44:49
R       10.11.201.0/24 [120/2] via 10.14.201.102, vd42link0, 02:45:15
```

```
R       10.11.202.0/24 [120/2] via 10.14.202.103, vd43link0, 02:45:15
R       10.12.101.0/24 [120/2] via 10.14.202.103, vd43link0, 02:45:15
C       10.14.101.0/24 is directly connected, LoAccounting
C       10.14.201.0/24 is directly connected, vd42link0
C       10.14.202.0/24 is directly connected, vd43link0
R       172.20.120.0/24 [120/2] via 10.14.201.102, vd42link0, 02:45:15
R       172.20.121.0/24 [120/2] via 10.14.202.103, vd43link0, 02:45:15
```

```
# execute traceroute 8.8.8.8
traceroute to 8.8.8.8 (8.8.8.8), 32 hops max, 3 probe packets per hop, 84 byte packets
 1  10.14.202.103  0.187 ms  0.054 ms  0.030 ms
 2  172.20.121.5  0.117 ms  0.062 ms  0.040 ms
 3  * * *
```

After:

- You might see different routes, and the routes might change, while convergence is occurring. During convergence, the metric for your default route increases to 16.

```
# get router info rip database
Codes: R - RIP, Rc - RIP connected, Rs - RIP static, K - Kernel,
       C - Connected, S - Static, O - OSPF, I - IS-IS, B - BGP
   Network            Next Hop         Metric From            If     Time
R  0.0.0.0/0          10.14.202.103        16 10.14.202.103   vd43link0 01:50
```

- After convergence is complete, the RIP database will look similar to the following:

```
# get router info rip database
Codes: R - RIP, Rc - RIP connected, Rs - RIP static, K - Kernel,
       C - Connected, S - Static, O - OSPF, I - IS-IS, B - BGP
   Network            Next Hop         Metric From            If     Time
R  0.0.0.0/0          10.14.201.102         3 10.14.201.102   vd42link0 02:53
R  10.11.101.0/24     10.14.202.103         3 10.14.202.103   vd43link0 03:00
R  10.11.201.0/24     10.14.201.102         2 10.14.201.102   vd42link0 02:53
R  10.11.202.0/24     10.14.202.103         2 10.14.202.103   vd43link0 03:00
R  10.12.101.0/24     10.14.202.103         2 10.14.202.103   vd43link0 03:00
Rc 10.14.101.0/24                           1                 LoAccounting
Rc 10.14.201.0/24                           1                 vd42link0
Rc 10.14.202.0/24                           1                 vd43link0
R  172.20.120.0/24    10.14.201.102         2 10.14.201.102   vd42link0 02:53
```

- The default router should point to Router2, with the same number of hops:

```
# get router info routing-table all
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default
Routing table for VRF=0
R*      0.0.0.0/0 [120/3] via 10.14.201.102, vd42link0, 00:05:24
R       10.11.101.0/24 [120/3] via 10.14.202.103, vd43link0, 02:58:13
R       10.11.201.0/24 [120/2] via 10.14.201.102, vd42link0, 02:58:39
R       10.11.202.0/24 [120/2] via 10.14.202.103, vd43link0, 02:58:39
R       10.12.101.0/24 [120/2] via 10.14.202.103, vd43link0, 02:58:39
C       10.14.101.0/24 is directly connected, LoAccounting
C       10.14.201.0/24 is directly connected, vd42link0
```

```
C       10.14.202.0/24 is directly connected, vd43link0
R       172.20.120.0/24 [120/2] via 10.14.201.102, vd42link0, 02:58:39
```

```
# execute traceroute 8.8.8.8
traceroute to 8.8.8.8 (8.8.8.8), 32 hops max, 3 probe packets per hop, 84 byte packets
 1  10.14.201.102  0.167 ms  0.063 ms  0.029 ms
 2  172.20.120.5  0.117 ms  0.073 ms  0.041 ms
 3  172.16.151.1  0.303 ms  0.273 ms  0.253 ms
```

Example 2- Additional link failures on Router2

In addition to the link failure on the ISP router in example, port1 and port3 on Router2 have also failed. This means that Router4 must go through Router3, Router1, Router2, then the ISP router to reach the internet. Note that, for a period of time, some routes' metrics increase to 16. If no better routes are found for these networks, then they eventually disappear.

After the convergence completes, the RIP database and routing table on Router4 should resemble the following:

```
# get router info rip database
Codes: R - RIP, Rc - RIP connected, Rs - RIP static, K - Kernel,
       C - Connected, S - Static, O - OSPF, I - IS-IS, B - BGP
   Network            Next Hop         Metric From            If     Time
R  0.0.0.0/0          10.14.202.103         5 10.14.202.103   vd43link0 02:54
R  10.11.101.0/24     10.14.202.103         3 10.14.202.103   vd43link0 02:54
R  10.11.201.0/24     10.14.202.103         3 10.14.202.103   vd43link0 02:54
R  10.11.202.0/24     10.14.202.103         2 10.14.202.103   vd43link0 02:54
Rc 10.14.101.0/24                           1                 LoAccounting
Rc 10.14.202.0/24                           1                 vd43link0
R  172.20.120.0/24    10.14.202.103         4 10.14.202.103   vd43link0 02:54
```

```
# get router info routing-table all
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default
Routing table for VRF=0
R*      0.0.0.0/0 [120/5] via 10.14.202.103, vd43link0, 00:03:54
R       10.11.101.0/24 [120/3] via 10.14.202.103, vd43link0, 03:10:12
R       10.11.201.0/24 [120/3] via 10.14.202.103, vd43link0, 00:03:54
R       10.11.202.0/24 [120/2] via 10.14.202.103, vd43link0, 03:10:38
C       10.14.101.0/24 is directly connected, LoAccounting
C       10.14.202.0/24 is directly connected, vd43link0
R       172.20.120.0/24 [120/4] via 10.14.202.103, vd43link0, 00:03:54
```

Reaching the internet on the default gateway now requires five hops from Router4:

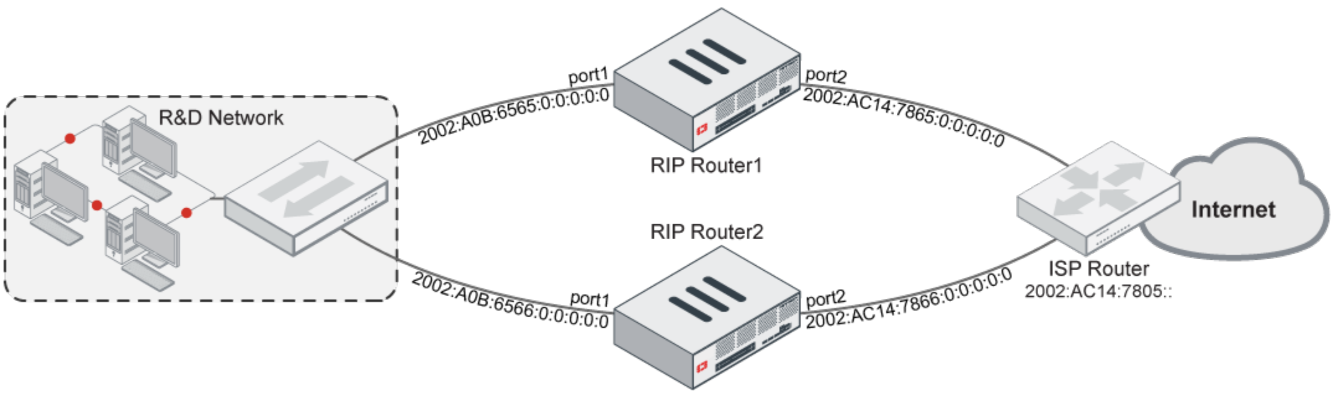
```
# execute traceroute 8.8.8.8
traceroute to 8.8.8.8 (8.8.8.8), 32 hops max, 3 probe packets per hop, 84 byte packets
 1  10.14.202.103  0.087 ms  0.026 ms  0.012 ms
 2  10.11.202.101  0.045 ms  0.024 ms  0.025 ms
 3  10.11.201.102  0.048 ms  0.024 ms  0.015 ms
 4  172.20.120.5  0.050 ms  0.028 ms  0.019 ms
 5  * * *
```

Basic RIPng example

In this example, a small network is configured with RIP next generation (RIPng). Two FortiGates are connected to the internal network and the ISP, providing some redundancy to help ensure that the internal network can always reach the internet.

The FortiGates are running in NAT mode with VDOMs disabled, and firewall policies have already been configured to allow traffic to flow across the interfaces.

All of the internal computers and other network devices support IPv6 addressing and are running RIPng (where applicable), so no static routing is required. Internal network devices only need to know the FortiGate's internal interface network addresses.



Router	Interface (alias)	IPv6 address
Router1	port1 (internal)	2002:A0B:6565:0:0:0:0
	port2 (ISP)	2002:AC14:7865:0:0:0:0
Router2	port1 (internal)	2002:A0B:6566:0:0:0:0
	port2 (ISP)	2002:AC14:7866:0:0:0:0

On each FortiGate, the interfaces are configured first, and then RIPng. No redistribution or authentication is configured.

In the RIPng configuration, only the interface names are required. The ISP router and the other FortiGate are configured as neighbors. Declaring the neighbors reduces the discovery traffic when the routers start. There is no specific command to include a subnet in the RIP broadcast, and RIPng can only be configured using the CLI.

To configure Router1:

1. Configure the interfaces:

```
config system interface
    edit port1
        set allowaccess ping https ssh
        set type physical
        set description "Internal RnD network"
        set alias "internal"
        config ipv6
            set ip6-address 2002:a0b:6565::/0
        end
```

```
        next
        edit port2
            set allowaccess ping https ssh
            set type physical
            set description "ISP and Internet"
            set alias "ISP"
            config ipv6
                set ip6-address 2002:ac14:7865::/0
            end
        next
    end
```

2. Configure RIPng:

```
config router ripng
    config neighbor
        edit 1
            set ip6 2002:a0b:6566::
            set interface port1
        next
        edit 2
            set ip6 2002:ac14:7805::
            set interface port2
        next
    end
    config interface
        edit port1
        next
        edit port2
        next
    end
end
```

To configure Router2:

1. Configure the interfaces:

```
config system interface
    edit port1
        set allowaccess ping https ssh
        set type physical
        set description "Internal RnD network"
        set alias "internal"
        config ipv6
            set ip6-address 2002:a0b:6566::/0
        end
    next
    edit port2
        set allowaccess ping https ssh
        set type physical
        set description "ISP and Internet"
        set alias "ISP"
        config ipv6
            set ip6-address 2002:ac14:7866::/0
        end
    next
end
```

2. Configure RIPng:

```
config router ripng
    config neighbor
        edit 1
            set ip6 2002:a0b:6565::
            set interface port1
        next
        edit 2
            set ip6 2002:ac14:7805::
            set interface port2
        next
    end
    config interface
        edit port1
        next
        edit port2
        next
    end
end
```

Testing the configuration

The following commands can be used to check the RIPng information on the FortiGates, and can help track down issues:

To view the local scope IPv6 addresses used as next-hops by RIPng on the FortiGate:

```
# diagnose ipv6 address list
```

To view IPv6 addresses that are installed in the routing table:

```
# diagnose ipv6 route list
```

To view the IPv6 routing table:

```
# get router info6 routing-table
```

This information is similar to the `diagnose ipv6 route list` command, but it is presented in an easier to read format.

To view the brief output on the RIP information for the interface listed:

```
# get router info6 rip interface external
```

This includes information such as, if the interface is up or down, what routing protocol is being used, and whether passive interface or split horizon is enabled.

OSPF

Open Shortest Path First (OSPF) is a link state routing protocol that is commonly used in large enterprise networks with L3 switches, routers, and firewalls from multiple vendors. It can quickly detect link failures, and converges network traffic without networking loops. It also has features to control which routes are propagated, allowing for smaller routing tables, and provides better load balancing on external links when compared to other routing protocols.

To configure OSPF in the GUI, go to *Network > OSPF*:

Option	Description
Router ID	A unique ID to identify your router in the network, typically in the format x.x.x.x.
Areas	The areas that the router is part of. For each are area, define the *Area ID*, *Type*, and *Authentication* method.
Networks	The networks that OSPF is enabled in, and the area that they belong to.
Interfaces	OSPF interfaces for transmitting and receiving packets. Configure interface properties, such as *Network Type*, *Cost*, *Hello interval*, and others.
Advanced Options	Settings for *Inject Default Route*, *Passive Interfaces*, and *Redistribute*. Redistribution can be enabled by protocol and the metric for each protocol can be configured.

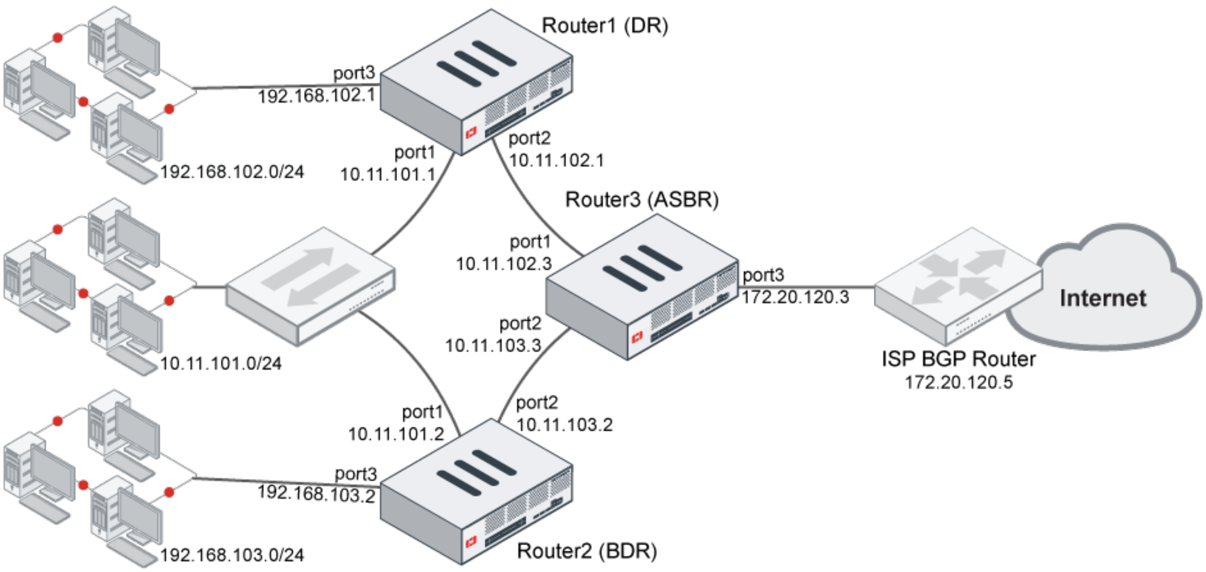
This section includes the following topics:

- Basic OSPF example on page 339
- OSPFv3 neighbor authentication on page 350
- OSPF graceful restart upon a topology change on page 352

Basic OSPF example

In this example, three FortiGate devices are configured in an OSPF network.

- Router1 is the Designated Router (DR). It has the highest priority and the lowest IP address, to ensure that it becomes the DR.
- Router2 is the Backup Designated Router (BDR). It has a high priority to ensure that it becomes the BDR.
- Router3 is the Autonomous System Border Router (ASBR). It routes all traffic to the ISP BGP router for internet access. It redistributes routes from BGP and advertises a default route to its neighbors. It can allow different types of routes, learned outside of OSPF, to be used in OSPF. Different metrics can be assigned to these routes to make them more or less preferred than regular OSPF routes. Route maps could be used to further control what prefixes are advertised or received from the ISP.



FortiGate	Interface	IP address
Router1 (DR)	port1	10.11.101.1
	port2	10.11.102.1
	port3	192.168.102.1
Router2 (BDR)	port1	10.11.101.2
	port2	10.11.103.2
	port3	192.168.103.2
Router3 (ASBR)	port1	10.11.102.3
	port2	10.11.103.3
	port3	172.20.120.3

- Firewall policies are already configured to allow unfiltered traffic in both directions between all of the connected interfaces.
- The interfaces are already configured, and NAT is only used for connections to public networks. The costs for all of the interfaces is left at 0.
- The OSPF network belongs to Area 0, and is not connected to any other OSPF networks. All of the routers are part of the backbone 0.0.0.0 area, so no inter-area communications are needed.
- Router3 redistributes BGP routes into the OSPF AS and peers with the ISP BGP Router over eBGP. For information about configuring BGP, see BGP on page 355.
- The advertised networks - 10.11.101.0, 10.11.102.0, and 10.11.103.0 - are summarized by 10.11.0.0/16. Additional networks are advertised individually by the /24 subnet.

Router1

To configure Router1 in the GUI:

1. Go to *Network > OSPF*.
2. Set *Router ID* to *10.11.101.1*.
3. In the *Areas* table, click *Create New* and set the following:

Area ID	0.0.0.0
Type	Regular
Authentication	None

4. Click *OK*.
5. In the *Networks* table, click *Create New* and set the following:

Area	0.0.0.0
IP/Netmask	10.11.0.0 255.255.0.0

6. Click *OK*.
7. In the *Networks* table, click *Create New* again and set the following:

Area	0.0.0.0
IP/Netmask	192.168.102.0 255.255.255.0

8. Click *OK*.
9. In the *Interfaces* table, click *Create New* and set the following:

Name	Router1-Internal-DR
Interface	port1
Cost	0
Priority	255
Authentication	None
Timers	• Hello Interval: 10 • Dead Interval: 40

10. Click *OK*.
11. In the *Interfaces* table, click *Create New* again and set the following:

Name	Router1-External
Interface	port2
Cost	0
Authentication	None
Timers	• Hello Interval: 10 • Dead Interval: 40

12. Click *OK*.
13. Click *Apply*.

To configure Router1 in the CLI:

```
config router ospf
    set router-id 10.11.101.1
    config area
        edit 0.0.0.0
        next
    end
    config ospf-interface
        edit "Router1-Internal-DR"
            set interface "port1"
            set priority 255
            set dead-interval 40
            set hello-interval 10
        next
        edit "Router1-External"
            set interface "port2"
            set dead-interval 40
            set hello-interval 10
        next
    end
    config network
        edit 1
            set prefix 10.11.0.0 255.255.0.0
        next
        edit 2
            set prefix 192.168.102.0 255.255.255.0
        next
    end
end
```

Router2

To configure Router2 in the GUI:

1. Go to *Network > OSPF*.
2. Set *Router ID* to *10.11.101.2*.
3. In the *Areas* table, click *Create New* and set the following:

Area ID	0.0.0.0
Type	Regular
Authentication	None

4. Click *OK*.
5. In the *Networks* table, click *Create New* and set the following:

Area	0.0.0.0
IP/Netmask	10.11.0.0 255.255.0.0

6. Click *OK*.
7. In the *Networks* table, click *Create New* again and set the following:

Area	0.0.0.0
IP/Netmask	192.168.103.0 255.255.255.0

8. Click *OK*.
9. In the *Interfaces* table, click *Create New* and set the following:

Name	Router2-Internal
Interface	port1
Cost	0
Priority	250
Authentication	None
Timers	• Hello Interval: 10 • Dead Interval: 40

10. Click *OK*.
11. In the *Interfaces* table, click *Create New* again and set the following:

Name	Router2-External
Interface	port2
Cost	0
Authentication	None
Timers	• Hello Interval: 10 • Dead Interval: 40

12. Click *OK*.
13. Click *Apply*.

To configure Router2 in the CLI:

```
config router ospf
    set router-id 10.11.101.1
    config area
        edit 0.0.0.0
        next
    end
    config ospf-interface
        edit "Router2-Internal"
            set interface "port1"
            set priority 250
            set dead-interval 40
            set hello-interval 10
        next
        edit "Router2-External"
            set interface "port2"
```

```
            set dead-interval 40
            set hello-interval 10
        next
    end
    config network
        edit 1
            set prefix 10.11.0.0 255.255.0.0
        next
        edit 2
            set prefix 192.168.103.0 255.255.255.0
        next
    end
end
```

Router3

To configure Router3 in the GUI:

1. Go to *Network > OSPF*.
2. Set *Router ID* to *10.11.103.3*.
3. Under *Default Settings*, set *Inject default route* to *Regular Areas*.
   A default route must be present on Router3 to advertise it to other routers.
4. Enable *Redistribute BGP* and use the default settings.
5. In the *Areas* table, click *Create New* and set the following:

Area ID	0.0.0.0
Type	Regular
Authentication	None

6. Click *OK*.
7. In the *Networks* table, click *Create New* and set the following:

Area	0.0.0.0
IP/Netmask	10.11.0.0 255.255.0.0

8. Click *OK*.
9. In the *Interfaces* table, click *Create New* and set the following:

Name	Router3-Internal
Interface	port1
Cost	0
Authentication	None
Timers	• Hello Interval: 10 • Dead Interval: 40

10. Click *OK*.
11. In the *Interfaces* table, click *Create New* again and set the following:

Name	Router3-Internal2
Interface	port2
Cost	0
Authentication	None
Timers	• Hello Interval: 10 • Dead Interval: 40

12. Click *OK*.
13. Click *Apply*.

To configure Router3 in the CLI:

```
config router ospf
    set default-information-originate enable
    set router-id 10.11.103.3
    config area
        edit 0.0.0.0
        next
    end
    config ospf-interface
        edit "Router3-Internal"
            set interface "port1"
            set dead-interval 40
            set hello-interval 10
        next
        edit "Router3-Internal2"
            set interface "port2"
            set dead-interval 40
            set hello-interval 10
        next
    end
    config network
        edit 1
            set prefix 10.11.0.0 255.255.0.0
        next
    end
    config redistribute "bgp"
        set status enable
    end
end
```

To configure BGP on Router3 in the CLI:

```
config router bgp
    set as 64511
    set router-id 1.1.1.1
    config neighbor
        edit "172.20.120.5"
            set remote-as 64512
        next
    end
    config network
```

```
        edit 1
            set prefix 172.20.120.0 255.255.255.0
        next
    end
end
```

For more information on configuring BGP, see BGP on page 355.

Testing the configuration

Both the network connectivity and OSPF routing are tested. When a link goes down, routes should converge as expected.

Working state

- Router3:

```
Router3 # get router info ospf neighbor
OSPF process 0, VRF 0:
Neighbor ID     Pri   State           Dead Time   Address         Interface
10.11.101.1       1   Full/Backup     00:00:34    10.11.102.1     port1
10.11.101.2       1   Full/Backup     00:00:38    10.11.103.2     port2
```

```
Router3 # get router info ospf status
 Routing Process "ospf 0" with ID 10.11.103.3
 Process uptime is 18 hours 52 minutes
 Process bound to VRF default
 Conforms to RFC2328, and RFC1583Compatibility flag is disabled
 Supports only single TOS(TOS0) routes
 Supports opaque LSA
 Do not support Restarting
 This router is an ASBR (injecting external routing information)
 SPF schedule delay 5 secs, Hold time between two SPFs 10 secs
 Refresh timer 10 secs
 Number of incomming current DD exchange neighbors 0/5
 Number of outgoing current DD exchange neighbors 0/5
 Number of external LSA 3. Checksum 0x021B78
 Number of opaque AS LSA 0. Checksum 0x000000
 Number of non-default external LSA 2
 External LSA database is unlimited.
 Number of LSA originated 16
 Number of LSA received 100
 Number of areas attached to this router: 1
    Area 0.0.0.0 (BACKBONE)
        Number of interfaces in this area is 2(2)
        Number of fully adjacent neighbors in this area is 2
        Area has no authentication
        SPF algorithm last executed 00:37:36.690 ago
        SPF algorithm executed 13 times
        Number of LSA 6. Checksum 0x03eafa
```

```
Router3 # get router info routing-table all
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
```

```
       * - candidate default
Routing table for VRF=0
B*      0.0.0.0/0 [20/0] via 172.20.120.5, port3, 01:10:12
O       10.11.101.0/24 [110/2] via 10.11.103.2, port2, 00:39:34
                       [110/2] via 10.11.102.1, port1, 00:39:34
C       10.11.102.0/24 is directly connected, port1
C       10.11.103.0/24 is directly connected, port2
C       172.20.120.0/24 is directly connected, port3
O       192.168.102.0/24 [110/2] via 10.11.102.1, port1, 02:24:59
O       192.168.103.0/24 [110/2] via 10.11.103.2, port2, 02:14:32
B       192.168.160.0/24 [20/0] via 172.20.120.5, port3, 19:08:39
B       192.168.170.0/24 [20/0] via 172.20.120.5, port3, 01:10:12
```

- Router2:

```
Router2 # get router info ospf neighbor
OSPF process 0, VRF 0:
Neighbor ID     Pri   State           Dead Time   Address         Interface
10.11.101.1     255   Full/DR         00:00:35    10.11.101.1     port1
10.11.103.3       1   Full/DR         00:00:38    10.11.103.3     port3
```

```
Router2 # get router info ospf status
 Routing Process "ospf 0" with ID 10.11.101.2
 Process uptime is 2 hours 53 minutes
 Process bound to VRF default
 Conforms to RFC2328, and RFC1583Compatibility flag is disabled
 Supports only single TOS(TOS0) routes
 Supports opaque LSA
 Do not support Restarting
 SPF schedule delay 5 secs, Hold time between two SPFs 10 secs
 Refresh timer 10 secs
 Number of incomming current DD exchange neighbors 0/5
 Number of outgoing current DD exchange neighbors 0/5
 Number of external LSA 3. Checksum 0x021979
 Number of opaque AS LSA 0. Checksum 0x000000
 Number of non-default external LSA 2
 External LSA database is unlimited.
 Number of LSA originated 5
 Number of LSA received 128
 Number of areas attached to this router: 1
    Area 0.0.0.0 (BACKBONE)
        Number of interfaces in this area is 3(3)
        Number of fully adjacent neighbors in this area is 2
        Area has no authentication
        SPF algorithm last executed 00:47:49.990 ago
        SPF algorithm executed 15 times
        Number of LSA 6. Checksum 0x03e8fb
```

```
Router2 # get router info routing-table all
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default
Routing table for VRF=0
O*E2    0.0.0.0/0 [110/10] via 10.11.103.3, port2, 01:03:58
```

```
C       10.11.101.0/24 is directly connected, port1
O       10.11.102.0/24 [110/2] via 10.11.103.3, port2, 00:49:01
                       [110/2] via 10.11.101.1, port1, 00:49:01
C       10.11.103.0/24 is directly connected, port2
O       192.168.102.0/24 [110/2] via 10.11.101.1, port1, 00:49:01
C       192.168.103.0/24 is directly connected, port3
O E2    192.168.160.0/24 [110/10] via 10.11.103.3, port2, 01:39:31
O E2    192.168.170.0/24 [110/10] via 10.11.103.3, port2, 01:19:39
```

The default route advertised by Router3 using `default-information-originate` is considered an OSPF E2 route. Other routes redistributed from BGP are also E2 routes.

- Router1:

```
Router1 # get router info ospf neighbor
OSPF process 0, VRF 0:
Neighbor ID     Pri   State           Dead Time   Address         Interface
10.11.101.2     250   Full/Backup     00:00:36    10.11.101.2     port1
10.11.103.3       1   Full/DR         00:00:37    10.11.102.3     port2
```

```
Router1 # get router info ospf status
 Routing Process "ospf 0" with ID 10.11.101.1
 Process uptime is 3 hours 7 minutes
 Process bound to VRF default
 Conforms to RFC2328, and RFC1583Compatibility flag is disabled
 Supports only single TOS(TOS0) routes
 Supports opaque LSA
 Do not support Restarting
 SPF schedule delay 5 secs, Hold time between two SPFs 10 secs
 Refresh timer 10 secs
 Number of incomming current DD exchange neighbors 0/5
 Number of outgoing current DD exchange neighbors 0/5
 Number of external LSA 3. Checksum 0x02157B
 Number of opaque AS LSA 0. Checksum 0x000000
 Number of non-default external LSA 2
 External LSA database is unlimited.
 Number of LSA originated 2
 Number of LSA received 63
 Number of areas attached to this router: 1
    Area 0.0.0.0 (BACKBONE)
        Number of interfaces in this area is 3(3)
        Number of fully adjacent neighbors in this area is 2
        Area has no authentication
        SPF algorithm last executed 00:54:08.160 ago
        SPF algorithm executed 11 times
        Number of LSA 6. Checksum 0x03e6fc
```

```
Router1 # get router info routing-table all
Routing table for VRF=0
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default
O*E2    0.0.0.0/0 [110/10] via 10.11.102.3, port2, 01:09:48
C       10.11.101.0/24 is directly connected, port1
C       10.11.102.0/24 is directly connected, port2
```

```
O       10.11.103.0/24 [110/2] via 10.11.102.3, port2, 00:54:49
                       [110/2] via 10.11.101.2, port1, 00:54:49
C       192.168.102.0/24 is directly connected, port3
O       192.168.103.0/24 [110/2] via 10.11.101.2, port1, 00:54:49
O E2    192.168.160.0/24 [110/10] via 10.11.102.3, port2, 01:45:21
O E2    192.168.170.0/24 [110/10] via 10.11.102.3, port2, 01:25:29
```

Link down state

If port1 is disconnected on Router3:

- Router3:

```
Router3 # get router info routing-table all
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default
Routing table for VRF=0
B*      0.0.0.0/0 [20/0] via 172.20.120.5, VLAN20, 01:29:25
O       10.11.101.0/24 [110/2] via 10.11.103.2, port2, 00:00:09
C       10.11.103.0/24 is directly connected, port2
C       172.20.120.0/24 is directly connected, port3
O       192.168.102.0/24 [110/3] via 10.11.103.2, port2, 00:00:09
O       192.168.103.0/24 [110/2] via 10.11.103.2, port2, 02:33:45
B       192.168.160.0/24 [20/0] via 172.20.120.5, port3, 19:27:52
B       192.168.170.0/24 [20/0] via 172.20.120.5, port3, 01:29:25
```

- Router2:

```
Router2 # get router info routing-table all
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default
Routing table for VRF=0
O*E2    0.0.0.0/0 [110/10] via 10.11.103.3, port2, 01:16:36
C       10.11.101.0/24 is directly connected, port1
O       10.11.102.0/24 [110/2] via 10.11.101.1, port1, 00:02:27
C       10.11.103.0/24 is directly connected, port2
O       192.168.102.0/24 [110/2] via 10.11.101.1, port1, 01:01:39
C       192.168.103.0/24 is directly connected, port3
O E2    192.168.160.0/24 [110/10] via 10.11.103.3, port2, 01:52:09
O E2    192.168.170.0/24 [110/10] via 10.11.103.3, port2, 01:32:17
```

- Router1:

```
Router1 # get router info routing-table all
Routing table for VRF=0
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
```

```
          * - candidate default
   O*E2    0.0.0.0/0 [110/10] via 10.11.101.2, port1, 00:05:14
   C       10.11.101.0/24 is directly connected, port1
   C       10.11.102.0/24 is directly connected, port2
   O       10.11.103.0/24 [110/2] via 10.11.101.2, port1, 00:05:15
   C       192.168.102.0/24 is directly connected, port3
   O       192.168.103.0/24 [110/2] via 10.11.101.2, port1, 01:03:50
   O E2    192.168.160.0/24 [110/10] via 10.11.101.2, port1, 00:05:14
   O E2    192.168.170.0/24 [110/10] via 10.11.101.2, port1, 00:05:14
```

OSPFv3 neighbor authentication

OSPFv3 neighbor authentication is available for enhanced IPv6 security.

To configure an OSPF6 interface:

```
config router ospf6
    config ospf6-interface
        edit <name>
            set authentication {none | ah | esp | area}
            set key-rollover-interval <integer>
            set ipsec-auth-alg {md5 | sha1 | sha256 | sha384 | sha512}
            set ipsec-enc-alg {null | des | 3des | aes128 | aes192 | aes256}
            config ipsec-keys
                edit <spi>
                    set auth-key <string>
                    set enc-key <string>
                next
            end
        next
    end
end
```

To configure an OSPF6 virtual link:

```
config router ospf6
    config area
        edit <id>
            config virtual-link
                edit <name>
                    set authentication {none | ah | esp | area}
                    set key-rollover-interval <integer>
                    set ipsec-auth-alg {md5 | sha1 | sha256 | sha384 | sha512}
                    set ipsec-enc-alg {null | des | 3des | aes128 | aes192 | aes256}
                    config ipsec-keys
                        edit <spi>
                            set auth-key <string>
                            set enc-key <string>
                        next
                    end
                next
            end
        next
    end
end
```

To configure an OSPF6 area:

```
config router ospf6
    config area
        edit <id>
            set authentication {none | ah | esp}
            set key-rollover-interval <integer>
            set ipsec-auth-alg {md5 | sha1 | sha256 | sha384 | sha512}
            set ipsec-enc-alg {null | des | 3des | aes128 | aes192 | aes256}
            config ipsec-keys
                edit <spi>
                    set auth-key <string>
                    set enc-key <string>
                next
            end
        next
    end
end
```

CLI command descriptions

Command	Description
<id>	Area entry IP address.
authentication {none | ah | esp | area}	Authentication mode: • `none`: Disable authentication • `ah`: Authentication Header • `esp`: Encapsulating Security Payload • `area`: Use the routing area authentication configuration
key-rollover-interval <integer>	Enter an integer value (300 - 216000, default = 300).
ipsec-auth-alg {md5 | sha1 | sha256 | sha384 | sha512}	Authentication algorithm.
ipsec-enc-alg {null | des | 3des | aes128 | aes192 | aes256}	Encryption algorithm.
<spi>	Security Parameters Index.
auth-key <string>	Authentication key should be hexadecimal numbers. Key length for each algorithm: • MD5: 16 bytes • SHA1: 20 bytes • SHA256: 32 bytes • SHA384:48 bytes • SHA512:84 bytes If the key is shorter than the required length, it will be padded with zeroes.
enc-key <string>	Encryption key should be hexadecimal numbers. Key length for each algorithm: • DES: 8 bytes • 3DES: 24 bytes

Command	Description
	• AES128: 16 bytes • AES192: 24 bytes • AES256: 32 bytes If the key is shorter than the required length, it will be padded with zeroes.

OSPF graceful restart upon a topology change

In OSPF graceful restart mode, the `restart-on-topology-change` option can be used to keep restarting the router in graceful restart mode when a topology change is detected during a restart.

```
config router ospf
    set restart-on-topology-change {enable | disable}
end
```

OSPFv3 graceful restart mode upon a topology change can be used in OSPF6:

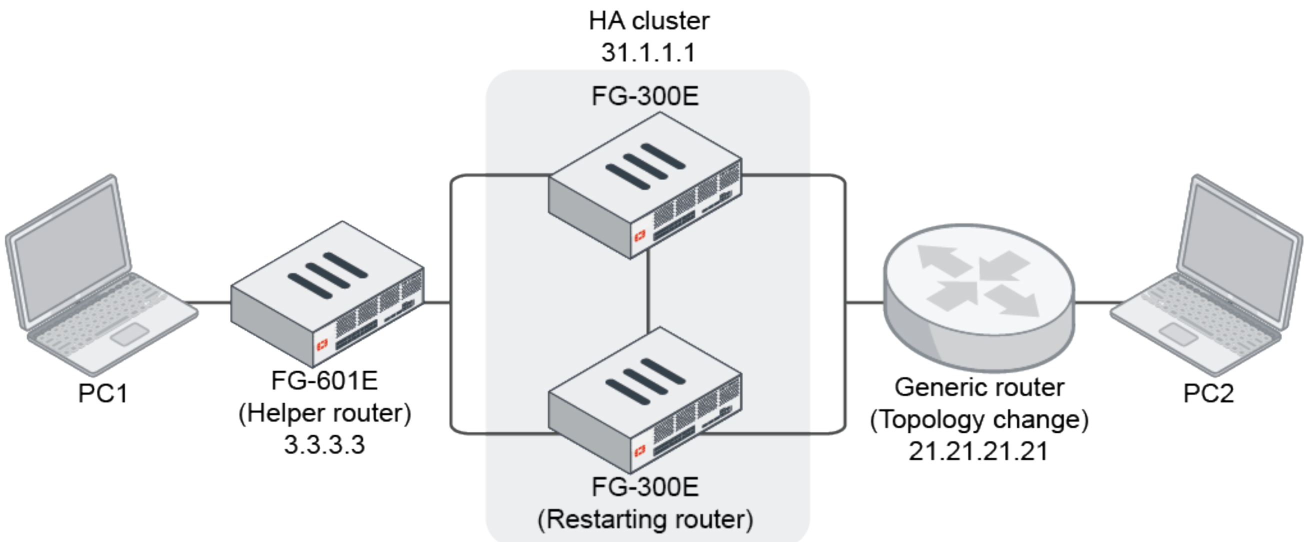
```
config router ospf6
    set restart-on-topology-change {enable | disable}
end
```

Example

In this example, a restarting router (one of the FG-300Es in the HA cluster) informs its neighbors using grace LSAs before restarting its OSPF process. When the helper router (the FG-601E) receives the grace LSAs, it enters helper mode to help with the graceful restart until the graceful period expires. It will act as though there are no changes on the restarting router (FG-300E). A generic router simulates a topology change during the restart event.

If `restart-on-topology-change` is enabled on the restarting router, it will not exit the graceful restart mode even when a topology change is detected.

If `restart-on-topology-change` is disabled on the restarting router, it will exit graceful restart mode when a topology change is detected.



To configure the restarting router:

```
config router ospf
    set router-id 31.1.1.1
    set restart-mode graceful-restart
    set restart-period 180
    set restart-on-topology-change enable
    config area
        edit 0.0.0.0
        next
    end
    config network
        edit 1
            set prefix 172.16.200.0 255.255.255.0
        next
        edit 2
            set prefix 31.1.1.1 255.255.255.255
        next
    end
end
```

To configure the restarting helper router:

```
config router ospf
    set router-id 3.3.3.3
    set restart-mode graceful-restart
    config area
        edit 0.0.0.0
        next
    end
    config network
        edit 1
            set prefix 172.16.200.0 255.255.255.0
        next
        edit 2
            set prefix 3.3.3.3 255.255.255.255
        next
    end
end
```

Testing the configuration

Topology change with continuing graceful restart enabled:

When `restart-on-topology-change` is enabled and there is a topology change during the HA OSPF graceful restart, the graceful restart will continue. The routes on the helper router (FG-601E) are still there and no traffic will drop.

```
# get router info ospf neighbor
OSPF process 0, VRF 0:
Neighbor ID     Pri   State           Dead Time   Address         Interface
31.1.1.1          1   Full/DR         00:14:47*   172.16.200.31   port1
```

```
# get router info routing-table ospf
Routing table for VRF=0
O       21.21.21.21/32 [110/300] via 172.16.200.31, port1, 00:09:55
```

```
O       31.1.1.1/32 [110/200] via 172.16.200.31, port1, 00:55:31
O       100.21.1.0/24 [110/200] via 172.16.200.31, port1, 00:12:31
```

```
# get router info ospf neighbor
OSPF process 0, VRF 0:
Neighbor ID     Pri   State           Dead Time   Address         Interface
31.1.1.1          1   Full/DR         00:14:47*   172.16.200.31   port1
```

```
# get router info routing-table ospf
Routing table for VRF=0
O       21.21.21.21/32 [110/300] via 172.16.200.31, port1, 00:10:07
O       31.1.1.1/32 [110/200] via 172.16.200.31, port1, 00:55:43
O       100.21.1.0/24 [110/200] via 172.16.200.31, port1, 00:12:43
```

```
# get router info ospf neighbor
OSPF process 0, VRF 0:
Neighbor ID     Pri   State           Dead Time   Address         Interface
31.1.1.1          1   Full/DR         00:14:38*   172.16.200.31   port1
```

```
# get router info routing-table ospf
Routing table for VRF=0
O       21.21.21.21/32 [110/300] via 172.16.200.31, port1, 00:10:17
O       31.1.1.1/32 [110/200] via 172.16.200.31, port1, 00:55:53
O       100.21.1.0/24 [110/200] via 172.16.200.31, port1, 00:12:53
```

```
# get router info ospf neighbor
OSPF process 0, VRF 0:
Neighbor ID     Pri   State           Dead Time   Address         Interface
31.1.1.1          1   Full/DR         00:00:38    172.16.200.31   port1
```

```
# get router info routing-table ospf
Routing table for VRF=0
O       21.21.21.21/32 [110/300] via 172.16.200.31, port1, 00:10:37
O       31.1.1.1/32 [110/200] via 172.16.200.31, port1, 00:56:13
O       100.21.1.0/24 [110/200] via 172.16.200.31, port1, 00:13:13
```

Topology change with continuing graceful restart disabled:

When `restart-on-topology-change` is disabled and there is a topology change during the HA OSPF graceful restart, the graceful restart will exit. The routes on the helper router (FG-601E) are lost and traffic will drop.

```
# get router info ospf neighbor
OSPF process 0, VRF 0:
Neighbor ID     Pri   State           Dead Time   Address         Interface
31.1.1.1          1   Full/DR         00:14:57*   172.16.200.31   port1
```

```
# get router info routing-table ospf
Routing table for VRF=0
O       21.21.21.21/32 [110/300] via 172.16.200.31, port1, 00:11:16
O       31.1.1.1/32 [110/200] via 172.16.200.31, port1, 00:56:52
O       100.21.1.0/24 [110/200] via 172.16.200.31, port1, 00:13:52
```

```
# get router info ospf neighbor
OSPF process 0, VRF 0:
Neighbor ID     Pri   State           Dead Time   Address         Interface
31.1.1.1          1   Full/DR         00:14:42*   172.16.200.31   port1
```

```
# get router info routing-table ospf
Routing table for VRF=0
O       21.21.21.21/32 [110/300] via 172.16.200.31, port1, 00:11:31
```

```
O       31.1.1.1/32 [110/200] via 172.16.200.31, port1, 00:57:07
O       100.21.1.0/24 [110/200] via 172.16.200.31, port1, 00:14:07

# get router info ospf neighbor
OSPF process 0, VRF 0:
Neighbor ID     Pri   State           Dead Time   Address         Interface
31.1.1.1          1   Full/DR         00:14:40*   172.16.200.31   port1
```

No routes are lost:

```
# get router info routing-table ospf
Routing table for VRF=0
O       31.1.1.1/32 [110/200] via 172.16.200.31, port1, 00:57:09

# get router info ospf neighbor
OSPF process 0, VRF 0:
Neighbor ID     Pri   State           Dead Time   Address         Interface
31.1.1.1          1   Full/DR         00:14:38*   172.16.200.31   port1
```

No routes are lost:

```
# get router info routing-table ospf
Routing table for VRF=0
O       31.1.1.1/32 [110/200] via 172.16.200.31, port1, 00:57:11
```

No routes are lost:

```
# get router info routing-table ospf
Routing table for VRF=0
O       21.21.21.21/32 [110/300] via 172.16.200.31, port1, 00:04:42
O       31.1.1.1/32 [110/200] via 172.16.200.31, port1, 01:01:59
O       100.21.1.0/24 [110/200] via 172.16.200.31, port1, 00:04:42
```

BGP

Border Gateway Protocol (BGP) is a standardized routing protocol that is used to route traffic across the internet. It exchanges routing information between Autonomous Systems (AS) on the internet and makes routing decisions based on path, network policies, and rule sets. BGP contains two distinct subsets: internal BGP (iBGP) and external BGP (eBGP). iBGP is intended for use within your own networks. eBGP is used to connect different networks together and is the main routing protocol for the internet backbone.

To configure BGP in the GUI, go to *Network > BGP*:

Option	Description
Local AS	The AS number for the local router.
Router ID	A unique ID to identify your router in the network, typically in the format x.x.x.x.
Neighbors	The neighbors that the FortiGate will be peering with. Configure the remote router's AS number, any other properties used for peering with the neighbor, and IPv4 and IPv6 filtering.
Neighbor Groups	The neighbor groups that share the same outbound policy configurations.
Neighbor Ranges	The source address range of BGP neighbors that will be automatically assigned to a neighbor group.

Option	Description
IPv4 & IPv6 Networks	The networks to be advertised to other BGP routers.
IPv4 & IPv6 Redistribute	Enable redistribution by protocol. Specify either *All* routes, or *Filter* by route map.
Dampening	Enable route flap dampening to reduce the propagation of flapping routes.
Graceful Restart	Enable BGP graceful restart, which causes the adjacent routers to keep routes active while the BGP peering is restarted on the FortiGate. This is useful in HA instances when failover occurs.
Advanced Options	Various advanced settings, such as *Local Preference*, *Distance internal*, *Keepalive*, *Holdtime*, and others
Best Path Selection	Configure path selection attributes on this router.

This section includes the following topics:

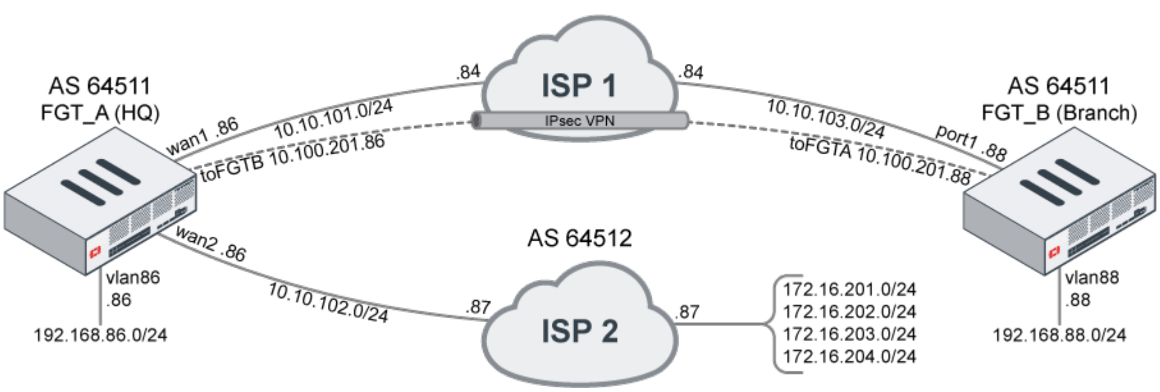
- Basic BGP example on page 356
- Route filtering with a distribution list on page 365
- Next hop recursive resolution using other BGP routes on page 369
- Next hop recursive resolution using ECMP routes on page 370
- BGP conditional advertisement on page 370
- BGP error handling per RFC 7606 on page 376
- BGP next hop tag-match mode on page 378
- Troubleshooting BGP on page 384

Basic BGP example

In this example, BGP is configured on two FortiGate devices. The FortiGates are geographically separated, and form iBGP peering over a VPN connection. FGT_A also forms eBGP peering with ISP2.

FGT_A learns routes from ISP2 and redistributes them to FGT_B while preventing any iBGP routes from being advertised.

The internal networks behind the FortiGates can communicate with each other, and the internal networks behind FGT_B can traverse FGT_A to reach networks that are advertised by ISP2.



- FGT_A and FGT_B have static routes to each other through ISP1. ISP1 does not participate in BGP.
- The IPsec VPN tunnel between FGT_A and FGT_B is configured with wildcard 0.0.0.0/0 networks for phase2 local and remote selectors. The VPN interfaces have IP addresses already configured and are used for peering between FGT_A and FGT_B.
- FGT_A is configure to peer with ISP2 on 10.10.108.86.
- The firewall policies between FGT_A and FGT_B are not NATed. The firewall policies egressing on wan2 are NATed.

Configuring iBGP peering

To configure FGT_A to establish iBGP peering with FGT_B in the GUI:

1. Go to *Network > BGP*.
2. Set *Local AS* to *64511*
3. Set *Router ID* to *1.1.1.1*.
4. In the *Neighbors* table, click *Create New* and set the following:

IP	10.100.201.88
Remote AS	64511

5. Click *OK*.
6. Under *Networks*, set *IP/Netmask* to *192.168.86.0/24*.
7. Click *Apply*.
8. In the CLI, set the interface used as the source IP address of the TCP connection (where the BGP session, TCP/179, is connecting from) for the neighbor (`update-source`) to `toFGTB`.

To configure FGT_A to establish iBGP peering with FGT_B in the CLI:

```
config router bgp
    set as 64511
    set router-id 1.1.1.1
    config neighbor
        edit "10.100.201.88"
            set remote-as 64511
            set update-source "toFGTB"
        next
    end
    config network
        edit 1
            set prefix 192.168.86.0 255.255.255.0
        next
    end
end
```

To configure FGT_B to establish iBGP peering with FGT_A in the GUI:

1. Go to *Network > BGP*.
2. Set *Local AS* to *64511*
3. Set *Router ID* to *2.2.2.2*.
4. In the *Neighbors* table, click *Create New* and set the following:

IP	10.100.201.86
Remote AS	64511

5. Click *OK*.
6. Under *Networks*, set *IP/Netmask* to *192.168.88.0/24*.
7. Click *Apply*.
8. In the CLI, set the interface used as the source IP address of the TCP connection (where the BGP session, TCP/179, is connecting from) for the neighbor (`update-source`) to `toFGTA`.

To configure FGT_B to establish iBGP peering with FGT_A in the CLI:

```
config router bgp
    set as 64511
    set router-id 2.2.2.2
    config neighbor
        edit "10.100.201.86"
            set remote-as 64511
            set update-source "toFGTA"
        next
    end
    config network
        edit 1
            set prefix 192.168.88.0 255.255.255.0
        next
    end
end
```

To check the FGT_A and FGT_B peering:

1. Check the BGP neighbors:

   ```
   # get router info bgp neighbors
   ```

2. Check the networks learned from neighbors:

   ```
   # get router info bgp network
   ```

3. Check that the routes are added to the routing table:

   ```
   # get router info routing-table all
   ```

To see the neighborship status, network, and routing table command outputs for the completed example, see Troubleshooting and debugging on page 361.

Configuring eBGP peering

By establishing eBGP peering with ISP2, learned routes will have a distance of 20 and will automatically be propagated to iBGP peers. iBGP peers do not change the next hop when they advertise a route. To make FGT_B receive a route with FGT_A as the next hop, and not ISP 2's network, *Next hop self* (`next-hop-self`) is enabled for routes advertised to FGT_B.

Additionally, to peer with another router that is multiple hops away, enable `ebg-enforce-multihop` in the neighbor configuration.

In this example, the iBGP routes are automatically advertised to the eBGP neighbor, so a route map is created to deny iBGP routes from being advertised to ISP 2. Prefixes from ISP 2 are advertised to FGT_A and FGT_B, but no prefixes are advertised from FGT_A to ISP 2.

To configure FGT_A to establish eBGP peering with ISP 2 in the GUI:

1. Configure a route map to prevent advertisement of iBGP routes to ISP 2:
   a. Go to *Network > Routing Objects* and click *Create New > Route Map*.
   b. Set *Name* to *exclude1*.
   c. In the *Rules* table, click *Create New*.
   d. Set *Action* to *Deny*.
   e. Under *Other Rule Variables*, enable *Match origin* and set it to *IGP*.
   f. Click *OK*.
   g. Click *OK*.
2. Update the BGP configuration:
   a. Go to *Network > BGP*.
   b. In the *Neighbors* table, click *Create New* and set the following:

IP	10.10.102.87
Remote AS	64512
Route map out	exclude1

   c. Click *OK*.
   d. In the *Neighbors* table, edit the previously created entry, *10.100.201.88*.
   e. Under *IPv4 Filtering*, select *Next hop self*.
   f. Click *OK*.
   g. Click *Apply*.

To configure FGT_A to establish eBGP peering with ISP 2 in the CLI:

1. Configure a route map to prevent advertisement of iBGP routes to ISP 2:

```
config router route-map
    edit "exclude1"
        config rule
            edit 1
                set action deny
                set match-origin igp
            next
        end
    next
end
```

2. Update the BGP configuration:

```
config router bgp
    config neighbor
        edit "10.10.102.87"
            set remote-as 64512
            set route-map-out "exclude1"
```

```
            next
            edit "10.100.201.88"
                set next-hop-self enable
            next
        end
    end
```

To see the neighborship status, network, and routing table command outputs for the completed example, see Troubleshooting and debugging on page 361.

Firewall policies

On FGT_A configure the following policies:

- Allow the internal subnet to the VPN interface. Do not enable NAT. Enable security profiles as required.
- Allow the VPN interface to the internal subnet. Do not enable NAT. Enable security profiles as required.
- Allow the internal subnet to wan2. Enable NAT and security profiles as required.
- Allow VPN traffic from toFGTA to wan2. Enable NAT and security profiles as required.

On FGT_B configure the following policies:

- Allow the internal subnet to the VPN interface. Do not enable NAT. Enable security profiles as required.
- Allow the VPN interface to the internal subnet. Do not enable NAT. Enable security profiles as required.

To verify that pinging from FGT_B to FGT_A is successful:

```
FGT_B # execute ping-options source 192.168.88.88
FGT_B # execute ping 192.168.86.86
PING 192.168.86.86 (192.168.86.86): 56 data bytes
64 bytes from 192.168.86.86: icmp_seq=0 ttl=255 time=0.5 ms
...
--- 192.168.86.86 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 0.3/0.3/0.5 ms
```

To verify that pinging from FGT_B to a subnet in ISP 2 is successful:

```
FGT_B # execute ping-options source 192.168.88.88
FGT_B # execute ping 172.16.201.87
PING 172.16.201.87 (172.16.201.87): 56 data bytes
64 bytes from 172.16.201.87: icmp_seq=0 ttl=254 time=0.6 ms
...
--- 172.16.201.87 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 0.4/0.4/0.6 ms

FGT_B # execute traceroute-options source 192.168.88.88
FGT_B # execute traceroute 172.16.201.87
traceroute to 172.16.201.87 (172.16.201.87), 32 hops max, 3 probe packets per hop, 84 byte
packets
 1  10.100.201.86  0.315 ms  0.143 ms  0.110 ms
 2  172.16.201.87  0.258 ms  0.144 ms  0.222 ms
```

Troubleshooting and debugging

When troubleshooting issues, logically step through the debugs. For example, if peering cannot be established between FGT_A and FGT_B:

1. Verify the basic connectivity between the FGT_A wan1 interface and the FGT_B port1 interface.
2. Verify that the VPN between FGT_A and FGT_B is established.
3. Verify the connectivity between the VPN interfaces.
4. Check the neighborship status on each peer. Use the BGP state to help determine the possible issue, for example:

Idle state	The local FortiGate has not started the BGP process with the neighbor. This could be because the eBGP peer is multiple hops away, but multihop is not enabled.
Connect	The local FortiGate has started the BGP process, but has not initiated a TCP connection, possibly due to improper routing.
Active	The local FortiGate has initiated a TCP connection, but there is no response. This might indicate issues with the delivery or the response from the remote peer.

5. If there are issues establishing the TCP connection, use the command `diagnose sniffer packet any 'tcp and port 179'` to identify the problem at the packet level.

The following outputs show instances where all of the configurations are completed, peering has formed, and routes have been exchanged. The debug output during each configuration step might differ from these outputs. These debug outputs can be used to help identify what might be missing or misconfigured on your device.

To verify the status of the neighbors:

```
FGT_A # get router info bgp neighbors
VRF 0 neighbor table:
BGP neighbor is 10.10.102.87, remote AS 64512, local AS 64511, external link
  BGP version 4, remote router ID 192.168.2.87
  BGP state = Established, up for 01:54:37
  Last read 00:00:29, hold time is 180, keepalive interval is 60 seconds
  Configured hold time is 180, keepalive interval is 60 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
    Address family IPv4 Unicast: advertised and received
    Address family IPv6 Unicast: advertised and received
  Received 513 messages, 1 notifications, 0 in queue
  Sent 517 messages, 2 notifications, 0 in queue
  Route refresh request: received 0, sent 0
  Minimum time between advertisement runs is 30 seconds
 For address family: IPv4 Unicast
  BGP table version 5, neighbor version 0
  Index 3, Offset 0, Mask 0x8
  Community attribute sent to this neighbor (both)
  Outbound path policy configured
  Route map for outgoing advertisements is *exclude1root
  4 accepted prefixes, 4 prefixes in rib
  0 announced prefixes
 For address family: IPv6 Unicast
  BGP table version 1, neighbor version 0
  Index 3, Offset 0, Mask 0x8
  Community attribute sent to this neighbor (both)
```

```
    0 accepted prefixes, 0 prefixes in rib
    0 announced prefixes
   Connections established 4; dropped 3
  Local host: 10.10.102.86, Local port: 20364
  Foreign host: 10.10.102.87, Foreign port: 179
  Nexthop: 10.10.102.86
  Nexthop interface: wan2
  Nexthop global: ::
  Nexthop local: ::
  BGP connection: non shared network
  Last Reset: 01:54:42, due to BGP Notification sent
  Notification Error Message: (CeaseUnspecified Error Subcode)
  BGP neighbor is 10.100.201.88, remote AS 64511, local AS 64511, internal link
    BGP version 4, remote router ID 2.2.2.2
    BGP state = Established, up for 01:54:07
    Last read 00:00:11, hold time is 180, keepalive interval is 60 seconds
    Configured hold time is 180, keepalive interval is 60 seconds
    Neighbor capabilities:
      Route refresh: advertised and received (old and new)
      Address family IPv4 Unicast: advertised and received
      Address family IPv6 Unicast: advertised and received
    Received 527 messages, 3 notifications, 0 in queue
    Sent 543 messages, 8 notifications, 0 in queue
    Route refresh request: received 0, sent 0
    Minimum time between advertisement runs is 30 seconds
    Update source is toFGTB
   For address family: IPv4 Unicast
    BGP table version 5, neighbor version 4
    Index 1, Offset 0, Mask 0x2
    NEXT_HOP is always this router
    Community attribute sent to this neighbor (both)
    1 accepted prefixes, 1 prefixes in rib
    5 announced prefixes
   For address family: IPv6 Unicast
    BGP table version 1, neighbor version 1
    Index 1, Offset 0, Mask 0x2
    Community attribute sent to this neighbor (both)
    0 accepted prefixes, 0 prefixes in rib
    0 announced prefixes
   Connections established 7; dropped 6
  Local host: 10.100.201.86, Local port: 179
  Foreign host: 10.100.201.88, Foreign port: 6245
  Nexthop: 10.100.201.86
  Nexthop interface: toFGTB
  Nexthop global: ::
  Nexthop local: ::
  BGP connection: non shared network
  Last Reset: 01:54:12, due to BGP Notification received
  Notification Error Message: (CeaseUnspecified Error Subcode)

  FGT_B # get router info bgp neighbors
  VRF 0 neighbor table:
  BGP neighbor is 10.100.201.86, remote AS 64511, local AS 64511, internal link
    BGP version 4, remote router ID 1.1.1.1
    BGP state = Established, up for 01:56:04
    Last read 00:00:48, hold time is 180, keepalive interval is 60 seconds
    Configured hold time is 180, keepalive interval is 60 seconds
```

```
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
    Address family IPv4 Unicast: advertised and received
    Address family IPv6 Unicast: advertised and received
  Received 532 messages, 3 notifications, 0 in queue
  Sent 526 messages, 3 notifications, 0 in queue
  Route refresh request: received 0, sent 0
  Minimum time between advertisement runs is 30 seconds
  Update source is toFGTA
 For address family: IPv4 Unicast
  BGP table version 4, neighbor version 3
  Index 1, Offset 0, Mask 0x2
  Community attribute sent to this neighbor (both)
  5 accepted prefixes, 5 prefixes in rib
  1 announced prefixes
 For address family: IPv6 Unicast
  BGP table version 1, neighbor version 1
  Index 1, Offset 0, Mask 0x2
  Community attribute sent to this neighbor (both)
  0 accepted prefixes, 0 prefixes in rib
  0 announced prefixes
 Connections established 7; dropped 6
Local host: 10.100.201.88, Local port: 6245
Foreign host: 10.100.201.86, Foreign port: 179
Nexthop: 10.100.201.88
Nexthop interface: toFGTA
Nexthop global: ::
Nexthop local: ::
BGP connection: non shared network
Last Reset: 01:56:09, due to BGP Notification sent
Notification Error Message: (CeaseUnspecified Error Subcode)
```

`# get router info bgp neighbors <neighbor's IP>` can also be used to verify the status of a specific neighbor.

To verify the networks learned from neighbors or a specific network:

```
FGT_A # get router info bgp network
VRF 0 BGP table version is 5, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network          Next Hop            Metric LocPrf Weight RouteTag Path
*> 172.16.201.0/24  10.10.102.87             0             0        0 64512 i <-/1>
*> 172.16.202.0/24  10.10.102.87             0             0        0 64512 i <-/1>
*> 172.16.203.0/24  10.10.102.87             0             0        0 64512 i <-/1>
*> 172.16.204.0/24  10.10.102.87             0             0        0 64512 i <-/1>
*> 192.168.86.0     0.0.0.0                       100  32768        0 i <-/1>
*>i192.168.88.0     10.100.201.88            0    100      0        0 i <-/1>
Total number of prefixes 6
FGT_A # get router info bgp network 172.16.201.0
VRF 0 BGP routing table entry for 172.16.201.0/24
Paths: (1 available, best #1, table Default-IP-Routing-Table)
  Advertised to non peer-group peers:
   10.100.201.88
  Original VRF 0
```

```
  64512
    10.10.102.87 from 10.10.102.87 (192.168.2.87)
      Origin IGP metric 0, localpref 100, valid, external, best
      Last update: Tue Dec 15 22:52:08 2020

FGT_A # get router info bgp network 192.168.88.0
VRF 0 BGP routing table entry for 192.168.88.0/24
Paths: (1 available, best #1, table Default-IP-Routing-Table)
  Not advertised to any peer
  Original VRF 0
  Local
    10.100.201.88 from 10.100.201.88 (2.2.2.2)
      Origin IGP metric 0, localpref 100, valid, internal, best
      Last update: Tue Dec 15 22:52:39 2020

FGT_B # get router info bgp network
VRF 0 BGP table version is 4, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network          Next Hop            Metric LocPrf Weight RouteTag Path
*>i172.16.201.0/24  10.100.201.86            0    100      0        0 64512 i <-/1>
*>i172.16.202.0/24  10.100.201.86            0    100      0        0 64512 i <-/1>
*>i172.16.203.0/24  10.100.201.86            0    100      0        0 64512 i <-/1>
*>i172.16.204.0/24  10.100.201.86            0    100      0        0 64512 i <-/1>
*>i192.168.86.0     10.100.201.86            0    100      0        0 i <-/1>
*> 192.168.88.0     0.0.0.0                       100  32768        0 i <-/1>
Total number of prefixes 6
```

To verify the routing tables on FGT_A and FGT_B:

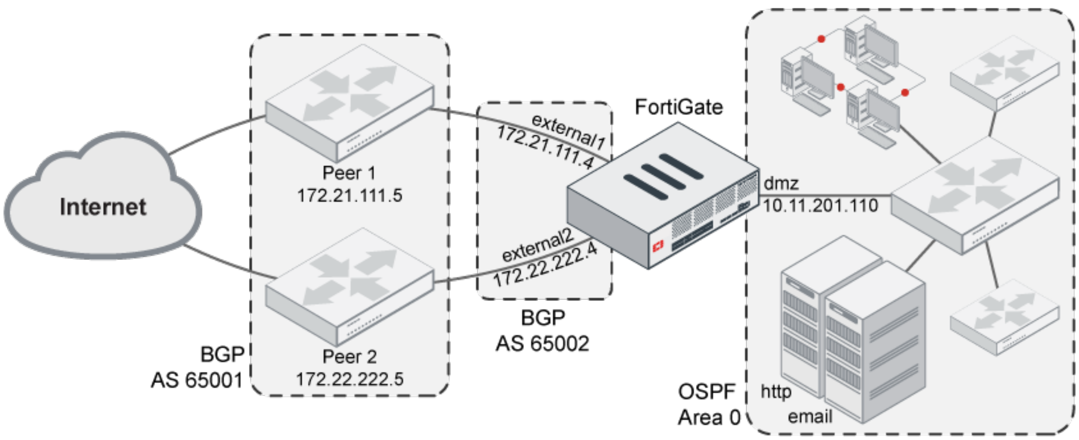
```
FGT_A # get router info routing-table all
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default
Routing table for VRF=0
S*      0.0.0.0/0 [10/0] via 172.16.151.1, port1, [5/0]
                  [10/0] via 192.168.2.1, port2, [10/0]
C       10.10.101.0/24 is directly connected, wan1
C       10.10.102.0/24 is directly connected, wan2
S       10.10.103.0/24 [10/0] via 10.10.101.84, wan1
C       10.100.201.0/24 is directly connected, toFGTB
C       10.100.201.86/32 is directly connected, toFGTB
C       172.16.151.0/24 is directly connected, port1
B       172.16.201.0/24 [20/0] via 10.10.102.87, wan2, 02:09:50
B       172.16.202.0/24 [20/0] via 10.10.102.87, wan2, 02:09:50
B       172.16.203.0/24 [20/0] via 10.10.102.87, wan2, 02:09:50
B       172.16.204.0/24 [20/0] via 10.10.102.87, wan2, 02:09:50
C       192.168.2.0/24 is directly connected, port2
C       192.168.86.0/24 is directly connected, vlan86
B       192.168.88.0/24 [200/0] via 10.100.201.88, toFGTB, 02:09:19

FGT_B # get router info routing-table all
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
```

```
        O - OSPF, IA - OSPF inter area
        N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
        E1 - OSPF external type 1, E2 - OSPF external type 2
        i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
        * - candidate default
Routing table for VRF=0
S*      0.0.0.0/0 [10/0] via 10.10.103.84, port1
C       10.10.103.0/24 is directly connected, port1
C       10.100.201.0/24 is directly connected, toFGTA
C       10.100.201.88/32 is directly connected, toFGTA
B       172.16.201.0/24 [200/0] via 10.100.201.86, toFGTA, 02:11:36
B       172.16.202.0/24 [200/0] via 10.100.201.86, toFGTA, 02:11:36
B       172.16.203.0/24 [200/0] via 10.100.201.86, toFGTA, 02:11:36
B       172.16.204.0/24 [200/0] via 10.100.201.86, toFGTA, 02:11:36
B       192.168.86.0/24 [200/0] via 10.100.201.86, toFGTA, 02:11:36
C       192.168.88.0/24 is directly connected, vlan88
```

Route filtering with a distribution list

During BGP operations, routes can be propagated between BGP peers and redistributed from other routing protocols. In some situations, advertising routes from one peer to another might need to be prevented.

The Basic BGP example on page 356 explains using a route map to filter routes that are learned from iBGP to prevent them from propagating to an eBGP peer. In this example, a distribution list is used to prevent certain routes from one peer from being advertised to another peer.



- A company has its own web and email servers in an OSPF area, and needs to advertise routes to these resources to external peers. Users, routers, and other server all reside in the OSPF area.
- The FortiGate acts as the BGP border router, redistributing routes from the company's network to its BGP peers. It is connected to the OSPF area using its DMZ interface.
- Two ISP managed BGP peers in an AS (Peer 1 and Peer 2) are used to access the internet, and routes must not to be advertised from Peer 1 to Peer 2. The manufacturers of these routers, and information about other devices on the external BGP AS, are not known.
- Routes to the BGP peers are redistributed so that external locations can access the web and email servers in the OSPF area. The FortiGate device's external interfaces and the BGP peers are in different ASs, and form eBGP peers.
- Other networking devices must be configured for BGP. The peer routers must be updated with the FortiGate device's BGP information, including IP addresses, AS number, and any specific capabilities that are used, such as

IPv6, graceful restart, BFD, and so on.

- It is assumed that security policies have been configured to allow traffic between the networks and NAT is not used. To tighten security, only the required services should be allowed inbound to the various servers.
- In a real life scenario, public IP addresses would be used in place of private IP addresses.

Configuring BGP

In this example, Peer 1 routes are blocked from being advertised to Peer 2 using an access list. All incoming routes from Peer 1 are blocked when updates are sent to Peer 2.

Routes learned from OSPF are redistributed into BGP. EBGP multi path is enabled to load-balance traffic between the peers using ECMP. See Equal cost multi-path on page 308 for more information.

To configure BGP in the GUI:

1. Configure an access list to block Peer 1 routes:
   a. Go to *Network > Routing Objects* and click *Create New > Access List*.
   b. Set *Name* to *block_peer1*.
   c. In the *Rules* table, click *Create New*.
   d. Set *Action* to *Deny*.
   e. Enable *Exact Match* and specify the prefix *172.21.111.0 255.255.255.0*.
   f. Click *OK*.
   g. Click *OK*.
2. Configure BGP:
   a. Go to *Network > BGP*.
   b. Set *Local AS* to *65001*
   c. Set *Router ID* to *10.11.201.110*.
   d. In the *Neighbors* table, click *Create New* and set the following:

IP	172.21.111.5
Remote AS	65001

   e. Click *OK*.
   f. In the *Neighbors* table, click *Create New* again and set the following:

IP	172.22.222.5
Remote AS	65001
Distribute list out	Enable, and select the *block_peer1* access list.

   g. Click *OK*.
   h. Under *IPv4 Redistribute*, enable *OSPF* and select *ALL*.
   i. Expand *Best Path Selection* and enable *EBGP multi path*.
   j. Click *Apply*.

To configure BGP in the CLI:

1. Configure an access list to block Peer 1 routes:

```
config router access-list
    edit "block_peer1"
        config rule
            edit 1
                set action deny
                set prefix 172.21.111.0 255.255.255.0
                set exact-match enable
            next
        end
    next
end
```

2. Configure BGP:

```
config router bgp
    set as 65001
    set router-id 10.11.201.110
    set ebgp-multipath enable
    config neighbor
        edit "172.21.111.5"
            set remote-as 65001
        next
        edit "172.22.222.5"
            set distribute-list-out "block_peer1"
            set remote-as 65001
        next
    end
    config redistribute "ospf"
        set status enable
    end
end
```

Configuring OSPF

In this example, all of the traffic is within the one OSPF area, and there are other OSPF routers in the network. When adjacencies are formed, other routers receive the routes advertised from the FortiGate that are redistributed from BGP.

To configure OSPF in the GUI:

1. Go to *Network > OSPF*.
2. Set *Router ID* to *10.11.201.110*.
3. In the *Areas* table, click *Create New* and set the following:

Area ID	0.0.0.0
Type	Regular
Authentication	None

4. Click *OK*.
5. In the *Networks* table, click *Create New* and set the following:

Area	0.0.0.0
IP/Netmask	10.11.201.0 255.255.255.0

6. Click *OK*.
7. In the *Interfaces* table, click *Create New* and set the following:

Name	OSPF_dmz_network
Interface	dmz

8. Click *OK*.
9. Enable *Redistribute BGP* and set *Metric value* to *1*.
10. Click *Apply*.

To configure OSPF in the CLI:

```
config router ospf
    set router-id 10.11.201.110
    config area
        edit 0.0.0.0
        next
    end
    config ospf-interface
        edit "OSPF_dmz_network"
            set interface "dmz"
        next
    end
    config network
        edit 1
            set prefix 10.11.201.0 255.255.255.0
        next
    end
    config redistribute "bgp"
        set status enable
        set metric 1
    end
end
```

Testing the configuration

To test this configuration, run the standard connectivity checks, and also make sure that routes are being passed between protocols as expected. Use the following checklist to help verify that the FortiGate is configured successfully:

1. Check that the FortiGate has established peering with BGP Peer 1 and Peer 2:

   ```
   # get router info bgp summary
   ```

   ```
   # get router info bgp neighbors
   ```

2. Check that the FortiGate has formed adjacency with OSPF neighbors:

   ```
   # get router info ospf status
   ```

   ```
   # get router info ospf neighbors
   ```

3. Check the routing table on the FortiGate to make sure that routes from both OSPF and BGP are included:

```
# get router info routing-table all
```

4. Check devices in the OSPF network for internet connectivity and to confirm that routes redistributed from BGP are in their routing tables.
5. Check the routing table on Peer 2 to confirm that no routes from Peer 1 are included.
6. Check that the routes from the internal OSPF network are redistributed to Peer 1 and Peer 2.
7. Verify connectivity to the HTTP and email servers.

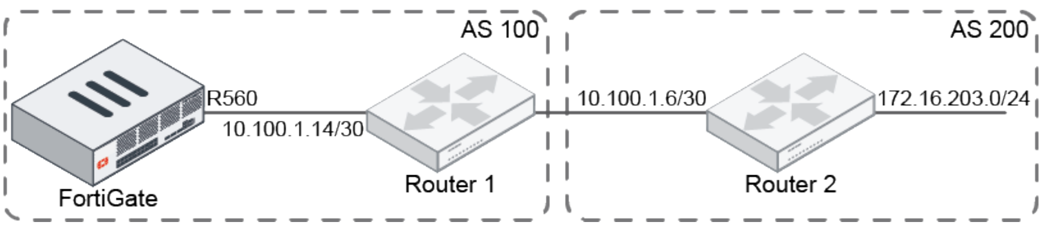
Next hop recursive resolution using other BGP routes

By default, BGP routes are not considered when a BGP next hop requires recursive resolution. They are considered when `recursive-next-hop` is enabled. Recursive resolution will resolve to one level.

To consider BGP routes for recursive resolution of next hops:

```
config router bgp
    set recursive-next-hop enable
end
```

Example



To see the change in the routing table when the option is enabled:

1. Check the BGP routing table:

```
# get router info routing-table bgp
Routing table for VRF=0
B       10.100.1.4/30 [200/0] via 10.100.1.14 (recursive is directly connected, R560),
00:02:06
```

2. Enable BGP routes for recursive resolution of next hops:

```
config router bgp
    set recursive-next-hop enable
end
```

3. Check the BGP routing table again:

```
# get router info routing-table bgp
Routing table for VRF=0
B       10.100.1.4/30 [200/0] via 10.100.1.14 (recursive is directly connected, R560),
00:02:15
B       172.16.203.0/24 [200/0] via 10.100.1.6 (recursive via 10.100.1.14, R560),
00:00:06
```

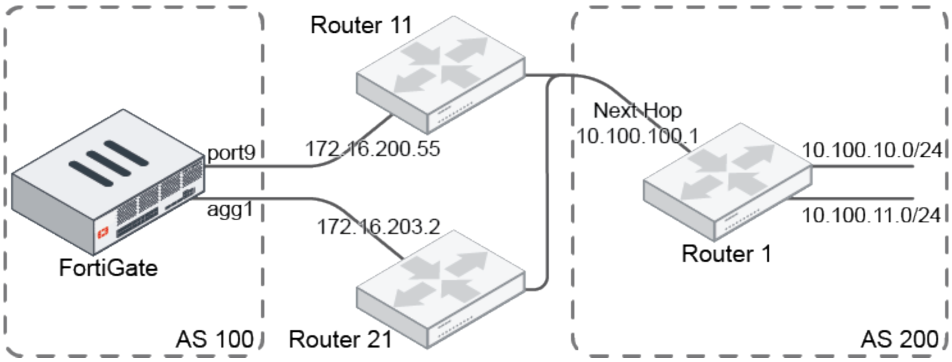
The second BGP route's next hop is now recursively resolved by another BGP route.

Next hop recursive resolution using ECMP routes

When there are multiple ECMP routes to a BGP next hop, all of them are considered for the next hop recursive resolution. This ensures that the outgoing traffic can be load balanced.

To support multipath, either EGBP or IGBP multipath must be enabled:

```
config router bgp
    set ebgp-multipath enable
    set ibgp-multipath enable
end
```



In this example, there are two static routes. The FortiGate has learned two BGP routes from Router 1 that have the same next hop at 10.100.100.1. The next hop is resolved by the two static routes.

To verify that the routes are added to the BGP routing table:

1. Check the two static routes:

```
# get router info routing-table static
Routing table for VRF=0
S       10.100.100.0/24 [10/0] via 172.16.200.55, port9
                        [10/0] via 172.16.203.2, agg1
```

2. Confirm that both routes are in the BGP routing table:

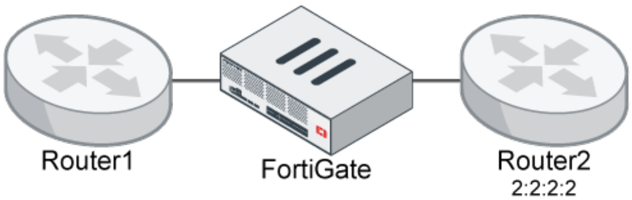
```
# get router info routing-table bgp
Routing table for VRF=0
B       10.100.10.0/24 [20/200] via 10.100.100.1 (recursive via 172.16.200.55, port9),
00:00:07
                                                 (recursive via 172.16.203.2, agg1),
00:00:07
B       10.100.11.0/24 [20/200] via 10.100.100.1 (recursive via 172.16.200.55, port9),
00:00:07
                                                 (recursive via 172.16.203.2, agg1),
00:00:07
```

BGP conditional advertisement

BGP conditional advertisement allows the router to advertise a route only when certain conditions are met. Multiple conditions can be used together, with conditional route map entries treated as an AND operator, and IPv6 is supported.

Multiple conditions example

In this example, the FortiGate only advertises routes to its neighbor 2.2.2.2 if it learns multiple BGP routes defined in its conditional route map entry. All conditionals must be met.



To configure multiple conditions in BGP conditional advertisements:

1. Configure the IPv4 prefix list:

```
config router prefix-list
    edit "281"
        config rule
            edit 1
                set prefix 172.28.1.0 255.255.255.0
                unset ge
                unset le
            next
        end
    next
    edit "222"
        config rule
            edit 1
                set prefix 172.22.2.0 255.255.255.0
                unset ge
                unset le
            next
        end
    next
end
```

2. Configure the community list:

```
config router community-list
    edit "30:5"
        config rule
            edit 1
                set action permit
                set match "30:5"
            next
        end
    next
end
```

3. Configure the IPv4 route maps:

```
config router route-map
    edit "comm1"
        config rule
            edit 1
                set match-community "30:5"
```

```
                set set-route-tag 15
            next
        end
    next
    edit "2224"
        config rule
            edit 1
                set match-ip-address "222"
            next
        end
    next
    edit "2814"
        config rule
            edit 1
                set match-ip-address "281"
            next
        end
    next
end
```

4. Configure the IPv6 prefix list:

```
config router prefix-list6
    edit "adv-222"
        config rule
            edit 1
                set prefix6 2003:172:22:1::/64
                unset ge
                unset le
            next
        end
    next
    edit "list6-2"
        config rule
            edit 1
                set prefix6 2003:172:28:2::/64
                unset ge
                unset le
            next
        end
    next
end
```

5. Configure the IPv6 route maps:

```
config router route-map
    edit "map-222"
        config rule
            edit 1
                set match-ip6-address "adv-222"
            next
        end
    next
    edit "map-282"
        config rule
            edit 1
                set action deny
```

```
                set match-ip6-address "list6-2"
            next
        end
    next
end
```

6. Configure the BGP settings:

```
config router bgp
    config neighbor
        edit "2.2.2.2"
            config conditional-advertise
                edit "2224"
                    set condition-routemap "2814" "2224" "comm1"
                    set condition-type non-exist
                next
            end
        next
        edit "2003::2:2:2:2"
            config conditional-advertise6
                edit "map-222"
                    set condition-routemap "map-222" "map-282"
                next
                end
            set route-reflector-client6 enable
        next
    end
end
```

To verify the IPv4 conditional advertisements:

```
# get router info bgp neighbors 2.2.2.2
...
  Conditional advertise-map:
        Adv-map 2224root 2814root, cond-state 0-1
                         2224root, cond-state 0-1
                        comm1root, cond-state 0-0
...
```

In this output, the condition is that the routes in route maps 2814, 2224 and comm1 do not exist. However, routes for 2814 and 2224 exist, so the conditions are not met.

To verify the IPv6 conditional advertisements:

```
# get router info6 bgp neighbors 2003::2:2:2:2
...
  Conditional advertise-map:
        Adv-map map-222root map-222root, cond-state 1-1
                            map-282root, cond-state 1-0
...
```

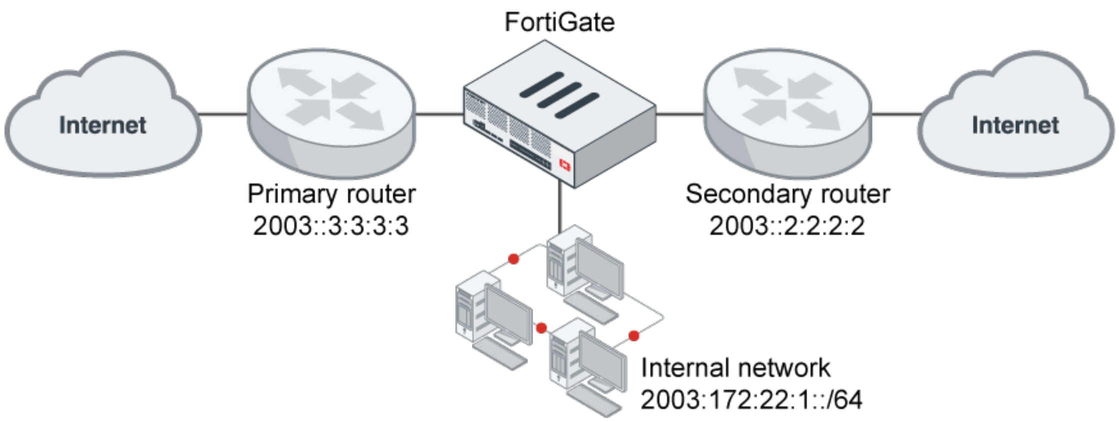
In this output, the condition is that the routes in route maps map-222 and map-282 exist. However, routes for map-222 exist, but map-282 does not, so the conditions are not met.

To view the conditional route maps:

```
# diagnose ip router command show-vrf root show running router bgp
...
 neighbor 2.2.2.2 advertise-map 2224root exist-map 2814root
 neighbor 2.2.2.2 advertise-map 2224root exist-map 2224root
 neighbor 2.2.2.2 advertise-map 2224root exist-map comm1root
... ...
 !
 address-family ipv6
 neighbor 2003::2:2:2:2 advertise-map map-222root non-exist-map map-222root
 neighbor 2003::2:2:2:2 advertise-map map-222root non-exist-map map-282root
!
```

IPv6 example 1

In this example, the FortiGate advertises its local network to the secondary router when the primary router is down. The FortiGate detects the primary router is down in the absence of a learned route.



- When the FortiGate learns route 2003:172:28:1::/64 from the primary router, it does not advertise its local route (2003:172:22:1::/64) to the secondary router.
- When the FortiGate does not learn route 2003:17:28:1::/64 from the primary router, advertises its local route (2003:172:22:1::/64) to the secondary router.
- The BGP conditional advertisement condition is set to be true if the condition route map (2003:172:28:1::/64) is not matched (`non-exist`).

To configure BGP conditional advertisement with IPv6:

1. Configure the IPv6 prefix lists:

```
config router prefix-list6
    edit "adv-222"
        config rule
            edit 1
                set prefix6 2003:172:22:1::/64
                unset ge
                unset le
            next
        end
    next
```

```
    edit "lrn-281"
        config rule
            edit 1
                set prefix6 2003:172:28:1::/64
                unset ge
                unset le
            next
        end
    next
end
```

2. Configure the route maps:

```
config router route-map
    edit "map-221"
        config rule
            edit 1
                set match-ip6-address "adv-222"
            next
        end
    next
    edit "map-281"
        config rule
            edit 1
                set match-ip6-address "lrn-281"
            next
        end
    next
end
```

3. Configure BGP:

```
config router bgp
    set as 65412
    set router-id 1.1.1.1
    set ibgp-multipath enable
    set network-import-check disable
    set graceful-restart enable
    config neighbor
        edit "2003::2:2:2:2"
            set soft-reconfiguration6 enable
            set remote-as 65412
            set update-source "loopback1"
            config conditional-advertise6
                edit "map-221"
                    set condition-routemap "map-281"
                    set condition-type non-exist
                next
            end
        next
        edit "2003::3:3:3:3"
            set soft-reconfiguration6 enable
            set remote-as 65412
            set update-source "loopback1"
        next
    end
end
```

In this configuration, if route map `map-281` does not exist, then the FortiGate advertises route map `map-221` to neighbor 2003::2:2:2:2.

4. Verify the routing table:

```
# get router info6 routing-table bgp
B       2003:172:28:1::/64 [200/0] via 2003::3:3:3:3 (recursive via
****::***:***:****:****, port9), 01:23:45
B       2003:172:28:2::/64 [200/0] via 2003::3:3:3:3 (recursive via
****::***:***:****:****, port9), 23:09:22
```

When the FortiGate learns 2003:172:28:1::/64, it will not advertise its local route 2003:172:22:1::/64 to neighbor 2003::2:2:2:2. If the FortiGate has not learned 2003:172:28:1::/64, it will advertise its local route 2003:172:22:1::/64 to neighbor 2003::2:2:2:2.

IPv6 example 2

With the same IPv6 prefix lists and route maps, when the FortiGate does learn 2003:172:28:1::/64, it advertises local route 2003:172:22:1::/64 to the secondary router. The BGP conditional advertisement condition is set to be true if the condition route map is matched (`exist`).

To configure BGP conditional advertisement with IPv6:

1. Configure BGP:

```
config router bgp
    config neighbor
        edit "2003::2:2:2:2"
            config conditional-advertise6
                edit "map-221"
                    set condition-routemap "map-281"
                    set condition-type exist
                next
            end
        next
    end
end
```

2. Verify the routing table:

```
# get router info6 routing-table bgp
B       2003:172:28:1::/64 [200/0] via 2003::3:3:3:3 (recursive via
****::***:***:****:****, port9), 01:23:45
B       2003:172:28:2::/64 [200/0] via 2003::3:3:3:3 (recursive via
****::***:***:****:****, port9), 23:09:22
```

When the FortiGate learns 2003:172:28:1::/64, it will advertise its local route 2003:172:22:1::/64 to neighbor 2003::2:2:2:2. If the FortiGate has not learned route 2003:172:28:1::/64, it will not advertise its local route 2003:172:22:1::/64 to neighbor 2003::2:2:2:2.

BGP error handling per RFC 7606

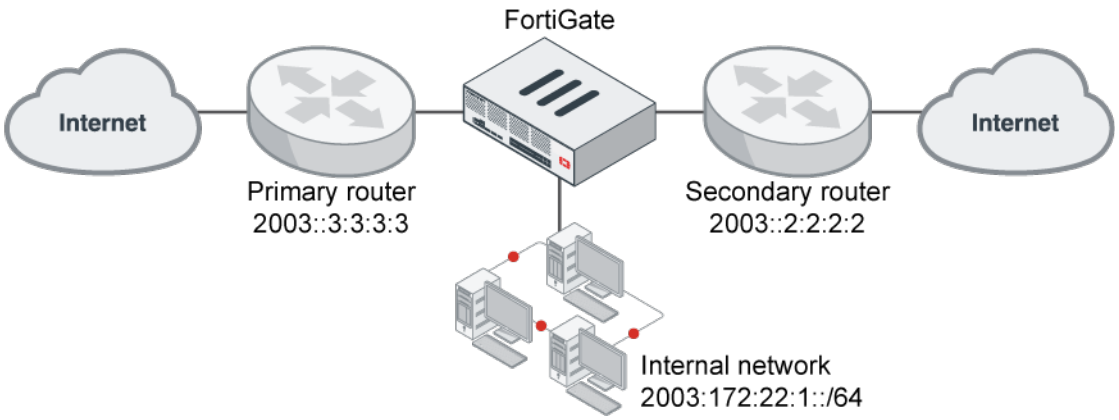
The FortiGate uses one of the three approaches to handle malformed attributes in BGP UPDATE messages, in order of decreasing severity:

1. Notification and Session reset
2. Treat-as-withdraw
3. Attribute discard

When a BGP UPDATE message contains multiple malformed attributes, the most severe approach that is triggered by one of the attributes is followed. See RFC 7606 for more information.

The following table lists the BGP attributes, and how FortiGate handles a malformed attribute in the UPDATE message:

BGP attribute	Handling
origin	Handled by the treat-as-withdraw approach.
AS path	Handled by the treat-as-withdraw approach.
AS 4 path	Handled by the attribute discard approach.
aggregator	Handled by the attribute discard approach.
aggregator 4	Handled by the attribute discard approach.
next-hop	Handled by the treat-as-withdraw approach.
multiple exit discriminator	Handled by the treat-as-withdraw approach.
local preference	Handled by the treat-as-withdraw approach.
atomic aggregate	Handled by the attribute discard approach.
community	Handled by the treat-as-withdraw approach.
extended community	Handled by the treat-as-withdraw approach.
originator	Handled by the treat-as-withdraw approach.
cluster	Handled by the treat-as-withdraw approach.
PMSI	Handled by the treat-as-withdraw approach.
MP reach	Handled by the notification message approach.
MP unreach	Handled by the notification message approach.
attribute set	Handled by the treat-as-withdraw approach.
AIGP	Handled by the treat-as-withdraw approach.
Unknown	If the BGP flag does not indicate that this is an optional attribute, this malformed attribute is handled by the notification message approach.



This example shows how the ORIGIN attribute can be malformed, and how it is handled.

Reason for malformed attribute	Handling
ORIGIN attribute length not one	The prefix will be gone and the BGP session will not be reset.
ORIGIN attribute value is invalid	The prefix will be gone and the BGP session will not be reset.
Two ORIGIN attributes with different values	The attributes are ignored, the BGP session will not be reset, and the BGP route will remain.
ORIGIN attribute is absent	The BGP session will be reset

For example, if the FortiGate receives a malformed UPDATE packet from the neighbor at 27.1.1.124 that has no ORIGIN attribute, the BGP session is reset and the state of the neighbor is shown as `Idle`, the first state of the BGP neighborship connection.

```
# get router info bgp summary
VRF 0 BGP router identifier 27.1.1.125, local AS number 125
BGP table version is 6
1 BGP AS-PATH entries
0 BGP community entries

Neighbor      V         AS  MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
3.3.3.3         4         33        0        0        0    0    0    never Active
27.1.1.124    4        124       94      126        0    0    0    never Idle

Total number of neighbors 2
```

BGP next hop tag-match mode

Tag-match mode can be configured to increase flexibility when controlling how BGP routes' next hops are resolved:

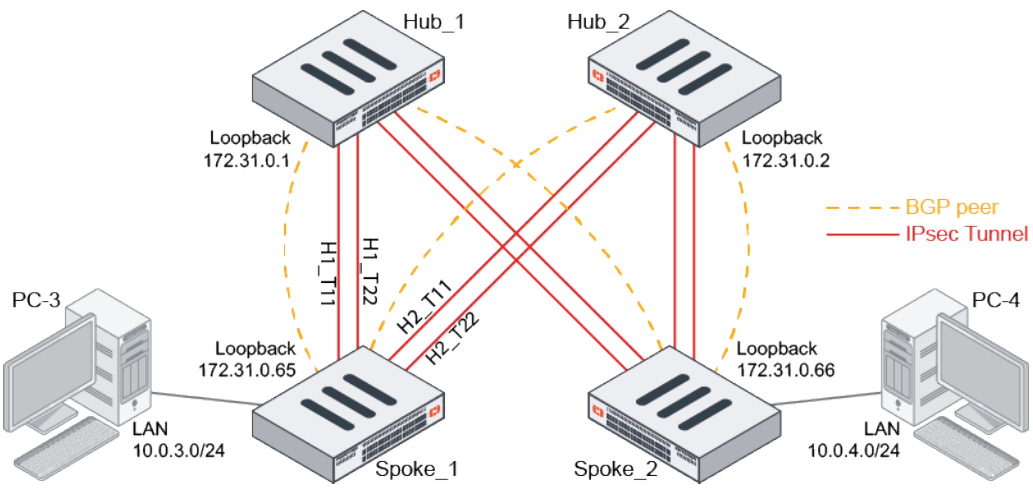
```
config router bgp
    set tag-resolve-mode {disable | preferred | merge}
end
```

Best-match (`disable`)	Resolve the BGP route's next hops with best-matched routes. This is the default setting.

Tag-match (`preferred`)	Resolve the BGP route's next hops with routes that have the same tag. If there are no results, resolve the next hops with best-matched routes.
Tag-and-best-match (`merge`)	Merge tag-match with best-match if they are using different routes, then let shortcuts hide their parents. The results exclude the next hops of tag-match whose interfaces have appeared in best-match.

In these examples:

- Each spoke has two IPsec tunnels to each hub, and one BGP peer on loopback interface to each hub (route-reflector).
- The loopbacks are exchanged with IKE between the spokes and hubs. They are installed as static routes that are used to provide reachability for establishing BGP neighbors.
- The summary BGP routes from the loopback IP address ranges that originated on the hubs are advertised to the spokes for resolving the BGP next hop s on the spokes.
- The spokes' PC LAN subnets are reflected by the hubs.
- Spoke_1 receives BGP routes (the LAN subnet and loopback IP summary) from Hub_1 with tag 1 and from Hub_2 with tag 2.
- SD-WAN is enabled on Spoke_1, and all of the tunnels are SD-WAN members.



Example 1: Connection between Hub and Spoke down

If the connections between Hub_1 and Spoke_2 are down, traffic from PC_3 to PC_4 can still go through Hub_1 because of the best-match resolving on Spoke_1, but packets will be dropped on Hub_1. When tag-match is enabled on Spoke_1, the spoke will resolve the PC_4 LAN route to Hub2, and traffic will be forwarded to Hub_2 and reach its destination.

To test the tag-match mode:

1. View the key routes on Spoke_1:

```
Spoke_1(root) # get router info routing-table all
C       10.0.3.0/24 is directly connected, port4
B       10.0.4.0/24 [200/0] via 172.31.0.66 [2] (recursive via H1_T11 tunnel
172.31.1.1), 20:09:52
                (recursive via H1_T22 tunnel 10.0.0.2), 20:09:52
```

```
              (recursive via H2_T11 tunnel 172.31.1.101), 20:09:52
              (recursive via H2_T22 tunnel 10.0.0.4), 20:09:52
B       172.31.0.0/25 [200/0] via 172.31.0.1 (recursive via H1_T11 tunnel 172.31.1.1),
23:25:37
              (recursive via H1_T22 tunnel 10.0.0.2), 23:25:37
            [200/0] via 172.31.0.2 (recursive via H2_T11 tunnel 172.31.1.101), 23:25:37
              (recursive via H2_T22 tunnel 10.0.0.4), 23:25:37
S       172.31.0.1/32 [15/0] via H1_T11 tunnel 172.31.1.1, [1/0]
            [15/0] via H1_T22 tunnel 10.0.0.2, [1/0]
S       172.31.0.2/32 [15/0] via H2_T11 tunnel 172.31.1.101, [1/0]
            [15/0] via H2_T22 tunnel 10.0.0.4, [1/0]
C       172.31.0.65/32 is directly connected, Loopback0
...
```

172.31.0.0/25 is the loopback IP summary originated by both Hub_1 and Hub_2. The next hop of the PC_4 LAN route is resolved to Hub_1 (H1_T11, H1_T22) and Hub_2 (H2_T11, H2_T22) based on the loopback IP summary route.

2. When connections between Spoke_2 and Hub_1 fails due to the BGP neighbor, tunnels, or physical ports going down, the PC_4 LAN route can be still resolved to Hub_1 and Hub_2 because the loopback IP summary can still be received from both Hub_1 and Hub_2:

```
Spoke_1(root) # get router info routing-table all
C       10.0.3.0/24 is directly connected, port4
B       10.0.4.0/24 [200/0] via 172.31.0.66 (recursive via H1_T11 tunnel 172.31.1.1),
00:03:06
              (recursive via H1_T22 tunnel 10.0.0.2), 00:03:06
              (recursive via H2_T11 tunnel 172.31.1.101), 00:03:06
              (recursive via H2_T22 tunnel 10.0.0.4), 00:03:06
B       172.31.0.0/25 [200/0] via 172.31.0.1 (recursive via H1_T11 tunnel 172.31.1.1),
23:55:34
              (recursive via H1_T22 tunnel 10.0.0.2), 23:55:34
            [200/0] via 172.31.0.2 (recursive via H2_T11 tunnel 172.31.1.101), 23:55:34
              (recursive via H2_T22 tunnel 10.0.0.4), 23:55:34
...
```

3. If traffic sent from PC_3 to PC_4 goes through Hub_1, packets are dropped because there is no PC_4 LAN route on Hub_1:

```
Spoke_1 (root) # diagnose sniffer packet any 'host 10.0.4.2' 4
interfaces=[any]
filters=[host 10.0.4.2]
11.261264 port4 in 10.0.3.2 -> 10.0.4.2: icmp: echo request
11.261349 H1_T11 out 10.0.3.2 -> 10.0.4.2: icmp: echo request
12.260268 port4 in 10.0.3.2 -> 10.0.4.2: icmp: echo request
12.260291 H1_T11 out 10.0.3.2 -> 10.0.4.2: icmp: echo request

Hub_1 (root) #  diagnose sniffer packet any 'host 10.0.4.2' 4
interfaces=[any]
filters=[host 10.0.4.2]
6.966064 EDGE_T1 in 10.0.3.2 -> 10.0.4.2: icmp: echo request
7.965012 EDGE_T1 in 10.0.3.2 -> 10.0.4.2: icmp: echo request
```

4. If the tag-match mode is set to tag-match (`preferred`) on Spoke_1, then the PC_4 LAN route can only be resolved to Hub_2 because of tag-match checking:

```
Spoke_1(root) # get router info routing-table all
C       10.0.3.0/24 is directly connected, port4
```

```
B       10.0.4.0/24 [200/0] via 172.31.0.66 tag 2 (recursive via H2_T11 tunnel 
172.31.1.101), 00:02:35
              (recursive via H2_T22 tunnel 10.0.0.4), 00:02:35
B       172.31.0.0/25 [200/0] via 172.31.0.1 tag 1 (recursive via H1_T11 tunnel 
172.31.1.1), 03:18:41
              (recursive via H1_T22 tunnel 10.0.0.2), 03:18:41
            [200/0] via 172.31.0.2 tag 2 (recursive via H2_T11 tunnel 172.31.1.101), 
03:18:41
              (recursive via H2_T22 tunnel 10.0.0.4), 03:18:41
...

Spoke_1 (root) # get router info routing-table details 10.0.4.0/24

Routing table for VRF=0
Routing entry for 10.0.4.0/24
  Known via "bgp", distance 200, metric 0, best
  Last update 00:01:11 ago
  * 172.31.0.66, tag 2 (recursive via H2_T11 tunnel 172.31.1.101), tag-match
              (recursive via H2_T22 tunnel 10.0.0.4), tag-match
```

5. If traffic is again sent from PC_3 to PC_4, it will go through Hub_2 and reach the destination:

```
Spoke_1 (root) # diagnose sniffer packet any 'host 10.0.4.2' 4
interfaces=[any]
filters=[host 10.0.4.2]
7.216948 port4 in 10.0.3.2 -> 10.0.4.2: icmp: echo request
7.217035 H2_T11 out 10.0.3.2 -> 10.0.4.2: icmp: echo request
7.217682 H2_T11 in 10.0.4.2 -> 10.0.3.2: icmp: echo reply
7.217729 port4 out 10.0.4.2 -> 10.0.3.2: icmp: echo reply
```

Example 2: SD-WAN failover when shortcut down

After the shortcut from Spoke_1 to Spoke_2 is established, Spoke_1 will only resolve the PC_4 LAN route to the shortcut, because of best-match resolving, prohibiting SD-WAN failover. When tag-and-best-match is enabled on Spoke_1, the spoke can resolve the PC_4 LAN route to the shortcut and to other alternative tunnels, allowing SD-WAN failover.

To test the tag-and-best-match mode:

1. Unset `tag-resolve-mode` and resume the connections between Spoke_2 and Hub_1. The routing table on Spoke_1 changes to the initial state:

```
Spoke_1(root) # get router info routing-table all
C       10.0.3.0/24 is directly connected, port4
B       10.0.4.0/24 [200/0] via 172.31.0.66 [2] (recursive via H1_T11 tunnel 
172.31.1.1), 00:01:54
              (recursive via H1_T22 tunnel 10.0.0.2), 00:01:54
              (recursive via H2_T11 tunnel 172.31.1.101), 00:01:54
              (recursive via H2_T22 tunnel 10.0.0.4), 00:01:54
B       172.31.0.0/25 [200/0] via 172.31.0.1 (recursive via H1_T11 tunnel 172.31.1.1), 
03:30:35
              (recursive via H1_T22 tunnel 10.0.0.2), 03:30:35
            [200/0] via 172.31.0.2 (recursive via H2_T11 tunnel 172.31.1.101), 03:30:35
              (recursive via H2_T22 tunnel 10.0.0.4), 03:30:35
S       172.31.0.1/32 [15/0] via H1_T11 tunnel 172.31.1.1, [1/0]
            [15/0] via H1_T22 tunnel 10.0.0.2, [1/0]
```

```
S       172.31.0.2/32 [15/0] via H2_T11 tunnel 172.31.1.101, [1/0]
            [15/0] via H2_T22 tunnel 10.0.0.4, [1/0]
C       172.31.0.65/32 is directly connected, Loopback0

...
```

2. Send traffic from PC_3 to PC_4.

   The shortcut from Spoke_1 o Spoke_2 is established.

   The PC_4 LAN route is only resolved to the shortcut because of best-match resolving. If the shortcut is out of SLA, then the traffic cannot switch over to another, alternative tunnel.

```
Spoke_1 (root) # get router info routing-table all
C       10.0.3.0/24 is directly connected, port4
B       10.0.4.0/24 [200/0] via 172.31.0.66 [2] (recursive via H1_T11_0 tunnel
10.0.0.40), 00:09:22
B       172.31.0.0/25 [200/0] via 172.31.0.1 (recursive via H1_T11 tunnel 172.31.1.1),
03:40:12
              (recursive via H1_T22 tunnel 10.0.0.2), 03:40:12
            [200/0] via 172.31.0.2 (recursive via H2_T11 tunnel 172.31.1.101), 03:40:12
              (recursive via H2_T22 tunnel 10.0.0.4), 03:40:12
S       172.31.0.1/32 [15/0] via H1_T11 tunnel 172.31.1.1, [1/0]
            [15/0] via H1_T22 tunnel 10.0.0.2, [1/0]
S       172.31.0.2/32 [15/0] via H2_T11 tunnel 172.31.1.101, [1/0]
            [15/0] via H2_T22 tunnel 10.0.0.4, [1/0]
C       172.31.0.65/32 is directly connected, Loopback0
S       172.31.0.66/32 [15/0] via H1_T11_0 tunnel 10.0.0.40, [1/0]
...
```

3. If the tag-match mode is set to tag-and-best-match (`merge`) on Spoke_1, then the PC_4 LAN route is resolved to the `H1_T11_0` shortcut based on best-match resolving, and to `H1_T11, H1_T22, H2_T11, H2_T22` based on tag-match resolving. It is then resolved to `H1_T11, H1_T22, H2_T11, H2_T22` after letting the shortcut hide its parent tunnel.

```
Spoke_1 (root) # get router info routing-table all
C       10.0.3.0/24 is directly connected, port4
B       10.0.4.0/24 [200/0] via 172.31.0.66 tag 1 (recursive via H1_T11_0 tunnel
10.0.0.40), 00:07:36
              (recursive via H1_T22 tunnel 10.0.0.2), 00:07:36
            [200/0] via 172.31.0.66 tag 2 (recursive via H1_T11_0 tunnel 10.0.0.40),
00:07:36
              (recursive via H2_T11 tunnel 172.31.1.101), 00:07:36
              (recursive via H2_T22 tunnel 10.0.0.4), 00:07:36
B       172.31.0.0/25 [200/0] via 172.31.0.1 tag 1 (recursive via H1_T11 tunnel
172.31.1.1), 03:48:26
              (recursive via H1_T22 tunnel 10.0.0.2), 03:48:26
            [200/0] via 172.31.0.2 tag 2 (recursive via H2_T11 tunnel 172.31.1.101),
03:48:26
              (recursive via H2_T22 tunnel 10.0.0.4), 03:48:26
S       172.31.0.1/32 [15/0] via H1_T11 tunnel 172.31.1.1, [1/0]
            [15/0] via H1_T22 tunnel 10.0.0.2, [1/0]
S       172.31.0.2/32 [15/0] via H2_T11 tunnel 172.31.1.101, [1/0]
            [15/0] via H2_T22 tunnel 10.0.0.4, [1/0]
C       172.31.0.65/32 is directly connected, Loopback0
S       172.31.0.66/32 [15/0] via H1_T11_0 tunnel 10.0.0.40, [1/0]

...
```

```
Spoke_1 (root) # get router info routing-table details 10.0.4.0/24

Routing table for VRF=0
Routing entry for 10.0.4.0/24
  Known via "bgp", distance 200, metric 0, best
  Last update 00:01:02 ago
  * 172.31.0.66, tag 1 (recursive via H1_T11_0 tunnel 10.0.0.42), best-match
              (recursive via H1_T22 tunnel 10.0.0.2), tag-match
  * 172.31.0.66, tag 2 (recursive via H1_T11_0 tunnel 10.0.0.42), best-match
              (recursive via H2_T11 tunnel 172.31.1.101), tag-match
              (recursive via H2_T22 tunnel 10.0.0.4), tag-match
```

4. If the `H1_T11_0` shortcut goes out of SLA, traffic will switch to tunnel `H1_T22` and shortcut `H1_T22_0` is triggered. The PC_4 LAN route is resolved to `H1_T11, H1_T22, H2_T11, H2_T22`.

```
Spoke_1 (root) # get router info routing-table all
C       10.0.3.0/24 is directly connected, port4
B       10.0.4.0/24 [200/0] via 172.31.0.66 tag 1 (recursive via H1_T11_0 tunnel
10.0.0.40), 00:18:50
              (recursive via H1_T22_0 tunnel 10.0.0.41), 00:18:50
            [200/0] via 172.31.0.66 tag 2 (recursive via H1_T11_0 tunnel 10.0.0.40),
00:18:50
              (recursive via H1_T22_0 tunnel 10.0.0.41), 00:18:50
              (recursive via H2_T11 tunnel 172.31.1.101), 00:18:50
              (recursive via H2_T22 tunnel 10.0.0.4), 00:18:50
B       172.31.0.0/25 [200/0] via 172.31.0.1 tag 1 (recursive via H1_T11 tunnel
172.31.1.1), 03:59:40
              (recursive via H1_T22 tunnel 10.0.0.2), 03:59:40
            [200/0] via 172.31.0.2 tag 2 (recursive via H2_T11 tunnel 172.31.1.101),
03:59:40
              (recursive via H2_T22 tunnel 10.0.0.4), 03:59:40
S       172.31.0.1/32 [15/0] via H1_T11 tunnel 172.31.1.1, [1/0]
            [15/0] via H1_T22 tunnel 10.0.0.2, [1/0]
S       172.31.0.2/32 [15/0] via H2_T11 tunnel 172.31.1.101, [1/0]
            [15/0] via H2_T22 tunnel 10.0.0.4, [1/0]
C       172.31.0.65/32 is directly connected, Loopback0
S       172.31.0.66/32 [15/0] via H1_T11_0 tunnel 10.0.0.40, [1/0]
            [15/0] via H1_T22_0 tunnel 10.0.0.41, [1/0]
...
```

```
Spoke_1 (root) # get router info routing-table details 10.0.4.0/24

Routing table for VRF=0
Routing entry for 10.0.4.0/24
  Known via "bgp", distance 200, metric 0, best
  Last update 00:06:40 ago
  * 172.31.0.66, tag 1 (recursive via H1_T11_0 tunnel 10.0.0.42), best-match
              (recursive via H1_T22_0 tunnel 10.0.0.43), best-match
  * 172.31.0.66, tag 2 (recursive via H1_T11_0 tunnel 10.0.0.42), best-match
              (recursive via H1_T22_0 tunnel 10.0.0.43), best-match
              (recursive via H2_T11 tunnel 172.31.1.101), tag-match
              (recursive via H2_T22 tunnel 10.0.0.4), tag-match
```

```
Spoke_1(root) # diagnose sys sdwan service

Service(1): Address Mode(IPV4) flags=0x200 use-shortcut-sla
  Gen(22), TOS(0x0/0x0), Protocol(0: 1->65535), Mode(sla), sla-compare-order
```

```
    Member sub interface(4):
      1: seq_num(1), interface(H1_T11):
         1: H1_T11_0(93)
      3: seq_num(4), interface(H1_T22):
         1: H1_T22_0(94)
    Members(4):
      1: Seq_num(1 H1_T11), alive, sla(0x1), gid(0), cfg_order(0), cost(0), selected
      2: Seq_num(4 H1_T22_0), alive, sla(0x1), gid(0), cfg_order(3), cost(0), selected
      3: Seq_num(4 H1_T22), alive, sla(0x1), gid(0), cfg_order(3), cost(0), selected
      4: Seq_num(1 H1_T11_0), alive, sla(0x0), gid(0), cfg_order(0), cost(0), selected
    Src address(1):
          10.0.0.0-10.255.255.255
    Dst address(1):
          10.0.0.0-10.255.255.255

  Service(2): Address Mode(IPV4) flags=0x200 use-shortcut-sla
    Gen(10), TOS(0x0/0x0), Protocol(0: 1->65535), Mode(sla), sla-compare-order
    Members(2):
      1: Seq_num(6 H2_T11), alive, sla(0x1), gid(0), cfg_order(0), cost(0), selected
      2: Seq_num(9 H2_T22), alive, sla(0x1), gid(0), cfg_order(3), cost(0), selected
    Src address(1):
          10.0.0.0-10.255.255.255
    Dst address(1):
          10.0.0.0-10.255.255.255
```

Troubleshooting BGP

There are some features in BGP that are used to deal with problems that may arise. Typically, the problems with a BGP network that has been configured involve routes going offline frequently. This is called route flap and causes problems for the routers using that route.

Clearing routing table entries

To see if a new route is being properly added to the routing table, you can clear all or some BGP neighbor connections (sessions) using the `execute router clear bgp` command.

For example, if you have 10 routes in the BGP routing table and you want to clear the specific route to IP address 10.10.10.1, enter the following CLI command:

```
# execute router clear bgp ip 10.10.10.1
```

To remove all routes for AS number 650001, enter the following CLI command:

```
# execute router clear bgp as 650001
```

Route flap

When routers or hardware along a route go offline and back online that is called a route flap. Flapping is the term that is used if these outages continue, especially if they occur frequently.

Route flap is a problem in BGP because each time a peer or a route goes down, all the peer routers that are connected to that out-of-service router advertise the change in their routing tables. This creates a lot of administration traffic on the network and the same traffic re-occurs when that router comes back online. If the problem is something like a faulty network cable that alternates online and offline every 10 seconds, there could easily be an overwhelming amount of routing updates sent out unnecessarily.

Another possible reason for route flap occurs with multiple FortiGate devices in HA mode. When an HA cluster fails over to the secondary unit, other routers on the network may see the HA cluster as being offline, resulting in route flap. While this doesn't occur often, or more than once at a time, it can still result in an interruption in traffic which is disruptive for network users. The easy solution for this problem is to increase the timers on the HA cluster, such as TTL timers, so they don't expire during the failover process. Also, configuring graceful restart on the HA cluster helps with a smooth failover.

The first method of dealing with route flap is to check your hardware. If a cable is loose or bad, it can easily be replaced and eliminate the problem. If an interface on the router is bad, either avoid using that interface or swap in a functioning router. If the power source is bad on a router, either replace the power supply or use a power conditioning backup power supply. These quick and easy fixes can save you from configuring more complex BGP options. However, if the route flap is from another source, configuring BGP to deal with the outages will ensure your network users uninterrupted service.

Some methods of dealing with route flap in BGP include:

- Holdtime timer on page 385
- Dampening on page 386
- Graceful restart on page 386
- BFD on page 388

Holdtime timer

The first step to troubleshooting a flapping route is the holdtime timer. This timer reduces how frequently a route going down will cause a routing update to be broadcast.

Once activated, the holdtime timer won't allow the FortiGate to accept any changes to that route for the duration of the timer. If the route flaps five times during the timer period, only the first outage will be recognized by the FortiGate. For the duration of the other outages, there won't be changes because the Fortigate is essentially treating this router as down. If the route is still flapping after the timer expires, it will start again.

If the route isn't flapping (for example, if it goes down, comes up, and stays back up) the timer will still count down and the route is ignored for the duration of the timer. In this situation, the route is seen as down longer than it really is but there will be only the one set of route updates. This isn't a problem in normal operation because updates are not frequent.

The potential for a route to be treated as down when it's really up can be viewed as a robustness feature. Typically, you don't want most of your traffic being routed over an unreliable route. So if there's route flap going on, it's best to avoid that route if you can. This is enforced by the holdtime timer.

How to configure the holdtime timer

There are three different route flapping situations that can occur: the route goes up and down frequently, the route goes down and back up once over a long period of time, or the route goes down and stays down for a long period of time. These can all be handled using the holdtime timer.

For example, your network has two routes that you want to set the timer for. One is your main route (to 10.12.101.4) that all of your Internet traffic goes through, and it can't be down for long if it's down. The second is a low speed connection to a custom network that's used infrequently (to 10.13.101.4). The timer for the main route should be fairly short (for example, 60 seconds). The second route timer can be left at the default, since it's rarely used.

To configure the BGP holdtime timer:

```
config router bgp
    config neighbor
        edit 10.12.101.4
            set holdtime-timer 60
```

```
            set keepalive-timer 60
        next
        edit 10.13.101.4
            set holdtime-timer 180
            set keepalive-timer 60
        next
    end
end
```

Dampening

Dampening is a method that's used to limit the amount of network problems due to flapping routes. With dampening, the flapping still occurs but the peer routers pay less and less attention to that route as it flaps more often. One flap doesn't start dampening, but the second flap starts a timer where the router won't use that route because it is considered unstable. If the route flaps again before the timer expires, the timer continues to increase. There's a period of time called the reachability half-life, after which a route flap will be suppressed for only half the time. This half-life comes into effect when a route has been stable for a while but not long enough to clear all the dampening completely. For the flapping route to be included in the routing table again, the suppression time must expire.

If the route flapping was temporary, you can clear the flapping or dampening from the FortiGate device's cache by using one of the `execute router clear bgp` CLI commands:

```
# execute router clear bgp dampening {<ip_address> | <ip_address/netmask>}
```

or

```
# execute router clear bgp flap-statistics {<ip_address> | <ip_address/netmask>}
```

For example, to remove route flap dampening information for the 10.10.0.0/16 subnet, enter the following CLI command:

```
# execute router clear bgp dampening 10.10.0.0/16
```

To configure BGP route dampening:

```
config router bgp
    set dampening {enable | disable}
    set dampening-max-suppress-time <minutes_integer>
    set dampening-reachability-half-life <minutes_integer>
    set dampening-reuse <reuse_integer>
    set dampening-route-map <routemap-name_str>
    set dampening-suppress <limit_integer>
    set dampening-unreachability-half-life <minutes_integer>
end
```

Graceful restart

BGP4 has the capability to gracefully restart.

In some situations, route flap is caused by routers that appear to be offline but the hardware portion of the router (control plane) can continue to function normally. One example of this is when some software is restarting or being upgraded but the hardware can still function normally.

Graceful restart is best used for these situations where routing won't be interrupted, but the router is unresponsive to routing update advertisements. Graceful restart doesn't have to be supported by all routers in a network, but the network will benefit when more routers support it.

FortiGate HA clusters can benefit from graceful restart. When a failover takes place, the HA cluster advertises that it is going offline, and will not appear as a route flap. It will also enable the new HA main unit to come online with an updated and usable routing table. If there is a flap, the HA cluster routing table will be out-of-date.

For example, the FortiGate is one of four BGP routers that send updates to each other. Any of those routers may support graceful starting. When a router plans to go offline, it sends a message to its neighbors stating how long it expects to be offline. This way, its neighboring routers don't remove it from their routing tables. However, if that router isn't back online when expected, the routers will mark it offline. This prevents routing flap and its associated problems.

FortiGate devices support both graceful restart of their own BGP routing software and neighboring BGP routers.

To configure BGP graceful restart:

```
config router bgp
    set graceful-restart {disable | enable}
    set graceful-restart-time <seconds_integer>
    set graceful-stalepath-time <seconds_integer>
    set graceful-update-delay <seconds_integer>
    config neighbor
        edit 10.12.101.4
            set capability-graceful-restart {enable | disable}
        next
    end
end
```

Before the restart, the router sends its peers a message to say it's restarting. The peers mark all the restarting router's routes as stale, but they continue to use the routes. The peers assume the router will restart, check its routes, and take care of them, if needed, after the restart is complete. The peers also know what services the restarting router can maintain during its restart. After the router completes the restart, the router sends its peers a message to say it's done restarting.

To restart the router:

```
# execute router restart
```

Scheduled time offline

Graceful restart is a means for a router to advertise that it is going to have a scheduled shutdown for a very short period of time. When neighboring routers receive this notice, they will not remove that router from their routing table until after a set time elapses. During that time, if the router comes back online, everything continues to function as normal. If that router remains offline longer than expected, then the neighboring routers will update their routing tables as they assume that the router will be offline for a long time.

The following example demonstrates if you want to configure graceful restart on the FortiGate where you expect the FortiGate to be offline for no more than two minutes, and after three minutes the BGP network should consider the FortiGate to be offline.

To configure graceful restart time settings:

```
config router bgp
    set graceful-restart enable
    set graceful-restart-time 120
    set graceful-stalepath-time 180
end
```

BFD

Bidirectional Forwarding Detection (BFD) is a protocol that you can use to quickly locate hardware failures in the network. Routers running BFD communicate with each other and if a timer runs out on a connection then that router is declared down. BFD then communicates this information to the routing protocol and the routing information is updated.

For more information about BFD, see BFD on page 388.

BFD

Bidirectional Forwarding Detection (BFD) is a protocol that you can use to quickly locate hardware failures in the network. Routers running BFD send packets to each other at a negotiated rate. If packets from a BFD-enabled router fail to arrive, that router is declared to be down. BFD communicates this information to the associated routing protocols and the routing information is updated. It helps one way device failure and is used for fast convergence of routing protocols.

BFD can run on an entire FortiGate, selected interfaces, or on a protocol, such as BGP, for all configured interfaces. The configuration hierarchy allows each lower level to override the BFD setting of the upper level. For example, if you enable BFD for an entire FortiGate, you can disable BFD for an interface or for BGP.

Echo mode and authentication are not supported for BFD on the FortiGate.

BFD can be enabled per device, VDOM, or interface. Once enabled, a BFD neighbor should be defined. Finally, enable BFD on a route or routing protocol.

To configure BFD for an entire FortiGate:

```
config system settings
    set bfd {enable | disable}
    set bfd-desired-min-tx <ms>
    set bfd-required-min-rx <ms>
    set bfd-detect-mult <multiplier>
    set bfd-dont-enforce-src-port {enable | disable}
end
```

To configure BFD for an interface:

```
config system interface
    edit <interface-name>
        set bfd {global | enable | disable}
        set bfd-desired-min-tx <ms>
        set bfd-required-min-rx <ms>
        set bfd-detect-mult <multiplier>
    next
end
```

To configure BFD neigbors:

```
config router {bfd | bfd6}
    config neighbor
        edit <IP-address>
```

```
            set interface <interface-name>
        next
    end
end
```

To show BFD neighbors:

```
# get router {info | info6} bfd neighbor
```

To show BFD requests:

```
# get router {info | info6} bfd requests
```

BFD and static routes

BFD for static routes allows you to configure routing failover based on remote path failure detection. BFD removes a static route from the routing table if the FortiGate can't reach the route's destination and returns the route to the routing table if the route's destination is restored.

For example, you can add two static routes with BFD enabled. If one of the routes has a higher priority, all matching traffic uses that route. If BFD determines that the link to the gateway of the route with the higher priority is down, the higher priority route is removed from the routing table and all matching traffic uses the lower priority route. If the link to the gateway for the higher priority route comes back up, BFD adds the route back into the routing table and all matching traffic switches to use the higher priority route.

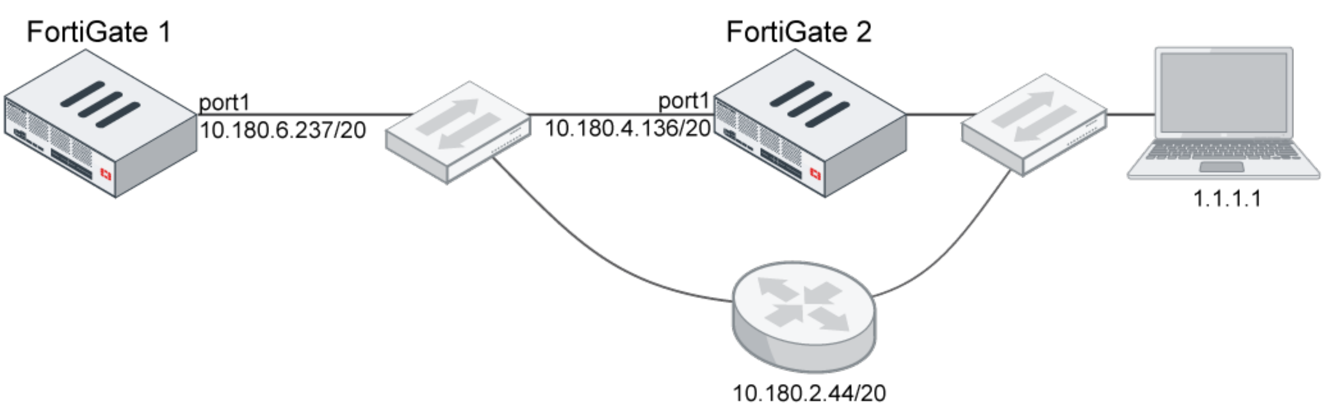
You can configure BFD for IPv4 and IPv6 static routes.

To configure BFD for static routes:

```
config router {static | static6}
    edit <sequence-number>
        set bfd {enable | disable}
        set device <gateway-out-interface>
    next
end
```

Example

The following example demonstrates the configuration of static routes between two FortiGates. There is a host behind FortiGate 2 with an IP address of 1.1.1.1. FortiGate 1 has multiple paths to reach the host.



To configure static routes:

1. Configure FortiGate 1:

```
config system interface
    edit "port1"
        set vdom "root"
        set ip 10.180.6.237 255.255.240.0
        set allowaccess ping
        set bfd enable
    next
end
config router bfd
    config neighbor
        edit 10.180.4.136
            set interface "port1"
        next
    end
end
```

2. Configure FortiGate 2:

```
config system interface
    edit "port1"
        set vdom "root"
        set ip 10.180.4.136 255.255.240.0
        set allowaccess ping
        set bfd enable
    next
end
config router bfd
    config neighbor
        edit 10.180.6.237
            set interface "port1"
        next
    end
end
```

3. Configure two static routes:

```
config router static
    edit 2
        set dst 1.1.1.1 255.255.255.255
        set gateway 10.180.4.136
        set device "port1"
        set bfd enable
    next
    edit 3
        set dst 1.1.1.1 255.255.255.255
        set gateway 10.180.2.44
        set distance 20
        set device "port1"
    next
end
```

4. Confirm that BFD neighborship is established:

```
# get router info bfd neighbor
OurAddress       NeighAddress     State        Interface       LDesc/RDesc
10.180.6.237     10.180.4.136     UP           port1           1/1
```

5. Review the active route in the routing table:

```
# get router info routing-table all
S       1.1.1.1/32 [10/0] via 10.180.4.136, port1
C       10.180.0.0/20 is directly connected, port1
```

The route with the lower distance is preferred in the routing table.

If port1 on FortiGate 2 goes down or FortiGate 1 is unable to reach 10.180.4.126, the BFD neighborship will go down.

```
# get router info bfd neighbor
OurAddress       NeighAddress     State        Interface       LDesc/RDesc
10.180.6.237     10.180.4.136     DOWN         port1           1/1
```

With BFD neighborship down, the FortiGate is unable to reach 1.1.1.1/32 through gateway 10.180.4.136. The routing table will be updated so that the route through gateway 10.180.2.44 is active in the routing table.

```
# get router info routing-table all
S       1.1.1.1/32 [20/0] via 10.180.2.44, port1
C       10.180.0.0/20 is directly connected, port1
```

BFD removes a static route from the routing table if the FortiGate cannot reach the route's destination. The static route will be returned to the routing table is the route's destination is restored.

BFD and OSPF

You can configure BFD for Open Shortest Path First (OSPF) on a FortiGate. FortiGate supports BFD for OSPF for both IPv4 and IPv6. BFD must be configured globally and per interface.

To configure BFD for OSPF:

```
config router {ospf | ospf6}
    set bfd {enable | disable}
end
```

To enable BFD on a specific OSPF interface:

```
config router {ospf | ospf6}
    set bfd enable
    config {ospf-interface | ospf6-interface}
        edit <ID>
            set bfd {global | enable | disable}
            set area-id <IP address>
        next
    end
end
```

If BFD is configured when OSPF is not, no BFD packets will be sent. When both BFD and OSFP are configured, the neighbors for both will be the same. Use the following commands to confirm that the neighbor IP addresses match:

```
# get router info ospf neighbor
# get router info bfd neighbor
```

BFD and BGP

While BGP can detect route failures, BFD can be configured to detect these failures more quickly, which allows for faster responses and improved convergence. This can be balanced with the bandwidth BFD uses in its frequent route checking.

The `config router bgp` commands allow you to set the addresses of the neighbor units that are also running BFD. Both units must be configured with BFD in order to use it.

To configure BFD for BGP:

```
config router bgp
    config neighbor
        edit <neighbor-IP-address>
            set bfd {enable | disable}
        next
    end
end
```

BFD for Multihop paths

FortiGate BFD can support neighbors connected over multiple hops. When BFD is down, BGP sessions will be reset and will try to re-establish neighbor connection immediately. See BFD for multihop path for BGP on page 393 for more information.

To configure BFD for multihop paths:

```
config router {bfd | bfd6}
    config multihop-template
        edit <ID>
            set src <IP address/netmask>
            set dst <IP address/netmask>
            set bfd-desired-min-tx <integer>
            set bfd-required-min-rx <integer>
            set bfd-detect-mult <integer>
            set auth-mode {none | md5}
            set md5-key <password>
        next
    end
end
```

Troubleshooting BFD

You can troubleshoot BFD using the following commands:

```
# get router {info | info6} bfd neighbor
# get router {info | info6} bfd requests
# diagnose sniffer packet any <filter> <sniffer count>
```

```
# diagnose debug application bfdd <debug level>
# diagnose debug enable
```

BFD for multihop path for BGP

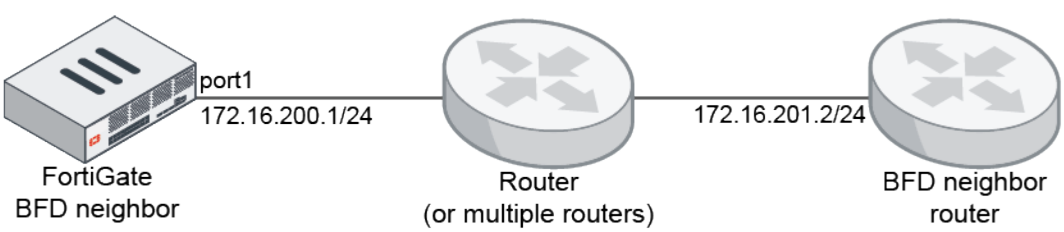
In BFD, a FortiGate can support neighbors connected over multiple hops. When BFD is down, BGP sessions are reset and will try to immediately re-establish neighbor connections. Previously, BFD was only supported when two routers or FortiGates were directly connected on the same network.

```
config router {bfd | bfd6}
    config multihop-template
        edit <ID>
            set src <class_IP/netmask>
            set dst <class_IP/netmask>
            set bfd-desired-min-tx <integer>
            set bfd-required-min-rx <integer>
            set bfd-detect-mult <integer>
            set auth-mode {none | md5}
            set md5-key <password>
        next
    end
end
```

`src <class_IP/netmask>`	Enter the source prefix.
`dst <class_IP/netmask>`	Enter the destination prefix.
`bfd-desired-min-tx <integer>`	Set the BFD desired minimal transmit interval, in milliseconds (100 - 30000, default = 250).
`bfd-required-min-rx <integer>`	Set the BFD required minimal transmit interval, in milliseconds (100 - 30000, default = 250).
`bfd-detect-mult <integer>`	Set the BFD detection multiplier (3 - 50, default = 3).
`auth-mode {none | md5}`	Set the authentication mode (none or meticulous MD5).
`md5-key <password>`	Enter the password.

Example

This example includes IPv4 and IPv6 BFD neighbor configurations. The BFD neighbor is also a BGP neighbor that is in a different AS.



To configure BFD with multihop BGP paths:

1. Enable BFD on all interfaces:

```
config system settings
    set bfd enable
end
```

2. Enable BFD on port1 and ignore the global configuration:

```
config system interface
    edit "port1"
        set bfd enable
    next
end
```

3. Configure the BGP neighbors:

```
config router bgp
    set as 65412
    set router-id 1.1.1.1
    config neighbor
        edit "172.16.201.2"
            set bfd enable
            set ebgp-enforce-multihop enable
            set soft-reconfiguration enable
            set remote-as 65050
        next
        edit "2000:172:16:201::2"
            set bfd enable
            set ebgp-enforce-multihop enable
            set soft-reconfiguration enable
            set remote-as 65050
        next
    end
end
```

4. Configure the IPv4 BFD:

```
config router bfd
    config multihop-template
        edit 1
            set src 172.16.200.0 255.255.255.0
            set dst 172.16.201.0 255.255.255.0
            set auth-mode md5
            set md5-key **********
        next
    end
end
```

5. Configure the IPv6 BFD:

```
config router bfd6
    config multihop-template
        edit 1
            set src 2000:172:16:200::/64
            set dst 2000:172:16:201::/64
        next
```

```
    end
end
```

Testing the connection

1. Verify the BFD status for IPv4 and IPv6:

```
# get router info bfd requests
BFD Peer Requests:
    client types(ct in 0x): 01=external 02=static
        04=ospf 08=bgp 10=pim-sm
src=172.16.200.1     dst=172.16.201.2     ct=08 ifi=9 type=SM
```

```
# get router info bfd neighbor
OurAddress      NeighAddress    State       Interface       LDesc/RDesc
172.16.200.1    172.16.201.2    UP          port1           5/3/M
```

```
# get router info6 bfd requests
BFD Peer Requests:
    client types(ct in 0x): 01=external 02=static
        04=ospf 08=bgp 10=pim-sm
src=2000:172:16:200::1
dst=2000:172:16:201::2
ct=08 ifi=9 type=SM
```

```
# get router info6 bfd neighbor
OurAddress: 2000:172:16:200::1
NeighAddress: 2000:172:16:201::2
State: UP Interface: port1 Desc: 6/4 Multi-hop
```

2. Verify the BGP status and the BGP routing table:

```
# get router info bgp summary
VRF 0 BGP router identifier 1.1.1.1, local AS number 65412
BGP table version is 11
3 BGP AS-PATH entries
0 BGP community entries

Neighbor           V         AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
172.16.201.2       4      65050     185     187       10    0    0 00:54:20        4
2000:172:16:201::2 4      65050     159     160       10    0    0 00:54:24        4

Total number of neighbors 2
```

```
# get router info routing-table bgp
Routing table for VRF=0
B       172.28.1.0/24 [20/0] via 172.16.201.2 (recursive via 172.16.200.4, port1),
00:54:32
B       172.28.2.0/24 [20/0] via 172.16.201.2 (recursive via 172.16.200.4, port1),
00:54:32
B       172.28.5.0/24 [20/0] via 172.16.201.2 (recursive via 172.16.200.4, port1),
00:54:32
B       172.28.6.0/24 [20/0] via 172.16.201.2 (recursive via 172.16.200.4, port1),
00:54:32
```

```
# get router info6 bgp summary
VRF 0 BGP router identifier 1.1.1.1, local AS number 65412
BGP table version is 8
3 BGP AS-PATH entries
```

```
0 BGP community entries

Neighbor          V         AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
172.16.201.2      4      65050     185     187        7    0    0 00:54:24        3
2000:172:16:201::2 4     65050     159     160        7    0    0 00:54:28        3

Total number of neighbors 2
```

```
# get router info6 routing-table bgp
Routing table for VRF=0
B       2000:172:28:1::/64 [20/0] via 2000:172:16:201::2 (recursive via
2000:172:16:200::4, port1), 00:54:40
B       2000:172:28:2::/64 [20/0] via 2000:172:16:201::2 (recursive via
2000:172:16:200::4, port1), 00:54:40
B       2000:172:28:3::/64 [20/0] via 2000:172:16:201::2 (recursive via
2000:172:16:200::4, port1), 00:54:40
```

3. Simulate a disruption to the BFD connection. The BFD neighbor is lost:

```
# get router info bfd neighbor
OurAddress      NeighAddress    State       Interface       LDesc/RDesc
```

```
# get router info6 bfd neighbor
```

4. The BGP neighbor is reset, and the FortiGate attempts to re-establish a connection with the neighbor. The timers are reset once the neighbor connection is re-established:

```
# get router info bgp summary
VRF 0 BGP router identifier 1.1.1.1, local AS number 65412
BGP table version is 12
4 BGP AS-PATH entries
0 BGP community entries

Neighbor          V         AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd

172.16.201.2      4      65050     189     192       11    0    0 00:00:11        4
2000:172:16:201::2 4     65050     165     167       12    0    0 00:00:08        4

Total number of neighbors 2
```

```
# get router info6 bgp summary
VRF 0 BGP router identifier 1.1.1.1, local AS number 65412
BGP table version is 10
4 BGP AS-PATH entries
0 BGP community entries

Neighbor          V         AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
172.16.201.2      4      65050     189     192        8    0    0 00:00:15        3
2000:172:16:201::2 4     65050     165     167        9    0    0 00:00:12        3

Total number of neighbors 2
```

5. The BGP routes are learned again, and there are new timers in the route tables:

```
# get router info routing-table bgp
Routing table for VRF=0
B       172.28.1.0/24 [20/0] via 172.16.201.2 (recursive via 172.16.200.4, port1),
00:00:15
B       172.28.2.0/24 [20/0] via 172.16.201.2 (recursive via 172.16.200.4, port1),
```

```
00:00:15
B       172.28.5.0/24 [20/0] via 172.16.201.2 (recursive via 172.16.200.4, port1),
00:00:15
B       172.28.6.0/24 [20/0] via 172.16.201.2 (recursive via 172.16.200.4, port1),
00:00:15
```

```
# get router info6 routing-table bgp
Routing table for VRF=0
B       2000:172:28:1::/64 [20/0] via 2000:172:16:201::2 (recursive via
2000:172:16:200::4, port1), 00:00:13
B       2000:172:28:2::/64 [20/0] via 2000:172:16:201::2 (recursive via
2000:172:16:200::4, port1), 00:00:13
B       2000:172:28:3::/64 [20/0] via 2000:172:16:201::2 (recursive via
2000:172:16:200::4, port1), 00:00:13
```

Multicast

The following topics include information about multicast:

- Multicast routing and PIM support on page 397
- Configuring multicast forwarding on page 398

Multicast routing and PIM support

Multicasting (also called IP multicasting) consists of using a single multicast source to send data to many receivers. Multicasting can be used to send data to many receivers simultaneously while conserving bandwidth and reducing network traffic. Multicasting can be used for one-way delivery of media streams to multiple receivers and for one-way data transmission for news feeds, financial information, and so on. Many dynamic routing protocols such as RIPv2, OSPF, and EIGRP use multicasting to share hello packets and routing information.

A FortiGate can operate as a Protocol Independent Multicast (PIM) version 2 router. FortiGates support PIM sparse mode (RFC 4601) and PIM dense mode (RFC 3973), and can service multicast servers or receivers on the network segment to which a FortiGate interface is connected. Multicast routing is not supported in transparent mode.

To support PIM communications, the sending and receiving applications, and all connecting PIM routers in between, must be enabled with PIM version 2. PIM can use static routes, RIP, OSPF, or BGP to forward multicast packets to their destinations. To enable source-to-destination packet delivery, sparse mode or dense mode must be enabled on the PIM router interfaces. Sparse mode routers cannot send multicast messages to dense mode routers. If the FortiGate is located between a source and a PIM router, between two PIM routers, or is connected directly to a receiver, you must manually create a multicast policy to pass encapsulated (multicast) packets or decapsulated data (IP traffic) between the source and destination.

PIM domains

A PIM domain is a logical area comprising a number of contiguous networks. The domain contains at least one bootstrap router (BSR), and if sparse mode is enabled, a number of rendezvous points (RPs) and designated routers (DRs). When PIM is enabled, the FortiGate can perform any of these functions at any time as configured.

A PIM domain can be configured in the GUI by going to *Network > Multicast*, or in the CLI using `config router multicast`. Note that PIM version 2 must be enabled on all participating routers between the source and receivers. Use `config router multicast` to set the global operating parameters.

When PIM is enabled, the FortiGate allocates memory to manage mapping information. The FortiGate communicates with neighboring PIM routers to acquire mapping information and, if required, processes the multicast traffic associated with specific multicast groups.

Instead of sending multiple copies of generated IP traffic to more than one specific IP destination address, PIM-enabled routers encapsulate the data and use a Class D multicast group address (224.0.0.0 to 239.255.255.255) to forward multicast packets to multiple destinations. A single stream of data can be sent because one destination address is used. Client applications receive multicast data by requesting that the traffic destined for a certain multicast group address be delivered to them.

Configuring multicast forwarding

There is sometimes confusion between the terms forwarding and routing. These two functions should not take place at the same time. Multicast forwarding should be enabled when the FortiGate is in NAT mode and you want to forward multicast packets between multicast routers and receivers. However, this function should not be enabled when the FortiGate itself is operating as a multicast router, or has an applicable routing protocol that uses multicast.

Multicast forwarding is not supported on enhanced MAC VLAN interfaces. To use multicast with enhanced MAC VLAN interfaces, use PIM (Multicast routing and PIM support on page 397).

There are two steps to configure multicast forwarding:

1. Enabling multicast forwarding on page 398
2. Configuring multicast policies on page 399

Enabling multicast forwarding

Multicast forwarding is enabled by default. If a FortiGate is operating in transparent mode, adding a multicast policy enables multicast forwarding. In NAT mode you must use the `multicast-forward` setting to enable or disable multicast forwarding.

Multicast forwarding in NAT mode

When `multicast-forward` is enabled, the FortiGate forwards any multicast IP packets in which the TTL is 2 or higher to all interfaces and VLAN interfaces, except the receiving interface. The TTL in the IP header will be reduced by 1. Even though the multicast packets are forwarded to all interfaces, you must add multicast policies to allow multicast packets through the FortiGate.

To enable multicast forwarding in NAT mode:

```
config system settings
    set multicast-forward enable
end
```

Prevent the TTL for forwarded packets from being changed

You can use the `multicast-ttl-notchange` option so that the FortiGate does not increase the TTL value for forwarded multicast packets. Use this option only if packets are expiring before reaching the multicast router.

To prevent the TTL for forwarded packets from being changed:

```
config system settings
    set multicast-ttl-notchange enable
end
```

Disable multicast traffic from passing through the FortiGate without a policy check in transparent mode

In transparent mode, the FortiGate does not forward frames with multicast destination addresses. The FortiGate should not interfere with the multicast traffic used by routing protocols, streaming media, or other multicast communication. To avoid any issues during transmission, you can disable `multicast-skip-policy` and configure multicast security policies.

To disable multicast traffic from passing through the FortiGate without a policy check in transparent mode:

```
config system settings
    set multicast-skip-policy disable
end
```

Configuring multicast policies

Multicast packets require multicast policies to allow packets to pass from one interface to another. Similar to firewall policies, in a multicast policy you specify the source and destination interfaces, and the allowed address ranges for the source and destination addresses of the packets. You can also use multicast policies to configure source NAT and destination NAT for multicast packets.

Keep the following in mind when configuring multicast policies:

- The matched forwarded (outgoing) IP multicast source IP address is changed to the configured IP address.
- The `snat` setting is optional. Use it when SNAT is needed.

IPv4 and IPv6 multicast policies can be configured in the GUI. Go to *System > Feature Visibility*, and enable *Multicast Policy* and *IPv6*.

Sample basic policy

In this basic policy, multicast packets received on an interface are flooded unconditionally to all interfaces on the forwarding domain, except the incoming interface.

```
config firewall multicast-policy
    edit 1
        set srcintf "any"
        set dstintf "any"
```

```
        set srcaddr "all"
        set dstaddr "all"
    next
end
```

The destination address (`dstaddr`) is a multicast address object. The `all` option corresponds to all multicast addresses in the range 224.0.0.0-239.255.255.255.

Sample policy with specific source and destination interfaces

This multicast policy only applies to the source port `wan1` and the destination port `internal`.

```
config firewall multicast-policy
    edit 1
        set srcintf "wan1"
        set dstintf "internal"
        set srcaddr "all"
        set dstaddr "all"
    next
end
```

Sample policy with specific source address object

In this policy, packets are allowed to flow from `wan1` to `internal`, and sourced by the address 172.20.120.129, which is represented by the `example_addr-1` address object.

```
config firewall multicast-policy
    edit 1
        set srcintf "wan1"
        set dstintf "internal"
        set srcaddr "example_addr-1"
        set dstaddr "all"
    next
end
```

Sample detailed policy

This policy accepts multicast packets that are sent from a PC with IP address 192.168.5.18 to destination address range 239.168.4.0-255. The policy allows the multicast packets to enter the `internal` interface and then exit the `external` interface. When the packets leave the external interface, their source address is translated to 192.168.18.10.

```
config firewall address
    edit "192.168.5.18"
        set subnet 192.168.5.18 255.255.255.255
    next
end

config firewall multicast-address
    edit "239.168.4.0"
        set start-ip 239.168.4.0
        set end-ip 239.168.4.255
    next
end

config firewall multicast-policy
    edit 1
        set srcintf "internal"
```

```
        set dstintf "external"
        set srcaddr "192.168.5.18"
        set dstaddr "239.168.4.0"
        set snat enable
        set snat-ip 192.168.18.10
    next
end
```

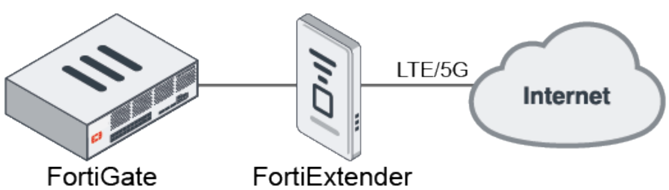
To configure multicast policies in the GUI, enable *Multicast Policy* in *System > Feature Visibility*.

FortiExtender

There are two configuration modes available on the FortiGate for FortiExtender integration: WAN extension mode and LAN extension mode.

WAN extension mode

In WAN extension mode, the FortiExtender works as an extended WAN interface in IP pass-through mode. The FortiGate manages FortiExtender over the CAPWAP protocol in IP pass-through mode, and is integrated into FortiOS as a manageable interface.

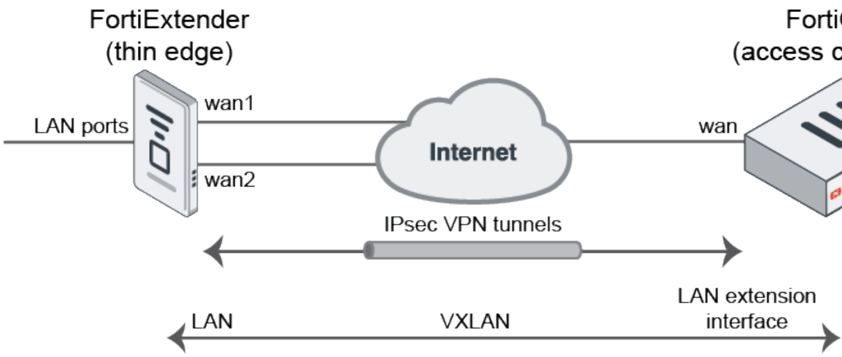


Sample configurations in WAN extension mode could include connecting a FortiExtender to two FortiGates in HA active-passive mode, or connecting two FortiExtenders to two FortiGates in HA active-active mode to provide dual active redundancy for wireless WAN access.

For more information, see FortiExtender and FortiGate integration in the FortiExtender (Managed) Administration Guide.

LAN extension mode

The LAN extension configuration mode allows FortiExtender to provide remote thin edge connectivity back to the FortiGate over a backhaul connection. A FortiExtender deployed at a remote location will discover the FortiGate access controller (AC) and form an IPsec tunnel (or multiple tunnels when multiple links exist on the FortiExtender) back to the FortiGate. A VXLAN is established over the IPsec tunnels to create an L2 network between the FortiGate and the network behind the remote FortiExtender.



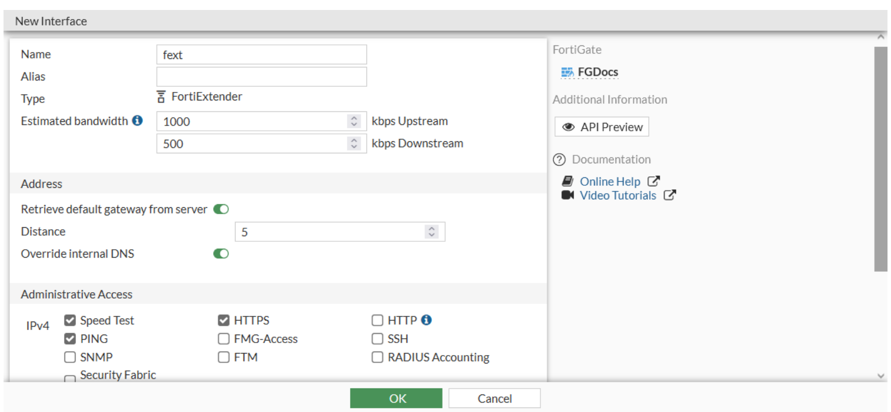
For more information, see FortiExtender as FortiGate LAN extension in the FortiExtender (Managed) Administration Guide.

Adding a FortiExtender

To add a FortiExtender to the FortiGate, create a virtual FortiExtender interface, then add a FortiExtender and assign the interface to the modem. Like other interface types, the FortiExtender interface can be used in static routes, SD-WAN (see Manage dual FortiExtender devices), policies, and other functions.

To create a virtual FortiExtender interface in the GUI:

1. Go to *Network > Interfaces* and click *Create New > FortiExtender*.
2. Enter a name for the interface.
3. Configure the remaining settings as needed. See Interface settings on page 137 for more details.

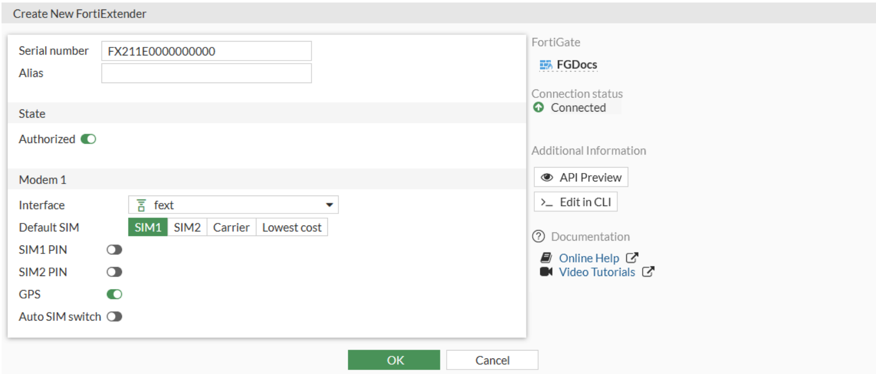


4. Click *OK*.

To add a FortiExtender in the GUI:

1. Go to *Network > FortiExtender* and click *Create New > Extenders*.
2. Enter your FortiExtender's serial number in the *Serial number* field.
3. Optionally, set an *Alias* for the FortiExtender.
4. In the *State* section, enable *Authorized*.
5. Set *Interface* to the FortiExtender interface.

6. Configure the remaining setting as required. See the FortiExtender Administration Guide (FGT-Managed) for more information.



7. Click *OK*.
8. In the extenders list, right-click on the FortiExtender and select *Diagnostics and Tools* to review the modem and SIM status, and other details about the FortiExtender.

To create a virtual FortiExtender interface in the CLI:

```
config system interface
    edit "fext"
        set vdom "root"
        set mode dhcp
        set allowaccess ping https speed-test
        set type fext-wan
        set estimated-upstream-bandwidth 1000
        set estimated-downstream-bandwidth 500
    next
end
```

To configure the FortiExtender in the CLI:

```
config extender-controller extender
    edit "FX211E0000000000"
        set id "FX211E0000000000"
        set authorized enable
        config modem1
            set ifname "fext"
        end
    next
end
```

To verify the modem settings in the CLI:

```
get extender modem-status FX211E0000000000 1
Modem 0:
    physical_port:          2-1.2
    manufacture:            Sierra Wireless, Incorporated
    product:                Sierra Wireless, Incorporated
    ....
```

Direct IP support for LTE/4G

Direct IP is a public IP address that is assigned to a computing device, which allows the device to directly access the internet.

When an LTE modem is enabled in FortiOS, a DHCP interface is created. As a result, the FortiGate can acquire direct IP (which includes IP, DNS, and gateway) from the LTE network carrier.

Since some LTE modems require users to input the access point name (APN) for the LTE network, the LTE modem configuration allows you to set the APN.

LTE modems can only be enabled by using the CLI.

To enable direct IP support using the CLI:

1. Enable the LTE modem:

```
config system lte-modem
  set status enable
end
```

2. Check that the LTE interface was created:

```
config system interface
  edit "wwan"
    set vdom "root"
    set mode dhcp
    set status down
    set distance 1
    set type physical
    set snmp-index 23
  next
end
```

Shortly after the LTE modem joins its carrier network, `wwan` is enabled and granted direct IP:

```
config system interface
    edit wwan
        get
name                 : wwan
....
ip                   : 100.112.75.43 255.255.255.248
....
status               : up
....
defaultgw            : enable
DHCP Gateway         : 100.112.75.41
Lease Expires        : Thu Feb 21 19:33:27 2019
dns-server-override : enable
Acquired DNS1        : 184.151.118.254
Acquired DNS2        : 70.28.245.227
....
```

PCs can reach the internet via the following firewall policy:

```
config firewall policy
  edit 5
    set name "LTE"
    set srcintf "port9"
    set dstintf "wwan"
    set srcaddr "all"
    set dstaddr "all"
    set action accept
    set schedule "always"
    set service "ALL"
    set utm-status enable
    set fsso disable
    set nat enable
  next
end
```

Sample LTE interface

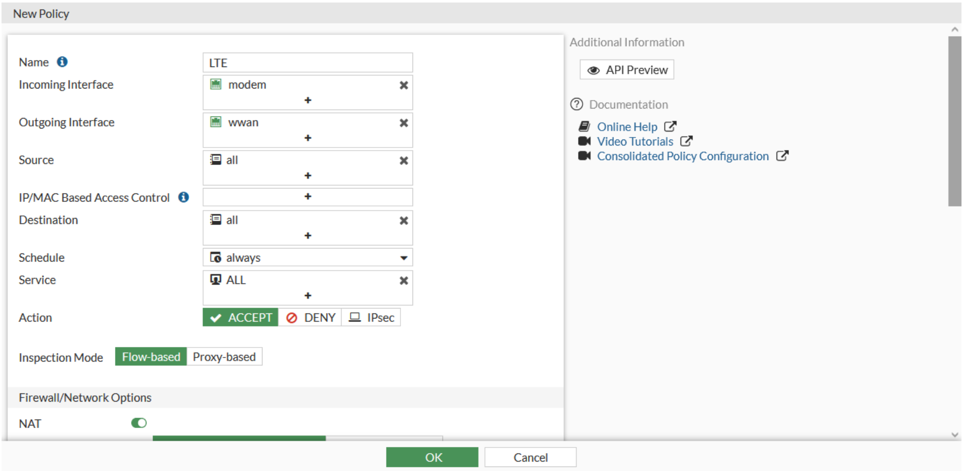
When an LTE modem is enabled, you can view the LTE interface in the GUI and check the acquired IP, DNS, and gateway.

To view the LTE interface in the GUI:

1. Go to *Network > Interfaces*.
2. Double-click the LTE interface (*wwan*) to view the properties.
3. Look in the *Address* section to see the *Obtained IP/Netmask*, *Acquired DNS*, and *Default Gateway*.
4. Click *Return*.

To configure the firewall policy that uses the LTE interface:

1. Go to *Policy & Objects > Firewall Policy*.
2. Edit the LTE policy.
3. In the *Outgoing Interface* field, select the interface (`wwan` in this example).
4. Configure the rest of the policy as needed.



5. Click *OK*.

Limitations

- Most LTE modems have a preset APN in their SIM card. Therefore, the APN does not need to be set in the FortiOS configuration. In cases where the internet cannot be accessed, consult with your carrier and set the APN in the LTE modem configuration (for example, inet.bell.ca):

```
config system lte-modem
  set status enable
  set apn "inet.bell.ca"
end
```

- Some models, such as the FortiGate 30E-3G4G, have built-in LTE modems. In this scenario, the LTE modem is enabled by default. The firewall policy via the LTE interface is also created by default. Once you plug in a SIM card, your network devices can connect to the internet.

Sample FortiGate 30E-3G4G default configuration:

```
config system lte-modem
  set status enable
  set extra-init ''
  set manual-handover disable
  set force-wireless-profile 0
  set authtype none
  set apn ''
  set modem-port 255
  set network-type auto
  set auto-connect disable
  set gpsd-enabled disable
  set data-usage-tracking disable
  set gps-port 255
end
```

```
config firewall policy
....
  edit 3
    set srcintf "internal"
```

```
        set dstintf "wwan"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set nat enable
      next
    end
```

LLDP reception

Device detection can scan LLDP as a source for device identification, but the FortiGate does not read or store the full information. Enabling LLDP reception allows the FortiGate to receive and store LLDP messages, learn about active neighbors, and makes the LLDP information available via the CLI, REST API, and SNMP.

You need to enable `device-identification` at the interface level, and then `lldp-reception` can be enabled on three levels: globally, per VDOM, or per interface.

To configure device identification on an interface:

```
config system interface
    edit <port>
        set device-identification enable
    next
end
```

To configure LLDP reception globally:

```
config system global
    set lldp-reception enable
end
```

To configure LLDP reception per VDOM:

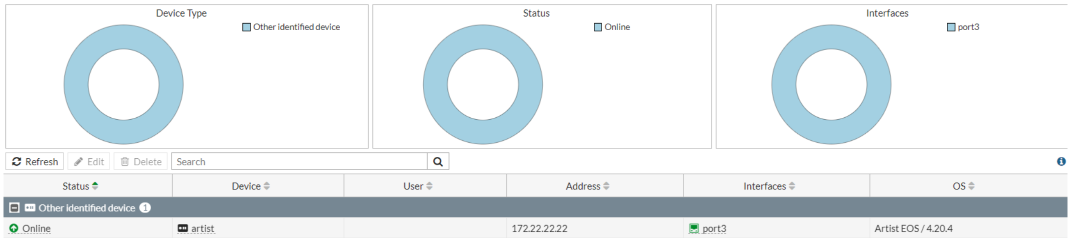
```
config system setting
    set lldp-reception enable
end
```

To configure LLDP reception per interface:

```
config system interface
    edit <port>
        set lldp-reception enable
    next
end
```

To view the LLDP information in the GUI:

1. Go to *Dashboard > Users & Devices*.
2. Expand the *Device Inventory* widget to full screen.



To view the received LLDP information in the CLI:

```
# diagnose user device list
    hosts
      vd root/0 44:0a:a0:0a:0a:0a gen 3 req S/2
        created 10290s gen 1 seen 0s port3 gen 1
        ip 172.22.22.22 src lldp
        type 20 'Other Network Device' src lldp id 155 gen 2
        os 'Artist EOS ' version '4.20.4' src lldp id 155
        host 'artist' src lldp
```

To view additional information about LLDP neighbors and ports:

```
# diagnose lldprx neighbor {summary | details | clear}
```

```
# diagnose lldprx port {details | summary | neighbor | filter}
```

```
# diagnose lldprx port neighbor {summary | details}
```

Note that the port index in the output corresponds to the port index from the following command:

```
# diagnose netlink interface list port2 port3 | grep index
      if=port2 family=00 type=1 index=4 mtu=1500 link=0 master=0
      if=port3 family=00 type=1 index=5 mtu=1500 link=0 master=0
```

To view the received LLDP information in the REST API:

```
{
  "http_method":"GET",
  "results":[
    {
      "mac":"90:9c:9c:c9:c9:90",
      "chassis_id":"90:9C:9C:C9:C9:90",
      "port":19,
      "port_id":"port12",
      "port_desc":"port12",
      "system_name":"S124DN3W00000000",
      "system_desc":"FortiSwitch-124D v3.6.6,build0416,180515 (GA)",
      "ttl":120,
      "addresses":[
        {
          "type":"ipv4",
          "address":"192.168.1.99"
        }
      ]
    }
  ],
  "vdom":"root",
```

```
    "path":"network",
    "name":"lldp",
    "action":"neighbors",
    "status":"success",
    "serial":"FG201E4Q00000000",
    "version":"v6.2.0",
    "build":866
}

{
    "http_method":"GET",
    "results":[
        {
            "name":"port1",
            "rx":320,
            "neighbors":1
        }
    ],
    "vdom":"root",
    "path":"network",
    "name":"lldp",
    "action":"ports",
    "mkey":"port1",
    "status":"success",
    "serial":"FG201E4Q00000000",
    "version":"v6.2.0",
    "build":866
}
```


Virtual routing and forwarding

Virtual Routing and Forwarding (VRF) is used to divide the FortiGate's routing functionality (layer 3), including interfaces, routes, and forwarding tables, into separate units. Packets are only forwarded between interfaces that have the same VRF.

VDOMs divide the FortiGate into two or more complete and independent virtual units that include all FortiGate functions. VDOMs can be used for routing segmentation, but that should not be the only reason to implement them when a less complex solution (VRFs) can be used. VDOMs also support administration boundaries, but VRFs do not.

Up to 64 VRFs can be configured per VDOM for any device, but only ten VDOMs can be configured by default on a FortiGate (more VDOMs can be configured on larger devices with additional licenses).

- Implementing VRF on page 410
- VRF routing support on page 411
- Route leaking between VRFs with BGP on page 415
- Route leaking between multiple VRFs on page 417
- VRF with IPv6 on page 428
- IBGP and EBGP support in VRF on page 432

Implementing VRF

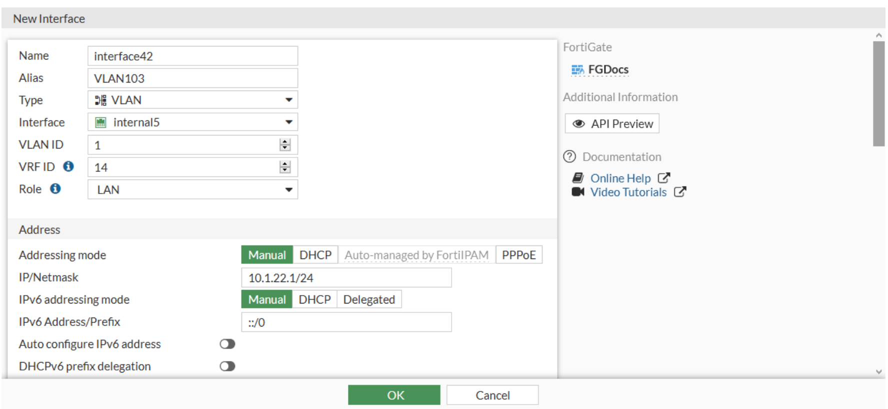
VRFs are always enabled and, by default, all routing is done in VRF 0. To use additional VRFs, assign a VRF ID to an interface. All routes relating to that interface are isolated to that VRF specific routing table. Interfaces in one VRF cannot reach interfaces in a different VRF.

If some traffic does have to pass between VRFs, route leaking can be used. See Route leaking between VRFs with BGP on page 415.

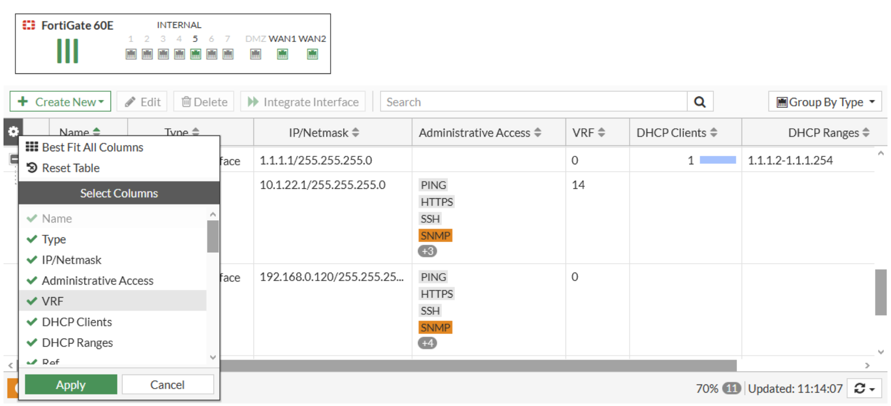
Enable *Advanced Routing* in *System > Feature Visibility* to configure VRFs.

To configure a VRF ID on an interface in the GUI:

1. Go to *Network > Interfaces* and click *Create New > Interface*.
2. Enter a value in the VRF ID field.
3. Configure the other settings as needed.



4. Click *OK*.
5. To add the VRF column in the interface table, click the gear icon, select *VRF*, and click *Apply*.

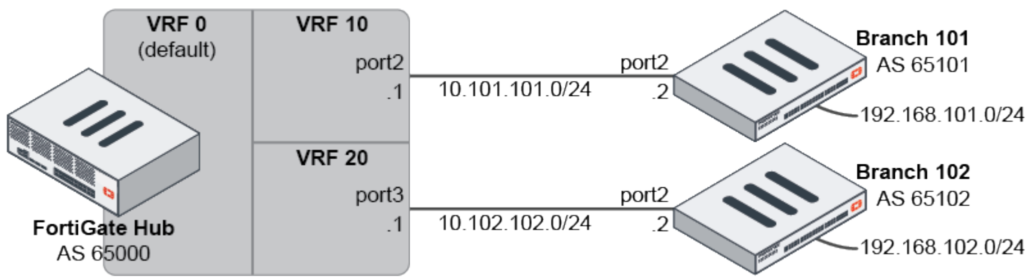


To configure a VRF ID on an interface in the CLI:

```
config system interface
    edit interface42
        ...
        set vrf 14
    next
end
```

VRF routing support

VRF supports static routing, OSPF, and BGP. Other routing protocols require using VDOMs.



BGP

In this example, BGP is used to update the VRF that it is neighbors with.

The hub is configured with two neighbors connected to two interfaces. The branches are configured to match the hub, with branch networks configured to redistribute into BGP.

Policies must be created on the hub and branches to allow traffic between them.

To configure the hub:

```
config router bgp
    set as 65000
    config neighbor
        edit "10.101.101.2"
            set soft-reconfiguration enable
            set interface "port2"
            set remote-as 65101
            set update-source "port2"
        next
        edit "10.102.102.2"
            set soft-reconfiguration enable
            set interface "port3"
            set remote-as 65102
            set update-source "port3"
        next
    end
end
```

To configure branch 101:

```
config router bgp
    set as 65101
    config neighbor
        edit "10.101.101.1"
            set soft-reconfiguration enable
            set interface "port2"
            set remote-as 65000
            set update-source "port2"
        next
    end
    config redistribute connected
        set status enable
    end
end
```

To configure branch 102:

```
config router bgp
    set as 65102
    config neighbor
        edit "10.102.102.1"
            set soft-reconfiguration enable
            set interface "port2"
            set remote-as 65000
            set update-source "port2"
        next
    end
    config redistribute connected
        set status enable
    end
end
```

To verify the BGP neighbors and check the routing table on the hub:

```
# get router info bgp summary
BGP router identifier 192.168.0.1, local AS number 65000
BGP table version is 2
2 BGP AS-PATH entries
0 BGP community entries

Neighbor            V       AS            MsgRcvd MsgSent TblVer  InQ     OutQ Up/Down    State/Pf
10.101.101.2   4       65101           4               4               2               0       0
10.102.102.2   4       65102           3               3               1               0       0

Total number of neighbors 2
```

```
# get router info routing-table all
Routing table for VRF=0
Codes (…)
S*     0.0.0.0/0 [10/0] via 192.168.0.254, port1
C      10.101.101.0/24 is directly connected, port2
C      10.102.102.0/24 is directly connected, port3
C      192.168.0.0/24 is directly connected, port1
```

```
B      192.168.101.0/24 [20/0] via 10.101.101.2, port2, 00:01:25
B      192.168.102.0/24 [20/0] via 10.102.102.2, port3, 00:00:50
```

To configure VRF on the hub:

1. Put the interfaces into VRF:

```
config system interface
    edit port2
        set vrf 10
    next
    edit port3
        set vrf 20
    next
end
```

2. Restart the router to reconstruct the routing tables:

```
# execute router restart
```

3. Check the routing tables:

```
# get router info routing-table all
Routing table for VRF=0
Codes (…)
S*     0.0.0.0/0 [10/0] via 192.168.0.254, port1
C      192.168.0.0/24 is directly connected, port1

Routing table for VRF=10
C      10.101.101.0/24 is directly connected, port2
B      192.168.101.0/24 [20/0] via 10.101.101.2, port2, 00:02:25

Routing table for VRF=20
C      10.102.102.0/24 is directly connected, port3
B      192.168.102.0/24 [20/0] via 10.102.102.2, port2, 00:01:50
```

4. Check the BGP summary:

```
# get router info bgp summary

VRF 10 BGP router identifier 10.101.101.1, local AS number 65000
BGP table version is 1
2 BGP AS-PATH entries
0 BGP community entries

Neighbor              V       AS              MsgRcvd MsgSent TblVer  InQ     OutQ Up/Down    State
10.101.101.2   4       65101           4               4               2               0       0

Total number of neighbors 1

VRF 10 BGP router identifier 10.101.101.1, local AS number 65000
BGP table version is 1
2 BGP AS-PATH entries
0 BGP community entries

Neighbor              V       AS              MsgRcvd MsgSent TblVer  InQ     OutQ Up/Down    State
10.102.102.2   4       65102           3               3               1               0       0
```

```
Total number of neighbors 1
```

OSPF

OSPF routes in VRFs work the same as BGP: the interface that OSPF is using is added to the VRF.

To configure the hub:

1. Configure OSPF:

```
config router ospf
    set router-id 1.1.1.1
    config area
        edit 0.0.0.0
        next
    end
    config ospf-interface
        edit Branch101
            set interface "port2"
            set dead-interval 40
            set hello-interval 10
        next
        edit Branch102
            set dead-interval 40
            set hello-interval 10
        next
    end
    config network
        edit 0
            set prefix 10.101.101.0 255.255.255.0
        next
        edit 0
            set prefix 10.102.102.0 255.255.255.0
        next
        edit 0
            set prefix 192.168.1.0 255.255.255.0
        next
    end
end
```

2. Put the interfaces into VRF:

```
config system interface
    edit port2
        set vrf 10
    next
    edit port3
        set vrf 20
    next
end
```

To configure branch 101:

```
config router ospf
    set router-id 101.101.101.101
    config area
        edit 0.0.0.0
        next
    end
    config ospf-interface
        edit HUB
            set interface port2
            set dead-interval 40
            set hello-interval 10
        next
    end
    config network
        edit 0
            set prefix 10.101.101.0 255.255.255.0
        next
        edit 0
            set prefix 192.168.101.0 255.255.255.0
        next
    end
end
```

To check the routing table and OSPF summary:

```
# get router info routing-table ospf
```

```
# get router info ospf interface
```

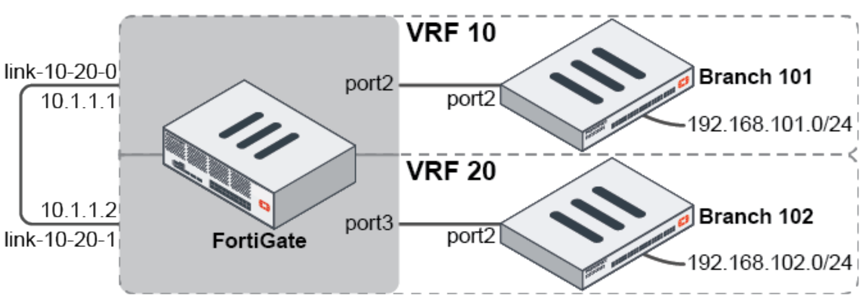
Route leaking between VRFs with BGP

Route leaking allows you to configure communication between VRFs. If route leaking is not configured, then the VRFs are isolated. This example shows route leaking with BGP using virtual inter-VDOM links.

In this example, a hub FortiGate forms BGP neighbors with two branches. It learns the networks 192.168.101.0/24 and 192.168.102.0/24 from the neighbors and separates them into VRF 10 and VRF 20.

To leak the learned routes to each other, an inter-VDOM link (IVL) is formed. An IVL normally bridges two VDOMs, but in this case the links reside on the same VDOM and are used to bridge the two VRFs. NPU links could also be used on models that support it to deliver better performance.

VRF 10 has a leaked route to 192.168.102.0/24 on IVL *link-10-20-0*, and VRF 20 has a leaked route to 192.168.101.0/24 on IVL *link-10-20-1*,



To configure route leaking:

1. Configure inter-VDOM links:

```
config global
    config system vdom-link
        edit link-10-20-
        next
    end
    config system interface
        edit link-10-20-0
            set vdom “root”
            set vrf 10
            set ip 10.1.1.1/30
        next
        edit link-10-20-1
            set vdom “root”
            set vrf 20
            set ip 10.1.1.2/30
        next
    end
end
```

2. Create prefix lists:
   These object define the subnet and mask that are leaked.

```
config router prefix-list
    edit VRF10_Route
        config rule
            edit 1
                set prefix 192.168.101.0 255.255.255.0
            next
        end
    next
    edit VRF20_Route
        config rule
            edit 1
                set prefix 192.168.102.0 255.255.255.0
            next
        end
    next
end
```

3. Create the route map:
   The route map can be used to group one or more prefix lists.

```
config router route-map
    edit "Leak_from_VRF10_to_VRF20"
        config rule
            edit 1
                set match-ip-address "VRF10_Route"
            next
        end
    next
    edit "Leak_from_VRF20_to_VRF10"
        config rule
            edit 1
```

```
                set match-ip-address "VRF20_Route"
            next
        end
    next
end
```

4. Configure the VRF leak in BGP, specifying a source VRF, destination VRF, an the route map to use:

```
config router bgp
    config vrf
        edit "10"
            config leak-target
                edit "20"
                    set route-map "Leak_from_VRF10_to_VRF20"
                    set interface "link-10-20-0"
                next
            end
        next
        edit "20"
            config leak-target
                edit "10"
                    set route-map "Leak_from_VRF20_to_VRF10"
                    set interface "link-10-20-1"
                next
            end
        next
    end
end
```

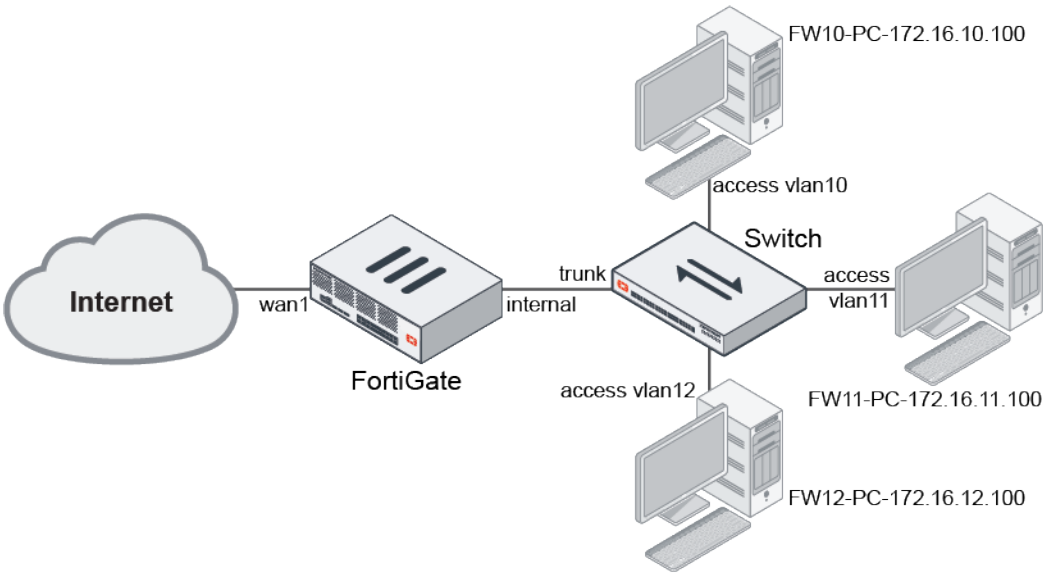
5. Create policies to allow traffic between the VRFs.
   Without a policy permitting traffic on the route between the VRFs, the VRFs are still isolated.

Route leaking between multiple VRFs

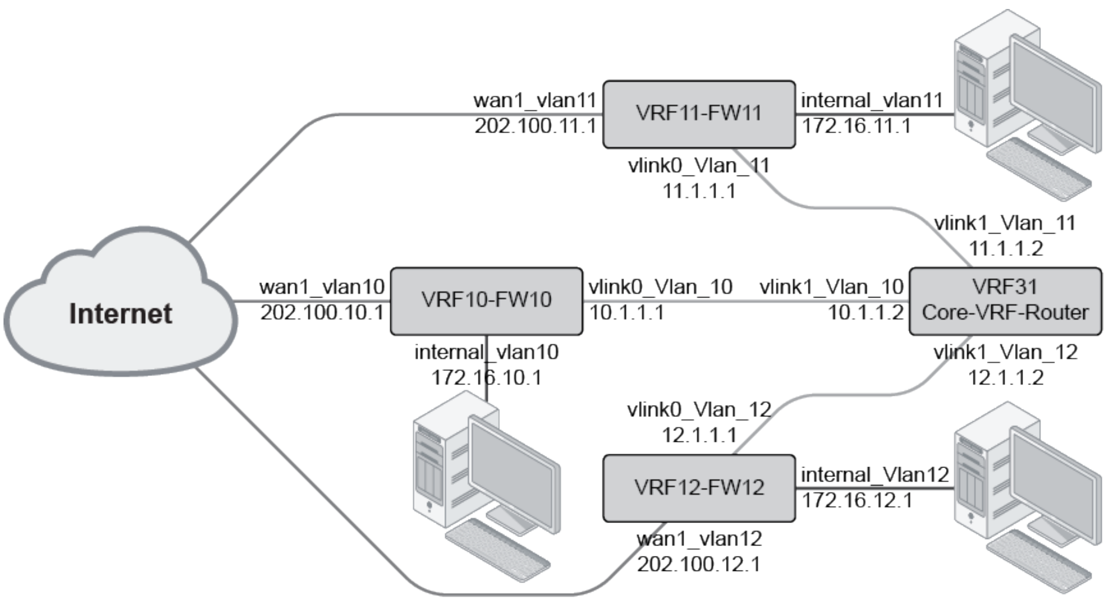
In this example, routing leaking between three VRFs in a star topology is configured. This allows the solution to be scaled to more VRFs without building full mesh, one-to-one connections between each pair of VRFs. VLAN subinterfaces are created on VDOM links to connect each VRF to the central VRF, allowing routes to be leaked from a VRF to the central VRF, and then to the other VRFs. Static routes are used for route leaking in this example.

For instructions on creating route leaking between two VRFs, see Route leaking between VRFs with BGP on page 415.

Physical topology:



Logical topology:



In this example, a specific route is leaked from each of the VRFs to each of the other VRFs. VLAN subinterfaces are created based on VDOM links to connect each VRF to the core VRF router.

Multi VDOM mode is enabled so that NP VDOM links can be used. The setup could be configured without enabling multi VDOM mode by manually creating non-NP VDOM links, but this is not recommended as the links are not offloaded to the NPU.

After VDOMs are enabled, all of the configuration is done in the *root* VDOM.

To configure the FortiGate:

1. Enable multi VDOM mode:

```
config system global
    set vdom-mode multi-vdom
end
```

   If the FortiGate has an NP, the VDOM links will be created:

```
# show system interface
config system interface
    ...
    edit "npu0_vlink0"
        set vdom "root"
        set type physical
    next
    edit "npu0_vlink1"
        set vdom "root"
        set type physical
    next
    ...
end
```

   If multi VDOM mode is not used, the VDOM links can be manually created:

```
config system vdom-link
    edit <name of vdlink>
    next
end
```

2. Allow interface subnets to use overlapping IP addresses:

```
config vdom
    edit root
        config system settings
            set allow-subnet-overlap enable
        end
```

3. Configure the inter-connecting VLAN subinterfaces between VRF based on VDOM-LINK:

```
config system interface
    edit "vlink0_Vlan_10"
        set vdom "root"
        set vrf 10
        set ip 10.1.1.1 255.255.255.252
        set allowaccess ping https ssh http
        set alias "vlink0_Vlan_10"
        set role lan
        set interface "npu0_vlink0"
        set vlanid 10
    next
    edit "vlink1_Vlan_10"
        set vdom "root"
        set vrf 31
        set ip 10.1.1.2 255.255.255.252
        set allowaccess ping https ssh http
        set alias "vlink1_Vlan_10"
        set role lan
```

```
        set interface "npu0_vlink1"
        set vlanid 10
    next
    edit "vlink0_Vlan_11"
        set vdom "root"
        set vrf 11
        set ip 11.1.1.1 255.255.255.252
        set allowaccess ping https ssh http
        set alias "vlink0_Vlan_11"
        set role lan
        set interface "npu0_vlink0"
        set vlanid 11
    next
    edit "vlink1_Vlan_11"
        set vdom "root"
        set vrf 31
        set ip 11.1.1.2 255.255.255.252
        set allowaccess ping https ssh http
        set alias "vlink1_Vlan_11"
        set role lan
        set interface "npu0_vlink1"
        set vlanid 11
    next
    edit "vlink0_Vlan_12"
        set vdom "root"
        set vrf 12
        set ip 12.1.1.1 255.255.255.252
        set allowaccess ping https ssh http
        set alias "vlink0_Vlan_12"
        set role lan
        set interface "npu0_vlink0"
        set vlanid 12
    next
    edit "vlink1_Vlan_12"
        set vdom "root"
        set vrf 31
        set ip 12.1.1.2 255.255.255.252
        set allowaccess ping https ssh http
        set alias "vlink1_Vlan_12"
        set role lan
        set interface "npu0_vlink1"
        set vlanid 12
    next
end
```

4. Configure a zone to allow intrazone traffic between VLANs in the central VRF:

```
config system zone
    edit "Core-VRF-Router"
        set intrazone allow
        set interface "vlink1_Vlan_10" "vlink1_Vlan_11" "vlink1_Vlan_12"
    next
end
```

5. Add allow policies for the VRF31 core router:

```
config firewall policy
    edit 0
        set name "any_to_core_vrf31"
        set srcintf "any"
        set dstintf "Core-VRF-Router"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set logtraffic all
    next
    edit 0
        set name "core_vrf31_to_any"
        set srcintf "Core-VRF-Router"
        set dstintf "any"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set logtraffic all
    next
end
```

6. Configure VRF10, VRF11, and VRF12 on the Internal and WAN VLAN sub-interfaces:

```
config system interface
    edit "Internal_VRF10"
        set vdom "root"
        set vrf 10
        set ip 172.16.10.1 255.255.255.0
        set allowaccess ping https ssh http
        set alias "Internal_VRF10"
        set role lan
        set interface "internal"
        set vlanid 10
    next
    edit "Internal_VRF11"
        set vdom "root"
        set vrf 11
        set ip 172.16.11.1 255.255.255.0
        set allowaccess ping https ssh http
        set alias "Internal_VRF11"
        set role lan
        set interface "internal"
        set vlanid 11
    next
    edit "Internal_VRF12"
        set vdom "root"
        set vrf 12
        set ip 172.16.12.1 255.255.255.0
        set allowaccess ping https ssh http
        set alias "Internal_VRF12"
        set role lan
        set interface "internal"
        set vlanid 12
```

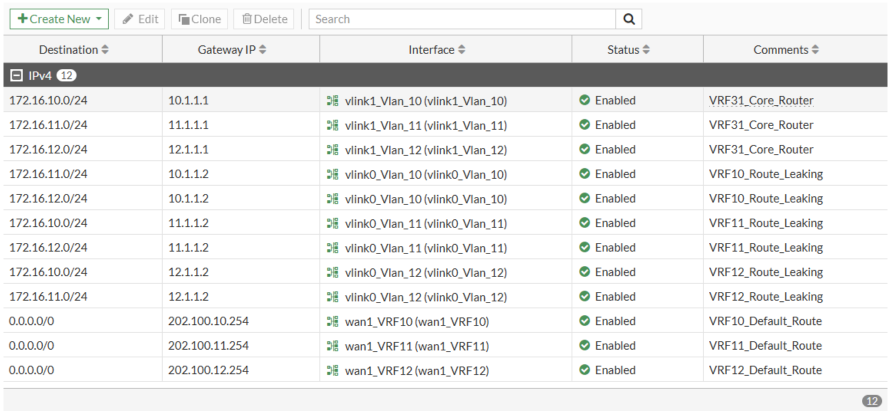
```
        next
        edit "wan1_VRF10"
            set vdom "root"
            set vrf 10
            set ip 202.100.10.1 255.255.255.0
            set allowaccess ping
            set alias "wan1_VRF10"
            set role wan
            set interface "wan1"
            set vlanid 10
        next
        edit "wan1_VRF11"
            set vdom "root"
            set vrf 11
            set ip 202.100.11.1 255.255.255.0
            set allowaccess ping
            set alias "wan1_VRF11"
            set role wan
            set interface "wan1"
            set vlanid 11
        next
        edit "wan1_VRF12"
            set vdom "root"
            set vrf 12
            set ip 202.100.12.1 255.255.255.0
            set allowaccess ping
            set alias "wan1_VRF12"
            set role wan
            set interface "wan1"
            set vlanid 12
        next
    end
```

7. Configure static routing and route leaking between each VRF and Core-VRF-Router:

```
config router static
    edit 1
        set dst 172.16.10.0 255.255.255.0
        set gateway 10.1.1.1
        set device "vlink1_Vlan_10"
        set comment "VRF31_Core_Router"
    next
    edit 2
        set dst 172.16.11.0 255.255.255.0
        set gateway 11.1.1.1
        set device "vlink1_Vlan_11"
        set comment "VRF31_Core_Router"
    next
    edit 3
        set dst 172.16.12.0 255.255.255.0
        set gateway 12.1.1.1
        set device "vlink1_Vlan_12"
        set comment "VRF31_Core_Router"
    next
    edit 4
        set dst 172.16.11.0 255.255.255.0
```

```
        set gateway 10.1.1.2
        set device "vlink0_Vlan_10"
        set comment "VRF10_Route_Leaking"
    next
    edit 5
        set dst 172.16.12.0 255.255.255.0
        set gateway 10.1.1.2
        set device "vlink0_Vlan_10"
        set comment "VRF10_Route_Leaking"
    next
    edit 6
        set dst 172.16.10.0 255.255.255.0
        set gateway 11.1.1.2
        set device "vlink0_Vlan_11"
        set comment "VRF11_Route_Leaking"
    next
    edit 7
        set dst 172.16.12.0 255.255.255.0
        set gateway 11.1.1.2
        set device "vlink0_Vlan_11"
        set comment "VRF11_Route_Leaking"
    next
    edit 8
        set dst 172.16.10.0 255.255.255.0
        set gateway 12.1.1.2
        set device "vlink0_Vlan_12"
        set comment "VRF12_Route_Leaking"
    next
    edit 9
        set dst 172.16.11.0 255.255.255.0
        set gateway 12.1.1.2
        set device "vlink0_Vlan_12"
        set comment "VRF12_Route_Leaking"
    next
    edit 10
        set gateway 202.100.10.254
        set device "wan1_VRF10"
        set comment "VRF10_Default_Route"
    next
    edit 11
        set gateway 202.100.11.254
        set device "wan1_VRF11"
        set comment "VRF11_Default_Route"
    next
    edit 12
        set gateway 202.100.12.254
        set device "wan1_VRF12"
        set comment "VRF12_Default_Route"
    next
end
```

In the GUI, go to *Network > Static Routes* to view the static routes:



Destination	Gateway IP	Interface	Status	Comments
IPv4 12				
172.16.10.0/24	10.1.1.1	vlink1_Vlan_10 (vlink1_Vlan_10)	Enabled	VRF31_Core_Router
172.16.11.0/24	11.1.1.1	vlink1_Vlan_11 (vlink1_Vlan_11)	Enabled	VRF31_Core_Router
172.16.12.0/24	12.1.1.1	vlink1_Vlan_12 (vlink1_Vlan_12)	Enabled	VRF31_Core_Router
172.16.11.0/24	10.1.1.2	vlink0_Vlan_10 (vlink0_Vlan_10)	Enabled	VRF10_Route_Leaking
172.16.12.0/24	10.1.1.2	vlink0_Vlan_10 (vlink0_Vlan_10)	Enabled	VRF10_Route_Leaking
172.16.10.0/24	11.1.1.2	vlink0_Vlan_11 (vlink0_Vlan_11)	Enabled	VRF11_Route_Leaking
172.16.12.0/24	11.1.1.2	vlink0_Vlan_11 (vlink0_Vlan_11)	Enabled	VRF11_Route_Leaking
172.16.10.0/24	12.1.1.2	vlink0_Vlan_12 (vlink0_Vlan_12)	Enabled	VRF12_Route_Leaking
172.16.11.0/24	12.1.1.2	vlink0_Vlan_12 (vlink0_Vlan_12)	Enabled	VRF12_Route_Leaking
0.0.0.0/0	202.100.10.254	wan1_VRF10 (wan1_VRF10)	Enabled	VRF10_Default_Route
0.0.0.0/0	202.100.11.254	wan1_VRF11 (wan1_VRF11)	Enabled	VRF11_Default_Route
0.0.0.0/0	202.100.12.254	wan1_VRF12 (wan1_VRF12)	Enabled	VRF12_Default_Route

8. Configure firewall policies for VRF10, VRF11, and VRF12

```
config firewall policy
    edit 6
        set name "VRF10_to_Internet_Policy"
        set srcintf "Internal_VRF10"
        set dstintf "wan1_VRF10"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set logtraffic all
        set nat enable
    next
    edit 7
        set name "VRF10_to_VRF_Leaking_Route"
        set srcintf "Internal_VRF10"
        set dstintf "vlink0_Vlan_10"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set logtraffic all
    next
    edit 8
        set name "VRF_Leaking_Route_to_VRF10"
        set srcintf "vlink0_Vlan_10"
        set dstintf "Internal_VRF10"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set logtraffic all
    next
    edit 9
        set name "VRF11_to_Internet_Policy"
        set srcintf "Internal_VRF11"
        set dstintf "wan1_VRF11"
        set srcaddr "all"
```

```
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set logtraffic all
        set nat enable
    next
    edit 10
        set name "VRF11_to_VRF_Leaking_Route"
        set srcintf "Internal_VRF11"
        set dstintf "vlink0_Vlan_11"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set logtraffic all
    next
    edit 11
        set name "VRF_Leaking_Route_to_VRF11"
        set srcintf "vlink0_Vlan_11"
        set dstintf "Internal_VRF11"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set logtraffic all
    next
    edit 12
        set name "VRF12_to_Internet_Policy"
        set srcintf "Internal_VRF12"
        set dstintf "wan1_VRF12"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set logtraffic all
        set nat enable
    next
    edit 13
        set name "VRF12_to_VRF_Leaking_Route"
        set uuid 92bccf8e-b27b-51eb-3c56-6d5259af6299
        set srcintf "Internal_VRF12"
        set dstintf "vlink0_Vlan_12"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set logtraffic all
    next
    edit 14
        set name "VRF_Leaking_Route_to_VRF12"
```

```
        set srcintf "vlink0_Vlan_12"
        set dstintf "Internal_VRF12"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set logtraffic all
    next
end
```

In the GUI, go to *Policy & Objects* > *Firewall Policy* to view the policies.

To check the results:

1. On the FortiGate, check the routing table to see each VRF:

```
# get router info routing-table all
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default

Routing table for VRF=0
C       10.6.30.0/24 is directly connected, mgmt

Routing table for VRF=10
S*      0.0.0.0/0 [10/0] via 202.100.10.254, wan1_VRF10
C       10.1.1.0/30 is directly connected, vlink0_Vlan_10
C       172.16.10.0/24 is directly connected, Internal_VRF10
S       172.16.11.0/24 [10/0] via 10.1.1.2, vlink0_Vlan_10
S       172.16.12.0/24 [10/0] via 10.1.1.2, vlink0_Vlan_10
C       202.100.10.0/24 is directly connected, wan1_VRF10

Routing table for VRF=11
S*      0.0.0.0/0 [10/0] via 202.100.11.254, wan1_VRF11
C       11.1.1.0/30 is directly connected, vlink0_Vlan_11
S       172.16.10.0/24 [10/0] via 11.1.1.2, vlink0_Vlan_11
C       172.16.11.0/24 is directly connected, Internal_VRF11
S       172.16.12.0/24 [10/0] via 11.1.1.2, vlink0_Vlan_11
C       202.100.11.0/24 is directly connected, wan1_VRF11

Routing table for VRF=12
S*      0.0.0.0/0 [10/0] via 202.100.12.254, wan1_VRF12
C       12.1.1.0/30 is directly connected, vlink0_Vlan_12
S       172.16.10.0/24 [10/0] via 12.1.1.2, vlink0_Vlan_12
S       172.16.11.0/24 [10/0] via 12.1.1.2, vlink0_Vlan_12
C       172.16.12.0/24 is directly connected, Internal_VRF12
C       202.100.12.0/24 is directly connected, wan1_VRF12

Routing table for VRF=31
C       10.1.1.0/30 is directly connected, vlink1_Vlan_10
C       11.1.1.0/30 is directly connected, vlink1_Vlan_11
C       12.1.1.0/30 is directly connected, vlink1_Vlan_12
```

```
S       172.16.10.0/24 [10/0] via 10.1.1.1, vlink1_Vlan_10
S       172.16.11.0/24 [10/0] via 11.1.1.1, vlink1_Vlan_11
S       172.16.12.0/24 [10/0] via 12.1.1.1, vlink1_Vlan_12
```

2. From the FW10-PC:

```
# ifconfig ens32
ens32: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 172.16.10.100  netmask 255.255.255.0  broadcast 172.16.10.255
        inet6 fe80::dbed:c7fe:170e:e61c  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:2a:3a:17  txqueuelen 1000  (Ethernet)
        RX packets 1632  bytes 160001 (156.2 KiB)
        RX errors 0  dropped 52  overruns 0  frame 0
        TX packets 2141  bytes 208103 (203.2 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

```
# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0         172.16.10.1     0.0.0.0         UG    100    0        0 ens32
172.16.10.0     0.0.0.0         255.255.255.0   U     100    0        0 ens32
192.168.122.0   0.0.0.0         255.255.255.0   U     0      0        0 virbr0
```

a. Ping a public IP address through VRF10:

```
# ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=113 time=4.33 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=113 time=4.17 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=113 time=4.04 ms
^C
--- 8.8.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2002ms
rtt min/avg/max/mdev = 4.049/4.188/4.336/0.117 ms
```

b. Ping the internet gateway through VRF10:

```
# ping 202.100.10.254
PING 202.100.10.254 (202.100.10.254) 56(84) bytes of data.
64 bytes from 202.100.10.254: icmp_seq=1 ttl=254 time=0.294 ms
64 bytes from 202.100.10.254: icmp_seq=2 ttl=254 time=0.225 ms
64 bytes from 202.100.10.254: icmp_seq=3 ttl=254 time=0.197 ms
^C
--- 202.100.10.254 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2000ms
rtt min/avg/max/mdev = 0.197/0.238/0.294/0.044 ms
```

c. Ping the FW11-PC on VRF11 from VRF10:

```
# ping 172.16.11.100
PING 172.16.11.100 (172.16.11.100) 56(84) bytes of data.
64 bytes from 172.16.11.100: icmp_seq=1 ttl=61 time=0.401 ms
64 bytes from 172.16.11.100: icmp_seq=2 ttl=61 time=0.307 ms
64 bytes from 172.16.11.100: icmp_seq=3 ttl=61 time=0.254 ms
64 bytes from 172.16.11.100: icmp_seq=4 ttl=61 time=0.277 ms
64 bytes from 172.16.11.100: icmp_seq=5 ttl=61 time=0.262 ms
^C
--- 172.16.11.100 ping statistics ---
```

```
5 packets transmitted, 5 received, 0% packet loss, time 3999ms
rtt min/avg/max/mdev = 0.254/0.300/0.401/0.054 ms
```

3. On the FortiGate, sniff traffic between VRF10 and VRF11:

```
# diagnose sniffer packet any "icmp and host 172.16.11.100" 4 l 0
interfaces=[any]
filters=[icmp and host 172.16.11.100]
10.086656 Internal_VRF10 in 172.16.10.100 -> 172.16.11.100: icmp: echo request
10.086705 vlink0_Vlan_10 out 172.16.10.100 -> 172.16.11.100: icmp: echo request
10.086706 npu0_vlink0 out 172.16.10.100 -> 172.16.11.100: icmp: echo request

10.086711 vlink1_Vlan_10 in 172.16.10.100 -> 172.16.11.100: icmp: echo request
10.086739 vlink1_Vlan_11 out 172.16.10.100 -> 172.16.11.100: icmp: echo request
10.086740 npu0_vlink1 out 172.16.10.100 -> 172.16.11.100: icmp: echo request

10.086744 vlink0_Vlan_11 in 172.16.10.100 -> 172.16.11.100: icmp: echo request
10.086929 Internal_VRF11 out 172.16.10.100 -> 172.16.11.100: icmp: echo request
10.086930 internal out 172.16.10.100 -> 172.16.11.100: icmp: echo request

10.087053 Internal_VRF11 in 172.16.11.100 -> 172.16.10.100: icmp: echo reply
10.087061 vlink0_Vlan_11 out 172.16.11.100 -> 172.16.10.100: icmp: echo reply
10.087062 npu0_vlink0 out 172.16.11.100 -> 172.16.10.100: icmp: echo reply

10.087066 vlink1_Vlan_11 in 172.16.11.100 -> 172.16.10.100: icmp: echo reply
10.087071 vlink1_Vlan_10 out 172.16.11.100 -> 172.16.10.100: icmp: echo reply
10.087072 npu0_vlink1 out 172.16.11.100 -> 172.16.10.100: icmp: echo reply

10.087076 vlink0_Vlan_10 in 172.16.11.100 -> 172.16.10.100: icmp: echo reply
10.087176 Internal_VRF10 out 172.16.11.100 -> 172.16.10.100: icmp: echo reply
10.087177 internal out 172.16.11.100 -> 172.16.10.100: icmp: echo reply
^C
20 packets received by filter
0 packets dropped by kernel
```

VRF with IPv6

IPv6 routes support VRF. Static, connected, OSPF, and BGP routes can be isolated in different VRFs. BGP IPv6 routes can be leaked from one VRF to another.

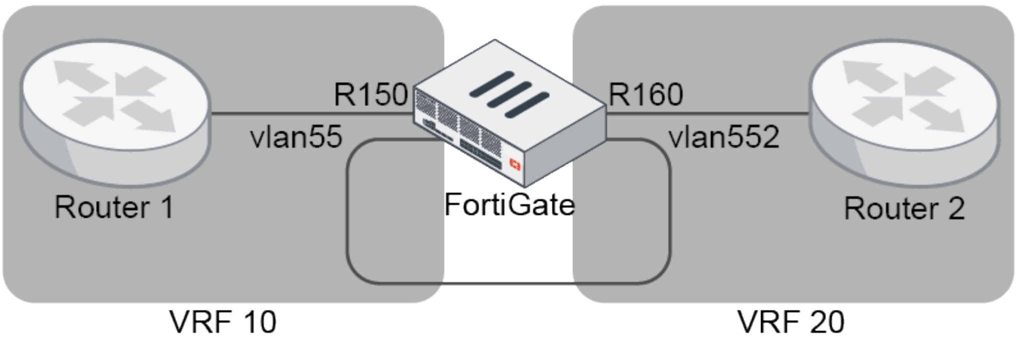
```
config router bgp
    config vrf6
        edit <origin vrf-id>
            config leak-target
                edit <target vrf-id>
                    set route-map <route-map>
                    set interface <interface>
                next
            end
        next
    end
end
```

The origin or target VRF ID is an integer value from 0 - 31.

```
config router static6
    edit <id>
        set vrf <vrf-id>
    next
end
```

Using a VRF leak on BGP

In this example, the route 2000:5:5:5::/64 learned from Router 1 is leaked to VRF 20 through the interface vlan552. Conversely, the route 2009:3:3:3::/64 learned from Router 2 is leaked to VRF 10 through interface vlan55.



To configure VRF leaking in BGP:

1. Configure the BGP neighbors:

```
config router bgp
    set as 65412
    config neighbor
        edit "2000:10:100:1::1"
            set activate disable
            set remote-as 20
            set update-source "R150"
        next
        edit "2000:10:100:1::5"
            set activate disable
            set soft-reconfiguration enable
            set interface "R160"
            set remote-as 20
        next
    end
end
```

2. Configure the VLAN interfaces:

```
config system interface
    edit "vlan55"
        set vdom "root"
        set vrf 10
        set ip 55.1.1.1 255.255.255.0
        set device-identification enable
        set role lan
        set snmp-index 51
        config ipv6
```

```
            set ip6-address 2000:55::1/64
        end
        set interface "npu0_vlink0"
        set vlanid 55
    next
    edit "vlan552"
        set vdom "root"
        set vrf 20
        set ip 55.1.1.2 255.255.255.0
        set device-identification enable
        set role lan
        set snmp-index 53
        config ipv6
            set ip6-address 2000:55::2/64
        end
        set interface "npu0_vlink1"
        set vlanid 55
    next
end
```

3. Configure the IPv6 prefixes:

```
config router prefix-list6
    edit "1"
        config rule
            edit 1
                set prefix6 2000:5:5:5::/64
                unset ge
                unset le
            next
        end
    next
    edit "2"
        config rule
            edit 1
                set prefix6 2009:3:3:3::/64
                unset ge
                unset le
            next
        end
    next
end
```

4. Configure the route maps:

```
config router route-map
    edit "from106"
        config rule
            edit 1
                set match-ip6-address "1"
            next
        end
    next
    edit "from206"
        config rule
            edit 1
                set match-ip6-address "2"
```

```
            next
        end
    next
end
```

5. Configure the IPv6 route leaking (leak route 2000:5:5:5::/64 learned from Router 1 to VRF 20, then leak route 2009:3:3:3::/64 learned from Router 2 to VRF 10):

```
config router bgp
    config vrf6
        edit "10"
            config leak-target
                edit "20"
                    set route-map "from106"
                    set interface "vlan55"
                next
            end
        next
        edit "20"
            config leak-target
                edit "10"
                    set route-map "from206"
                    set interface "vlan552"
                next
            end
        next
    end
end
```

To verify the VRF leaking:

1. Check the routing table before the leak:

```
# get router info6 routing-table bgp
Routing table for VRF=10
B       2000:5:5:5::/64 [20/0] via fe00::2000:0000:0000:00, R150, 00:19:45

Routing table for VRF=20
B       2008:3:3:3::/64 [20/0] via fe00::3000:0000:0000:00, R160, 00:18:49
B       2009:3:3:3::/64 [20/0] via fe00::3000:0000:0000:00, R160, 00:18:49
```

2. Check the routing table after the leak:

```
# get router info6 routing-table bgp
Routing table for VRF=10
B       2000:5:5:5::/64 [20/0] via fe00::2000:0000:0000:0, R150, 00:25:45
B       2009:3:3:3::/64 [20/0] via fe80::10:0000:0000:4245, vlan55, 00:00:17

Routing table for VRF=20
B       2000:5:5:5::/64 [20/0] via fe80::10:0000:0000:4244, vlan552, 00:00:16
B       2008:3:3:3::/64 [20/0] via fe00::3000:0000:0000:00, R160, 00:24:49
B       2009:3:3:3::/64 [20/0] via fe00::3000:0000:0000:00, R160, 00:24:49
```

Using VRF on a static route

In this example, a VRF is defined on static route 22 so that it will only appear in the VRF 20 routing table.

To configure the VRF on the static route:

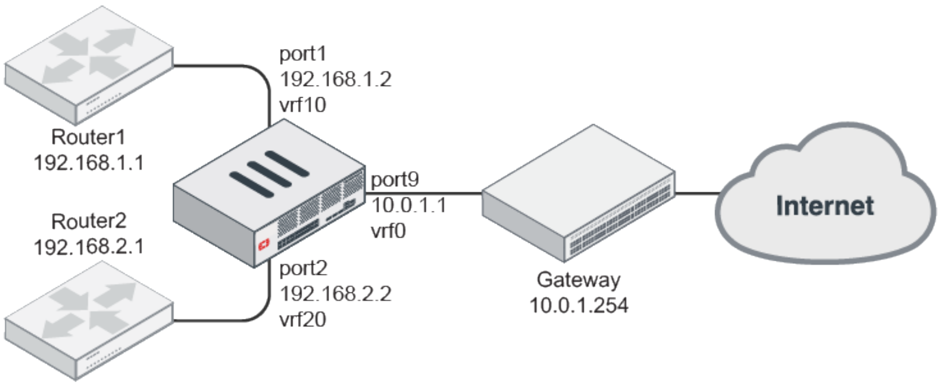
```
config router static6
    edit 22
        set dst 2010:2:2:2::/64
        set blackhole enable
        set vrf 20
    next
end
```

IBGP and EBGP support in VRF

Support is included for internal and external border gateway protocols (IBGP and EBGP) in virtual routing and forwarding (VRF).

FortiGate can establish neighbor connections with other FortiGates or routers, and the learned routes are put into different VRF tables according to the neighbor's settings.

This example uses the following topology:



- BGP routes learned from the Router1 neighbor are put into vrf10.
- BGP routes learned from the Router2 neighbor are put into vrf20.

To configure this example:

```
config system interface
    edit port1
        set vrf 10
    next
    edit port2
        set vrf 20
    next
end

config router bgp
    config neighbor
        edit "192.168.1.1"
            set update-source port1
        next
        edit "192.168.2.1"
            set interface port2
        next
```

```
    end
end
```

Results

Using the above topology:

- Both Router1 and Router2 establish OSPF and BGP neighbor with the FortiGate.
- Router1 advertises 10.10.1.0/24 into OSPF and 10.10.2.0/24 into BGP.
- Router2 advertises 20.20.1.0/24 into OSPF and 20.20.2.0/24 into BGP.

When port1 and port2 have not set VRF, all of the routing is in VRF=0:

```
# get router info routing-table all
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default

Routing table for VRF=0
S*      0.0.0.0/0 [5/0] via 10.0.1.254, port9
C       10.0.1.0/24 is directly connected, port9
O       10.10.1.0/24 [110/10] via 192.168.1.1, port1, 00:18:31
B       10.10.2.0/24 [20/200] via 192.168.1.1, port1, 00:01:31
O       20.20.1.0/22 [110/10] via 192.168.2.1, port2, 00:19:05
B       20.20.2.0/24 [20/200] via 192.168.2.1, port2, 00:01:31
C       192.168.1.0/24 is directly connected, port1
C       192.168.2.0/24 is directly connected, port2
```

After VRF is set for BGP, BGP routes are added to the VRF tables along with OSPF and connected routes:

```
# get router info routing-table all
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default

Routing table for VRF=0
S*      0.0.0.0/0 [5/0] via 10.0.1.254, port9
C       10.0.1.0/24 is directly connected, port9

Routing table for VRF=10
O       10.10.1.0/24 [110/10] via 192.168.1.1, port1, 00:18:31
B       10.10.2.0/24 [20/200] via 192.168.1.1, port1, 00:01:31
C       192.168.1.0/24 is directly connected, port1

Routing table for VRF=20
O       20.20.1.0/22 [110/10] via 192.168.2.1, port2, 00:19:05
B       20.20.2.0/24 [20/200] via 192.168.2.1, port2, 00:01:31
C       192.168.2.0/24 is directly connected, port2
```

BGP neighbor groups

This feature is also supported in the BGP neighbor groups. For example:

```
config router bgp
    config neighbor-group
        edit "FGT"
            set update-source "port1"
        next
    end
    config neighbor-range
        edit 1
            set prefix 172.16.201.0 255.255.255.0
            set neighbor-group "FGT"
        next
    end
end
```

Note that the `set interface` command is not supported.

NetFlow

NetFlow allows you to collect IP network traffic statistics for an interface, and then export those statistics for analysis. NetFlow samplers, that sample every packet, are configured per interface. Full NetFlow is supported through the information maintained in the firewall session.

To configure NetFlow:

```
config system netflow
    set collector-ip <ip>
    set collector-port <port>
    set source-ip <ip>
    set active-flow-timeout <integer>
    set inactive-flow-timeout <integer>
    set template-tx-timeout <integer>
    set template-tx-counter <integer>
end
```

`collector-ip <ip>`	Collector IP address.
`collector-port <port>`	NetFlow collector port number (0 - 65535).
`source-ip <ip>`	Source IP address, for communication with the NetFlow agent.
`active-flow-timeout <integer>`	Timeout to report active flows, in seconds (60 - 3600, default = 1800).
`inactive-flow-timeout <integer>`	Timeout for periodic report of finished flows, in seconds (10 - 600, default = 15).
`template-tx-timeout <integer>`	Timeout for periodic template flowset transmission, in seconds (60 - 86400, default = 1800).
`template-tx-counter <integer>`	Counter of flowset records, before resending a template flowset record (10 - 6000, default = 20).

To configure NetFlow in a specific VDOM:

```
config vdom
    edit <vdom>
        config system vdom-netflow
            set vdom-netflow enable
            set collector-ip <ip>
            set collector-port <port>
            set source-ip <ip>
        end
    next
end
```

To configure a NetFlow sampler on an interface:

```
config system interface
    edit <interface>
        set netflow-sampler {disable | tx | rx | both}
    next
end
```

`disable`	Disable the NetFlow protocol on this interface (default).
`tx`	Monitor transmitted traffic on this interface.
`rx`	Monitor received traffic on this interface.
`both`	Monitor transmitted/received traffic on this interface.

Verification and troubleshooting

If data are not seen on the NetFlow collector after it has been configured, use the following sniffer commands to verify if the FortiGate and the collector are communicating:

- By collector port:

  ```
  # diagnose sniffer packet 'port <collector-port>'  6 0 a
  ```

- By collector IP address:

  ```
  # diagnose sniffer packet 'host <collector-ip>' 6 0 a
  ```

NetFlow uses the sflow daemon. The current NetFlow configuration can be viewed using test level 3 or 4:

```
# diagnose test application sflowd 3
```

```
# diagnose test application sflowd 4
Netflow Cache Stats:
vdoms=1 Collectors=1 Cached_intf=2 Netflow_enabled_intf=1 Live_sessions=0 Session cache max
count:71950
```

NetFlow templates

NetFlow uses templates to capture and categorize the data that it collects. FortiOS supports the following NetFlow templates:

Name	Template ID	Description
STAT_OPTIONS	256	Statistics information about exporter
APP_ID_OPTIONS	257	Application information
IPV4	258	No NAT IPv4 traffic
IPV6	259	No NAT IPv6 traffic
ICMP4	260	No NAT ICMPv4 traffic
ICMP6	261	No NAT ICMPv6 traffic
IPV4_NAT	262	Source/Destination NAT IPv4 traffic
IPV4_AF_NAT	263	AF NAT IPv4 traffic (4->6)
IPV6_NAT	264	Source/Destination NAT IPv6 traffic
IPV6_AF_NAT	265	AF NAT IPv6 traffic (6->4)
ICMP4_NAT	266	Source/Destination NAT ICMPv4 traffic
ICMP4_AF_NAT	267	AF NAT ICMPv4 traffic (4->6)
ICMP6_NAT	268	Source/Destination NAT ICMPv6 traffic
ICMPv6_AF_NAT	269	AF NAT ICMPv6 traffic (6->4)

256 - STAT_OPTIONS

Description	Statistics information about exporter
Scope Field Count	1
Data Field Count	7
Option Scope Length	4
Option Length	28
Padding	0000

Scope fields

Field #	Field	Type	Length
1	System	System (1)	2

Data fields

Field #	Field	Type	Length
1	TOTAL_BYTES_EXP	TOTAL_BYTES_EXP (40)	8

Field #	Field	Type	Length
2	TOTAL_PKTS_EXP	TOTAL_PKTS_EXP (41)	8
3	TOTAL_FLOWS_EXP	TOTAL_FLOWS_EXP (42)	8
4	FLOW_ACTIVE_TIMEOUT	FLOW_ACTIVE_TIMEOUT (36)	2
5	FLOW_INACTIVE_TIMEOUT	FLOW_INACTIVE_TIMEOUT (37)	2
6	SAMPLING_INTERVAL	SAMPLING_INTERVAL (34)	4
7	SAMPLING_ALGORITHM	SAMPLING_ALGORITHM (35)	1

257 - APP_ID_OPTIONS

Description	Application information
Scope Field Count	1
Data Field Count	4
Option Scope Length	4
Option Length	16
Padding	0000

Scope fields

Field #	Field	Type	Length
1	System	System (1)	2

Data fields

Field #	Field	Type	Length
1	APPLICATION_ID	APPLICATION_ID (95)	9
2	APPLICATION_NAME	APPLICATION_NAME (96)	64
3	APPLICATION_DESC	APPLICATION_DESC (94)	64
4	applicationCategoryName	applicationCategoryName (372)	32

258 - IPV4

Description	No NAT IPv4 traffic
Data Field Count	17

Data fields

Field #	Field	Type	Length
1	BYTES	BYTES (1)	8
2	OUT_BYTES	OUT_BYTES (23)	8
3	PKTS	PKTS (2)	4
4	OUT_PKTS	OUT_PKTS (24)	4
5	FIRST_SWITCHED	FIRST_SWITCHED (22)	4
6	LAST_SWITCHED	LAST_SWITCHED (21)	4
7	L4_SRC_PORT	L4_SRC_PORT (7)	2
8	L4_DST_PORT	L4_DST_PORT (11)	2
9	INPUT_SNMP	INPUT_SNMP (10)	2
10	OUTPUT_SNMP	OUTPUT_SNMP (14)	2
11	PROTOCOL	PROTOCOL (4)	1
12	APPLICATION_ID	APPLICATION_ID (95)	9
13	FLOW_FLAGS	FLOW_FLAGS (65)	2
14	FORWARDING_STATUS	FORWARDING_STATUS (89)	1
15	flowEndReason	flowEndReason (136)	1
16	IP_SRC_ADDR	IP_SRC_ADDR (8)	4
17	IP_DST_ADDR	IP_DST_ADDR (12)	4

259 - IPV6

Description	No NAT IPv6 traffic
Data Field Count	17

Data fields

Field #	Field	Type	Length
1	BYTES	BYTES (1)	8
2	OUT_BYTES	OUT_BYTES (23)	8
3	PKTS	PKTS (2)	4
4	OUT_PKTS	OUT_PKTS (24)	4
5	FIRST_SWITCHED	FIRST_SWITCHED (22)	4

Field #	Field	Type	Length
6	LAST_SWITCHED	LAST_SWITCHED (21)	4
7	L4_SRC_PORT	L4_SRC_PORT (7)	2
8	L4_DST_PORT	L4_DST_PORT (11)	2
9	INPUT_SNMP	INPUT_SNMP (10)	2
10	OUTPUT_SNMP	OUTPUT_SNMP (14)	2
11	PROTOCOL	PROTOCOL (4)	1
12	APPLICATION_ID	APPLICATION_ID (95)	9
13	FLOW_FLAGS	FLOW_FLAGS (65)	2
14	FORWARDING_STATUS	FORWARDING_STATUS (89)	1
15	flowEndReason	flowEndReason (136)	1
16	IPV6_SRC_ADDR	IPV6_SRC_ADDR (27)	16
17	IPV6_DST_ADDR	IPV6_DST_ADDR (28)	16

260 - ICMP4

Description	No NAT ICMPv4 traffic
Data Field Count	16

Data fields

Field #	Field	Type	Length
1	BYTES	BYTES (1)	8
2	OUT_BYTES	OUT_BYTES (23)	8
3	PKTS	PKTS (2)	4
4	OUT_PKTS	OUT_PKTS (24)	4
5	FIRST_SWITCHED	FIRST_SWITCHED (22)	4
6	LAST_SWITCHED	LAST_SWITCHED (21)	4
7	INPUT_SNMP	INPUT_SNMP (10)	2
8	OUTPUT_SNMP	OUTPUT_SNMP (14)	2
9	ICMP_TYPE	ICMP_TYPE (32)	2
10	PROTOCOL	PROTOCOL (4)	1
11	APPLICATION_ID	APPLICATION_ID (95)	9

Field #	Field	Type	Length
12	FLOW_FLAGS	FLOW_FLAGS (65)	2
13	FORWARDING_STATUS	FORWARDING_STATUS (89)	1
14	flowEndReason	flowEndReason (136)	1
15	IP_SRC_ADDR	IP_SRC_ADDR (8)	4
16	IP_DST_ADDR	IP_DST_ADDR(12)	4

261 - ICMP6

Description	No NAT ICMPv6 traffic
Data Field Count	16

Data fields

Field #	Field	Type	Length
1	BYTES	BYTES (1)	8
2	OUT_BYTES	OUT_BYTES (23)	8
3	PKTS	PKTS (2)	4
4	OUT_PKTS	OUT_PKTS (24)	4
5	FIRST_SWITCHED	FIRST_SWITCHED (22)	4
6	LAST_SWITCHED	LAST_SWITCHED (21)	4
7	INPUT_SNMP	INPUT_SNMP (10)	2
8	OUTPUT_SNMP	OUTPUT_SNMP (14)	2
9	ICMP_TYPE	ICMP_TYPE (32)	2
10	PROTOCOL	PROTOCOL (4)	1
11	APPLICATION_ID	APPLICATION_ID (95)	9
12	FLOW_FLAGS	FLOW_FLAGS (65)	2
13	FORWARDING_STATUS	FORWARDING_STATUS (89)	1
14	flowEndReason	flowEndReason (136)	1
15	IPV6_SRC_ADDR	IPV6_SRC_ADDR (27)	16
16	IPV6_DST_ADDR	IPV6_DST_ADDR (28)	16

262 - IPV4_NAT

Description	Source/Destination NAT IPv4 traffic
Data Field Count	25

Data fields

Field #	Field	Type	Length
1	BYTES	BYTES (1)	8
2	OUT_BYTES	OUT_BYTES (23)	8
3	PKTS	PKTS (2)	4
4	OUT_PKTS	OUT_PKTS (24)	4
5	FIRST_SWITCHED	FIRST_SWITCHED (22)	4
6	LAST_SWITCHED	LAST_SWITCHED (21)	4
7	L4_SRC_PORT	L4_SRC_PORT (7)	2
8	L4_DST_PORT	L4_DST_PORT (11)	2
9	INPUT_SNMP	INPUT_SNMP (10)	2
10	OUTPUT_SNMP	OUTPUT_SNMP (14)	2
11	PROTOCOL	PROTOCOL (4)	1
12	postIpDiffServCodePoint	postIpDiffServCodePoint (98)	1
13	IP_TOS	ipClassofService (5)	1
14	DST_DOS	postIpClassOfService (55)	1
15	APPLICATION_ID	APPLICATION_ID (95)	9
16	INTERNET_APPLICATION_ID	INTERNET_APPLICATION_ID(66)	4
17	FLOW_FLAGS	FLOW_FLAGS (65)	2
18	FORWARDING_STATUS	FORWARDING_STATUS (89)	1
19	flowEndReason	flowEndReason (136)	1
20	IP_SRC_ADDR	IP_SRC_ADDR (8)	4
21	IP_DST_ADDR	IP_DST_ADDR (12)	4
22	postNATSourceIPv4Address	postNATSourceIPv4Address (225)	4
23	postNATDestinationIPv4Address	postNATDestinationIPv4Address (226)	4
24	postNAPTSourceTransportPort	postNAPTSourceTransportPort (227)	2
25	postNAPTDestinationTransportPort	postNAPTDestinationTransportPort (228)	2

263 - IPV4_AF_NAT

Description	AF NAT IPv4 traffic (4->6)
Data Field Count	21

Data fields

Field #	Field	Type	Length
1	BYTES	BYTES (1)	8
2	OUT_BYTES	OUT_BYTES (23)	8
3	PKTS	PKTS (2)	4
4	OUT_PKTS	OUT_PKTS (24)	4
5	FIRST_SWITCHED	FIRST_SWITCHED (22)	4
6	LAST_SWITCHED	LAST_SWITCHED (21)	4
7	L4_SRC_PORT	L4_SRC_PORT (7)	2
8	L4_DST_PORT	L4_DST_PORT (11)	2
9	INPUT_SNMP	INPUT_SNMP (10)	2
10	OUTPUT_SNMP	OUTPUT_SNMP (14)	2
11	PROTOCOL	PROTOCOL (4)	1
12	APPLICATION_ID	APPLICATION_ID (95)	9
13	FLOW_FLAGS	FLOW_FLAGS (65)	2
14	FORWARDING_STATUS	FORWARDING_STATUS (89)	1
15	flowEndReason	flowEndReason (136)	1
16	IPV6_SRC_ADDR	IPV6_SRC_ADDR (27)	16
17	IPV6_DST_ADDR	IPV6_DST_ADDR (28)	16
18	postNATSourceIPv6Address	postNATSourceIPv6Address (281)	16
19	postNATDestinationIPv6Address	postNATDestinationIPv6Address (282)	16
20	postNAPTSourceTransportPort	postNAPTSourceTransportPort (227)	2
21	postNAPTDestinationTransportPort	postNAPTDestinationTransportPort (228)	2

264 - IPV6_NAT

Description	Source/Destination NAT IPv6 traffic
Data Field Count	21

Data fields

Field #	Field	Type	Length
1	BYTES	BYTES (1)	8
2	OUT_BYTES	OUT_BYTES (23)	8
3	PKTS	PKTS (2)	4
4	OUT_PKTS	OUT_PKTS (24)	4
5	FIRST_SWITCHED	FIRST_SWITCHED (22)	4
6	LAST_SWITCHED	LAST_SWITCHED (21)	4
7	L4_SRC_PORT	L4_SRC_PORT (7)	2
8	L4_DST_PORT	L4_DST_PORT (11)	2
9	INPUT_SNMP	INPUT_SNMP (10)	2
10	OUTPUT_SNMP	OUTPUT_SNMP (14)	2
11	PROTOCOL	PROTOCOL (4)	1
12	APPLICATION_ID	APPLICATION_ID (95)	9
13	FLOW_FLAGS	FLOW_FLAGS (65)	2
14	FORWARDING_STATUS	FORWARDING_STATUS (89)	1
15	flowEndReason	flowEndReason (136)	1
16	IP_SRC_ADDR	IP_SRC_ADDR (8)	4
17	IP_DST_ADDR	IP_DST_ADDR (12)	4
18	postNATSourceIPv6Address	postNATSourceIPv6Address (281)	16
19	postNATDestinationIPv6Address	postNATDestinationIPv6Address (282)	16
20	postNAPTSourceTransportPort	postNAPTSourceTransportPort (227)	2
21	postNAPTDestinationTransportPort	postNAPTDestinationTransportPort (228)	2

265 - IPV6_AF_NAT

Description	AF NAT IPv6 traffic (6->4)
Data Field Count	21

Data fields

Field #	Field	Type	Length
1	BYTES	BYTES (1)	8
2	OUT_BYTES	OUT_BYTES (23)	8
3	PKTS	PKTS (2)	4
4	OUT_PKTS	OUT_PKTS (24)	4
5	FIRST_SWITCHED	FIRST_SWITCHED (22)	4
6	LAST_SWITCHED	LAST_SWITCHED (21)	4
7	L4_SRC_PORT	L4_SRC_PORT (7)	2
8	L4_DST_PORT	L4_DST_PORT (11)	2
9	INPUT_SNMP	INPUT_SNMP (10)	2
10	OUTPUT_SNMP	OUTPUT_SNMP (14)	2
11	PROTOCOL	PROTOCOL (4)	1
12	APPLICATION_ID	APPLICATION_ID (95)	9
13	FLOW_FLAGS	FLOW_FLAGS (65)	2
14	FORWARDING_STATUS	FORWARDING_STATUS (89)	1
15	flowEndReason	flowEndReason (136)	1
16	IPV6_SRC_ADDR	IPV6_SRC_ADDR (27)	16
17	IPV6_DST_ADDR	IPV6_DST_ADDR (28)	16
18	postNATSourceIPv4Address	postNATSourceIPv4Address (225)	4
19	postNATDestinationIPv4Address	postNATDestinationIPv4Address (226)	4
20	postNAPTSourceTransportPort	postNAPTSourceTransportPort (227)	2
21	postNAPTDestinationTransportPort	postNAPTDestinationTransportPort (228)	2

266 - ICMPV4_NAT

Description	Source/Destination NAT ICMPv4 traffic
Data Field Count	20

Data fields

Field #	Field	Type	Length
1	BYTES	BYTES (1)	8
2	OUT_BYTES	OUT_BYTES (23)	8
3	PKTS	PKTS (2)	4
4	OUT_PKTS	OUT_PKTS (24)	4
5	FIRST_SWITCHED	FIRST_SWITCHED (22)	4
6	LAST_SWITCHED	LAST_SWITCHED (21)	4
7	INPUT_SNMP	INPUT_SNMP (10)	2
8	OUTPUT_SNMP	OUTPUT_SNMP (14)	2
9	ICMP_TYPE	ICMP_TYPE (32)	2
10	PROTOCOL	PROTOCOL (4)	1
11	APPLICATION_ID	APPLICATION_ID (95)	9
12	FLOW_FLAGS	FLOW_FLAGS (65)	2
13	FORWARDING_STATUS	FORWARDING_STATUS (89)	1
14	flowEndReason	flowEndReason (136)	1
15	IP_SRC_ADDR	IP_SRC_ADDR (8)	4
16	IP_DST_ADDR	IP_DST_ADDR (12)	4
17	postNATSourceIPv4Address	postNATSourceIPv4Address (225)	4
18	postNATDestinationIPv4Address	postNATDestinationIPv4Address (226)	4
19	postNAPTSourceTransportPort	postNAPTSourceTransportPort (227)	2
20	postNAPTDestinationTransportPort	postNAPTDestinationTransportPort (228)	2

267 - ICMPV4_AF_NAT

Description	AF NAT ICMPv4 traffic (4->6)
Data Field Count	20

Data fields

Field #	Field	Type	Length
1	BYTES	BYTES (1)	8

Field #	Field	Type	Length
2	OUT_BYTES	OUT_BYTES (23)	8
3	PKTS	PKTS (2)	4
4	OUT_PKTS	OUT_PKTS (24)	4
5	FIRST_SWITCHED	FIRST_SWITCHED (22)	4
6	LAST_SWITCHED	LAST_SWITCHED (21)	4
7	INPUT_SNMP	INPUT_SNMP (10)	2
8	OUTPUT_SNMP	OUTPUT_SNMP (14)	2
9	ICMP_TYPE	ICMP_TYPE (32)	2
10	PROTOCOL	PROTOCOL (4)	1
11	APPLICATION_ID	APPLICATION_ID (95)	9
12	FLOW_FLAGS	FLOW_FLAGS (65)	2
13	FORWARDING_STATUS	FORWARDING_STATUS (89)	1
14	flowEndReason	flowEndReason (136)	1
15	IPV6_SRC_ADDR	IPV6_SRC_ADDR (27)	16
16	IPV6_DST_ADDR	IPV6_DST_ADDR (28)	16
17	postNATSourceIPv6Address	postNATSourceIPv6Address (281)	16
18	postNATDestinationIPv6Address	postNATDestinationIPv6Address (282)	16
19	postNAPTSourceTransportPort	postNAPTSourceTransportPort (227)	2
20	postNAPTDestinationTransportPort	postNAPTDestinationTransportPort (228)	2

268 - ICMPV6_NAT

Description	Source/Destination NAT ICMPv6 traffic
Data Field Count	20

Data fields

Field #	Field	Type	Length
1	BYTES	BYTES (1)	8
2	OUT_BYTES	OUT_BYTES (23)	8
3	PKTS	PKTS (2)	4
4	OUT_PKTS	OUT_PKTS (24)	4

Field #	Field	Type	Length
5	FIRST_SWITCHED	FIRST_SWITCHED (22)	4
6	LAST_SWITCHED	LAST_SWITCHED (21)	4
7	INPUT_SNMP	INPUT_SNMP (10)	2
8	OUTPUT_SNMP	OUTPUT_SNMP (14)	2
9	ICMP_TYPE	ICMP_TYPE (32)	2
10	PROTOCOL	PROTOCOL (4)	1
11	APPLICATION_ID	APPLICATION_ID (95)	9
12	FLOW_FLAGS	FLOW_FLAGS (65)	2
13	FORWARDING_STATUS	FORWARDING_STATUS (89)	1
14	flowEndReason	flowEndReason (136)	1
15	IP_SRC_ADDR	IP_SRC_ADDR (8)	4
16	IP_DST_ADDR	IP_DST_ADDR (12)	4
17	postNATSourceIPv6Address	postNATSourceIPv6Address (281)	16
18	postNATDestinationIPv6Address	postNATDestinationIPv6Address (282)	16
19	postNAPTSourceTransportPort	postNAPTSourceTransportPort (227)	2
20	postNAPTDestinationTransportPort	postNAPTDestinationTransportPort (228)	2

269 - ICMPV6_AF_NAT

Description	AF NAT ICMPv6 traffic (6->4)
Data Field Count	20

Data fields

Field #	Field	Type	Length
1	BYTES	BYTES (1)	8
2	OUT_BYTES	OUT_BYTES (23)	8
3	PKTS	PKTS (2)	4
4	OUT_PKTS	OUT_PKTS (24)	4
5	FIRST_SWITCHED	FIRST_SWITCHED (22)	4
6	LAST_SWITCHED	LAST_SWITCHED (21)	4
7	INPUT_SNMP	INPUT_SNMP (10)	2

Field #	Field	Type	Length
8	OUTPUT_SNMP	OUTPUT_SNMP (14)	2
9	ICMP_TYPE	ICMP_TYPE (32)	2
10	PROTOCOL	PROTOCOL (4)	1
11	APPLICATION_ID	APPLICATION_ID (95)	9
12	FLOW_FLAGS	FLOW_FLAGS (65)	2
13	FORWARDING_STATUS	FORWARDING_STATUS (89)	1
14	flowEndReason	flowEndReason (136)	1
15	IPV6_SRC_ADDR	IPV6_SRC_ADDR (27)	16
16	IPV6_DST_ADDR	IPV6_DST_ADDR (28)	16
17	postNATSourceIPv4Address	postNATSourceIPv4Address (225)	4
18	postNATDestinationIPv4Address	postNATDestinationIPv4Address (226)	4
19	postNAPTSourceTransportPort	postNAPTSourceTransportPort (227)	2
20	postNAPTDestinationTransportPort	postNAPTDestinationTransportPort (228)	2

NetFlow on FortiExtender and tunnel interfaces

NetFlow sampling is supported on FortiExtender and VPN tunnel interfaces.

VPN tunnel interfaces can be IPsec, IP in IP, or GRE tunnels. NetFlow sampling is supported on both NPU and non-NPU offloaded tunnels.

Examples

In the following examples, a FortiExtender and a VPN tunnel interface are configured with NetFlow sampling.

To configure a FortiExtender interface with NetFlow sampling:

1. Configure a FortiExtender interface with NetFlow sampling enabled for both transmitted and received traffic:

```
config system interface
    edit "fext-211"
        set vdom "root"
        set mode dhcp
        set type fext-wan
        set netflow-sampler both
        set role wan
        set snmp-index 8
        set macaddr 2a:4e:68:a3:f4:6a
    next
end
```

2. Check the NetFlow status and configuration:

Device index 26 is the FortiExtender interface `fext-211`.

```
# diagnose test application sflowd 3
===== Netflow Vdom Configuration =====
Global collector:172.18.60.80:[2055] source ip: 0.0.0.0 active-timeout(seconds):60
inactive-timeout(seconds):600
____ vdom: root, index=0, is master, collector: disabled (use global config) (mgmt vdom)
   |_ coll_ip:172.18.60.80[2055],src_ip:10.6.30.105,seq_num:300,pkts/time to next
template: 18/29
   |_ exported: Bytes:3026268, Packets:11192, Sessions:290 Flows:482
   |____ interface:fext-211 sample_direction:both device_index:26 snmp_index:8
```

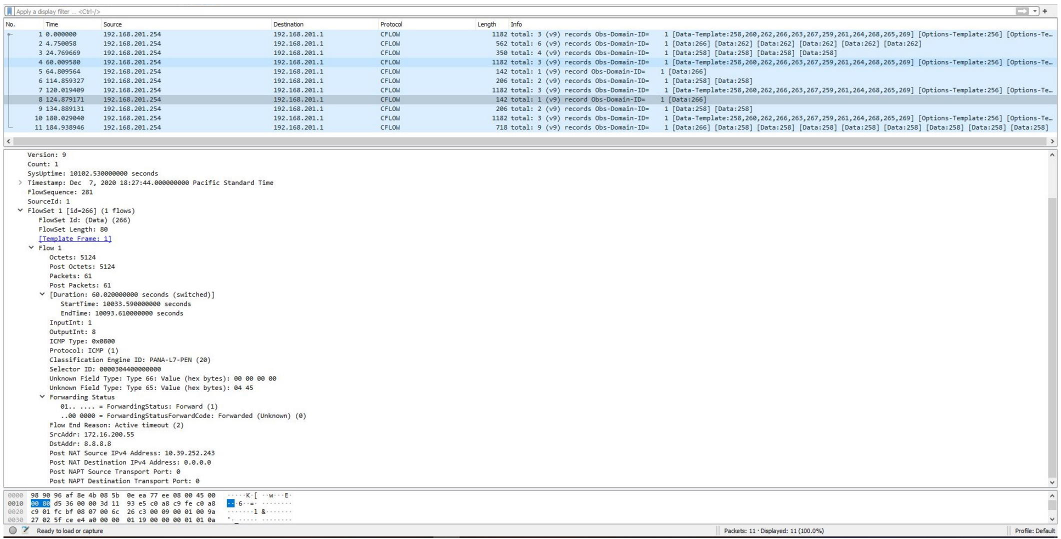
3. Check the network interface list:

```
# diagnose netlink interface list
...
if=fext-211 family=00 type=1 index=26 mtu=1500 link=0 master=0
ref=27 state=start present fw_flags=60000 flags=up broadcast run multicast
...
```

4. Check the session list for the FortiExtender interface and NetFlow flowset packet:

```
# diagnose sys session list
session info: proto=1 proto_state=00 duration=1732 expire=59 timeout=0 flags=00000000
socktype=0 sockport=0 av_idx=0 use=3
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/255
state=may_dirty netflow-origin netflow-reply
statistic(bytes/packets/allow_err): org=145572/1733/1 reply=145572/1733/1 tuples=2
tx speed(Bps/kbps): 83/0 rx speed(Bps/kbps): 83/0
orgin->sink: org pre->post, reply pre->post dev=5->26/26->5
gwy=10.39.252.244/172.16.200.55
hook=post dir=org act=snat 172.16.200.55:61290->8.8.8.8:8(10.39.252.243:61290)
hook=pre dir=reply act=dnat 8.8.8.8:61290->10.39.252.243:0(172.16.200.55:61290)
misc=0 policy_id=1 auth_info=0 chk_client_info=0 vd=0
serial=00001298 tos=ff/ff app_list=0 app=0 url_cat=0
sdwan_mbr_seq=0 sdwan_service_id=0
rpdb_link_id=00000000 rpdb_svc_id=0 ngfwid=n/a
npu_state=0x040000
no_ofld_reason:  non-npu-intf
total session 1
```

5. The flowset packet can be captured on UDP port 2055 by a packet analyzer, such as Wireshark:



To configure a VPN tunnel interface with NetFlow sampling:

1. Configure a VPN interface with NetFlow sampling enabled for both transmitted and received traffic:

```
config system interface
    edit "A-to-B_vpn"
        set vdom "vdom1"
        set type tunnel
        set netflow-sampler both
        set snmp-index 42
        set interface "port3"
    next
end
```

2. Configure the VPN tunnel:

```
config vpn ipsec phase1-interface
    edit "A-to-B_vpn"
        set interface "port3"
        set peertype any
        set net-device disable
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set comments "VPN: A-to-B_vpn [Created by VPN wizard]"
        set wizard-type static-fortigate
        set remote-gw 10.2.2.2
        set psksecret ENC
    next
end
```

```
config vpn ipsec phase2-interface
    edit "A-to-B_vpn"
        set phase1name "A-to-B_vpn"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
aes256gcm chacha20poly1305
        set comments "VPN: A-to-B_vpn [Created by VPN wizard]"
        set src-addr-type name
        set dst-addr-type name
        set src-name "A-to-B_vpn_local"
```

```
        set dst-name "A-to-B_vpn_remote"
    next
end
```

3. Check the NetFlow status and configuration:

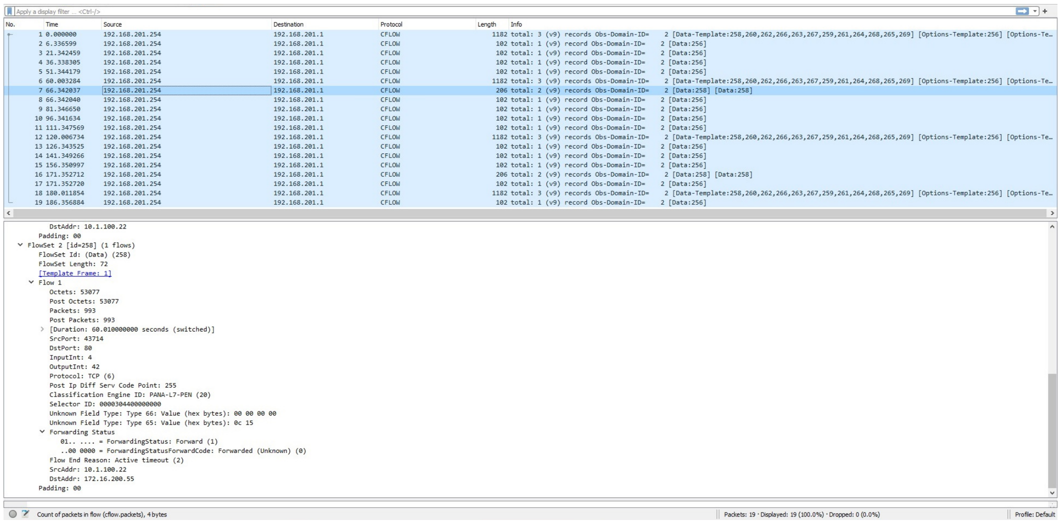
   Device index 52 is the VPN interface `A-to-B_vpn`.

```
# diagnose test application sflowd 3
===== Netflow Vdom Configuration =====
Global collector:172.18.60.80:[2055] source ip: 0.0.0.0 active-timeout(seconds):60
inactive-timeout(seconds):15
____ vdom: vdom1, index=1, is master, collector: disabled (use global config) (mgmt
vdom)
   |_ coll_ip:172.18.60.80[2055],src_ip:10.1.100.1,seq_num:60,pkts/time to next
template: 15/6
   |_ exported: Bytes:11795591, Packets:48160, Sessions:10 Flows:34
   |____ interface:A-to-B_vpn sample_direction:both device_index:52 snmp_index:42
```

4. Check the session list for the VPN interface and NetFlow flowset packet (unencapsulated traffic going through the VPN tunnel):

```
# diagnose sys session list
session info: proto=6 proto_state=01 duration=6 expire=3599 timeout=3600 flags=00000000
socktype=0 sockport=0 av_idx=0 use=3
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/255
state=may_dirty npu netflow-origin netflow-reply
statistic(bytes/packets/allow_err): org=6433/120/1 reply=884384/713/1 tuples=2
tx speed(Bps/kbps): 992/7 rx speed(Bps/kbps): 136479/1091
orgin->sink: org pre->post, reply pre->post dev=10->52/52->10 gwy=10.2.2.2/10.1.100.22
hook=pre dir=org act=noop 10.1.100.22:43714->172.16.200.55:80(0.0.0.0:0)
hook=post dir=reply act=noop 172.16.200.55:80->10.1.100.22:43714(0.0.0.0:0)
pos/(before,after) 0/(0,0), 0/(0,0)
src_mac=00:0c:29:ac:ae:4f
misc=0 policy_id=5 auth_info=0 chk_client_info=0 vd=1
serial=00003b6c tos=ff/ff app_list=0 app=0 url_cat=0
sdwan_mbr_seq=0 sdwan_service_id=0
rpdb_link_id=00000000 rpdb_svc_id=0 ngfwid=n/a
npu_state=0x000001 no_offload
npu info: flag=0x82/0x00, offload=0/0, ips_offload=0/0, epid=0/0, ipid=0/0,
vlan=0x0000/0x0000
vlifid=0/0, vtag_in=0x0000/0x0000 in_npu=0/0, out_npu=0/0, fwd_en=0/0, qid=0/0
no_ofld_reason:  disabled-by-policy
total session 1
```

5. The flowset packet can be captured on UDP port 2055 by a packet analyzer, such as Wireshark:



sFlow

sFlow is a method of monitoring the traffic on your network to identify areas on the network that may impact performance and throughput. FortiGate supports sFlow v5. sFlow collector software is available from a number of third-party software vendors. For more information about sFlow, see www.sflow.org.

The packet information that the FortiGate's sFlow agent collects depends on the interface type:

- On an internal interface, when the interface receives packets from devices with private IP addresses, the collected information includes the private IP addresses.
- On an external, or WAN, interface, when the interface receives to route to or from the internet, the collected information includes the IP address of the WAN interface as the source or destination interface, depending on the direction of the traffic. It does not include IP addresses that are NATed on another interface.

sFlow datagrams contain the following information:

- Packet headers, such as MAC, IPv4, and TCP
- Sample process parameters, such as rate and pool
- Input and output ports
- Priority (802.1p and ToS)
- VLAN (802.1Q)
- Source prefixes, destination prefixes, and next hop addresses
- BGP source AS, source peer AS, destination peer AS, communities, and local preference
- User IDs (TACACS, RADIUS) for source and destination
- Interface statistics (RFC 1573, RFC 2233, and RFC 2358)

Configuring sFlow

sFlow can be configured globally, then on traffic VDOMs and individual interfaces.

When configuring sFlow on a VDOM, the collector can be specified, or the collector that is configured globally can be used.

sFlow is supported on some interface types, such as physical, VLAN, and aggregate. It is not supported on virtual interfaces, such as VDOM link, IPsec, GRE, or SSL. When configuring sFlow on an interface, the rate that the agent samples traffic, the direction of that traffic, and the frequency that the agent sends sFlow datagrams to the sFlow collector can be specified. If sFlow is configured on the VDOM that the interface belongs to, the agent sends datagrams to the collector configured for the VDOM. Otherwise, the datagrams are sent to the collector that is configured globally.

Configuring sFlow for an interface disables NP offloading for all traffic on that interface.

To configure sFlow globally:

```
config system sflow
    set collector-ip <ipv4_address>
    set collector-port <port>
    set source-ip <ipv4_address>
    set interface-select-method {auto | sdwan | specify}
    set interface <interface>
end
```

`collector-ip <ipv4_address>`	The IPv4 address of the sFlow collector that sFlow agents added to interface (default = 0.0.0.0).
`collector-port <port>`	The UDP port number used for sending sFlow datagrams (0 - 65535, default = 6343). Only configured this option if required by the sFlow collector or your network configuration.
`source-ip <ipv4_address>`	The source IPv4 address that the sFlow agent used to send datagrams to the collector (default = 0.0.0.0). If this option is not configured, the FortiGate uses the IP address of the interface that it sends the datagram through.
`interface-select-method {auto | sdwan | specify}`	How the outgoing interface to reach the server is selected (default = auto).
`interface <interface>`	The outgoing interface used to reach the server. This option is only available when `interface-select-method` is `specify`.

To configure sFlow for a VDOM:

```
config vdom
    edit <vdom>
        config system vdom-sflow
            set vdom-sflow {enable | disable}
            set collector-ip <ipv4_address>
            set collector-port <port>
            set source-ip <ipv4_address>
            set interface-select-method {auto | sdwan | specify}
            set interface <interface>
        end
    next
end
```

`vdom-sflow {enable | disable}`	Enable/disable the sFlow configuration for the current VDOM (default = disable).
`collector-ip <ipv4_address>`	The IPv4 address of the sFlow collector that sFlow agents added to interface (default = 0.0.0.0). If this option is not configured, the global setting will be used.
`collector-port <port>`	The UDP port number used for sending sFlow datagrams (0 - 65535, default = 6343). Only configured this option if required by the sFlow collector or your network configuration. If this option is not configured, the global setting will be used.
`source-ip <ipv4_address>`	The source IPv4 address that the sFlow agent used to send datagrams to the collector (default = 0.0.0.0). If this option is not configured, the FortiGate uses the IP address of the interface that it sends the datagram through.
`interface-select-method {auto | sdwan | specify}`	How the outgoing interface to reach the server is selected (default = auto).
`interface <interfae>`	The outgoing interface used to reach the server. This option is only available when `interface-select-method` is `specify`.

To configure sFlow on an interface:

```
config system interface
    edit <interface>
        set sflow-sampler {enable | disable}
        set sample-rate <integer>
        set polling-interval <integer>
        set sample-direction {tx | rx | both}
    next
end
```

`sflow-sampler {enable | disable}`	Enable/disable sFlow on this interface (default = disable).
`sample-rate <integer>`	The average number of packets that the agent lets pass before taking a sample (10 - 99999, default = 2000). Setting a lower rate will sample a higher number of packets, increasing the accuracy or the sampling data, but also increasing the CPU and network bandwidth usage. The default value is recommended.
`polling-interval <integer>`	The amount of time that the agent waits between sending datagrams to the collector, in seconds (1 - 255, default = 20). Setting a higher value lowers the amount of data that the agent sends across the network, but makes the collector's view of the network less current.
`sample-direction {tx | rx | both}`	The direction of the traffic that the agent collects (default = both).

Link monitor

The link monitor is a mechanism that allows the FortiGate to probe the status of a detect server in order to determine the health of the link, next hop, or the path to the server. Ping, TCP echo, UDP echo, HTTP, and TWAMP protocols can be used for the probes. Typically, the detect server is set to a stable server several hops away. Multiple servers can also be configured with options to define the protocol and weights for each server.

The link monitor serves several purposes. In the most basic configuration, it can be used to detect failures and remove routes associated with the interface and gateway to prevent traffic from routing out the failed link. More granularity is added in 7.0 that allows only the routes specified in the link monitor to be removed from the routing table. With this benefit, only traffic to specific routing destinations are removed, rather than all routing destinations.

Another enhancement starting in 7.0.1 is an option to toggle between enabling or disabling policy route updates when a link health monitor fails.

The link monitor can also monitor remote servers for HA failover. Using the HA built-in link monitor, it is only able to detect physical link failovers to trigger HA link failover. With the link monitor, remote servers can be used to monitor the health of the path to the server in order to trigger HA failover.

Finally, the link monitor can cascade the failure to other interfaces. When the `update-cascade-interface` option is enabled, the interface can be configured in conjunction with `fail-detect` enabled to trigger a link down event on other interfaces.

The following topics provide more information about the link monitor:

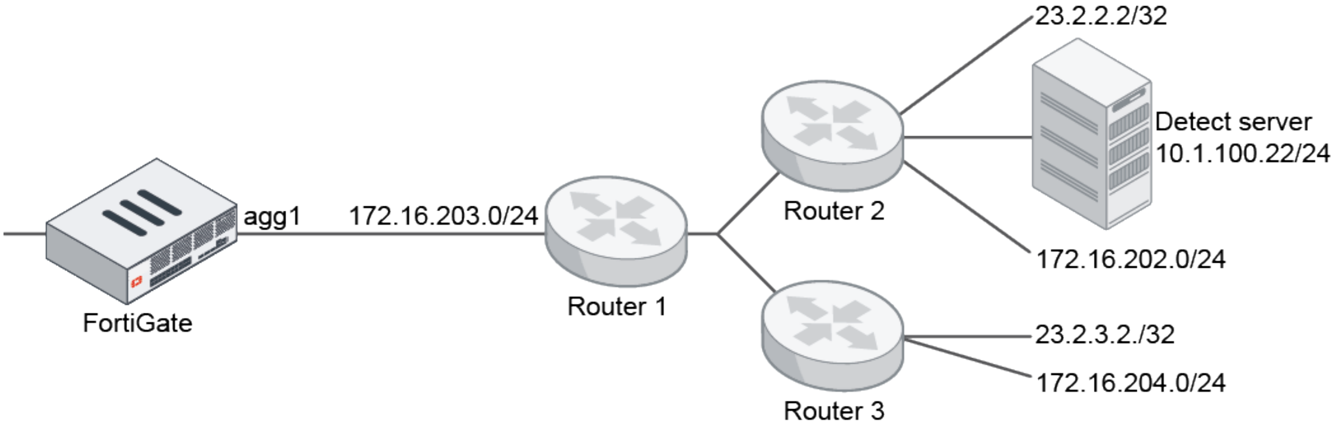
- Link monitor with route updates on page 455
- Enable or disable updating policy routes when link health monitor fails on page 457
- Add weight setting on each link health monitor server on page 459
- Dual internet connections on page 312
- SLA link monitoring for dynamic IPsec and SSL VPN tunnels on page 462

Link monitor with route updates

When a link monitor fails, only the routes that are specified in the link monitor are removed from the routing table, instead of all the routes with the same interface and gateway. If no routes are specified, then all of the routes are removed. Only IPv4 routes are supported.

Example

In this example, the FortiGate has several routes to 23.2.2.2/32 and 172.16.202.2/24, and is monitoring the link *agg1* by pinging the server at 10.1.100.22. The link monitor uses the gateway 172.16.203.2.



When the link monitor fails, only the routes to the specified subnet using interface *agg1* and gateway 172.16.203.2 are removed.

To configure the link monitor:

```
config system link-monitor
    edit "22"
        set srcintf "agg1"
        set server "10.1.100.22"
        set gateway-ip 172.16.203.2
        set route "23.2.2.2/32" "172.16.202.0/24"
    next
end
```

To check the results:

1. When the link monitor is alive:

```
# get router info routing-table static
Routing table for VRF=0
S*      0.0.0.0/0 [5/0] via 10.100.1.249, port12
S       10.1.100.0/24 [10/0] via 172.16.203.2, agg1
S       23.2.2.2/32 [10/0] via 172.16.203.2, agg1
S       23.2.3.2/32 [10/0] via 172.16.203.2, agg1
S       172.16.201.0/24 [10/0] via 172.16.200.4, port9
S       172.16.202.0/24 [10/0] via 172.16.203.2, agg1
S       172.16.204.0/24 [10/0] via 172.16.200.4, port9
                        [10/0] via 172.16.203.2, agg1
                        [10/0] via 172.16.206.2, vlan100, [100/0]
```

2. When the link monitor is dead:

```
# get router info routing-table static
Routing table for VRF=0
S*      0.0.0.0/0 [5/0] via 10.100.1.249, port12
S       10.1.100.0/24 [10/0] via 172.16.203.2, agg1
S       23.2.3.2/32 [10/0] via 172.16.203.2, agg1
S       172.16.201.0/24 [10/0] via 172.16.200.4, port9
S       172.16.204.0/24 [10/0] via 172.16.200.4, port9
                        [10/0] via 172.16.203.2, agg1
                        [10/0] via 172.16.206.2, vlan100, [100/0]
```

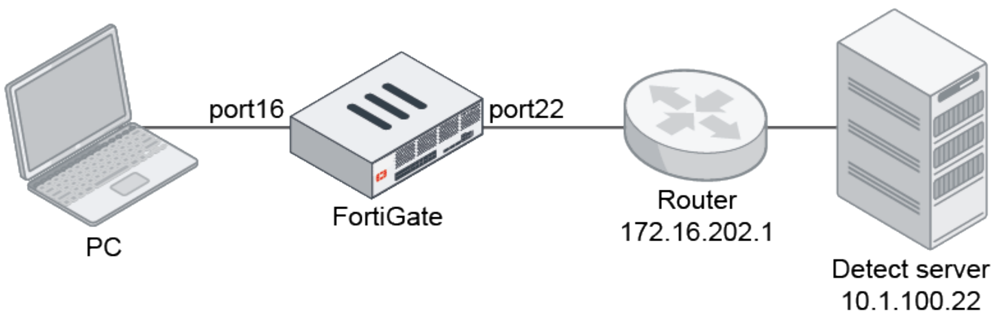
Enable or disable updating policy routes when link health monitor fails

An option has been added to toggle between enabling or disabling policy route updates when a link health monitor fails. By disabling policy route updates, a link health monitor failure will not cause corresponding policy-based routes to be removed.

```
config system link-monitor
    edit <name>
        set update-policy-route {enable | disable}
    next
end
```

Example

In the following topology, the FortiGate is monitoring the detect server, 10.1.100.22. The FortiGate has a policy-based route to destination 172.16.205.10 using the same gateway (172.16.202.1) and interface (port22). By configuring `update-policy-route disable`, the policy-based route is not removed when the link health monitor detects a failure.



To disable updating policy routes when the link health monitor fails:

1. Configure the link health monitor:

```
config system link-monitor
    edit "test-1"
        set srcintf "port22"
        set server "10.1.100.22"
        set gateway-ip 172.16.202.1
        set failtime 3
        set update-policy-route disable
    next
end
```

2. Configure the policy route:

```
config router policy
    edit 1
        set input-device "port16"
        set dst "172.16.205.10/255.255.255.255"
        set gateway 172.16.202.1
        set output-device "port22"
        set tos 0x14
        set tos-mask 0xff
```

```
    next
end
```

3. When the health link monitor status is up, verify that the policy route is active.
   a. Verify the link health monitor status:

```
# diagnose sys link-monitor status
Link Monitor: test-1, Status: alive, Server num(1), HA state: local(alive), shared
(alive)
Flags=0x1 init, Create time: Fri May 28 15:20:15 2021
Source interface: port22 (14)
Gateway: 172.16.202.1
Interval: 500 ms
Service-detect: disable
Diffservcode: 000000
Class-ID: 0
  Peer: 10.1.100.22(10.1.100.22)
        Source IP(172.16.202.2)
        Route: 172.16.202.2->10.1.100.22/32, gwy(172.16.202.1)
        protocol: ping, state: alive
                Latency(Min/Max/Avg): 0.374/0.625/0.510 ms
                Jitter(Min/Max/Avg): 0.008/0.182/0.074
                Packet lost: 0.000%
                Number of out-of-sequence packets: 0
                Fail Times(0/3)
                Packet sent: 7209, received: 3400, Sequence(sent/rcvd/exp):
7210/7210/7211
```

   b. Verify the policy route list:

```
# diagnose firewall proute  list
list route policy info(vf=root):
id=1 dscp_tag=0xff 0xff flags=0x0 tos=0x14 tos_mask=0xff protocol=0 sport=0-0 iif=41
dport=0-65535 oif=14(port22) gwy=172.16.202.1
source wildcard(1): 0.0.0.0/0.0.0.0
destination wildcard(1): 172.16.205.10/255.255.255.255
hit_count=1 last_used=2021-05-27 23:04:33
```

4. When the health link monitor status is down, verify that the policy route is active:
   a. Verify the link health monitor status:

```
# diagnose sys link-monitor status
Link Monitor: test-1, Status: die, Server num(1), HA state: local(die), shared(die)
Flags=0x9 init log_downgateway, Create time: Fri May 28 15:20:15 2021
Source interface: port22 (14)
Gateway: 172.16.202.1
Interval: 500 ms
Service-detect: disable
Diffservcode: 000000
Class-ID: 0
  Peer: 10.1.100.22(10.1.100.22)
        Source IP(172.16.202.2)
        Route: 172.16.202.2->10.1.100.22/32, gwy(172.16.202.1)
        protocol: ping, state: die
                Packet lost: 11.000%
                Number of out-of-sequence packets: 0
                Recovery times(0/5) Fail Times(0/3)
```

```
                Packet sent: 7293, received: 3471, Sequence(sent/rcvd/exp):
7294/7281/7282
```

b. Verify the policy route list:

```
# diagnose firewall proute list
list route policy info(vf=root):
id=1 dscp_tag=0xff 0xff flags=0x0 tos=0x14 tos_mask=0xff protocol=0 sport=0-0 iif=41
dport=0-65535 oif=14(port22) gwy=172.16.202.1
source wildcard(1): 0.0.0.0/0.0.0.0
destination wildcard(1): 172.16.205.10/255.255.255.255
hit_count=1 last_used=2021-05-27 23:04:33
```

If the `update-policy-route` setting is enabled, the link health monitor would be down and the policy-based route would be disabled:

```
# diagnose firewall proute list
list route policy info(vf=root):
id=1 dscp_tag=0xff 0xff flags=0x8 disable tos=0x14 tos_mask=0xff protocol=0 sport=0-0
iif=41 dport=0-65535 oif=14(port22) gwy=172.16.202.1
source wildcard(1): 0.0.0.0/0.0.0.0
destination wildcard(1): 172.16.205.10/255.255.255.255
hit_count=1 last_used=2021-05-27 23:04:33
```

Add weight setting on each link health monitor server

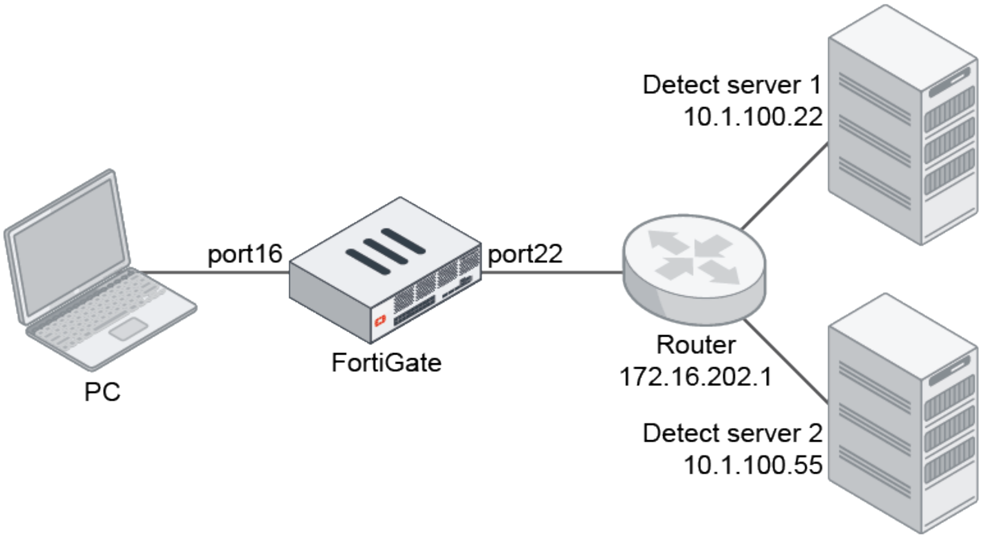
Prior to FortiOS 7.0.1, the link health monitor is determined to be dead when all servers are unreachable. Starting in 7.0.1, the link health monitor can configure multiple servers and allow each server to have its own weight setting. When the link health monitor is down, it will trigger static route updates and cascade interface updates if the weight of all dead servers exceeds the monitor's fail weight threshold.

```
config system link-monitor
    edit <name>
        set srcintf <interface>
        set server-config {default | individual}
        set fail-weight <integer>
        config server-list
            edit <id>
                set dst <address>
                set weight <integer>
            next
        end
    next
end
```

`server-config`	Set the server configuration mode: • default: all servers share the same attributes. • individual: some attributes can be specified for individual servers.
`fail-weight <integer>`	Threshold weight to trigger link failure alert (0 - 255, default = 0).
`server-list`	Configure the servers to be monitored by the link monitor.
`dst <address>`	Enter the IP address of the server to be monitored.
`weight <integer>`	Weight of the monitor to this destination (0 - 255, default = 0).

Examples

In the following topology, there are two detect servers that connect to the FortiGate through a router: server 1 (10.1.100.22) and server 2 (10.1.100.55).



Alive link health monitor

In this configuration, one server is dead and one server alive. The failed server weight is not over the threshold, so the link health monitor status is alive.

To configure the weight settings on the link health monitor:

1. Configure the link health monitor:

```
config system link-monitor
    edit "test-1"
        set srcintf "port22"
        set server-config individual
        set gateway-ip 172.16.202.1
        set failtime 3
        set fail-weight 40
        config server-list
            edit 1
                set dst "10.1.100.22"
                set weight 60
            next
            edit 2
                set dst "10.1.100.55"
                set weight 30
            next
        end
    next
end
```

2. Trigger server 2 to go down. The link monitor is still alive because the fail weight threshold has not been reached.

3. Verify the link health monitor status:

```
# diagnose sys link-monitor status test-1
Link Monitor: test-1, Status: alive, Server num(2), HA state: local(alive), shared
(alive)
Flags=0x1 init, Create time: Fri Jun  4 17:23:29 2021
Source interface: port22 (14)
Gateway: 172.16.202.1
Interval: 500 ms
Service-detect: disable
Diffservcode: 000000
Class-ID: 0
Fail-weight (40): not activated
  Peer: 10.1.100.22(10.1.100.22)
        Source IP(172.16.202.2)
        Route: 172.16.202.2->10.1.100.22/32, gwy(172.16.202.1)
        protocol: ping, state: alive
                Latency(Min/Max/Avg): 0.417/0.585/0.530 ms
                Jitter(Min/Max/Avg): 0.007/0.159/0.057
                Packet lost: 0.000%
                Number of out-of-sequence packets: 0
                Fail Times(0/3)
                Packet sent: 239, received: 236, Sequence(sent/rcvd/exp): 240/240/241
  Peer: 10.1.100.55(10.1.100.55)
        Source IP(172.16.202.2)
        Route: 172.16.202.2->10.1.100.55/32, gwy(172.16.202.1)
        Fail weight 30 applied
        protocol: ping, state: dead
                Packet lost: 100.000%
                Number of out-of-sequence packets: 0
                Recovery times(0/5) Fail Times(1/3)
                Packet sent: 239, received: 3, Sequence(sent/rcvd/exp): 240/4/5
```

Dead link health monitor

In this configuration, one server is dead and one server alive. The failed server weight is over the threshold, so the link health monitor status is dead.

To configure the weight settings on the link health monitor:

1. Configure the link health monitor:

```
config system link-monitor
    edit "test-1"
        set srcintf "port22"
        set server-config individual
        set gateway-ip 172.16.202.1
        set failtime 3
        set fail-weight 40
        config server-list
            edit 1
                set dst "10.1.100.22"
                set weight 30
            next
            edit 2
                set dst "10.1.100.55"
```

```
                    set weight 50
                next
            end
        next
    end
```

2. Trigger server 2 to go down. The link monitor is dead because the fail weight threshold has been reached.
3. Verify the link health monitor status:

```
# diagnose sys link-monitor status  test-1
Link Monitor: test-1, Status: dead, Server num(2), HA state: local(dead), shared(dead)
Flags=0x9 init log_downgateway, Create time: Fri Jun  4 17:23:29 2021
Source interface: port22 (14)
Gateway: 172.16.202.1
Interval: 500 ms
Service-detect: disable
Diffservcode: 000000
Class-ID: 0
Fail-weight (40): activated
  Peer: 10.1.100.22(10.1.100.22)
        Source IP(172.16.202.2)
        Route: 172.16.202.2->10.1.100.22/32, gwy(172.16.202.1)
        protocol: ping, state: alive
                Latency(Min/Max/Avg): 0.393/0.610/0.520 ms
                Jitter(Min/Max/Avg): 0.009/0.200/0.095
                Packet lost: 0.000%
                Number of out-of-sequence packets: 0
                Fail Times(0/3)
                Packet sent: 680, received: 677, Sequence(sent/rcvd/exp): 681/681/682
  Peer: 10.1.100.55(10.1.100.55)
        Source IP(172.16.202.2)
        Route: 172.16.202.2->10.1.100.55/32, gwy(172.16.202.1)
        Fail weight 50 applied
        protocol: ping, state: dead
                Packet lost: 100.000%
                Number of out-of-sequence packets: 0
                Recovery times(0/5) Fail Times(1/3)
                Packet sent: 680, received: 3, Sequence(sent/rcvd/exp): 681/4/5
```

SLA link monitoring for dynamic IPsec and SSL VPN tunnels

The link health monitor settings can measure SLA information of dynamic VPN interfaces, which assign IP addresses to their clients during tunnel establishment. This includes SSL VPN tunnels, IPsec remote access, and IPsec site-to-site tunnels.

This feature currently only supports IPv4 and the ICMP monitoring protocol. In the IPsec tunnel settings, `net-device` must be disabled.

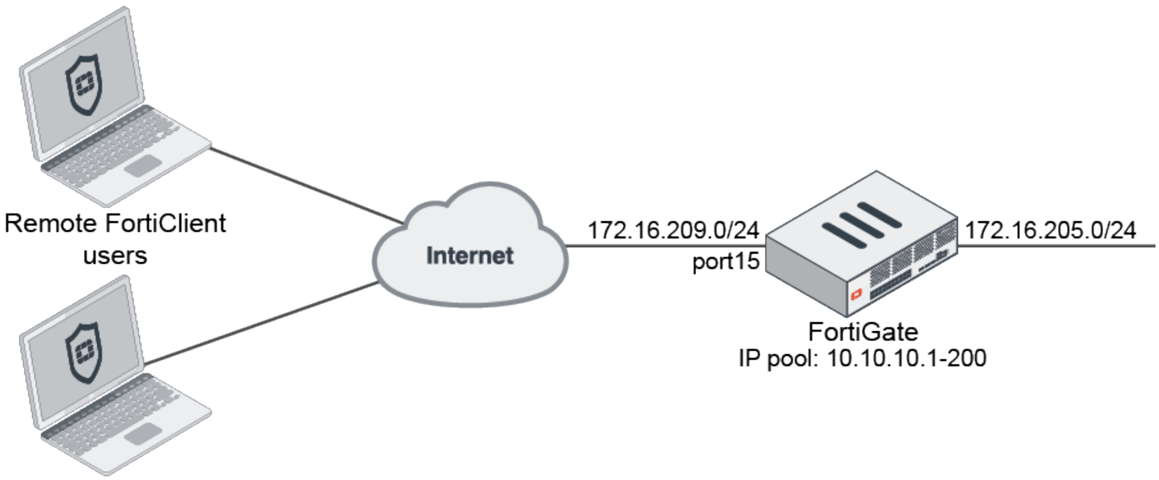
```
config system link-monitor
    edit <name>
        set server-type {static | dynamic}
    next
end
```

To view the dial-up tunnel statistics:

```
# diagnose sys link-monitor tunnel {name | all} [<tunnel_name>]
```

Example

In this example, endpoint users dial up using FortiClient to create IPSec tunnels with the FortiGate and obtain IP addresses. The link monitor on the FortiGate's dynamic VPN interface detects the path quality to the endpoints.



To configure SLA link health monitoring in dynamic IPsec tunnels:

1. Configure the IPsec phase 1 interface:

```
config vpn ipsec phase1-interface
    edit "for_Branch"
        set type dynamic
        set interface "port15"
        set mode aggressive
        set peertype any
        set net-device disable
        set mode-cfg enable
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set dpd on-idle
        set dhgrp 5
        set xauthtype auto
        set authusrgrp "vpngroup"
        set assign-ip-from name
        set ipv4-netmask 255.255.255.0
        set dns-mode auto
        set ipv4-split-include "172.16.205.0"
        set ipv4-name "client_range"
        set save-password enable
        set psksecret **********
        set dpd-retryinterval 60
    next
end
```

2. Configure the IPsec phase 2 interface:

```
config vpn ipsec phase2-interface
    edit "for_Branch_p2"
        set phase1name "for_Branch"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
aes256gcm chacha20poly1305
        set dhgrp 5
    next
end
```

3. Configure the dynamic interface:

```
config system interface
    edit "for_Branch"
        set vdom "root"
        set ip 10.10.10.254 255.255.255.255
        set type tunnel
        set remote-ip 10.10.10.253 255.255.255.0
        set snmp-index 100
        set interface "port15"
    next
end
```

4. Add the IPsec dial-up tunnel to the link health monitor:

```
config system link-monitor
    edit "1"
        set srcintf "for_Branch"
        set server-type dynamic
    next
end
```

5. Once endpoint users have connected using FortiClient, verify the tunnel information:

```
# get vpn ipsec tunnel summary
'for_Branch_0' 10.1.100.23:0  selectors(total,up): 1/1  rx(pkt,err): 21091/0  tx
(pkt,err): 20741/0
'for_Branch_1' 10.1.100.13:0  selectors(total,up): 1/1  rx(pkt,err): 19991/0  tx
(pkt,err): 20381/0
```

6. Verify the link health monitor status:

```
# diagnose sys link-monitor tunnel all
for_Branch_0 (1): state=alive, peer=10.10.10.1, create_time=2022-02-08 10:43:11,
srcintf=for_Branch, latency=0.162, jitter=0.018, pktloss=0.000%
for_Branch_1 (1): state=alive, peer=10.10.10.2, create_time=2022-02-08 10:49:24,
srcintf=for_Branch, latency=0.266, jitter=0.015, pktloss=0.000%
```

7. Manually add 200 ms latency on the path between the FortiGate and FortiClients.
8. Verify the link health monitor status again:

```
# diagnose sys link-monitor tunnel all
for_Branch_0 (1): state=alive, peer=10.10.10.1, create_time=2022-02-08 10:43:11,
srcintf=for_Branch, latency=200.177, jitter=0.021, pktloss=0.000%
for_Branch_1 (1): state=alive, peer=10.10.10.2, create_time=2022-02-08 10:49:24,
srcintf=for_Branch, latency=200.257, jitter=0.017, pktloss=0.000%
```

IPv6

The following topics provide information about IPv6:

IPv6 tunneling

IPv6 tunneling involves tunneling IPv6 packets from an IPv6 network through an IPv4 network to another IPv6 network. This is different than NAT because once the packet reaches its final destination, the true originating address of the sender is still readable. The IPv6 packets are encapsulated within packets with IPv4 headers that carry their IPv6 payload through the IPv4 network. IPv6 tunneling is suitable in networks that have completely transitioned over to IPv6 but need an internet connection, which is still mostly IPv4 addresses.

Both IPv6 tunneling devices, whether they are a host or a network device, must be dual stack compatible. The tunneling process is as follows:

1. The tunnel entry node creates an encapsulating IPv4 header and transmits the encapsulated packet.
2. The tunnel exit node receives the encapsulated packet.
3. The IPv4 header is removed.
4. The IPv6 header is updated and the IPv6 packet is processed.

There are two types of tunnels in IPv6 tunneling, automatic and configured. Automatic tunnels are configured by using IPv4 address information embedded in an IPv6 address. The IPv6 address of the destination host includes information about which IPv4 address the packet should be tunneled to. Configured tunnels are manually configured, and they are used for IPv6 addresses that do not have any embedded IPv4 information. The IPv6 and IPv4 addresses of the tunnel endpoints must be specified.

Tunnel configurations

There are four tunneling configurations available depending on which segment of the path between the endpoints of the session the encapsulation takes place.

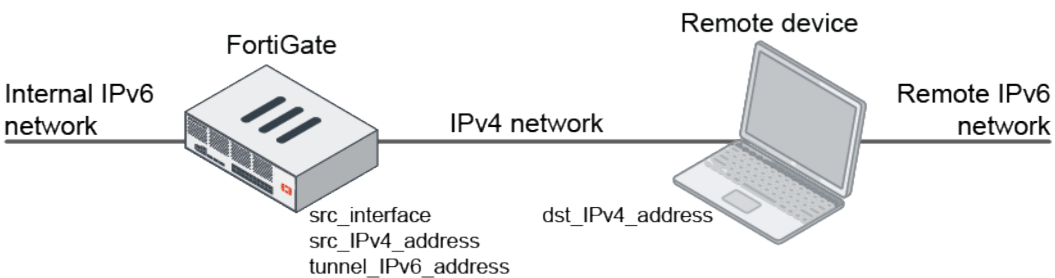
Type	Description
Network device-to-network device	Dual stack capable devices connected by an IPv4 infrastructure can tunnel IPv6 packets between themselves. The tunnel spans one segment of the path taken by the IPv6 packets.
Host-to-network device	Dual stack capable hosts can tunnel IPv6 packets to an intermediary IPv6 or IPv4 network device that is reachable through an IPv4 infrastructure. The tunnel spans the first segment of the path taken by the IPv6 packets.
Host-to-host	Dual stack capable hosts that are interconnected by an IPv4 infrastructure can tunnel IPv6 packets between themselves. The tunnel spans the entire path taken by the IPv6 packets.

Type	Description
Network device-to-host	Dual stack capable network devices can tunnel IPv6 packets to their final destination IPv6 or IPv4 host. The tunnel spans only the last segment of the path taken by the IPv6 packets.

Regardless of whether the tunnel starts at a host or a network device, the node that does the encapsulation needs to maintain soft state information, such as the maximum transmission unit (MTU), about each tunnel in order to process the IPv6 packets.

6in4 tunnel

The following tunnel configuration tunnels IPv6 traffic over an IPv4 network. An internal IPv6 interface can be configured under `config system interface`.



To configure an IPv6 tunnel over IPv4:

```
config system sit-tunnel
    edit <name>
        set source <src_IPv4_address>
        set destination <dst_IPv4_address>
        set interface <src_interface>
        set ip6 <tunnel_IPv6_address>
    next
end
```

4in6 tunnel

Conversely, the following tunnel configuration tunnels IPv4 traffic over an IPv6 network.

To configure an IPv4 tunnel over IPv6:

```
config system ipv6-tunnel
    edit <name>
        set source <src_IPv6_address>
        set destination <dst_IPv6_address>
        set interface <src_interface>
    next
end
```

The preceding configurations are not available in transparent mode.

IPv6 tunnel inherits MTU based on physical interface

The MTU of an IPv6 tunnel interface is calculated from the MTU of its parent interface minus headers.

Example

In this topology, FortiGate B and FortiGate D are connected over an IPv6 network. An IPv6 tunnel is formed, and IPv4 can be used over the IPv6 tunnel. The tunnel interface MTU is based on the physical interface MTU minus the IP and TCP headers (40 bytes). On FortiGate B's physical interface port5, the MTU is set to 1320. The IPv6 tunnel is based on port5, and its MTU value of 1280 is automatically calculated from the MTU value of its physical interface minus the header. The same is true for port3 on FortiGate D.

To verify the MTU for the IPv6 tunnel on FortiGate B:

1. Configure port5:

```
config system interface
    edit "port5"
        set vdom "root"
        set type physical
        set snmp-index 7
        config ipv6
            set ip6-address 2000:172:16:202::1/64
            set ip6-allowaccess ping
        end
        set mtu-override enable
        set mtu 1320
    next
end
```

2. Configure the IPv6 tunnel:

```
config system ipv6-tunnel
    edit "B_2_D"
        set source 2000:172:16:202::1
        set destination 2000:172:16:202::2
        set interface "port5"
    next
end
```

3. Configure the tunnel interface:

```
config system interface
    edit "B_2_D"
        set vdom "root"
        set ip 172.16.210.1 255.255.255.255
        set allowaccess ping https http
        set type tunnel
```

```
        set remote-ip 172.16.210.2 255.255.255.255
        set snmp-index 33
        config ipv6
            set ip6-address 2000:172:16:210::1/64
            set ip6-allowaccess ping
            config ip6-extra-addr
                edit fe80::2222/10
                next
            end
        end
        set interface "port5"
    next
end
```

4. Verify the interface lists:

```
# diagnose netlink interface list port5
if=port5 family=00 type=1 index=13 mtu=1320 link=0 master=0
ref=68 state=start present fw_flags=0 flags=up broadcast run multicast
Qdisc=mq hw_addr=**:**:**:**:**:** broadcast_addr=**:**:**:**:**:**
stat: rxp=1577 txp=1744 rxb=188890 txb=203948 rxe=0 txe=0 rxd=0 txd=0 mc=825 collision=0
@ time=1631647112
re: rxl=0 rxo=0 rxc=0 rxf=0 rxfi=0 rxm=0
te: txa=0 txc=0 txfi=0 txh=0 txw=0
misc rxc=0 txc=0
input_type=0 state=3 arp_entry=0 refcnt=68
```

```
# diagnose netlink interface list B_2_D
if=B_2_D family=00 type=769 index=41 mtu=1280 link=0 master=0
ref=25 state=start present fw_flags=0 flags=up p2p run noarp multicast
Qdisc=noqueue local=0.0.0.0 remote=0.0.0.0
stat: rxp=407 txp=417 rxb=66348 txb=65864 rxe=0 txe=61 rxd=0 txd=0 mc=0 collision=60 @
time=1631647126
re: rxl=0 rxo=0 rxc=0 rxf=0 rxfi=0 rxm=0
te: txa=0 txc=0 txfi=0 txh=0 txw=0
misc rxc=0 txc=0
input_type=0 state=3 arp_entry=0 refcnt=25
```

To verify the MTU for the IPv6 tunnel on FortiGate D:

1. Configure port3:

```
config system interface
    edit "port3"
        set vdom "root"
        set type physical
        set snmp-index 5
        config ipv6
            set ip6-address 2000:172:16:202::2/64
            set ip6-allowaccess ping
        end
        set mtu-override enable
        set mtu 1320
    next
end
```

2. Configure the IPv6 tunnel:

```
config system ipv6-tunnel
    edit "D_2_B"
        set source 2000:172:16:202::2
        set destination 2000:172:16:202::1
        set interface "port3"
    next
end
```

3. Configure the tunnel interface:

```
config system interface
    edit "D_2_B"
        set vdom "root"
        set ip 172.16.210.2 255.255.255.255
        set allowaccess ping https http
        set type tunnel
        set remote-ip 172.16.210.1 255.255.255.255
        set snmp-index 36
        config ipv6
            set ip6-address 2000:172:16:210::2/64
            set ip6-allowaccess ping
            config ip6-extra-addr
                edit fe80::4424/10
                next
            end
        end
        set interface "port3"
    next
end
```

4. Verify the interface lists:

```
# diagnose netlink interface list port3
```

```
# diagnose netlink interface list D_2_B
```

Configuring IPv4 over IPv6 DS-Lite service

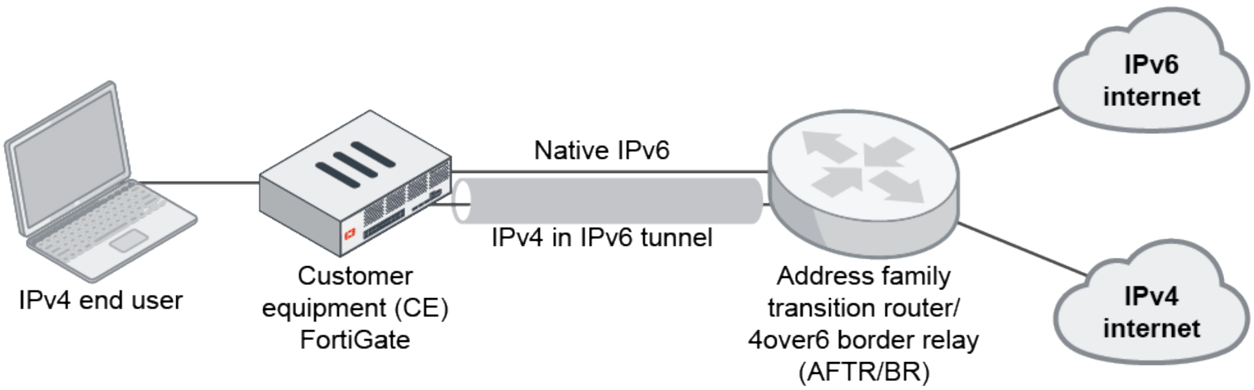
IPv4 over IPv6 DS-Lite service can be configured on a virtual network enabler (VNE) tunnel. In addition, VNE tunnel fixed IP mode supports username and password authentication.

```
config system vne-tunnel
    set status enable
    set mode {map-e | fixed-ip | ds-lite}
    set ipv4-address <IPv4_address>
    set br <IPv6_address or FQDN>
    set http-username <string>
    set http-password <password>
end
```

`mode {map-e	fixed-ip	ds-lite}`	Set the VNE tunnel mode: • map-e: MAP-E • fixed-ip: fixed IP • ds-lite: DS-Lite

`ipv4-address <IPv4_address>`	Enter the tunnel IPv4 address and netmask. This setting is optional.
`br <IPv6_address or FQDN>`	Enter the IPv6 or FQDN of the border relay.
`http-username <string>`	Enter the HTTP authentication user name.
`http-password <password>`	Enter the HTTP authentication password.

DS-Lite allows applications using IPv4 to access the internet with IPv6. DS-Lite is supported by internet providers that do not have enough public IPv4 addresses for their customers, so DS-Lite is used for IPv6 internet connections. When a DS-Lite internet connections is used, the FortiGate encapsulates all data from IPv4 applications into IPv6 packets. The packets are then transmitted to the internet service provider using the IPv6 connection. Next, a dedicated server unpacks the IPv6 packets and forwards the IPv4 data to the actual destination on the internet.



DS-Lite example

In this example, DS-Lite VNE tunnel mode is used between the FortiGate and the BR.

To configure a DS-Lite tunnel between the FortiGate and the BR:

1. Configure the IPv6 interface:

```
config system interface
    edit "wan1"
        set vdom "root"
        set mode dhcp
        set allowaccess ping fgfm
        set type physical
        set role wan
        set snmp-index 1
        config ipv6
            set ip6-allowaccess ping
            set dhcp6-information-request enable
            set autoconf enable
            set unique-autoconf-addr enable
        end
    next
end
```

2. Configure the VNE tunnel:

```
config system vne-tunnel
    set status enable
    set interface "wan1"
    set ssl-certificate "Fortinet_Factory"
    set auto-asic-offload enable
    set ipv4-address 192.168.1.99 255.255.255.255
    set br "dgw.xxxxx.jp"
    set mode ds-lite
end
```

3. View the wan1 IPv6 configuration details:

```
config system interface
    edit "wan1"
        config ipv6
            get
                ip6-mode            : static
                nd-mode             : basic
                ip6-address         : 2001:f70:2880:xxxx:xxxx:xxxx:fe39:ccd2/64
                ip6-allowaccess     : ping
                icmp6-send-redirect : enable
                ra-send-mtu         : enable
                ip6-reachable-time  : 0
                ip6-retrans-time    : 0
                ip6-hop-limit       : 0
                dhcp6-information-request: enable
                cli-conn6-status    : 1
                vrrp-virtual-mac6   : disable
                vrip6_link_local    : ::
                ip6-dns-server-override: enable
                Acquired DNS1       : 2001:f70:2880:xxxx:xxxx:xxxx:fe40:9082
                Acquired DNS2       : ::
                ip6-extra-addr:
                ip6-send-adv        : disable
                autoconf            : enable
                prefix      : 2001:f70:2880:xxxx::/64
                preferred-life-time         : 942735360
                valid-life-time     : 1077411840
                unique-autoconf-addr: enable
                interface-identifier: ::
                dhcp6-relay-service : disable
        end
    next
end
```

4. Verify the IPv6 address list:

```
# diagnose ipv6 address list
dev=5 devname=wan1 flag= scope=0 prefix=64 addr=2001:f70:2880:xxxx:xxxx:xxxx:fe39:ccd2
preferred=11525 valid=13325 cstamp=6520 tstamp=6892
dev=5 devname=wan1 flag=P scope=253 prefix=64 addr=fe80::xxxx:xxxx:fe39:ccd2
```

```
preferred=4294967295 valid=4294967295 cstamp=6373 tstamp=6373
dev=18 devname=root flag=P scope=254 prefix=128 addr=::1 preferred=4294967295
valid=4294967295 cstamp=3531 tstamp=3531
dev=25 devname=vsys_ha flag=P scope=254 prefix=128 addr=::1 preferred=4294967295
valid=4294967295 cstamp=5604 tstamp=5604
dev=27 devname=vsys_fgfm flag=P scope=254 prefix=128 addr=::1 preferred=4294967295
valid=4294967295 cstamp=6377 tstamp=6377
```

5. Test the tunnel connection by pinging the Google public DNS IPv6 address:

```
# execute ping6 2001:4860:4860::8888
PING 2001:4860:4860::8888(2001:4860:4860::8888) 56 data bytes
64 bytes from 2001:4860:4860::8888: icmp_seq=1 ttl=114 time=6.89 ms
64 bytes from 2001:4860:4860::8888: icmp_seq=2 ttl=114 time=3.39 ms
64 bytes from 2001:4860:4860::8888: icmp_seq=3 ttl=114 time=3.46 ms
64 bytes from 2001:4860:4860::8888: icmp_seq=4 ttl=114 time=3.34 ms
64 bytes from 2001:4860:4860::8888: icmp_seq=5 ttl=114 time=3.39 ms
--- 2001:4860:4860::8888 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss, time 4079ms
rtt min/avg/max/mdev = 3.340/4.097/6.895/1.400 ms
```

Fixed IP mode example

In this example, fixed IP VNE tunnel mode with HTTP authentication is used between the FortiGate and the BR.

To configure a fixed IP mode with HTTP authentication between the FortiGate and the BR:

1. Configure the IPv6 interface:

```
config system interface
    edit "wan1"
        set vdom "root"
        set mode dhcp
        set allowaccess ping fgfm
        set type physical
        set role wan
        set snmp-index 1
        config ipv6
            set ip6-allowaccess ping
            set dhcp6-information-request enable
            set autoconf enable
        end
    next
end
```

2. Configure the VNE tunnel:

```
config system vne-tunnel
    set status enable
    set interface "wan1"
    set ipv4-address 120.51.xxx.xxx1 255.255.255.255
    set br "2001:f60:xxxx:xxxx::1"
    set update-url "https://ddnsweb1.ddns.xxxxxx.jp/cgi-bin/ddns_
api.cgi?d=xxxxxx.v4v6.xxxxx.jp&p=**********&a=[IP6]&u=xxxxxx.v4v6.xxxxx.jp"
```

```
    set mode fixed-ip
    set http-username "laptop-1"
    set http-password **********
end
```

3. Verify the wan1 IPv6 configuration details:

```
config system interface
    edit "wan1"
        config ipv6
            get
                ....
```

4. Verify the VNE daemon:

```
# diagnose test application vned 1
------------------------------------------------------------------------
vdom: root/0, is master, devname=wan1 link=0 tun=vne.root mode=fixed-ip ssl_
cert=Fortinet_Factory
end user ipv6 perfix: 2001:f70:2880:xxxx::/64
interface ipv6 addr: 2001:f70:2880:xxxx:xxxx:xxxx:fe39:ccd2
config ipv4 perfix: 120.51.xxx.xxx1/255.255.255.255
config br: 2001:f60:xxxx:xxxx::1
HTTP username: laptop-1
update url: https://ddnsweb1.ddns.xxxxxx.jp/cgi-bin/ddns_
api.cgi?d=xxxxxx.v4v6.xxxxx.jp&p=**********&a=[IP6]&u=xxxxxx.v4v6.xxxxx.jp
host: ddnsweb1.ddns.xxxxxx.jp path: /cgi-bin/ddns_
api.cgi?d=xxxxxx.v4v6.xxxxx.jp&p=**********&a=[IP6]&u=xxxxxx.v4v6.xxxxx.jp port:443 ssl:
1
tunnel br: 2001:f60:xxxx:xxxx::1
tunnel ipv6 addr: 2001:f70:2880:xxxx:xxxx:xxxx:fe39:ccd2
tunnel ipv4 addr: 120.51.xxx.xxx1/255.255.255.255
update result: <H1>DDNS API</H1><HR><H2>* Query parameter check :
OK</H2>FQDN=xxxxxx.v4v6.xxxxx.jp<BR>Password=**********<BR>IPv6=2001:f70:2880:xxxx:xxxx:
xxxx:fe39:ccd2<BR>UID=xxxxxx.v4v6.xxxxx.jp<BR>Address=2001:f70:2880:xxxx:xxxx:xxxx:fe39:
ccd2<BR><H2>* routerinfo check : OK</H2><H2>* records check : OK</H2><H2>* routerinfo
update : OK</H2><H2>* records update : OK</H2><H2>* DDNS API update : Success [2022-01-
18 18:37:58 1642498678]</H2>
Fixed IP rule client: state=succeed retries=0 interval=0 expiry=0 reply_code=0
fqdn=2001:f60:xxxx:xxxx::1 num=1 cur=0 ttl=4294967295 expiry=0
2001:f60:xxxx:xxxx::1
Fixed IP DDNS client: state=succeed retries=0 interval=10 expiry=0 reply_code=200
fqdn=ddnsweb1.ddns.xxxxxx.jp num=1 cur=0 ttl=6 expiry=0
2001:f61:0:2a::18
```

5. Test the tunnel connection by pinging the Google public DNS IPv4 and IPv6 addresses:

```
# execute ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8): 56 data bytes
64 bytes from 8.8.8.8: icmp_seq=0 ttl=119 time=3.7 ms
64 bytes from 8.8.8.8: icmp_seq=1 ttl=119 time=3.6 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=119 time=3.6 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=119 time=3.6 ms
```

```
64 bytes from 8.8.8.8: icmp_seq=4 ttl=119 time=3.5 ms
--- 8.8.8.8 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 3.5/3.6/3.7 ms
```

```
# execute ping6 2001:4860:4860::8888
PING 2001:4860:4860::8888(2001:4860:4860::8888) 56 data bytes
64 bytes from 2001:4860:4860::8888: icmp_seq=1 ttl=114 time=6.99 ms
64 bytes from 2001:4860:4860::8888: icmp_seq=2 ttl=114 time=3.61 ms
64 bytes from 2001:4860:4860::8888: icmp_seq=3 ttl=114 time=3.34 ms
64 bytes from 2001:4860:4860::8888: icmp_seq=4 ttl=114 time=3.27 ms
64 bytes from 2001:4860:4860::8888: icmp_seq=5 ttl=114 time=3.75 ms
--- 2001:4860:4860::8888 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss, time 4039ms
rtt min/avg/max/mdev = 3.276/4.195/6.992/1.409 ms
```

Diagnostics

Administrators can use the *Diagnostics* page to access the following tools:

Capture packet	Captures packet streams in real-time to let you view header and payload information. See Using the packet capture tool on page 474.
Debug flow	Traces packet flow through FortiOS to help you diagnose and debug issues. See Using the debug flow tool on page 478.

See also Performing a sniffer trace or packet capture on page 2658 and Debugging the packet flow on page 2659.

Using the packet capture tool

Administrators can use the packet capture tool to select a packet and view its header and payload information in real-time. Once completed, packets can be filtered by various fields or through the search bar. The capture can be saved as a PCAP file that you can use with a third-party application, such as Wireshark, for further analysis.

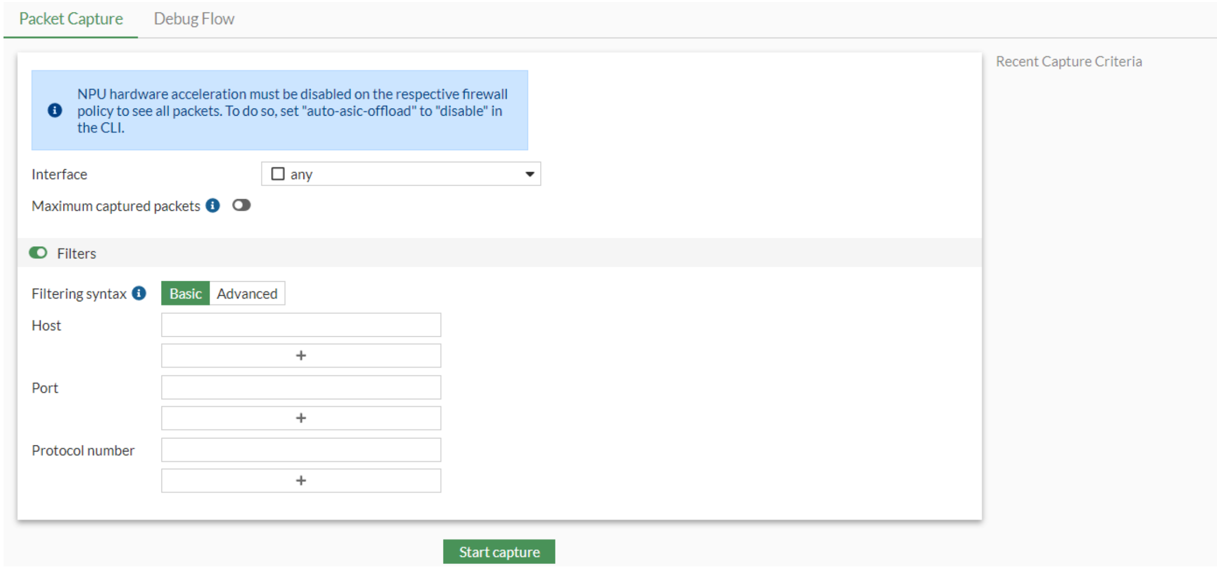
Recent capture criteria is saved after the packet capture, and you can select and use the same criteria again.

For information about running a packet capture in the CLI, see Performing a sniffer trace or packet capture on page 2658.

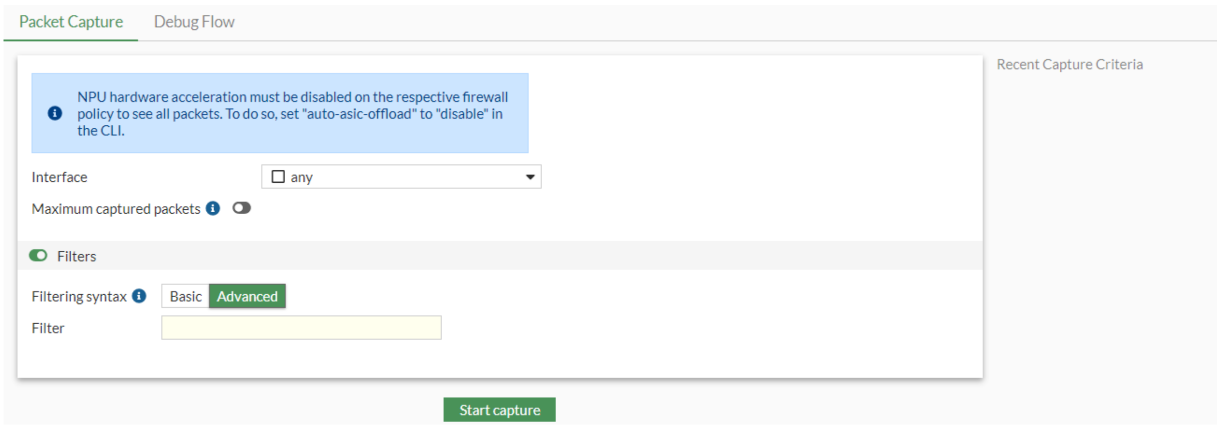
To use the packet capture tool in the GUI:

1. Go to *Network > Diagnostics* and select the *Packet Capture* tab.
2. Optionally, select an *Interface* (*any* is the default).
3. Optionally, enable *Filters* and select a *Filtering syntax*:

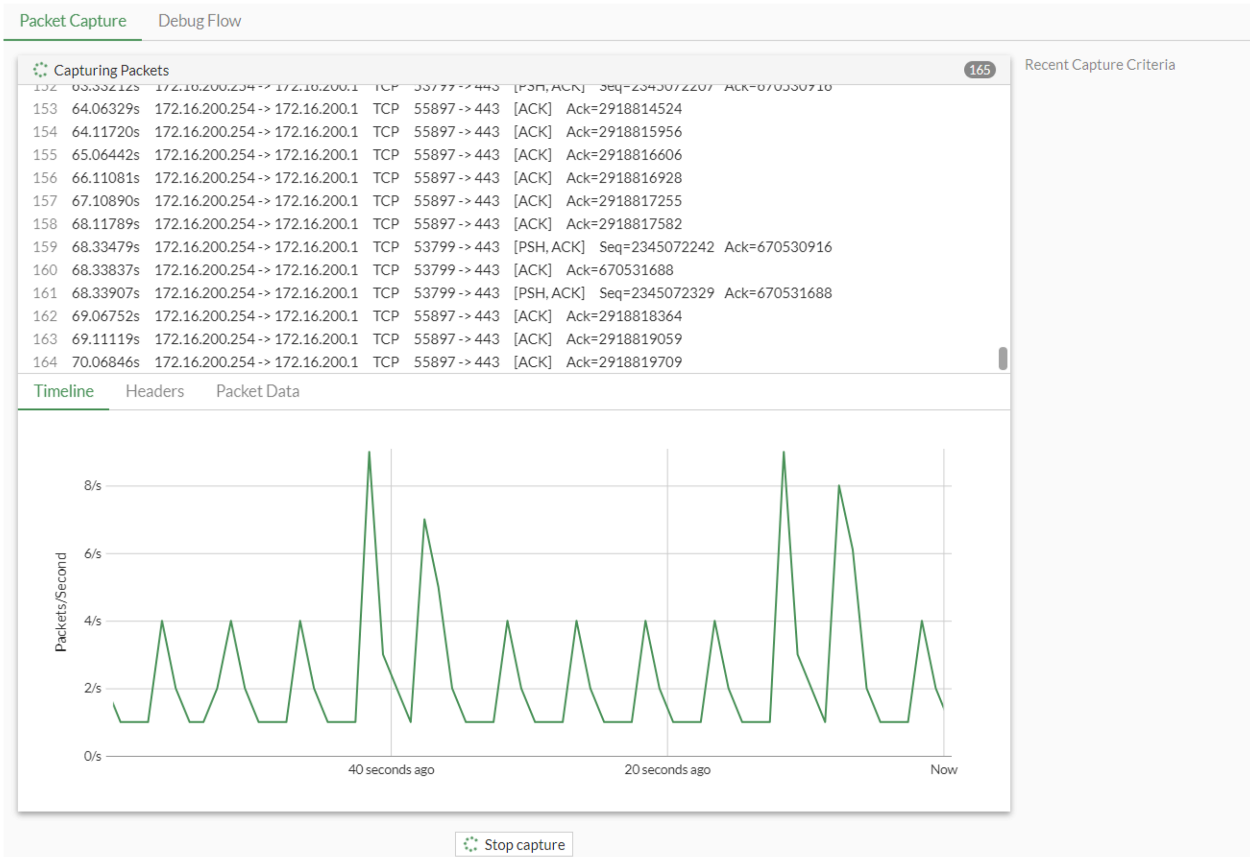
a. *Basic*: enter criteria for the *Host*, *Port*, and *Protocol number*.



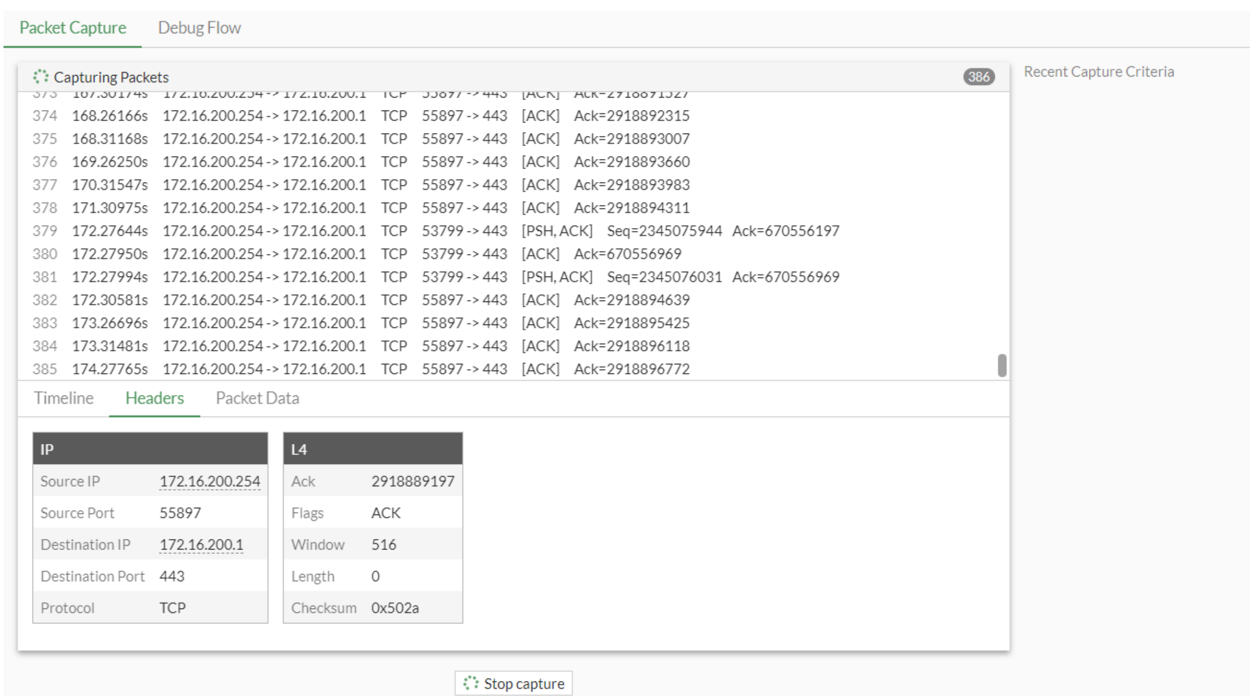
b. *Advanced*: enter a string, such as *src host 172.16.200.254 and dst host 172.16.200.1 and dst port 443*.

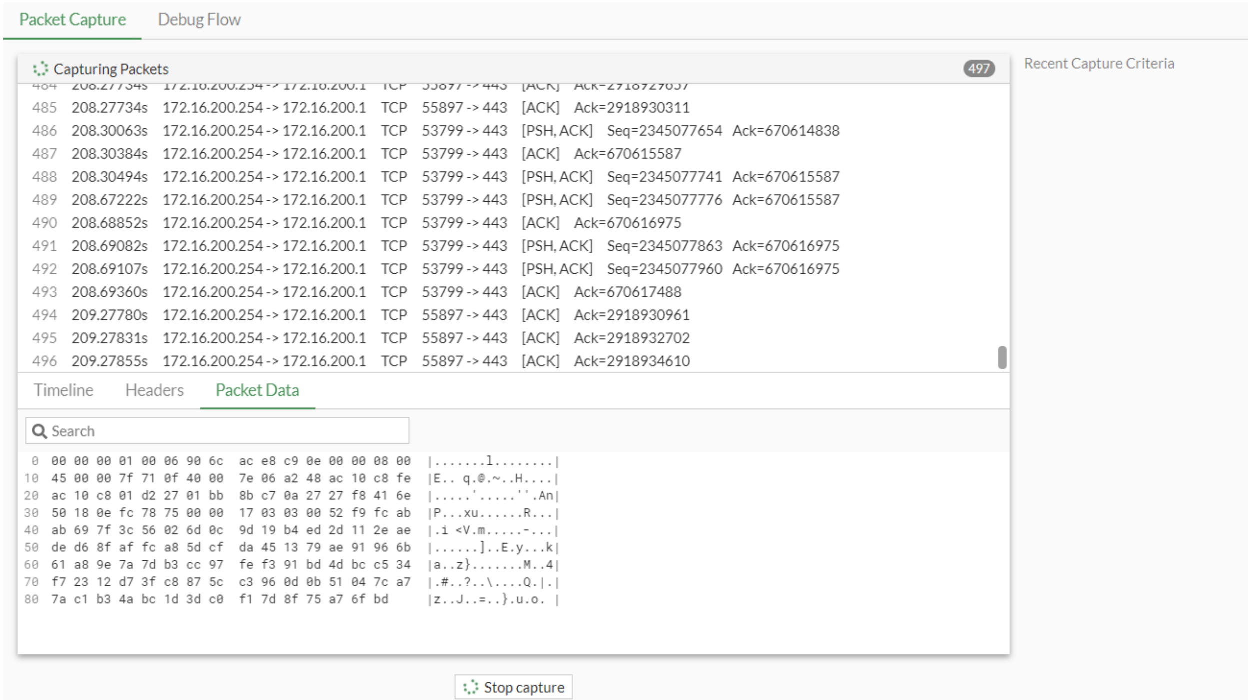


4. Click *Start capture*. The capture is visible in real-time.

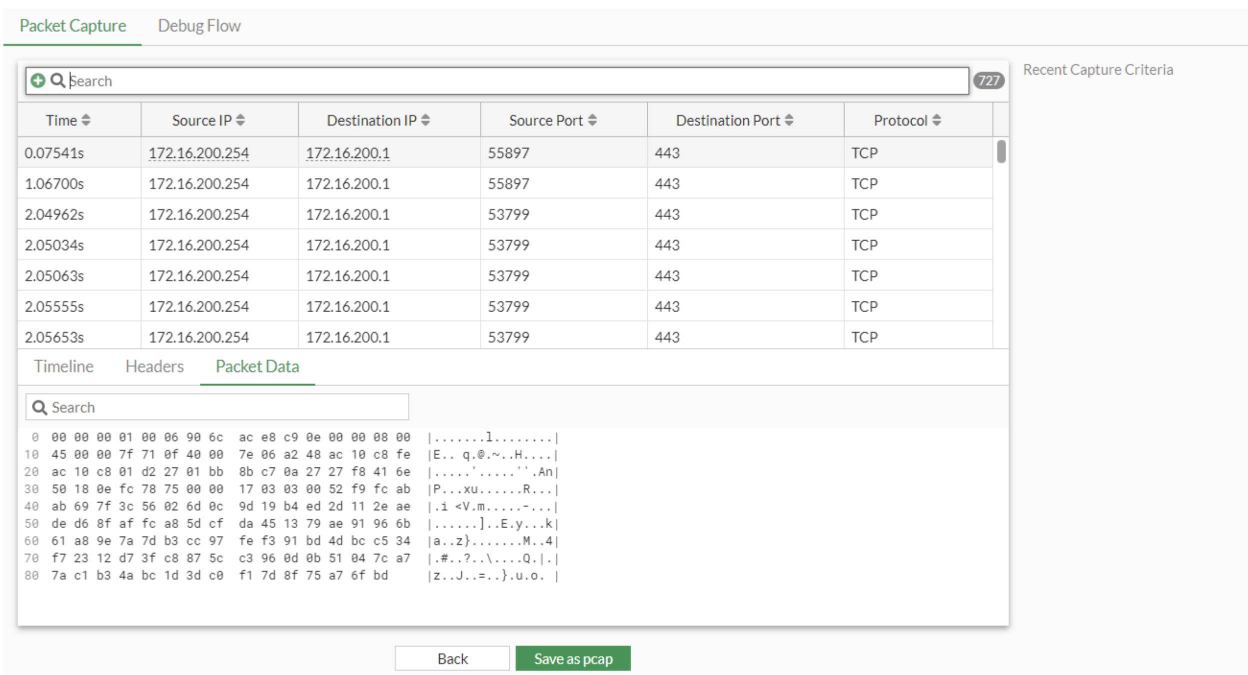


5. While the capture is running, select a packet, then click the *Headers* or *Packet Data* tabs to view more information.



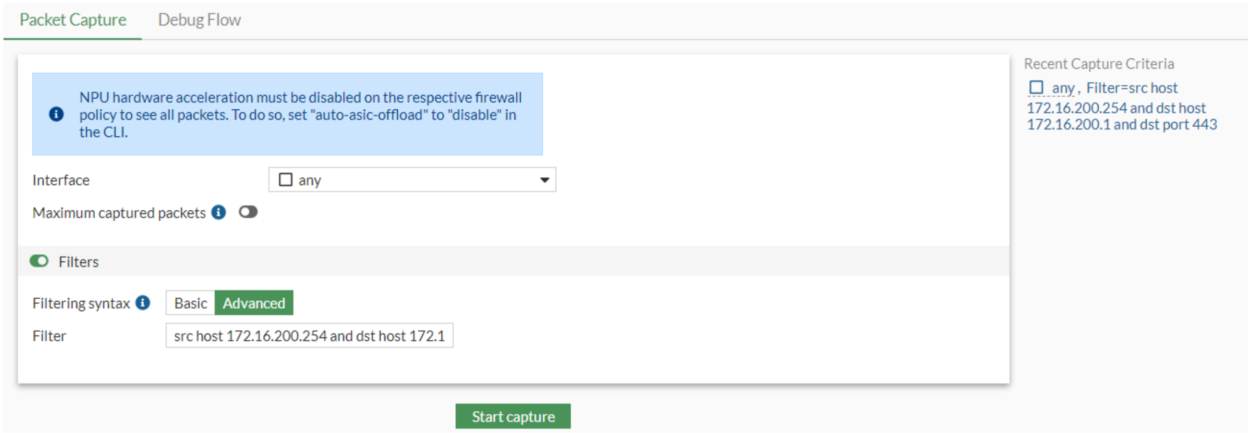


6. When the capture is finished, click *Save as pcap*. The PCAP file is automatically downloaded.



7. Optionally, use the *Search* bar or the column headers to filter the results further.

   The packet capture history is listed under *Recent Capture Criteria* in the right-side of the screen. Clicking the hyperlink will take you back to the main page with the interface and filter settings already populated.



For more granular sniffer output with various verbose settings, use `diagnose sniffer packet <interface> <'filter'> <verbose> <count> <tsformat>`. See Performing a sniffer trace or packet capture on page 2658.

To use recent capture criteria:

1. Go to *Network > Diagnostics* and select the *Packet Capture* tab.
2. Under *Recent Capture Criteria*, click one of the saved capture criteria. The criteria populate the fields.
3. Click *Start Capture*.

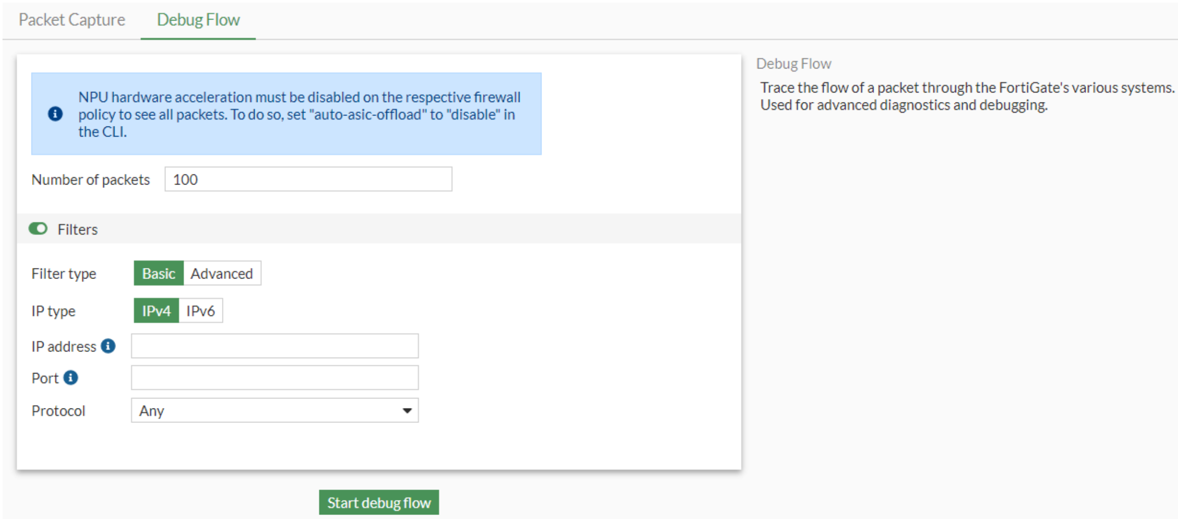
Using the debug flow tool

Administrators can use the debug flow tool to display debug flow output in real-time until it is stopped. The completed output can be filtered by time, message, or function. The output can be exported as a CSV file.

For information about using the debug flow tool in the CLI, see Debugging the packet flow on page 2659.

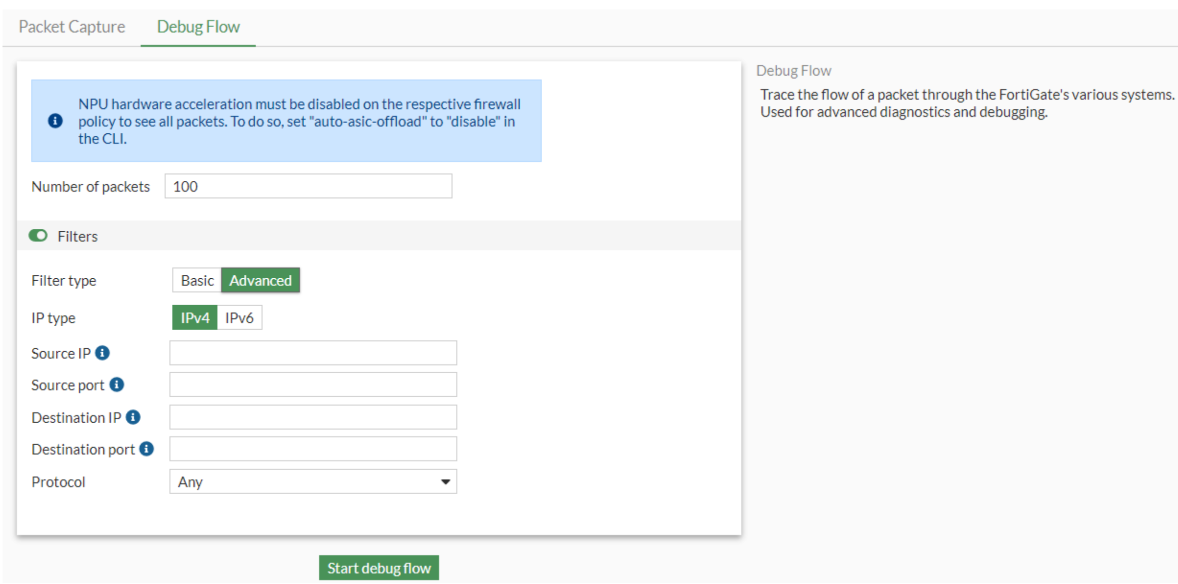
To run a debug flow:

1. Go to *Network > Diagnostics* and select the *Debug Flow* tab.
2. Optionally, enable *Filters* and select a *Filter type*:
   a. *Basic*: filter by *IP address*, *Port*, and *Protocol*, which is the equivalent of:
      - `# diagnose debug flow filter addr <addr/range>`
      - `# diagnose debug flow filter port <port/range>`
      - `# diagnose debug flow filter proto <protocol>`

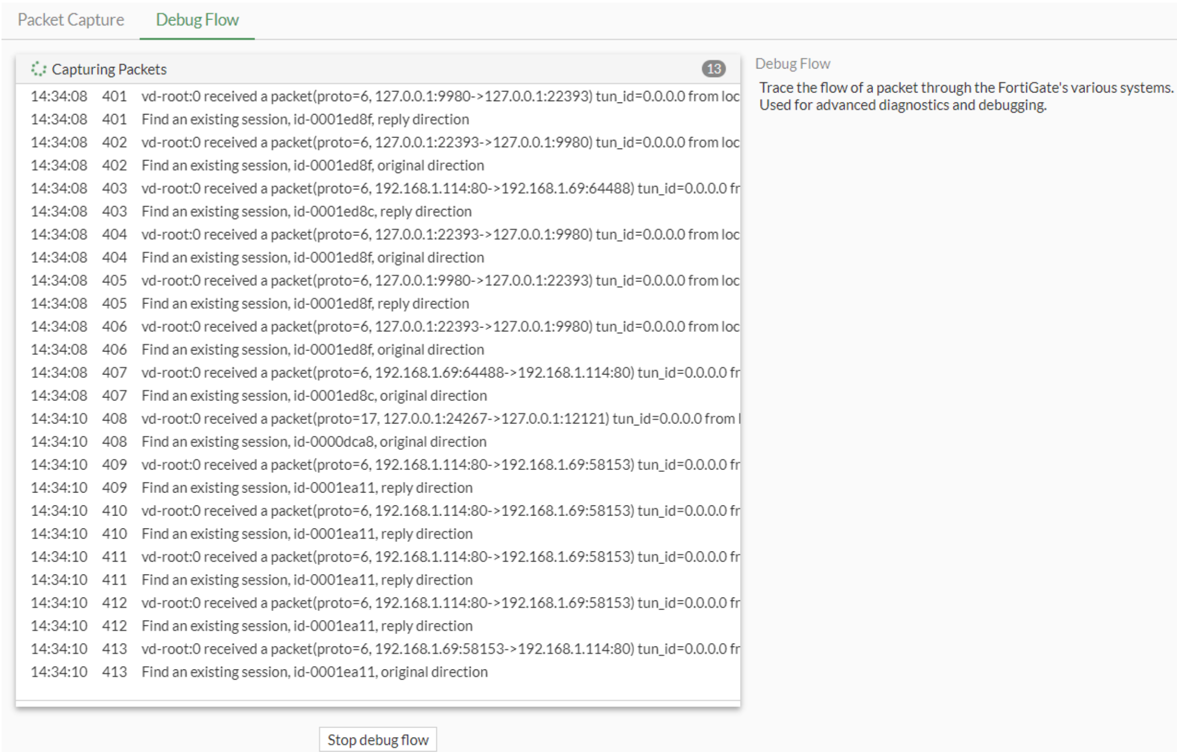


b. *Advanced*: filter by *Source IP*, *Source port*, *Destination IP*, *Destination port*, and *Protocol*, which is the equivalent of:

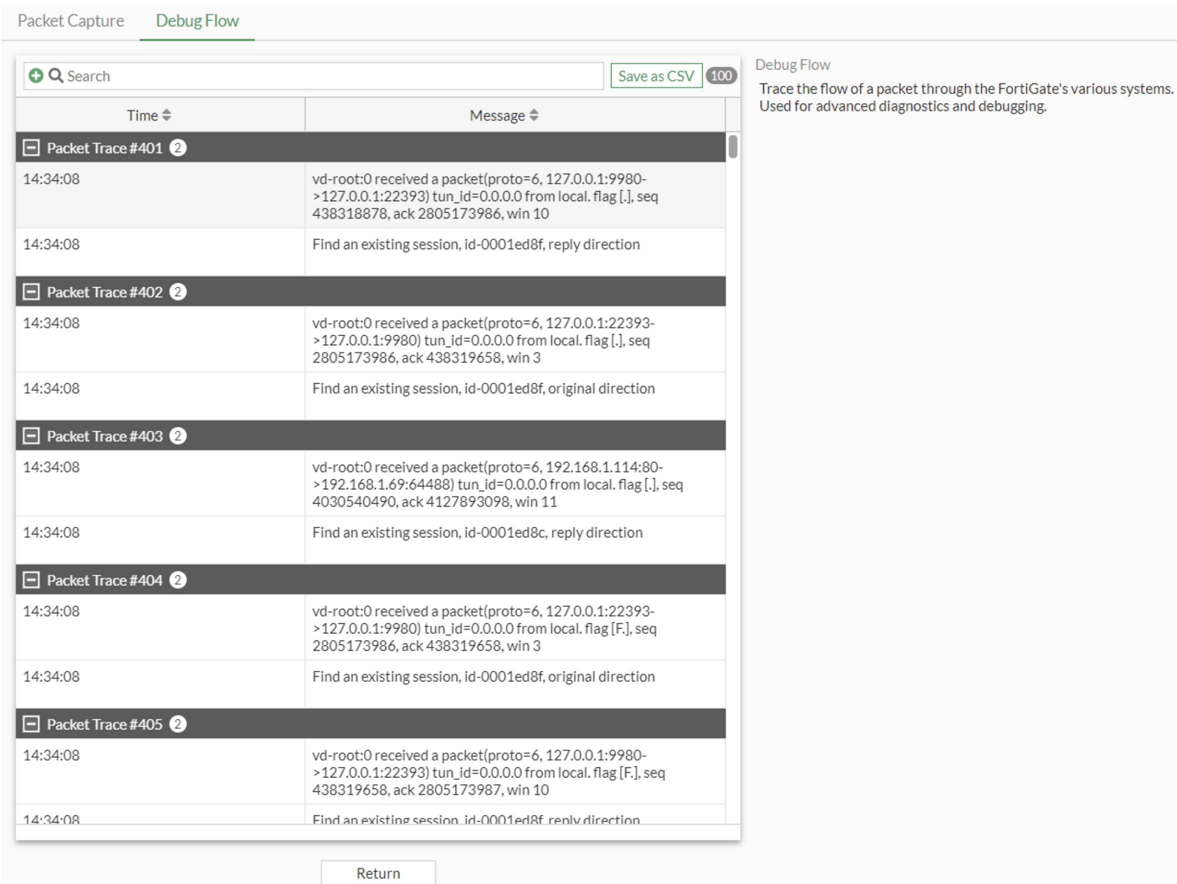
- `# diagnose debug flow filter saddr <addr/range>`
- `# diagnose debug flow filter sport <port/range>`
- `# diagnose debug flow filter daddr <addr/range>`
- `# diagnose debug flow filter dport <port/range>`
- `# diagnose debug flow filter proto <protocol>`



3. Click *Start debug flow*. The debug messages are visible in real-time.

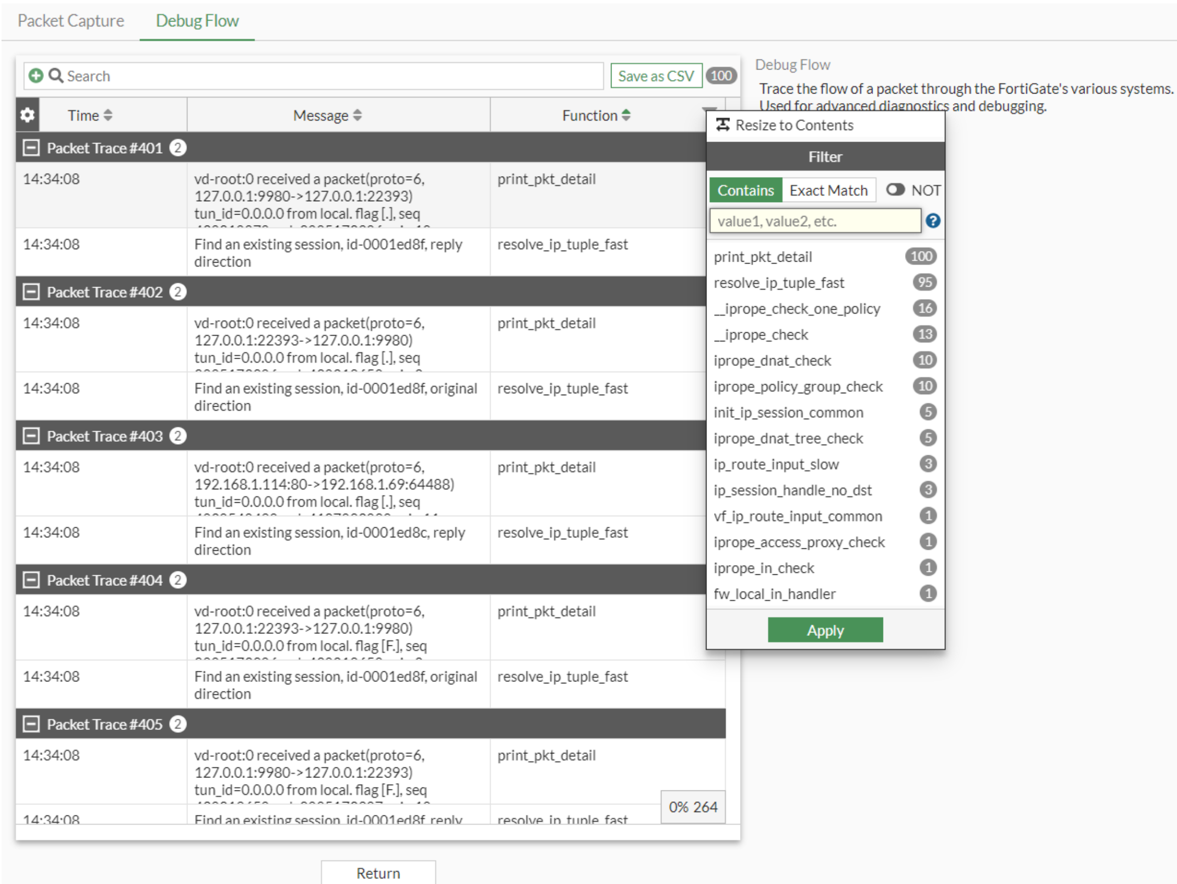


4. When the debug flow is finished (or you click *Stop debug flow*), click *Save as CSV*. The CSV file is automatically downloaded.



The current output can be filtered by *Time* and *Message*. The *Function* field can be added.

5. Hover over the table header and click the gear icon (*Configure Table*).
6. Select *Function* and click *Apply*. The *Function* column is displayed and can be used to filter the output for further analysis.


Packet Capture
Debug Flow
Search
Save as CSV
100
Time
Message
Function
Packet Trace #401
14:34:08
vd-root:0 received a packet(proto=6, 127.0.0.1:9980->127.0.0.1:22393) tun_id=0.0.0.0 from local. flag [.], seq
print_pkt_detail
14:34:08
Find an existing session, id-0001ed8f, reply direction
resolve_ip_tuple_fast
Packet Trace #402
14:34:08
vd-root:0 received a packet(proto=6, 127.0.0.1:22393->127.0.0.1:9980) tun_id=0.0.0.0 from local. flag [.], seq
print_pkt_detail
14:34:08
Find an existing session, id-0001ed8f, original direction
resolve_ip_tuple_fast
Packet Trace #403
14:34:08
vd-root:0 received a packet(proto=6, 192.168.1.114:80->192.168.1.69:64488) tun_id=0.0.0.0 from local. flag [.], seq
print_pkt_detail
14:34:08
Find an existing session, id-0001ed8c, reply direction
resolve_ip_tuple_fast
Packet Trace #404
14:34:08
vd-root:0 received a packet(proto=6, 127.0.0.1:22393->127.0.0.1:9980) tun_id=0.0.0.0 from local. flag [F.], seq
print_pkt_detail
14:34:08
Find an existing session, id-0001ed8f, original direction
resolve_ip_tuple_fast
Packet Trace #405
14:34:08
vd-root:0 received a packet(proto=6, 127.0.0.1:9980->127.0.0.1:22393) tun_id=0.0.0.0 from local. flag [F.], seq
print_pkt_detail
0% 264
Resize to Contents
Filter
Contains
Exact Match
NOT
value1, value2, etc.
print_pkt_detail 100
resolve_ip_tuple_fast 95
__iprope_check_one_policy 16
__iprope_check 13
iprope_dnat_check 10
iprope_policy_group_check 10
init_ip_session_common 5
iprope_dnat_tree_check 5
ip_route_input_slow 3
ip_session_handle_no_dst 3
vf_ip_route_input_common 1
iprope_access_proxy_check 1
iprope_in_check 1
fw_local_in_handler 1
Apply
Debug Flow
Trace the flow of a packet through the FortiGate's various systems.
and debugging.
Return

SD-WAN

The following topics provide information about SD-WAN:

SD-WAN overview

SD-WAN is a software-defined approach to managing Wide-Area Networks (WAN). It consolidates the physical transport connections, or underlays, and monitors and load-balances traffic across the links. VPN overlay networks can be built on top of the underlays to control traffic across different sites.

Health checks and SD-WAN rules define the expected performance and business priorities, allowing the FortiGate to automatically and intelligently route traffic based on the application, internet service, or health of a particular connection.

WAN security and intelligence can be extended into the LAN by incorporating wired and wireless networks under the same domain. FortiSwitch and FortiAP devices integrate seamlessly with the FortiGate to form the foundation of an SD-Branch.

Some of the key benefits of SD-WAN include:

- Reduced cost with transport independence across MPLS, 4G/5G LTE, and others.
- Reduced complexity with a single vendor and single-pane-of-glass management.
- Improve business application performance thanks to increased availability and agility.
- Optimized user experience and efficiency with SaaS and public cloud applications.

SD-WAN components

SD-WAN can be broken down into three layers:

- Management and orchestration
- Control, data plane, and security
- Network access

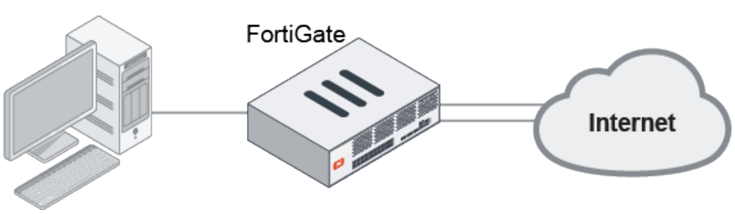
The control, data plane, and security layer can only be deployed on a FortiGate. The other two layers can help to scale and enhance the solution. For large deployments, FortiManager and FortiAnalyzer provide the management and orchestration capabilities FortiSwitch and FortiAP provide the components to deploy an SD-Branch.

Layer	Functions	Devices	
Management and orchestration	• Unified management • Template based solution • Zero touch provisioning • Logging, monitoring, and analysis • Automated orchestration using the REST API	FortiManager	FortiAnalyzer
Control, data plane, and security	• Consolidation of underlays and overlays into SD-WAN zones • Scalable VPN solutions using ADVPN • Static and dynamic routing definition • SD-WAN health-checks and monitoring • Application-aware steering and intelligence • NGFW firewalling	FortiGate Underlay, overlay, routing SD-WAN Security	
Network access	• Wired and wireless network segmentation • Built-in network access control	FortiSwitch	FortiAP

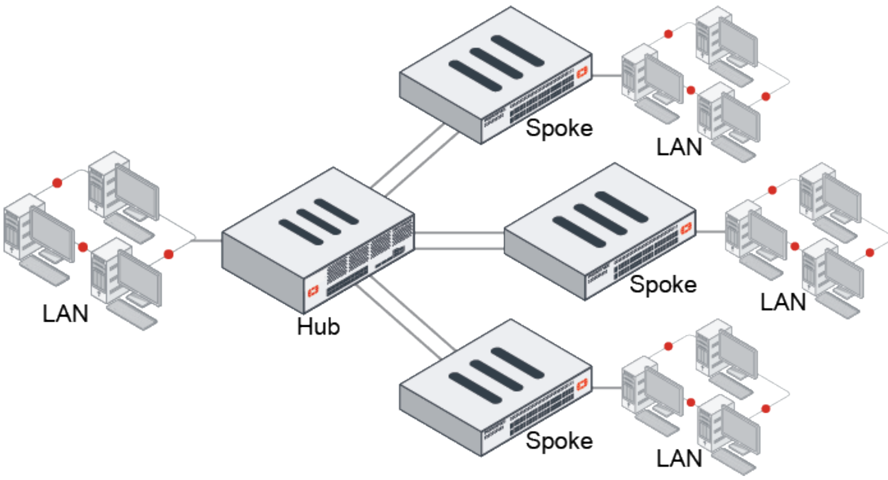
SD-WAN designs and architectures

The core functionalities of Fortinet's SD-WAN solution are built into the FortiGate. Whether the environment contains one FortiGate, or one hundred, you can use SD-WAN by enabling it on the individual FortiGates.

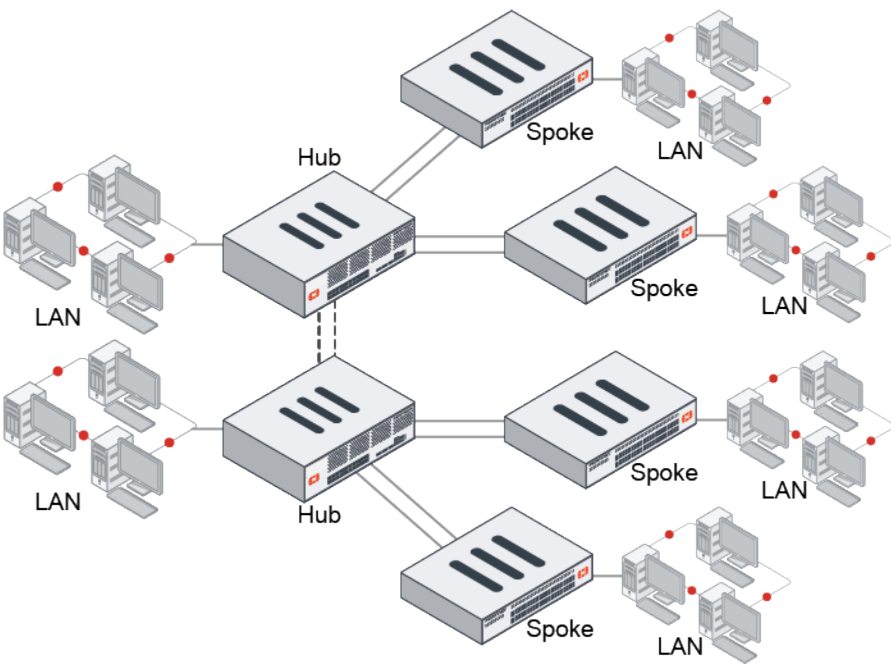
At a basic level, SD-WAN can be deployed on a single device in a single site environment:



At a more advanced level, SD-WAN can be deployed in a multi-site, hub and spoke environment:



At an enterprise or MSSP level, the network can include multiple hubs, possibly across multiple regions:



For more details, see the SD-WAN / SD-Branch Architecture for MSSPs guide.

SD-WAN designs principles

The Five-pillar approach, described in the SD-WAN / SD-Branch Architecture for MSSPs guide, is recommended when designing a secure SD-WAN solution.

Pillar	Overview
Underlay	Choose the WAN links to use.
Overlay	Choose the topology to interconnect your sites.
Routing	Choose how to propagate routes between your sites.

Pillar	Overview
Security	Choose how to protect each of the available paths.
SD-WAN	Choose the strategy used to pick one of the available paths.

Underlay

Determine the WAN links that will be used for the underlay network, such as your broadband link, MPLS, 4G/5G LTE connection, and others.

For each link, determine the bandwidth, quality and reliability (packet loss, latency, and jitter), and cost. Use this information to determine which link to prefer, what type of traffic to send across the each link, and to help you the baselines for health-checks.

Overlay

VPN overlays are needed when traffic must travel across multiple sites. These are usually site-to-site IPsec tunnels that interconnect branches, datacenters, and the cloud, forming a hub-and-spoke topology.

The management and maintenance of the tunnels should be considered when determining the overlay network requirements. Manual tunnel configuration might be sufficient in a small environment, but could become unmanageable as the environment size increases. ADVPN can be used to help scale the solution; see ADVPN on page 1530 for more information.

Routing

Traditional routing designs manipulate routes to steer traffic to different links. SD-WAN uses traditional routing to build the basic routing table to reach different destinations, but uses SD-WAN rules to steer traffic. This allows the steering to be based on criteria such as destination, internet service, application, route tag, and the health of the link. Routing in an SD-WAN solution is used to identify all possible routes across the underlays and overlays, which the FortiGate balances using ECMP.

In the most basic configuration, static gateways that are configured on an SD-WAN member interface automatically provide the basic routing needed for the FortiGate to balance traffic across the links. As the number of sites and destinations increases, manually maintaining routes to each destination becomes difficult. Using dynamic routing to advertise routes across overlay tunnels should be considered when you have many sites to interconnect.

Security

Security involves defining policies for access control and applying the appropriate protection using the FortiGate's NGFW features. Efficiently grouping SD-WAN members into SD-WAN zones must also be considered. Typically, underlays provide direct internet access and overlays provide remote internet or network access. Grouping the underlays together into one zone, and the overlays into one or more zones could be an effective method.

SD-WAN

The SD-WAN pillar is the intelligence that is applied to traffic steering decisions. It is comprised of four primary elements:

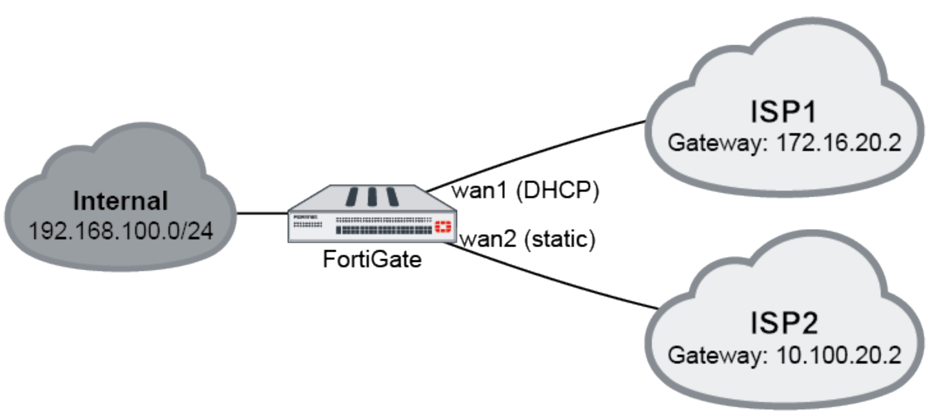
- **SD-WAN zones**
  SD-WAN is divided into zones. SD-WAN member interfaces are assigned to zones, and zones are used in policies as source and destination interfaces. You can define multiple zones to group SD-WAN interfaces together, allowing logical groupings for overlay and underlay interfaces. Routing can be configured per zone.
  See SD-WAN zones on page 497.
- **SD-WAN members**
  Also called interfaces, SD-WAN members are the ports and interfaces that are used to run traffic. At least one interface must be configured for SD-WAN to function.
  See Configuring the SD-WAN interface on page 488.
- **Performance SLAs**
  Also called health-checks, performance SLAs are used to monitor member interface link quality, and to detect link failures. When the SLA falls below a configured threshold, the route can be removed, and traffic can be steered to different links in the SD-WAN rule.
  SLA health-checks use active or passive probing:
  - Active probing requires manually defining the server to be probed, and generates consistent probing traffic.
  - Passive probing uses active sessions that are passing through firewall policies used by the related SD-WAN interfaces to derive health measurements. It reduces the amount of configuration, and eliminates probing traffic. See Passive WAN health measurement on page 517 for details.

  See Performance SLA on page 506.
- **SD-WAN rules**
  Also called services, SD-WAN rules control path selection. Specific traffic can be dynamically sent to the best link, or use a specific route
  Rules control the strategy that the FortiGate uses when selecting the outbound traffic interface, the SLAs that are monitored when selecting the outgoing interface, and the criteria for selecting the traffic that adheres to the rule. When no SD-WAN rules match the traffic, the implicit rule applies.
  See SD-WAN rules on page 538.

SD-WAN quick start

This section provides an example of how to start using SD-WAN for load balancing and redundancy.

In this example, two ISP internet connections, wan1 (DHCP) and wan2 (static), use SD-WAN to balance traffic between them at 50% each.



Configuring the SD-WAN interface

First, SD-WAN must be enabled and member interfaces must be selected and added to a zone. The selected FortiGate interfaces can be of any type (physical, aggregate, VLAN, IPsec, and others), but must be removed from any other configurations on the FortiGate.

In this step, two interfaces are configured and added to the default SD-WAN zone (virtual-wan-link) as SD-WAN member interfaces. This example uses a mix of static and dynamic IP addresses; your deployment could also use only one or the other.

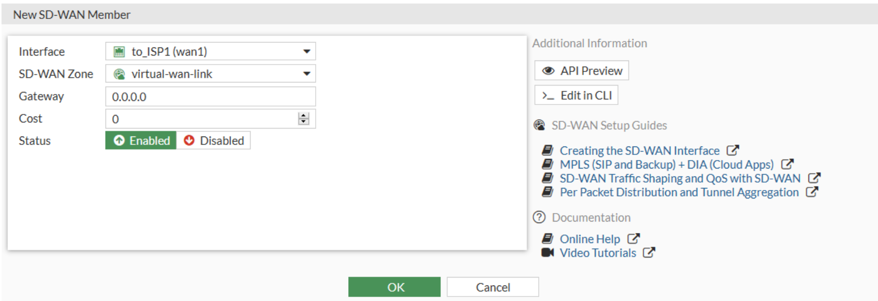
Once the SD-WAN members are created and added to a zone, the zone can be used in firewall policies, and the whole SD-WAN can be used in static routes.

To configure SD-WAN members:

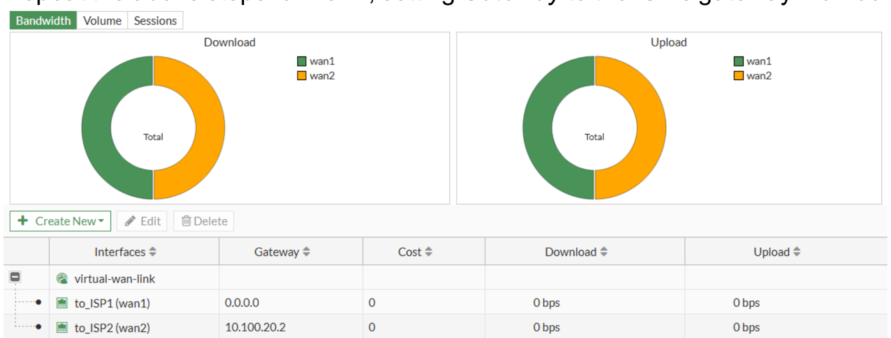
1. Configure the wan1 and wan2 interfaces. See Interface settings on page 137 for details.
   a. Set the wan1 interface *Addressing mode* to *DHCP* and *Distance* to *10*.

   By default, a DHCP interface has a distance of 5, and a static route has a distance of 10. It is important to account for this when configuring your SD-WAN for 50/50 load balancing by setting the DHCP interface's distance to 10.

   b. Set the wan2 interface *IP/Netmask* to *10.100.20.1 255.255.255.0*.
2. Go to *Network > SD-WAN*, select the *SD-WAN Zones* tab, and click *Create New > SD-WAN Member*.
3. Set the *Interface* to *wan1*.
4. Leave *SD-WAN Zone* as *virtual-wan-link*.
5. As wan1 uses DHCP, leave *Gateway* set to *0.0.0.0*.
   If IPv6 visibility is enabled in the GUI, an IPv6 gateway can also be added for each member. See Feature visibility on page 2163 for details.
6. Leave *Cost* as *0*.
   The *Cost* field is used by the Lowest Cost (SLA) strategy. The link with the lowest cost is chosen to pass traffic. The lowest possible *Cost* is *0*.



7. Set *Status* to *Enable*, and click *OK*.
8. Repeat the above steps for wan2, setting *Gateway* to the ISP's gateway: *10.100.20.2*.

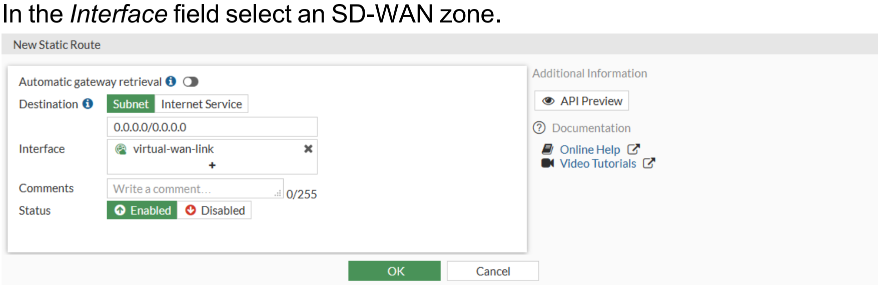


Adding a static route

You must configure a default route for the SD-WAN. The default gateways for each SD-WAN member interface do not need to be defined in the static routes table. FortiGate will decide what route or routes are preferred using Equal Cost Multi-Path (ECMP) based on distance and priority.

To create a static route for SD-WAN:

1. Go to *Network > Static Routes*.
2. Click *Create New*. The *New Static Route* page opens.
3. Set *Destination* to *Subnet*, and leave the IP address and subnet mask as *0.0.0.0/0.0.0.0*.
4. In the *Interface* field select an SD-WAN zone.



5. Ensure that *Status* is *Enabled*.
6. Click *OK*.

Selecting the implicit SD-WAN algorithm

SD-WAN rules define specific routing options to route traffic to an SD-WAN member.

If no routing rules are defined, the default *Implicit* rule is used. It can be configured to use one of five different load balancing algorithms. See Implicit rule on page 546 for more details and examples.

This example shows four methods to equally balance traffic between the two WAN connections. Go to *Network > SD-WAN*, select the *SD-WAN Rules* tab, and edit the *sd-wan* rule to select the method that is appropriate for your requirements.

- *Source IP* (CLI command: `source-ip-based`):
  Select this option to balance traffic equally between the SD-WAN members according to a hash algorithm based on the source IP addresses.
- *Session* (`weight-based`):
  Select this option to balance traffic equally between the SD-WAN members by the session numbers ratio among its members. Use weight 50 for each of the 2 members.
- *Source-Destination IP* (`source-dest-ip-based`):
  Select this option to balance traffic equally between the SD-WAN members according to a hash algorithm based on the source and destination IP addresses.
- *Volume* (`measured-volume-based`):
  Select this option to balance traffic equally between the SD-WAN members according to the bandwidth ratio among its members.

Configuring firewall policies for SD-WAN

SD-WAN zones can be used in policies as source and destination interfaces. Individual SD-WAN members cannot be used in policies.

You must configure a policy that allows traffic from your organization's internal network to the SD-WAN zone. Policies configured with the SD-WAN zone apply to all SD-WAN interface members in that zone.

To create a firewall policy for SD-WAN:

1. Go to *Policy & Objects > Firewall Policy*.
2. Click *Create New*. The *New Policy* page opens.

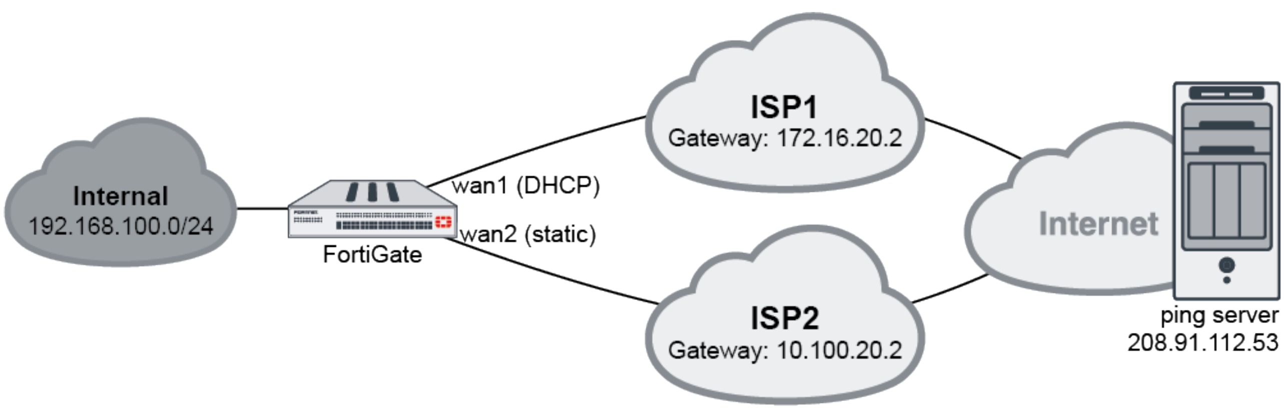
3. Configure the following:

Name	Enter a name for the policy.
Incoming Interface	*internal*
Outgoing Interface	*virtual-wan-link*
Source	*all*
Destination	*all*
Schedule	*always*
Service	*ALL*
Action	*ACCEPT*
Firewall / Network Options	Enable *NAT* and set *IP Pool Configuration* to *Use Outgoing Interface Address*.
Security Profiles	Apply profiles as required.
Logging Options	Enable *Log Allowed Traffic* and select *All Sessions*. This allows you to verify results later.

4. Enable the policy, then click *OK*.

Link monitoring and failover

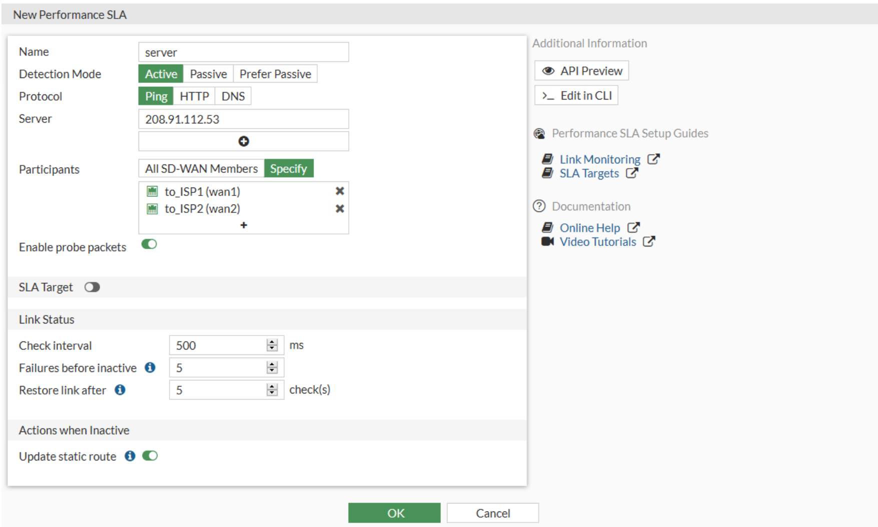
Performance SLA link monitoring measures the health of links that are connected to SD-WAN member interfaces by sending probing signals through each link to a server, and then measuring the link quality based on latency, jitter, and packet loss. If a link is broken, the routes on that link are removed and traffic is routed through other links. When the link is working again, the routes are re-enabled. This prevents traffic being sent to a broken link and lost.



In this example, the detection server IP address is 208.91.112.53. A performance SLA is created so that, if ping fails per the metrics defined, the routes to that interface are removed and traffic is detoured to the other interface. The ping protocol is used, but other protocols could also be selected as required.

To configure a performance SLA:

1. Go to *Network > SD-WAN*, select the *Performance SLAs* tab, and click *Create New*.
2. Enter a name for the SLA and set *Protocol* to *Ping*.
3. In the *Server* field, enter the detection server IP address (208.91.112.53 in this example).
4. In the *Participants* field, select *Specify* and add wan1 and wan2.



SLA targets are not required for link monitoring.

5. Configure the required metrics in *Link Status*.
6. Ensure that *Update static route* is enabled. This disables static routes for the inactive interface and restores routes on recovery.
7. Click *OK*.

Results

The following GUI pages show the function of the SD-WAN and can be used to confirm that it is setup and running correctly:

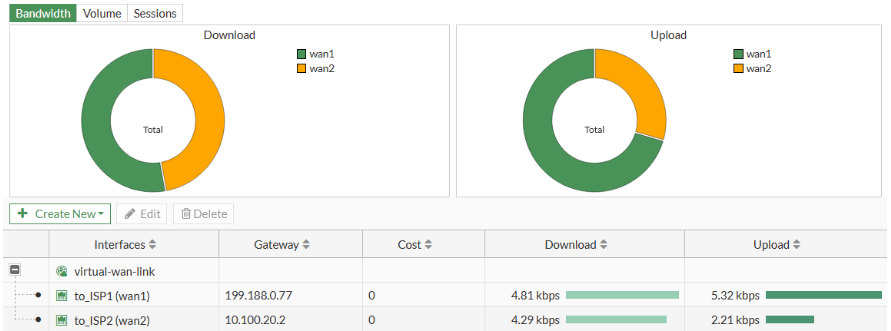
- Interface usage on page 492
- Performance SLA on page 493
- Routing table on page 495
- Firewall policy on page 495

Interface usage

Go to *Network > SD-WAN* and select the *SD-WAN Zones* tab to review the SD-WAN interfaces' usage.

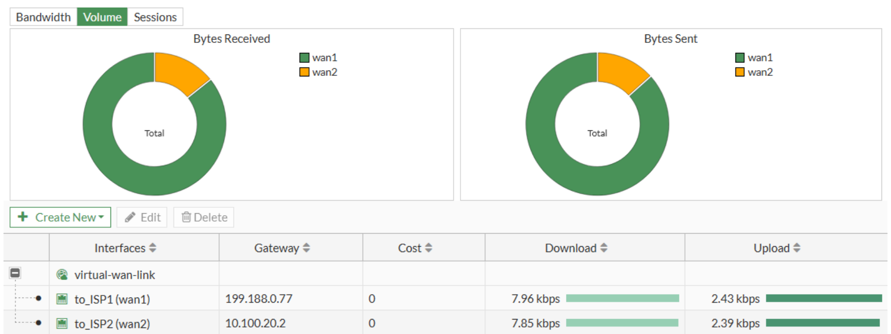
Bandwidth

Select *Bandwidth* to view the amount of downloaded and uploaded data for each interface.



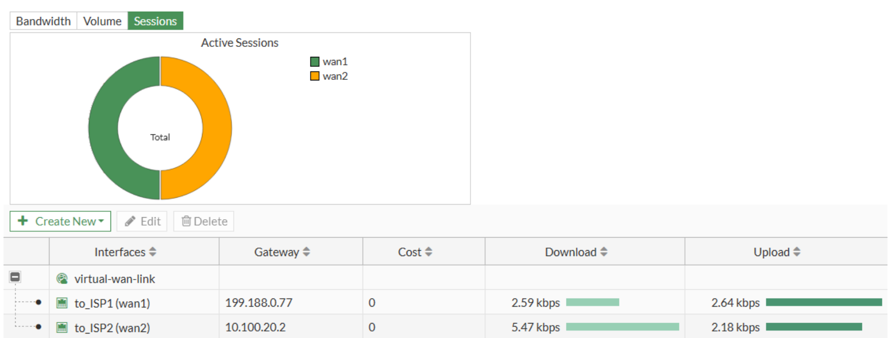
Volume

Select *Volume* to see donut charts of the received and sent bytes on the interfaces.



Sessions

Select *Sessions* to see a donut chart of the number of active sessions on each interface.

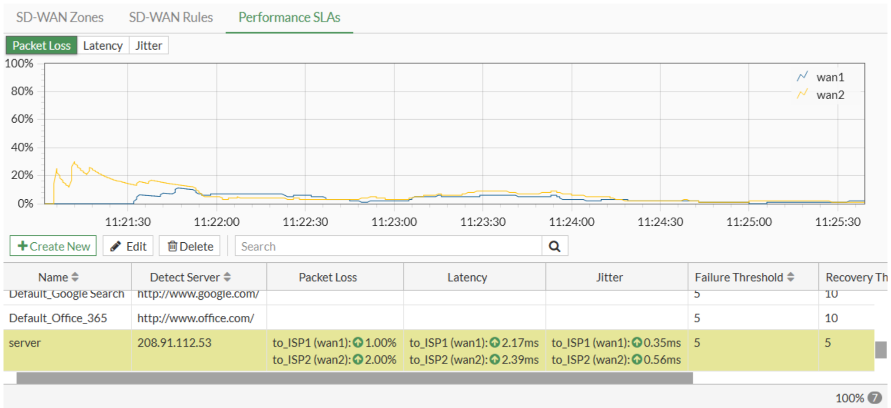


Performance SLA

Go to *Network > SD-WAN*, select the *Performance SLAs* tab, and select the SLA from the table (*server* in this example) to view the packet loss, latency, and jitter on each SD-WAN member in the health check server.

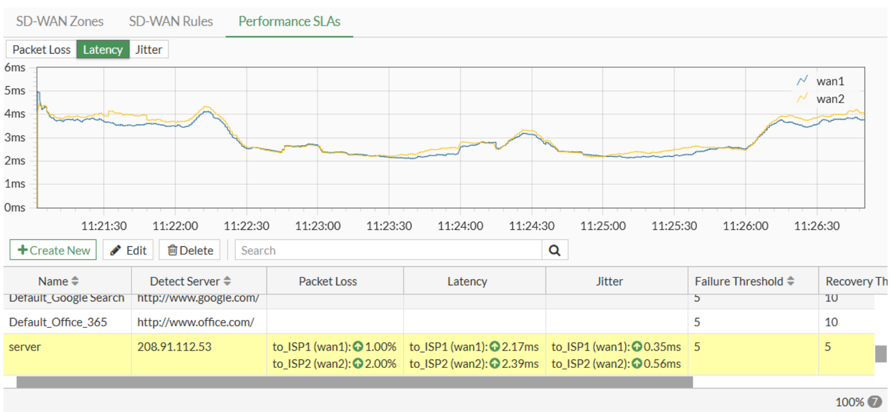
Packet loss

Select *Packet Loss* to see the percentage of packets lost for each member.



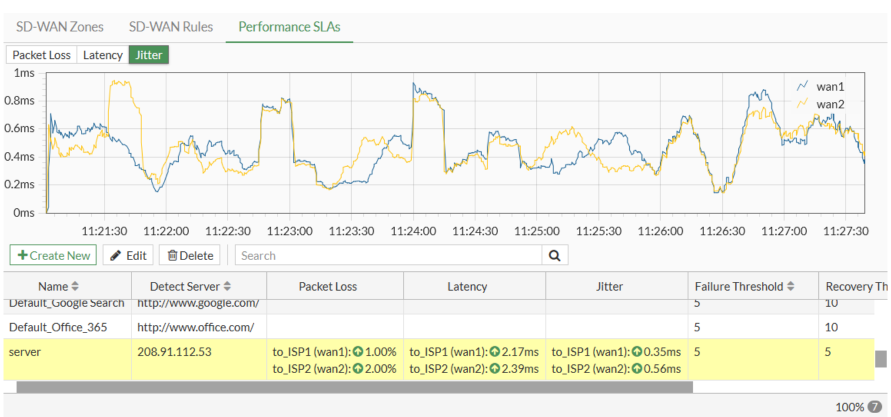
Latency

Select *Latency* to see the current latency, in milliseconds, for each member.



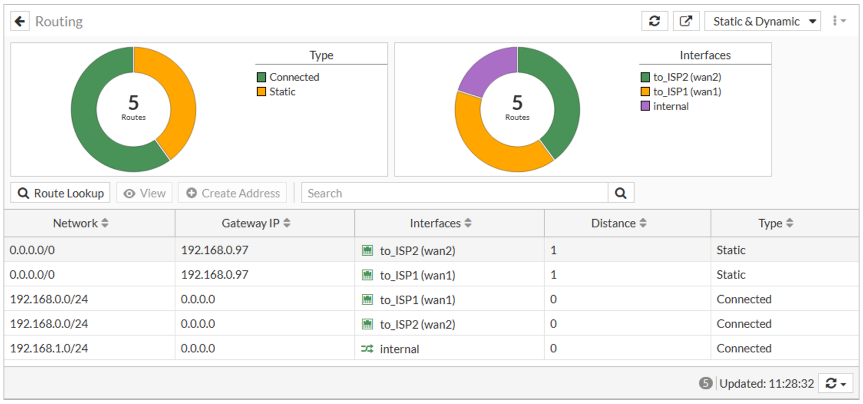
Jitter

Select *Jitter* to see the jitter, in milliseconds, for each member.



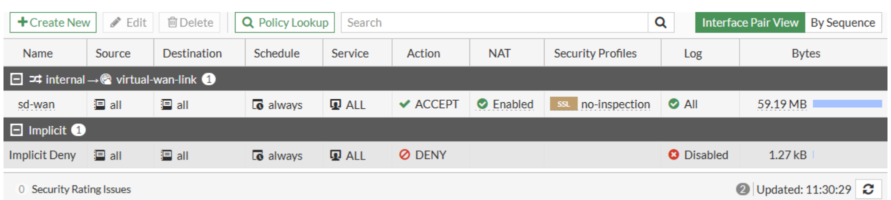
Routing table

Go to *Dashboard* > *Network*, expand the *Routing* widget, and select *Static & Dynamic* to review all static and dynamic routes. For more information about the widget, see Static & Dynamic Routing monitor on page 89.



Firewall policy

Go to *Policy & Objects* > *Firewall Policy* to review the SD-WAN policy.



Configuring SD-WAN in the CLI

This example can be entirely configured using the CLI.

To configure SD-WAN in the CLI:

1. Configure the wan1 and wan2 interfaces:

```
config system interface
    edit "wan1"
        set alias to_ISP1
        set mode dhcp
        set distance 10
    next
    edit "wan2"
        set alias to_ISP2
        set ip 10.100.20.1 255.255.255.0
    next
end
```

2. Enable SD-WAN and add the interfaces as members:

```
config system sdwan
    set status enable
    config members
        edit 1
            set interface "wan1"
        next
        edit 2
            set interface "wan2"
            set gateway 10.100.20.2
        next
    end
end
```

If no SD-WAN zone is specified, members are added to the default *virtual-wan-link* zone.

3. Create a static route for SD-WAN:

```
config router static
    edit 1
        set sdwan-zone "virtual-wan-link"
    next
end
```

4. Select the implicit SD-WAN algorithm:

```
config system sdwan
    set load-balance-mode {source-ip-based | weight-based | source-dest-ip-based |
measured-volume-based}
end
```

5. Create a firewall policy for SD-WAN:

```
config firewall policy
    edit <policy_id>
        set name <policy_name>
        set srcintf "internal"
        set dstintf "virtual-wan-link"
        set srcaddr all
        set dstaddr all
        set action accept
        set schedule always
        set service ALL
        set utm-status enable
        set ssl-ssh-profile <profile_name>
        set av-profile <profile_name>
        set webfilter-profile <profile_name>
        set dnsfilter-profile <profile_name>
        set emailfilter-profile <profile_name>
        set ips_sensor <sensor_name>
        set application-list <app_list>
        set voip-profile <profile_name>
        set logtraffic all
        set nat enable
```

```
        set status enable
    next
end
```

6. Configure a performance SLA:

```
config system sdwan
    config health-check
        edit "server"
            set server "208.91.112.53"
            set update-static-route enable
            set members 1 2
        next
    end
end
```

Results

To view the routing table:

```
# get router info routing-table all

Routing table for VRF=0
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default

S*      0.0.0.0/0 [1/0] via 172.16.20.2, wan1
                  [1/0] via 10.100.20.2, wan2
C       10.100.20.0/24 is directly connected, wan2
C       172.16.20.2/24 is directly connected, wan1
C       192.168.0.0/24 is directly connected, internal
```

To diagnose the Performance SLA status:

```
FGT # diagnose sys sdwan health-check
Health Check(server):
Seq(1): state(alive), packet-loss(0.000%) latency(15.247), jitter(5.231) sla_map=0x0
Seq(2): state(alive), packet-loss(0.000%) latency(13.621), jitter(6.905) sla_map=0x0
```

SD-WAN zones

SD-WAN is divided into zones. SD-WAN member interfaces are assigned to zones, and zones are used in policies, static routes, and SD-WAN rules.

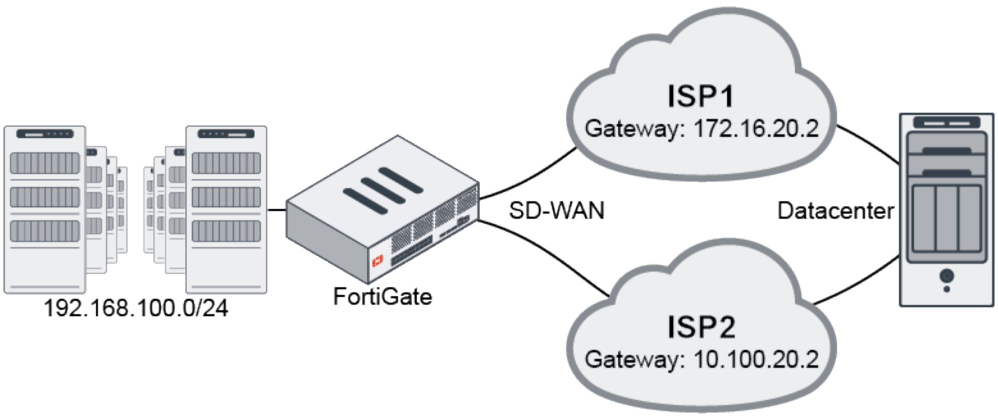
You can define multiple zones to group SD-WAN interfaces together, allowing logical groupings for overlay and underlay interfaces. Zones are used in firewall policies, as source and destination interfaces, to allow for more granular control. SD-WAN members cannot be used directly in policies.

SD-WAN zones and members can both be used in IPv4 and IPv6 static routes to make route configuration more flexible, and in SD-WAN rules to simplify the rule configuration. See Specify an SD-WAN zone in static routes and SD-WAN rules on page 502 for more information.

In the CLI:

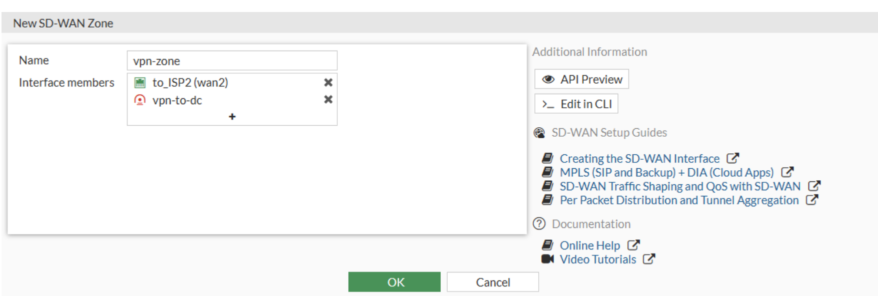
- `config system sdwan` has replaced `config system virtual-wan-link`.
- `diagnose sys sdwan` has replaced `diagnose sys virtual-wan-link`.
- When configuring a static route, the `sdwan-zone` variable has replaced the `sdwan` variable.

When the Security Fabric is configured, SD-WAN zones are included in the Security Fabric topology views.

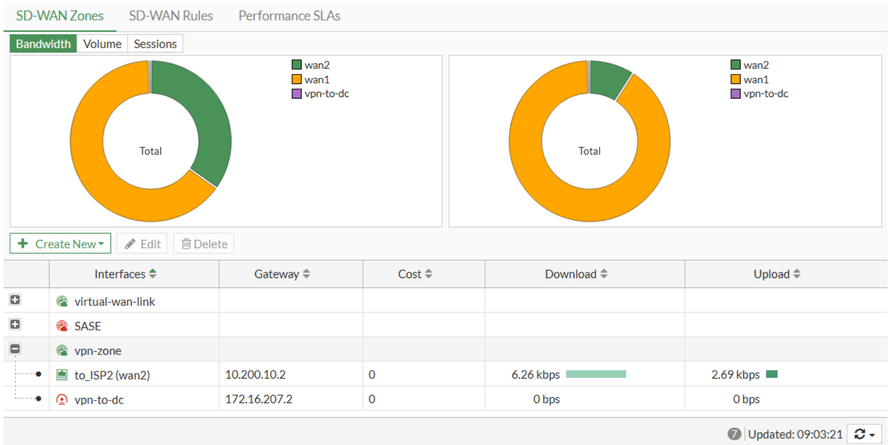


To create an SD-WAN zone in the GUI:

1. Go to *Network > SD-WAN* and select the *SD-WAN Zones* tab.
   The default SD-WAN zones are *virtual-wan-link* and *SASE*.
2. Click *Create New > SD-WAN Zone*.
3. Enter a name for the new zone, such as *vpn-zone*.
4. If SD-WAN members have already been created, add the required members to the zone.
   Members can also be added to the zone after it has been created by editing the zone, or when creating or editing the member.

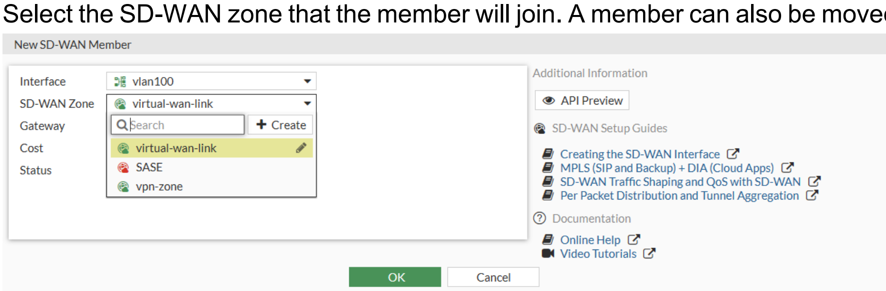


5. Click *OK*.

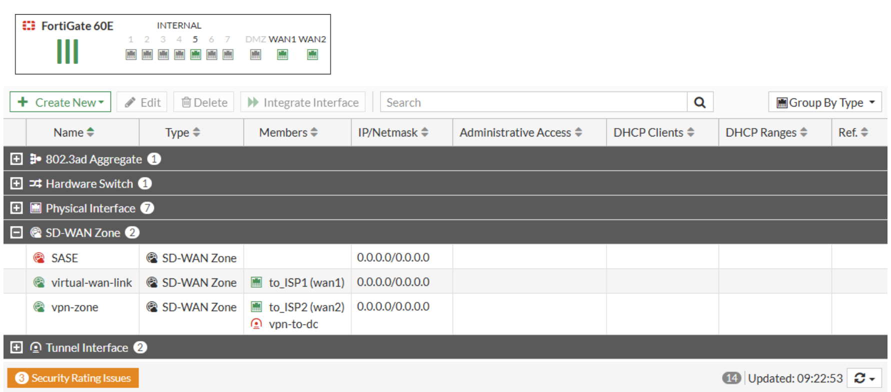


To create an SD-WAN interface member in the GUI:

1. Go to *Network > SD-WAN*, select the *SD-WAN Zones* tab, and click *Create New > SD-WAN Member*.
2. Select an interface.
   The interface can also be left as *none* and selected later, or click *+VPN* to create an IPsec VPN for the SD-WAN member.
3. Select the SD-WAN zone that the member will join. A member can also be moved to a different zone at any time.

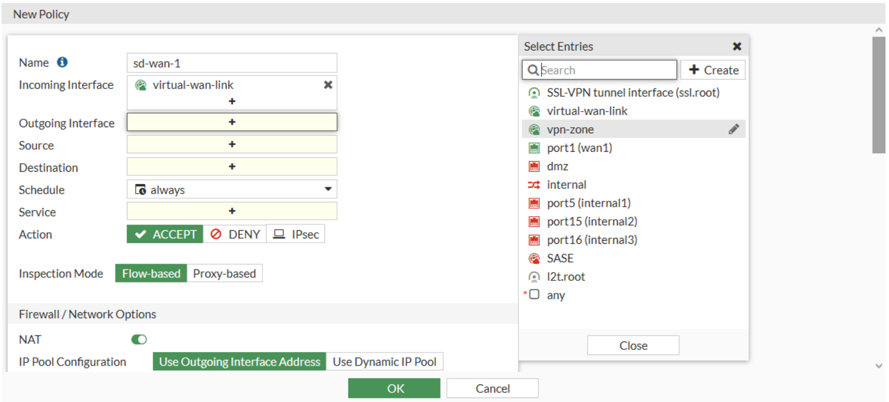


4. Set the *Gateway*, *Cost*, and *Status* as required.
5. Click *OK*.
   The interface list at *Network > Interfaces* shows the SD-WAN zones and their members.



To create a policy using the SD-WAN zone in the GUI:

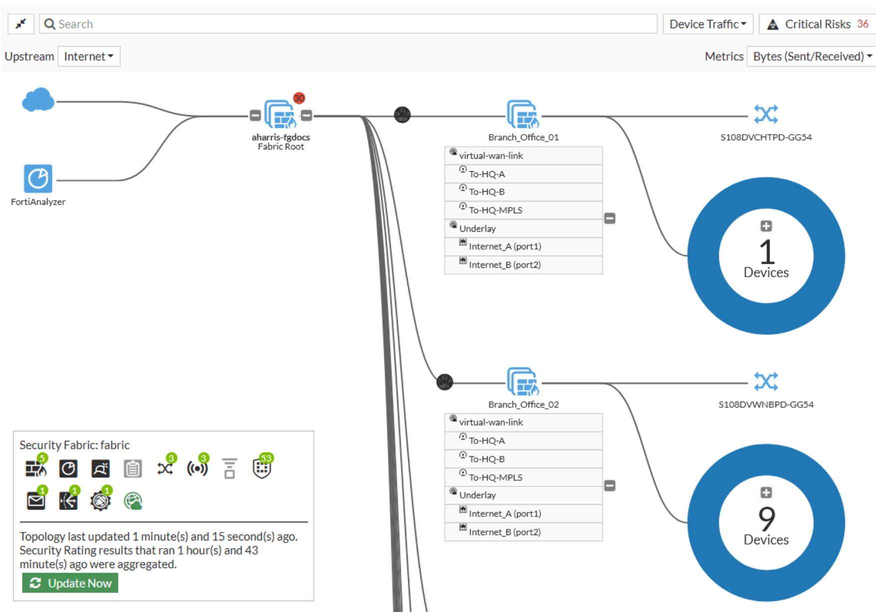
1. Go to *Policy & Objects > Firewall Policy*, *Policy & Objects > Proxy Policy*, or *Policy & Objects > Security Policy*.
2. Click *Create New* .
3. Configure the policy settings as needed, selecting an SD-WAN zone or zones for the incoming and/or outgoing interface.

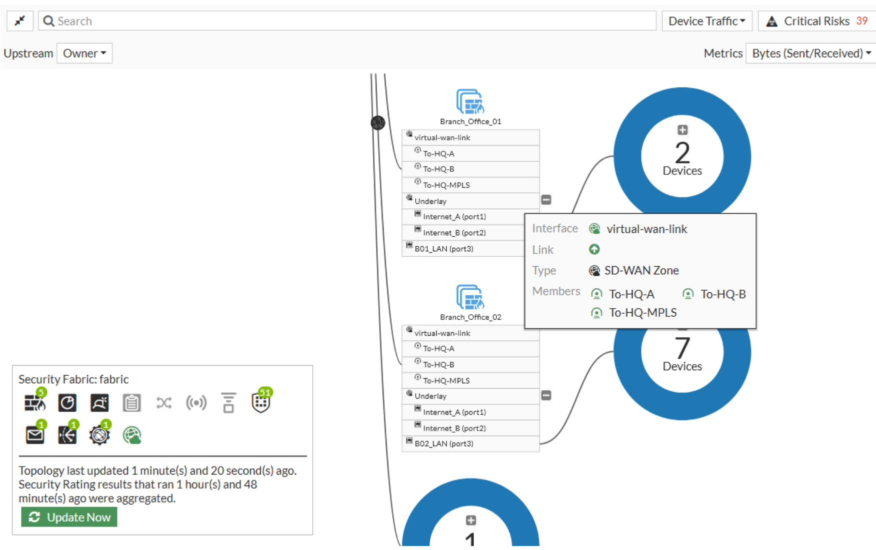


4. Click *OK*.

To view SD-WAN zones in a Security Fabric topology:

1. Go to *Security Fabric > Physical Topology* or *Security Fabric > Logical Topology*. The SD-WAN zones and their members are shown.





To configure SD-WAN in the CLI:

1. Enable SD-WAN and create a zone:

```
config system sdwan
    set status enable
    config zone
        edit "vpn-zone"
        next
    end
end
```

2. Configure SD-WAN members and add them to a zone:

```
config system sdwan
    config members
        edit 1
            set interface "to_ISP2"
            set zone "vpn-zone"
        next
        edit 2
            set interface "vpn-to-dc"
            set zone "vpn-zone"
        next
    end
end
```

To create a policy using the SD-WAN zone in the CLI:

```
config firewall policy
    edit 1
        set name sd-wan-1
        set srcintf internal
        set dstintf vpn-zone
        set srcaddr all
        set dstaddr all
        set action accept
```

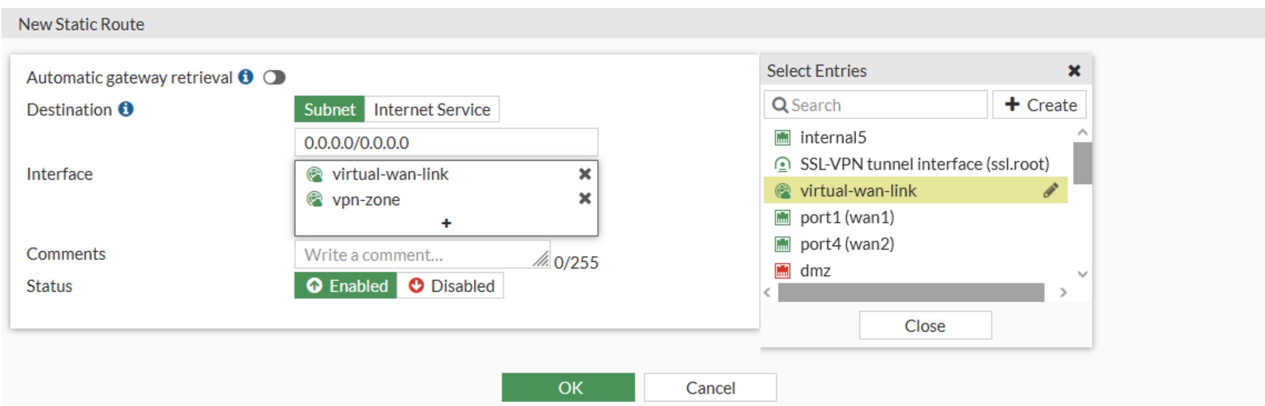
```
        set schedule always
        set service ALL
        set utm-status enable
        set logtraffic all
        set nat enable
        set status enable
    next
end
```

Specify an SD-WAN zone in static routes and SD-WAN rules

SD-WAN zones can be used in IPv4 and IPv6 static routes, and in SD-WAN service rules. This makes route configuration more flexible, and simplifies SD-WAN rule configuration.

To configure an SD-WAN zone in a static route in the GUI:

1. Go to *Network > Static Routes*
2. Edit an existing static route, or click *Create New* to create a new route.
3. Set *Interface* to one or more SD-WAN zones.



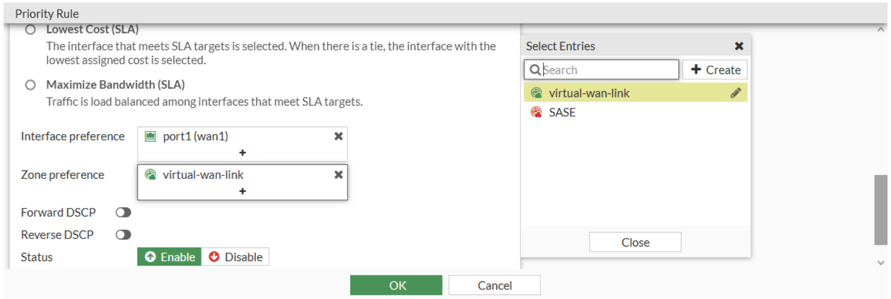
4. Configure the remaining settings are required.
5. Click *OK*.

To configure an SD-WAN zone in a static route in the CLI:

```
config router {static | static6}
    edit 1
        set sdwan-zone <zone> <zone> ...
    next
end
```

To configure an SD-WAN zone in an SD-WAN rule in the GUI:

1. Go to *Network > SD-WAN* and select the *SD-WAN Rules* tab
2. Edit an existing rule, or click *Create New* to create a new rule.
3. In the *Zone preference* field add one or more SD-WAN zones.



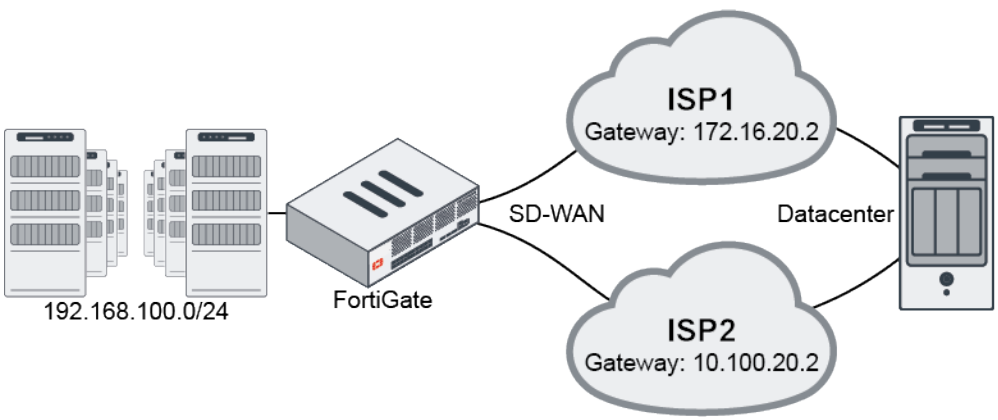
4. Configure the remaining settings are needed.
5. Click *OK*.

To configure an SD-WAN zone in an SD-WAN rule in the CLI:

```
config system sdwan
    config service
        edit 1
            set priority-zone <zone>
        next
    end
end
```

Examples

In these two examples, three SD-WAN members are created. Two members, port13 and port15, are in the default zone (*virtual-wan-link*), and the third member, to_FG_B_root, is in the *SASE* zone.



Example 1

In this example:

- Two service rules are created. Rule 1 uses the *virtual-wan-link* zone, and rule 2 uses the *SASE* zone.
- Two IPv4 static routes are created. The first route uses the *virtual-wan-link* zone, and the second route uses the *SASE* zone.

To configure the SD-WAN:

1. Assign port13 and port15 to the *virtual-wan-link* zone and to_FG_B_root to the *SASE* zone:

```
config system sdwan
    set status enable
    config members
        edit 1
            set interface "port13"
            set zone "virtual-wan-link"
            set gateway 10.100.1.1
        next
        edit 2
            set interface "port15"
            set zone "virtual-wan-link"
            set gateway 10.100.1.5
        next
        edit 3
            set interface "to_FG_B_root"
            set zone "SASE"
        next
    end
end
```

2. Create two service rules, one for each SD-WAN zone:

```
config system sdwan
    config service
        edit 1
            set dst "10.100.20.0"
            set priority-zone "virtual-wan-link"
        next
        edit 2
            set internet-service enable
            set internet-service-name "Fortinet-FortiGuard"
            set priority-zone "SASE"
        next
    end
end
```

3. Configure static routes for each of the SD-WAN zones:

```
config router static
    edit 1
        set distance 1
        set sdwan-zone "virtual-wan-link"
    next
    edit 2
        set dst 172.16.109.0 255.255.255.0
        set distance 1
        set sdwan-zone "SASE"
    next
end
```

To verify the results:

1. Check the service rule 1 diagnostics:

```
# diagnose sys sdwan service 1

Service(1): Address Mode(IPV4) flags=0x200 use-shortcut-sla
  Gen(1), TOS(0x0/0x0), Protocol(0: 1->65535), Mode(manual)
  Members(2):
    1: Seq_num(1 port13), alive, selected
    2: Seq_num(2 port15), alive, selected
  Dst address(1):
        10.100.20.0-10.100.20.255
```

   Both members of the *virtual-wan-link* zone are selected. In manual mode, the interface members are selected based on the member configuration order. In SLA and priority mode, the order depends on the link status. If all of the link statuses pass, then the members are selected based on the member configuration order.

2. Check the service rule 2 diagnostics:

```
# diagnose sys sdwan service 2

Service(2): Address Mode(IPV4) flags=0x200 use-shortcut-sla
  Gen(1), TOS(0x0/0x0), Protocol(0: 1->65535), Mode(manual)
  Members(1):
    1: Seq_num(3 to_FG_B_root), alive, selected
  Internet Service(1): Fortinet-FortiGuard(1245324,0,0,0)
```

   The member of the *SASE* zone is selected.

3. Review the routing table:

```
# get router info routing-table static
Routing table for VRF=0
S*      0.0.0.0/0 [1/0] via 10.100.1.1, port13
                  [1/0] via 10.100.1.5, port15
S       172.16.109.0/24 [1/0] via 172.16.206.2, to_FG_B_root
```

   The default gateway has the members from the *virtual-wan-link* zone, and the route to 172.16.10.9.0/24 has the single member from the *SASE* zone.

Example 2

In this example, two IPv6 static routes are created. The first route uses the *virtual-wan-link* zone, and the second route uses the *SASE* zone.

To configure the SD-WAN:

1. Configure port13 and port15 with IPv6 addresses and assign them to the *virtual-wan-link* zone, and assign to_FG_B_root to the *SASE* zone:

```
config system sdwan
    set status enable
    config members
        edit 1
            set interface "port13"
            set zone "virtual-wan-link"
            set gateway6 2004:10:100:1::1
```

```
            set source6 2004:10:100:1::2
        next
        edit 2
            set interface "port15"
            set zone "virtual-wan-link"
            set gateway6 2004:10:100:1::5
            set source6 2004:10:100:1::6
        next
        edit 3
            set interface "to_FG_B_root"
            set zone "SASE"
        next
    end
end
```

2. Configure IPv6 static routes for each of the SD-WAN zones:

```
config router static6
    edit 1
        set distance 1
        set sdwan-zone "virtual-wan-link"
    next
    edit 2
        set dst 2003:172:16:109::/64
        set distance 1
        set sdwan-zone "SASE"
    next
end
```

To verify the results:

1. Review the routing table:

```
# get router info6 routing-table static
Routing table for VRF=0
S*      ::/0 [1/0] via 2004:10:100:1::1, port13, 00:20:51, [1024/0]
             [1/0] via 2004:10:100:1::5, port15, 00:20:51, [1024/0]
S       2003:172:16:109::/64 [1/0] via ::ac10:ce02, to_FG_B_root, 00:20:51, [1024/0]
S       2003:172:16:209::/64 [5/0] via ::ac10:ce02, to_FG_B_root, 14:40:14, [1024/0]
```

The IPv6 default route includes the members from the *virtual-wan-link* zone, and the route to 2003:172:16:109::/64 has the single member from the *SASE* zone.

Performance SLA

The following topics provide instructions on configuring performance SLA:

- Link health monitor on page 507
- Factory default health checks on page 509
- Health check options on page 512
- Link monitoring example on page 514
- SLA targets example on page 515
- Passive WAN health measurement on page 517

Link health monitor

Performance SLA link health monitoring measures the health of links that are connected to SD-WAN member interfaces by either sending probing signals through each link to a server, or using session information that is captured on firewall policies (see Passive WAN health measurement on page 517 for information), and measuring the link quality based on latency, jitter, and packet loss. If a link fails all of the health checks, the routes on that link are removed from the SD-WAN link load balancing group, and traffic is routed through other links. When the link is working again the routes are reestablished. This prevents traffic being sent to a broken link and lost.

When an SD-WAN member has multiple health checks configured, all of the checks must fail for the routes on that link to be removed from the SD-WAN link load balancing group.

Two health check servers can be configured to ensure that, if there is a connectivity issue, the interface is at fault and not the server. A server can only be used in one health check.

The FortiGate uses the first server configured in the health check server list to perform the health check. If the first server is unavailable, then the second server is used. The second server continues to be used until it becomes unavailable, and then the FortiGate returns to the first server, if it is available. If both servers are unavailable, then the health check fails.

You can configure the protocol that is used for status checks, including: Ping, HTTP, DNS, TCP echo, UDP echo, two-way active measurement protocol (TWAMP), TCP connect, and FTP. In the GUI, only Ping, HTTP, and DNS are available.

You can view link quality measurements by going to *Network > SD-WAN* and selecting the *Performance SLAs* tab. The table shows the default health checks, the health checks that you configured, and information about each health check. The values shown in the *Packet Loss*, *Latency*, and *Jitter* columns are for the health check server that the FortiGate is currently using. The green up arrows indicate that the server is responding, and does not indicate if the health checks are being met. See Results on page 492 for more information.

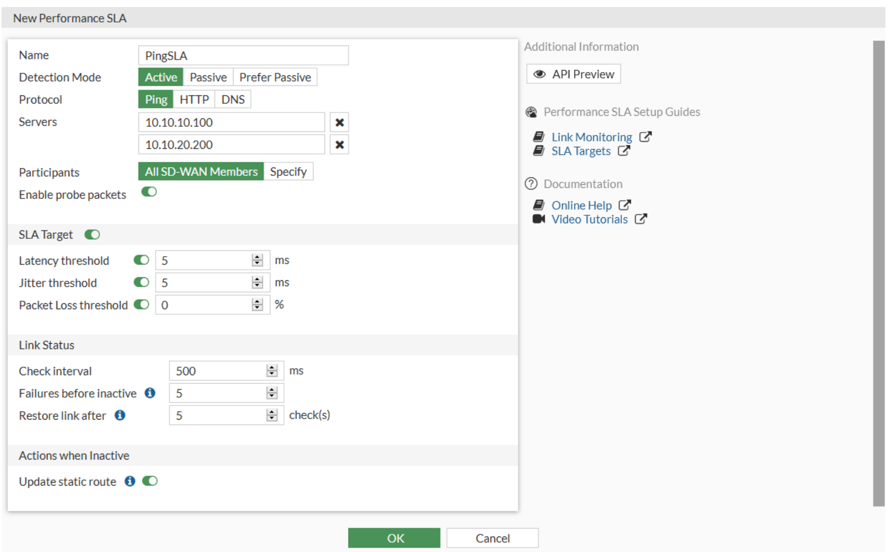
To configure a link health monitor in the GUI:

1. Go to *Network > SD-WAN*, select the *Performance SLAs* tab, and click *Create New*.
2. Set a *Name* for the SLA.
3. Set the *Protocol* that you need to use for status checks: *Ping*, *HTTP*, or *DNS*.
4. Set *Server* to the IP addresses of up to two servers that all of the SD-WAN members in the performance SLA can reach.
5. Set *Participants* to *All SD-WAN Members*, or select *Specify* to choose specific SD-WAN members.
6. Set *Enable probe packets* to enable or disable sending probe packets.
7. Configure *SLA Target*:
   If the health check is used in an SD-WAN rule that uses *Manual* or *Best Quality* strategies, enabling *SLA Target* is optional. If the health check is used in an SD-WAN rule that uses *Lowest Cost (SLA)* or *Maximum Bandwidth (SLA)* strategies, then *SLA Target* is enabled.

When *SLA Target* is enabled, configure the following:

- *Latency threshold*: Calculated based on last 30 probes (default = 5ms).
- *Jitter threshold*: Calculated based on last 30 probes (default = 5ms).
- *Packet Loss threshold*: Calculated based on last 100 probes (default = 0%).

8. In the *Link Status* section configure the following:
   - *Check interval*: The interval in which the FortiGate checks the interface, in milliseconds (500 - 3600000, default = 500).
   - *Failures before inactive*: The number of failed status checks before the interface shows as inactive (1 - 3600, default =5). This setting helps prevent flapping, where the system continuously transfers traffic back and forth between links
   - *Restore link after*: The number of successful status checks before the interface shows as active (1 - 3600, default = 5). This setting helps prevent flapping, where the system continuously transfers traffic back and forth between links
9. In the *Actions when Inactive* section, enable *Update static route* to disable static routes for inactive interfaces and restore routes when interfaces recover.



10. Click *OK*.

To configure a link health monitor in the CLI:

```
config system sdwan
    config health-check
        edit "PingSLA"
            set addr-mode {ipv4 | ipv6}
            set server <server1_IP_address> <server2_IP_address>
            set detect-mode {active | passive | prefer-passive}
            set protocol {ping | tcp-echo | udp-echo | http | twamp | dns | tcp-connect |
ftp}
            set ha-priority <integer>
            set probe-timeout <integer>
            set probe-count <integer>
            set probe-packets {enable | disable}
            set interval <integer>
            set failtime <integer>
            set recoverytime <integer>
            set diffservcode <binary>
```

```
            set update-static-route {enable | disable}
            set update-cascade-interface {enable | disable}
            set sla-fail-log-period <integer>
            set sla-pass-log-period <integer>
            set threshold-warning-packetloss <integer>
            set threshold-alert-packetloss <integer>
            set threshold-warning-latency <integer>
            set threshold-alert-latency <integer>
            set threshold-warning-jitter <integer>
            set threshold-alert-jitter <integer>
            set members <member_number> ... <member_number>
            config sla
                edit 1
                    set link-cost-factor {latency jitter packet-loss}
                    set latency-threshold <integer>
                    set jitter-threshold <integer>
                    set packetloss-threshold <integer>
                next
            end
        next
    end
end
```

Additional settings are available for some of the protocols:

Protocol	Additional options

http	`port <port_number>`
	`http-get <url>`
	`http-match <response_string>`

twamp	`port <port_number>`
	`security mode {none | authentication}`
	`password <password>`
	`packet-size <size>`

ftp	`ftp {passive | port}` `ftp-file <path>`

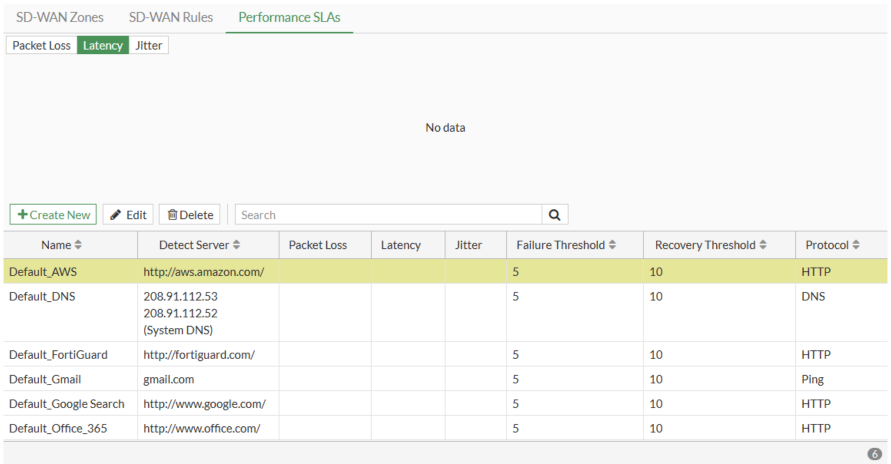
For more examples see Health check options on page 512.

Factory default health checks

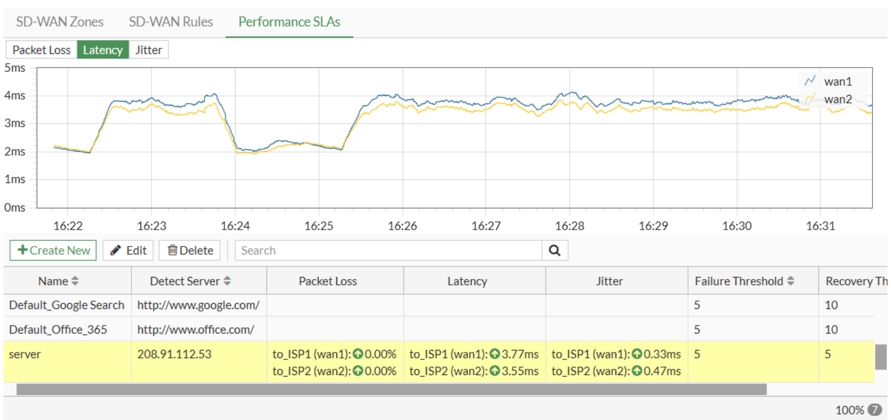
There are six predefined performance SLA profiles for newly created VDOMs or factory reset FortiGate devices:

- AWS
- System DNS
- FortiGuard
- Gmail
- Google Search
- Office 365

You can view and configure the SLA profiles by going to *Network > SD-WAN* and selecting the *Performance SLAs* tab.



After configuring a health check, you will be able to view packet loss, latency, and jitter data for the SLA profiles. If a value is colored red, it means that it failed to meet the SLA requirements.



To configure the performance SLA profiles in the CLI:

```
config system sdwan
    config health-check
        edit "Default_DNS"
            set system-dns enable
            set interval 1000
            set probe-timeout 1000
            set recoverytime 10
            config sla
                edit 1
                    set latency-threshold 250
                    set jitter-threshold 50
                    set packetloss-threshold 5
                next
            end
        next
        edit "Default_Office_365"
            set server "www.office.com"
            set protocol http
            set interval 1000
            set probe-timeout 1000
```

```
        set recoverytime 10
        config sla
            edit 1
                set latency-threshold 250
                set jitter-threshold 50
                set packetloss-threshold 5
            next
        end
    next
    edit "Default_Gmail"
        set server "gmail.com"
        set interval 1000
        set probe-timeout 1000
        set recoverytime 10
        config sla
            edit 1
                set latency-threshold 250
                set jitter-threshold 50
                set packetloss-threshold 2
            next
        end
    next
    edit "Default_AWS"
        set server "aws.amazon.com"
        set protocol http
        set interval 1000
        set probe-timeout 1000
        set recoverytime 10
        config sla
            edit 1
                set latency-threshold 250
                set jitter-threshold 50
                set packetloss-threshold 5
            next
        end
    next
    edit "Default_Google Search"
        set server "www.google.com"
        set protocol http
        set interval 1000
        set probe-timeout 1000
        set recoverytime 10
        config sla
            edit 1
                set latency-threshold 250
                set jitter-threshold 50
                set packetloss-threshold 5
            next
        end
    next
    edit "Default_FortiGuard"
        set server "fortiguard.com"
        set protocol http
        set interval 1000
        set probe-timeout 1000
        set recoverytime 10
```

```
            config sla
                edit 1
                    set latency-threshold 250
                    set jitter-threshold 50
                    set packetloss-threshold 5
                next
            end
        next
    end
end
```

Health check options

Health checks include several protocols and protocol specific options.

The health check protocol options include:

ping	Use PING to test the link with the server.
tcp-echo	Use TCP echo to test the link with the server.
udp-echo	Use UDP echo to test the link with the server.
http	Use HTTP-GET to test the link with the server.
twamp	Use TWAMP to test the link with the server.
dns	Use DNS query to test the link with the server. The FortiGate sends a DNS query for an A Record and the response matches the expected IP address.
tcp-connect	Use a full TCP connection to test the link with the server. The method to measure the quality of the TCP connection can be: • `half-open`: FortiGate sends SYN and gets SYN-ACK. The latency is based on the round trip between SYN and SYN-ACK (default). • `half-close`: FortiGate sends FIN and gets FIN-ACK. The latency is based on the round trip between FIN and FIN-ACK.
ftp	Use FTP to test the link with the server. The FTP mode can be: • `passive`: The FTP health-check initiates and establishes the data connection (default). • `port`: The FTP server initiates and establishes the data connection.

SD-WAN health checks can generate traffic that becomes quite high as deployments grow. Please take this into consideration when setting DoS policy thresholds. For details on setting DoS policy thresholds, refer to DoS policy on page 790.

To use UDP-echo and TCP-echo as health checks:

```
config system sdwan
    set status enable
    config health-check
```

```
        edit "h4_udp1"
            set protocol udp-echo
            set port 7
            set server <server>
        next
        edit "h4_tcp1"
            set protocol tcp-echo
            set port 7
            set server <server>
        next
        edit "h6_udp1"
            set addr-mode ipv6
            set server "2032::12"
            set protocol udp-echo
            set port 7
        next
    end
end
```

To use DNS as a health check, and define the IP address that the response must match:

```
config system sdwan
    set status enable
    config health-check
        edit "h4_dns1"
            set protocol dns
            set dns-request-domain "ip41.forti2.com"
            set dns-match-ip 1.1.1.1
        next
        edit "h6_dns1"
            set addr-mode ipv6
            set server "2000::15.1.1.4"
            set protocol dns
            set port 53
            set dns-request-domain "ip61.xxx.com"
        next
    end
end
```

To use TCP Open (SYN/SYN-ACK) and TCP Close (FIN/FIN-ACK) to verify connections:

```
config system sdwan
    set status enable
    config health-check
        edit "h4_tcpconnect1"
            set protocol tcp-connect
            set port 443
            set quality-measured-method {half-open | half-close}
            set server <server>
        next
        edit "h6_tcpconnect1"
            set addr-mode ipv6
            set server "2032::13"
            set protocol tcp-connect
            set port 444
            set quality-measured-method {half-open | half-close}
```

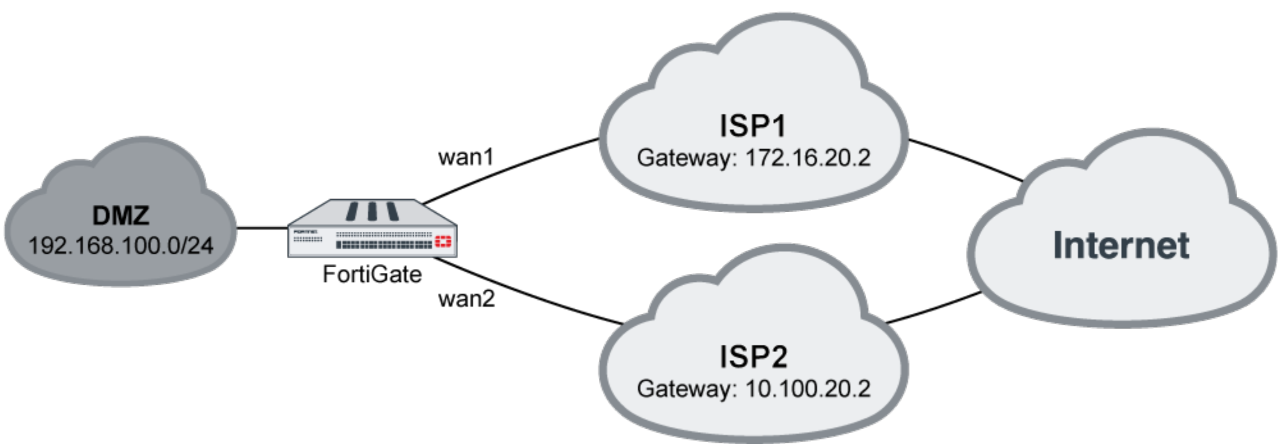
```
        next
    end
end
```

To use active or passive mode FTP to verify connections:

```
config system sdwan
    set status enable
    config health-check
        edit "h4_ftp1"
            set protocol ftp
            set port 21
            set user "root"
            set password ***********
            set ftp-mode {passive | port}
            set ftp-file "1.txt"
            set server <server>
        next
        edit "h6_ftp1"
            set addr-mode ipv6
            set server "2032::11"
            set protocol ftp
            set port 21
            set user "root"
            set password ***********
            set ftp-mode {passive | port}
            set ftp-file "2.txt"
        next
    end
end
```

Link monitoring example

Performance SLA link monitoring measures the health of links that are connected to SD-WAN member interfaces by sending probing signals through each link to a server and measuring the link quality based on latency, jitter, and packet loss. If a link is broken, the routes on that link are removed, and traffic is routed through other links. When the link is working again, the routes are reenabbled. This prevents traffic being sent to a broken link and lost.



In this example:

- Interfaces wan1 and wan2 connect to the internet through separate ISPs
- The detection server IP address is 208.91.114.182

A performance SLA is created so that, if one link fails, its routes are removed and traffic is detoured to the other link.

To configure a Performance SLA using the GUI:

1. On the FortiGate, add wan1 and wan2 as SD-WAN members, then add a policy and static route. See SD-WAN quick start on page 487 for details.
2. Go to *Network > SD-WAN*, select the *Performance SLAs* tab, and click *Create New*.
3. Enter a name for the SLA and select a protocol.
4. In the *Server* field, enter the detection server IP address (208.91.114.182 in this example).
5. In the *Participants* field, select both wan1 and wan2.
6. Configured the remaining settings as needed, then click *OK*.

To configure a Performance SLA using the CLI:

```
config system sdwan
    config health-check
        edit "server"
            set server "208.91.114.182"
            set update-static-route enable
            set members 1 2
        next
    end
end
```

To diagnose the Performance SLA status:

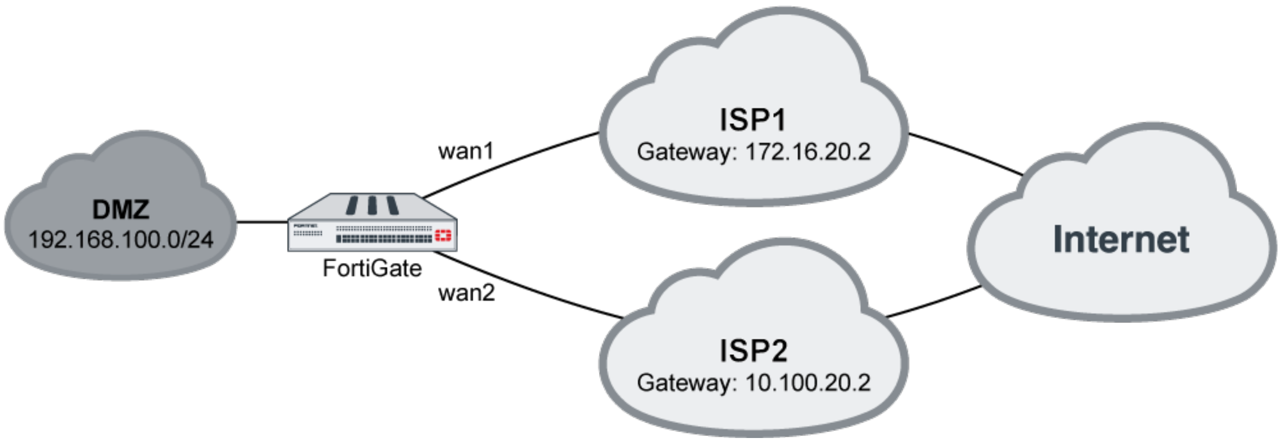
```
# diagnose sys sdwan health-check
Health Check(server):
Seq(1): state(alive), packet-loss(0.000%) latency(15.247), jitter(5.231) sla_map=0x0
Seq(2): state(alive), packet-loss(0.000%) latency(13.621), jitter(6.905) sla_map=0x0
```

SLA targets example

SLA targets are a set of constraints that are used in SD-WAN rules to control the paths that traffic take.

The available constraints are:

- *Latency threshold*: Latency for SLA to make decision, in milliseconds (0 - 10000000, default = 5).
- *Jitter threshold*: Jitter for SLA to make decision, in milliseconds (0 - 10000000, default = 5).
- *Packet loss threshold*: Packet loss for SLA to make decision, in percentage (0 - 100, default = 0).



To configure Performance SLA targets using the GUI:

1. On the FortiGate, add wan1 and wan2 as SD-WAN members, then add a policy and static route. See SD-WAN quick start on page 487 for details.
2. Go to *Network > SD-WAN* and select the *Performance SLAs* tab.
3. Create a new Performance SLA or edit an existing one. See Link monitoring example on page 514.
4. Enable *SLA Targets*and configure the constraints. To add multiple SLA targets, use the CLI.
5. Configured the remaining settings as needed, then click *OK*.

To configure Performance SLA targets using the CLI:

```
config system sdwan
    config health-check
        edit "server"
            set server "208.91.114.182"
            set members 1 2
            config sla
                edit 1
                    set link-cost-factor latency jitter packet-loss
                    set latency-threshold 10
                    set jitter-threshold 10
                    set packetloss-threshold 1
                next
                edit 2
                    set link-cost-factor latency packet-loss
                    set latency-threshold 15
                    set packetloss-threshold 2
                next
            end
        next
    end
end
```

The `link-cost-factor` variable is used to select which constraints are enabled.

Passive WAN health measurement

SD-WAN passive WAN health measurement determines the health check measurements using session information that is captured on firewall policies that have *Passive Health Check* (`passive-wan-health-measurement`) enabled. Passive measurements analyze session information that is gathered from various TCP sessions to determine the jitter, latency, and packet loss.

Using passive WAN health measurement reduces the amount of configuration required and decreases the traffic that is produced by health check monitor probes doing active measurements. Passive WAN health measurement analyzes real-life traffic; active WAN health measurement using a detection server might not reflect the real-life traffic.

By default, active WAN health measurement is enabled when a new health check is created. It can be changed to passive or prefer passive:

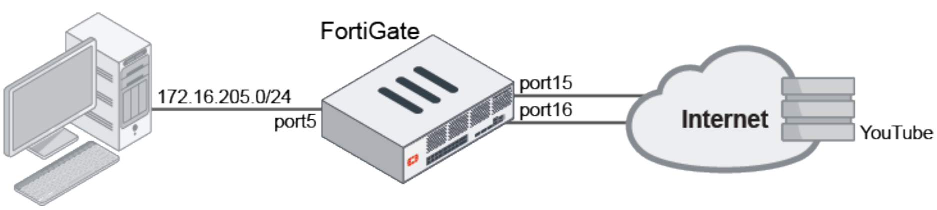
passive	Health is measured using traffic, without probes. No link health monitor needs to be configured.
prefer-passive	Health is measured using traffic when there is traffic, and using probes when there is no traffic. A link health monitor must be configured, see Link health monitor for details.

When `passive-wan-health-measurement` is enabled, `auto-asic-offload` will be disabled.

Example

In this example, the FortiGate is configured to load-balance between two WAN interfaces, port15 and port16. A health check is configured in passive mode, and SLA thresholds are set. Passive WAN health measurement is enabled on the SD-WAN policy.

Measurements are taken from YouTube traffic generated by the PC. When latency is introduced to the traffic on port15, the passive health check trigger threshold is exceeded and traffic is rerouted to port16.

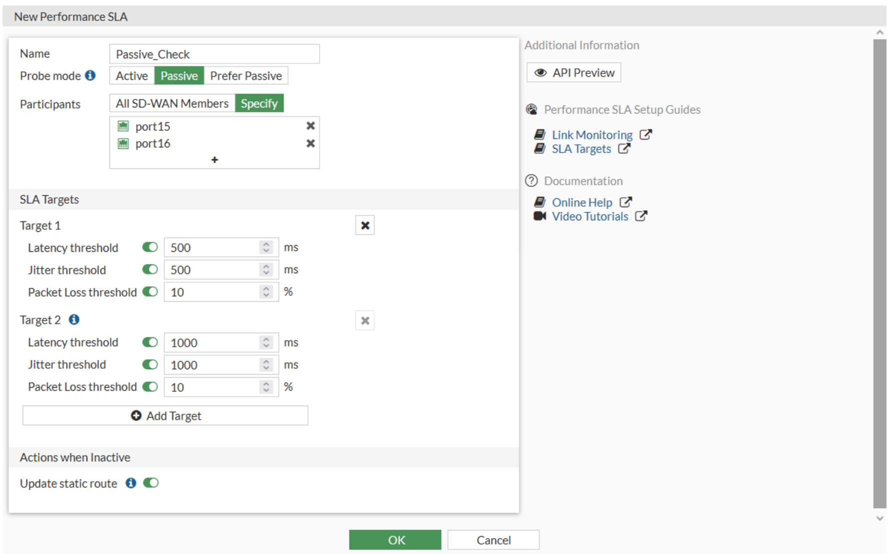


To configure the SD-WAN in the GUI:

1. Create the SD-WAN zone:
    a. Go to *Network > SD-WAN* and select the *SD-WAN Zones* tab.
    b. Click *Create New > SD-WAN Zone*.
    c. Enter a name for the zone, such as *SD-WAN*.
    d. Click *OK*.
2. Create the SD-WAN members:

   a. Go to *Network > SD-WAN* and select the *SD-WAN Zones* tab.
   b. Click *Create New > SD-WAN Member*.
   c. Set *Interface* to *port15*, *SD-WAN Zone* to *SD-WAN*, and *Gateway* set to *172.16.209.2*.
   d. Click *OK*.
   e. Click *Create New > SD-WAN Member* again.
   f. Set *Interface* to *port16*, *SD-WAN Zone* to *SD-WAN*, and *Gateway* set to *172.16.210.2*.
   g. Click *OK*.

3. Create a performance SLA:
   a. Go to *Network > SD-WAN* and select the *Performance SLAs* tab.
   b. Edit an existing health check, or create a new one.
   c. Set *Probe mode* to *Passive*.
   d. Set *Participants* to *Specify* and add *port15* and *port16*.
   e. Configure two SLA targets. Note that the second SLA target must be configured in the CLI.



   f. Configure the remaining settings as needed.
   g. Click *OK*.
   The SLA list shows the probe mode in the *Detect Server* column, if the probe mode is passive or prefer passive.

Probe packets can only be disabled in the CLI and when the probe mode is not passive.

---

4. Create SD-WAN rules:
   a. Go to *Network > SD-WAN*, select the *SD-WAN Rules* tab, and click *Create New*.
   b. Configure the first rule:

Name	Background_Traffic
Source address	172.16.205.0

Application	Click in the field, and in the *Select Entries* pane search for *YouTube* and select all of the entries
Strategy	Maximize Bandwidth (SLA)
Interface preference	port15 and port16
Required SLA target	Passive_Check#2

c. Click *OK*.
d. Click *Create New* again and configure the second rule:

Name	Foreground_Traffic
Source address	172.16.205.0
Address	all
Protocol number	Specify - 1
Strategy	Lowest Cost (SLA)
Interface preference	port15 and port16
Required SLA target	Passive_Check#1

e. Click *OK*.

To configure the firewall policy in the GUI:

1. Go to *Policy & Objects > Firewall Policy* and click *Create New*.
2. Configure the policy:

Name	SD-WAN-HC-policy
Incoming Interface	port5
Outgoing Interface	SD-WAN
Source	all
Destination	all
Schedule	always
Service	ALL
Action	ACCEPT
Passive Health Check	Enabled Passive health check can only be enabled in a policy when the outgoing interface is an SD-WAN zone.

3. Click *OK*.

To configure the SD-WAN in the CLI:

```
config system sdwan
    set status enable
```

```
    config zone
        edit "SD-WAN"
        next
    end
    config members
        edit 1
            set zone "SD-WAN"
            set interface "port15"
            set gateway 172.16.209.2
        next
        edit 2
            set zone "SD-WAN"
            set interface "port16"
            set gateway 172.16.210.2
        next
    end
    config health-check
        edit "Passive_Check"
            set detect-mode passive
            set members 1 2
            config sla
                edit 1
                    set latency-threshold 500
                    set jitter-threshold 500
                    set packetloss-threshold 10
                next
                edit 2
                    set latency-threshold 1000
                    set jitter-threshold 1000
                    set packetloss-threshold 10
                next
            end
        next
    end
    config service
        edit 1
            set name "Background_Traffic"
            set mode load-balance
            set src "172.16.205.0"
            set internet-service enable
            set internet-service-app-ctrl 31077 33321 41598 31076 33104 23397 30201 16420
17396 38569 25564
            config sla
                edit "Passive_Check"
                    set id 2
                next
            end
            set priority-member 1 2
        next
        edit 2
            set name "Foreground_Traffic"
            set mode sla
            set src "172.16.205.0"
            set protocol 1
            set dst "all"
            config sla
```

```
                edit "Passive_Check"
                    set id 1
                next
            end
            set priority-member 1 2
        next
    end
end
```

To configure the firewall policy in the CLI:

```
config firewall policy
    edit 1
        set name "SD-WAN-HC-policy"
        set srcintf "port5"
        set dstintf "SD-WAN"
        set nat enable
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set passive-wan-health-measurement enable
        set auto-asic-offload disable
    next
end
```

Results

When both links pass the SLA:

```
# diagnose sys link-monitor-passive interface
Interface port16 (28):
  Latency 10.000 Jitter 5.000 Packet_loss 0.000% Last_updated Fri Mar 5 10:09:21 2021

Interface port15 (27):
  Latency 60.000 Jitter 0.000 Packet_loss 0.000% Last_updated Fri Mar 5 10:39:24 2021
```

```
# diagnose sys sdwan health-check
Health Check(Passive_Check):
Seq(1 port15): state(alive), packet-loss(0.000%) latency(60.000), jitter(0.750) sla_map=0x3
Seq(2 port16): state(alive), packet-loss(0.000%) latency(10.000), jitter(5.000) sla_map=0x3
```

```
# diagnose sys sdwan service 2

Service(2): Address Mode(IPV4) flags=0x200
  Gen(1), TOS(0x0/0x0), Protocol(1: 1->65535), Mode(sla), sla-compare-order
  Members(2):
    1: Seq_num(1 port15), alive, sla(0x1), gid(0), cfg_order(0), cost(0), selected
    2: Seq_num(2 port16), alive, sla(0x1), gid(0), cfg_order(1), cost(0), selected
  Src address(1):
        172.16.205.0-172.16.205.255

  Dst address(1):
        8.8.8.8-8.8.8.8
```

When the latency is increased to 610ms on port15, the SLA is broken and pings are sent on port16:

```
# diagnose sys sdwan health-check
Health Check(Passive_Check):
Seq(1 port15): state(alive), packet-loss(0.000%) latency(610.000), jitter(2.500) sla_map=0x3
Seq(2 port16): state(alive), packet-loss(0.000%) latency(50.000), jitter(21.000) sla_map=0x3
```

```
# diagnose sys sdwan service 2

Service(2): Address Mode(IPV4) flags=0x200
  Gen(6), TOS(0x0/0x0), Protocol(1: 1->65535), Mode(sla), sla-compare-order
  Members(2):
    1: Seq_num(2 port16), alive, sla(0x1), gid(1), cfg_order(1), cost(0), selected
    2: Seq_num(1 port15), alive, sla(0x0), gid(2), cfg_order(0), cost(0), selected
  Src address(1):
        172.16.205.0-172.16.205.255

  Dst address(1):
        8.8.8.8-8.8.8.8
```

Passive health-check measurement by internet service and application

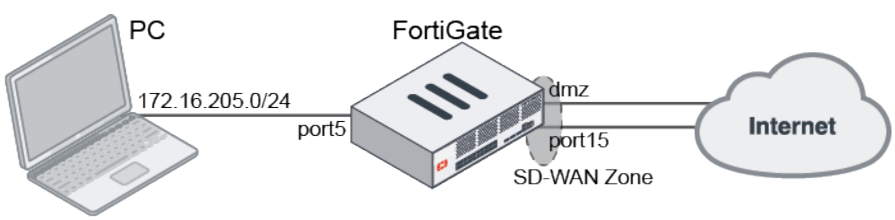
Passive health measurement supports passive detection for each internet service and application.

If internet services or applications are defined in an SD-WAN rule with passive health check, SLA information for each service or application will be differentiated and collected. SLA metrics (latency, jitter, and packet loss) on each SD-WAN member in the rule are then calculated based on the relevant internet service's or application's SLA information.

In this example, three SD-WAN rules are created:

- Rule 1: Best quality (latency) using passive SLA for the internet services Alibaba and Amazon.
- Rule 2: Best quality (latency) using passive SLA for the applications Netflix and YouTube.
- Rule 3: Best quality (latency) using passive SLA for all other traffic.

After passive application measurement is enabled for rules one and two, the SLA metric of rule one is the average latency of the internet services Alibaba and Amazon, and the SLA metric of rule two is the average latency of the applications Netflix and YouTube.



To configure the SD-WAN:

1. Configure the SD-WAN members:

```
config system sdwan
    set status enable
    config zone
        edit "virtual-wan-link"
        next
```

```
    end
    config members
        edit 1
            set interface "dmz"
            set gateway 172.16.208.2
        next
        edit 2
            set interface "port15"
            set gateway 172.16.209.2
        next
    end
end
```

2. Configure the passive mode health check:

```
config health-check
    edit "Passive_HC"
        set detect-mode passive
        set members 1 2
    next
end
```

3. Configure SD-WAN service rules:

```
config service
    edit 1
        set name "1"
        set mode priority
        set src "172.16.205.0"
        set internet-service enable
        set internet-service-name "Alibaba-Web" "Amazon-Web"
        set health-check "Passive_HC"
        set priority-members 1 2
        set passive-measurement enable    //Enable "passive application measurement", it
is a new command which is introduced in this project.
    next
    edit 2
        set name "2"
        set mode priority
        set src "172.16.205.0"
        set internet-service enable
        set internet-service-app-ctrl 18155 31077
        set health-check "Passive_HC"
        set priority-members 1 2
        set passive-measurement enable    ////Enable "passive application measurement"
    next
    edit 3
        set name "3"
        set mode priority
        set dst "all"
        set src "172.16.205.0"
        set health-check "Passive_HC"
        set priority-members 1 2
    next
end
```

4. Configure SD-WAN routes:

```
config router static
    edit 1
        set distance 1
        set sdwan-zone "virtual-wan-link"
    next
end
```

5. Configure the firewall policy with passive WAN health measurement enabled:

```
config firewall policy
    edit 1
        set uuid 972345c6-1595-51ec-66c5-d705d266f712
        set srcintf "port5"
        set dstintf "virtual-wan-link"
        set action accept
        set srcaddr "172.16.205.0"
        set dstaddr "all"
        set schedule "always"
        set service "ALL"
        set passive-wan-health-measurement enable
        set utm-status enable
        set ssl-ssh-profile "certificate-inspection"
        set application-list "g-default"
        set auto-asic-offload disable
    next
end
```

To verify the results:

1. On the PC, open the browser and visit the internet services and applications.
2. On the FortiGate, check the collected SLA information to confirm that each server or application on the SD-WAN members was measured individually:

```
# diagnose sys link-monitor-passive interface

Interface dmz (5):
       Default(0x00000000): latency=3080.0  11:57:54, jitter=5.0     11:58:08,
pktloss=0.0  % NA
   Alibaba-Web(0x00690001): latency=30.0    11:30:06, jitter=25.0    11:29:13,
pktloss=0.0  % NA
       YouTube(0x00007965): latency=100.0   12:00:35, jitter=2.5     12:00:30,
pktloss=0.0  % NA
       Netflix(0x000046eb): latency=10.0    11:31:24, jitter=10.0    11:30:30,
pktloss=0.0  % NA
    Amazon-Web(0x00060001): latency=80.0    11:31:52, jitter=35.0    11:32:07,
pktloss=0.0  % NA

Interface port15 (27):
       Default(0x00000000): latency=100.0   12:00:42, jitter=0.0     12:00:42,
pktloss=0.0  % NA
    Amazon-Web(0x00060001): latency=30.0    11:56:05, jitter=0.0     11:55:21,
pktloss=0.0  % NA
   Alibaba-Web(0x00690001): latency=0.0     11:26:08, jitter=35.0    11:27:08,
pktloss=0.0  % NA
       YouTube(0x00007965): latency=100.0   11:33:34, jitter=0.0     11:33:50,
pktloss=0.0  % NA
```

```
       Netflix(0x000046eb): latency=0.0     11:26:29, jitter=0.0     11:29:03,
pktloss=0.0  % NA
```

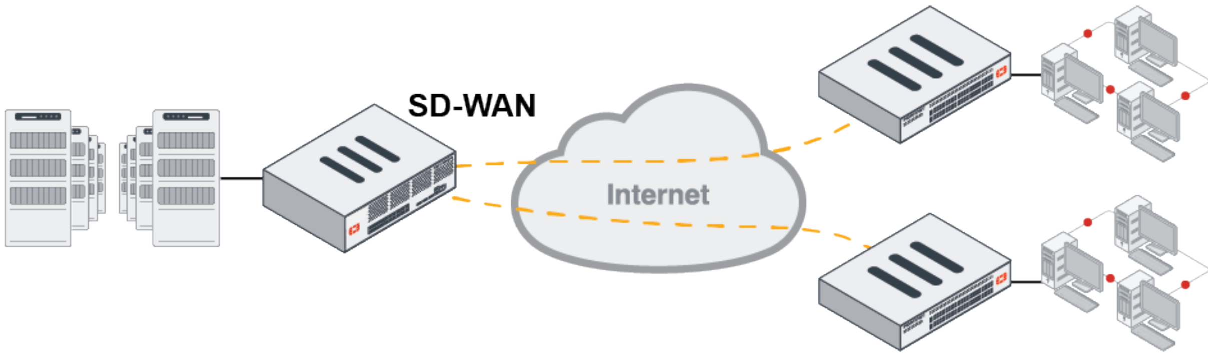
3. Verify that the SLA metrics on the members are calculated as expected:

```
# diagnose sys sdwan service

Service(1): Address Mode(IPV4) flags=0x600 use-shortcut-sla
  Gen(1), TOS(0x0/0x0), Protocol(0: 1->65535), Mode(priority), link-cost-factor
(latency), link-cost-threshold(10), heath-check(Passive_HC)
  Members(2):
    1: Seq_num(2 port15), alive, latency: 15.000, selected           // Average latency
of "Alibaba-Web" and "Amazon-Web" on port15:     15.000 = (0.0+30.0)/2
    2: Seq_num(1 dmz), alive, latency: 55.000, selected              // Average latency
of "Alibaba-Web" and "Amazon-Web" on dmz:         55.000 = (30.0+80.0)/2
  Internet Service(2): Alibaba-Web(6881281,0,0,0) Amazon-Web(393217,0,0,0)
  Src address(1):
        172.16.205.0-172.16.205.255

Service(2): Address Mode(IPV4) flags=0x600 use-shortcut-sla
  Gen(2), TOS(0x0/0x0), Protocol(0: 1->65535), Mode(priority), link-cost-factor
(latency), link-cost-threshold(10), heath-check(Passive_HC)
  Members(2):
    1: Seq_num(1 dmz), alive, latency: 55.000, selected               // Average latency
of "Netflix" and "YouTube" on dmz:        55.000 = (10.0+100.0)/2
    2: Seq_num(2 port15), alive, latency: 50.000, selected           // Average latency
of "Netflix" and "YouTube" on port15:     50.000 = (0.0+100.0)/2
  Internet Service(2): Netflix(4294837427,0,0,0 18155) YouTube(4294838283,0,0,0 31077)
  Src address(1):
        172.16.205.0-172.16.205.255

Service(3): Address Mode(IPV4) flags=0x200 use-shortcut-sla
  Gen(9), TOS(0x0/0x0), Protocol(0: 1->65535), Mode(priority), link-cost-factor
(latency), link-cost-threshold(10), heath-check(Passive_HC)
  Members(2):
    1: Seq_num(2 port15), alive, latency: 46.000, selected          // Average latency
of all TCP traffic on port15:       46   = (100.0+30.0+0.0+100.0+0.0)/5
    2: Seq_num(1 dmz), alive, latency: 660.000, selected           // Average latency of
all TCP traffic on dmz:           660 = (3080.0+30.0+100.0+10.0+80.0)/5
  Src address(1):
        172.16.205.0-172.16.205.255

  Dst address(1):
        0.0.0.0-255.255.255.255
```

Health check packet DSCP marker support

SD-WAN health check probe packets support Differentiated Services Code Point (DSCP) markers for accurate evaluation of the link performance for high priority applications by upstream devices.

When the SD-WAN health check packet is sent out, the DSCP can be set with a CLI command.



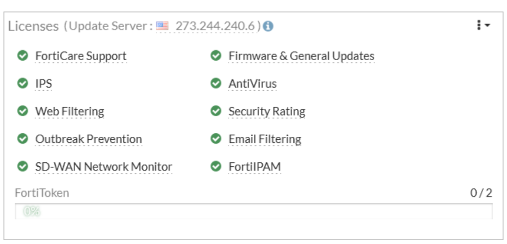
To mark health-check packets with DSCP:

```
config system sdwan
   config health-check
      edit <name>
         set diffservcode <6 bits binary, range 000000-111111>
      next
   end
end
```

Manual interface speedtest

An interface speedtest can be manually performed on WAN interfaces in the GUI. The results of the test can be added to the interface's *Estimated bandwidth*. The estimated upstream and downstream bandwidths can be used in SD-WAN service rules to determine the best link to use when either Maximize Bandwidth or Best Quality strategies are selected.

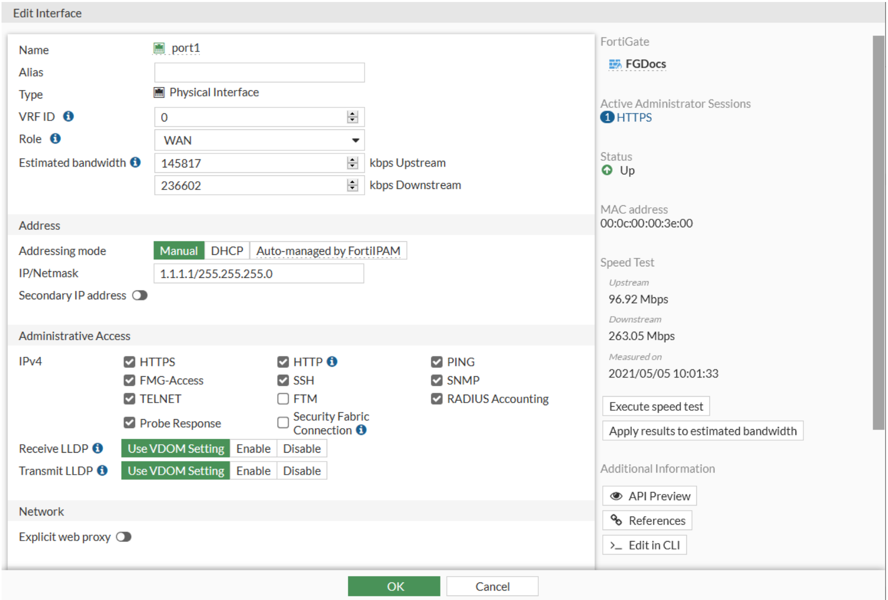
An SD-WAN Network Monitor license is required to use the speedtest. The *License* widget and the *System* > *FortiGuard* page show the license status.



To run an interface speedtest in the GUI:

1. Go to *Network* > *Interfaces*.
2. Edit a WAN interface. The interfaces can be grouped by role using the grouping dropdown on the right side of the toolbar.

3. Click *Execute speed test* in the right pane.



4. When the test completes, click *OK* in the *Confirm* pane to apply the results to the estimated bandwidth.
   The results can also be applied later by clicking *Apply results to estimated bandwidth*.
   The speedtest results are used to populate the *Estimated bandwidth* fields.
5. Click *OK*.

The FortiGate must be connected to FortiGuard, and able to reach either the AWS or Google speedtest servers.

Scheduled interface speedtest

The SD-WAN Network Monitor service supports running a speed test based on a schedule. The test results are automatically updated in the interface `measured-upstream-bandwidth` and `measured-downstream-bandwidth` fields. These fields do not impact the interface inbound bandwidth, outbound bandwidth, estimated upstream bandwidth, or estimated downstream bandwidth settings.

An SD-WAN Network Monitor license is required to use the speedtest. The *License* widget and the *System* > *FortiGuard* page show the license status.

When the scheduled speed tests run, it is possible to temporarily bypass the bandwidth limits set on the interface and configure custom maximum or minimum bandwidth limits. These configurations are optional.

```
config system speed-test-schedule
    edit <interface>
        set schedules <schedule> ...
        set update-inbandwidth enable {enable | disable}
        set update-outbandwidth enable {enable | disable}
        set update-inbandwidth-maximum <integer>
        set update-inbandwidth-minimum <integer>
        set update-outbandwidth-maximum <integer>
        set update-outbandwidth-minimum <integer>
```

```
    next
end
```

`update-inbandwidth enable {enable | disable}`	Enable/disable bypassing the interface's inbound bandwidth setting.
`update-outbandwidth enable {enable | disable}`	Enable/disable bypassing the interface's outbound bandwidth setting.
`update-inbandwidth-maximum <integer>`	Maximum downloading bandwidth to be used in a speed test, in Kbps (0 - 16776000).
`update-inbandwidth-minimum <integer>`	Minimum downloading bandwidth to be considered effective, in Kbps (0 - 16776000).
`update-outbandwidth-maximum <integer>`	Maximum uploading bandwidth to be used in a speed test, in Kbps (0 - 16776000).
`update-outbandwidth-minimum <integer>`	Minimum uploading bandwidth to be considered effective, in Kbps (0 - 16776000).

In the following example, a speed test is scheduled on port1 at 10:00 AM, and another one at 14:00 PM.

To run a speed test based on a schedule:

1. Configure the recurring schedules:

```
config firewall schedule recurring
    edit "10"
        set start 10:00
        set end 12:00
        set day monday tuesday wednesday thursday friday
    next
    edit "14"
        set start 14:00
        set end 16:00
        set day monday tuesday wednesday thursday friday
    next
end
```

2. Configure the speed test schedule:

```
config system speed-test-schedule
    edit "port1"
        set schedules "10" "14"
        set update-inbandwidth enable
        set update-outbandwidth enable
        set update-inbandwidth-maximum 60000
        set update-inbandwidth-minimum 10000
        set update-outbandwidth-maximum 50000
        set update-outbandwidth-minimum 10000
    next
end
```

3. View the speed test results:

```
config system interface
    edit port1
```

```
            get | grep measure
                measured-upstream-bandwidth: 23691
                measured-downstream-bandwidth: 48862
                bandwidth-measure-time:  Wed Jan 27 14:00:39 2021
        next
    end
```

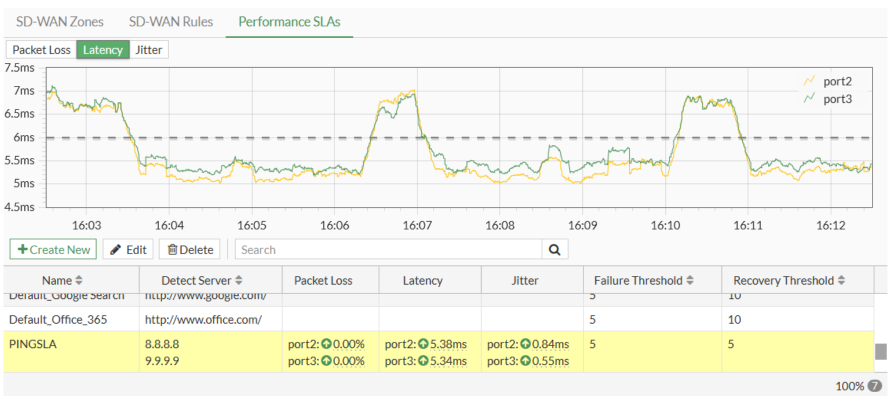
Monitor performance SLA

SD-WAN diagnostics can be used to help maintain your SD-WAN solution

Monitoring SD-WAN link quality status

Link quality plays a significant role in link selection for SD-WAN. Investigate any prolonged issues with packet loss, latency, or jitter to ensure that your network does not experience degraded performance or an outage.

You can monitor the link quality status of SD-WAN interface members by going to *Network > SD-WAN* and selecting the *Performance SLAs* tab.



The live charts show the packet loss, latency, or jitter for the selected health check. Hover the cursor over a line in the chart to see the specific value for that interface at that specific time.

The table shows information about each health check, including the configured servers, link quality data, and thresholds. The colored arrow indicates the status of the interface when the last status check was performed: green means that the interface was active, and red means that the interface was inactive. Hover the cursor over the arrow for additional information.

Monitoring system event logs

The features adds an SD-WAN daemon function to keep a short, 10 minute history of SLA that can be viewed in the CLI.

Performance SLA results related to interface selection, session failover, and other information, can be logged. These logs can then be used for long-term monitoring of traffic issues at remote sites, and for reports and views in FortiAnalyzer.

The time intervals that Performance SLA fail and pass logs are generated in can be configured.

To configure the fail and pass logs' generation time interval:

```
config system sdwan
    config health-check
        edit "PingSLA"
            set sla-fail-log-period 30
            set sla-pass-log-period 60
        next
    end
end
```

To view the 10 minute Performance SLA link status history:

```
FGDocs # diagnose sys sdwan sla-log PingSLA 1
Timestamp: Fri Sep  4 10:32:37 2020, vdom root, health-check PingSLA, interface: wan2,
status: up, latency: 4.455, jitter: 0.430, packet loss: 0.000%.
Timestamp: Fri Sep  4 10:32:37 2020, vdom root, health-check PingSLA, interface: wan2,
status: up, latency: 4.461, jitter: 0.436, packet loss: 0.000%.
Timestamp: Fri Sep  4 10:32:38 2020, vdom root, health-check PingSLA, interface: wan2,
status: up, latency: 4.488, jitter: 0.415, packet loss: 0.000%.
...
Timestamp: Fri Sep  4 10:42:36 2020, vdom root, health-check PingSLA, interface: wan2,
status: up, latency: 6.280, jitter: 0.302, packet loss: 0.000%.
Timestamp: Fri Sep  4 10:42:37 2020, vdom root, health-check PingSLA, interface: wan2,
status: up, latency: 6.261, jitter: 0.257, packet loss: 0.000%.
Timestamp: Fri Sep  4 10:42:37 2020, vdom root, health-check PingSLA, interface: wan2,
status: up, latency: 6.229, jitter: 0.245, packet loss: 0.000%.
```

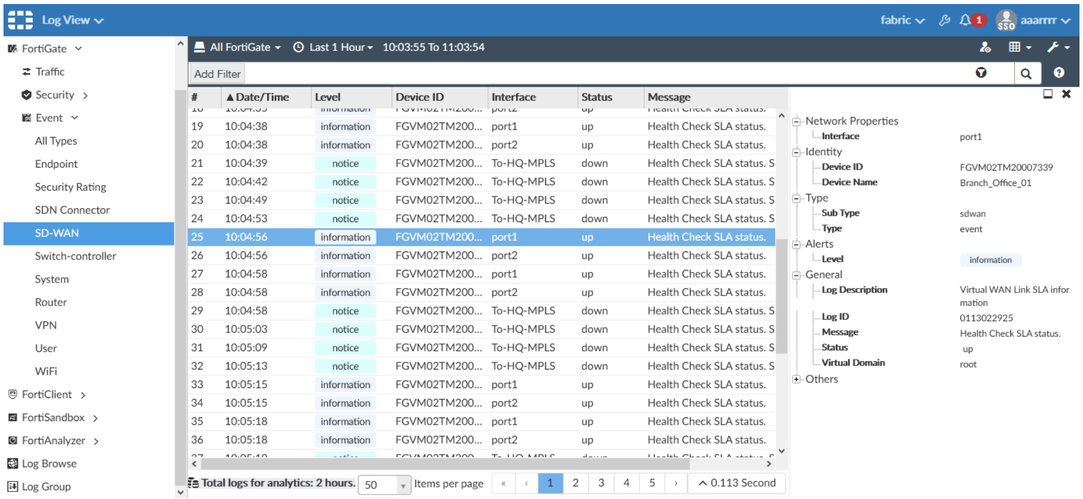
SLA pass logs

The FortiGate generates Performance SLA logs at the specified pass log interval (`sla-pass-log-period`) when SLA passes.

```
date="2021-04-15" time="10:04:56" id=6951431609690095758 bid=52507 dvid=1047
itime=1618506296 euid=3 epid=3 dsteuid=3 dstepid=3 logver=700000066 logid="0113022925"
type="event" subtype="sdwan" level="information" msg="Health Check SLA status."
logdesc="SDWAN SLA information" status="up" interface="port1" eventtime=1618506296222639301
tz="-0700" eventtype="SLA" jitter="0.277" inbandwidthavailable="10.00Gbps"
outbandwidthavailable="10.00Gbps" bibandwidthavailable="20.00Gbps" packetloss="1.000%"
latency="186.071" slamap="0x1" healthcheck="BusinessCritical_CloudApps" slatargetid=1
outbandwidthused="40kbps" inbandwidthused="24kbps" bibandwidthused="64kbps"
devid="FGVM02TM20000000" vd="root" devname="Branch_Office_01" csf="fabric"
```

```
date="2021-04-15" time="10:04:56" id=6951431609690095759 bid=52507 dvid=1047
itime=1618506296 euid=3 epid=3 dsteuid=3 dstepid=3 logver=700000066 logid="0113022925"
type="event" subtype="sdwan" level="information" msg="Health Check SLA status."
logdesc="SDWAN SLA information" status="up" interface="port2" eventtime=1618506296223163068
tz="-0700" eventtype="SLA" jitter="0.204" inbandwidthavailable="10.00Gbps"
outbandwidthavailable="10.00Gbps" bibandwidthavailable="20.00Gbps" packetloss="0.000%"
latency="185.939" slamap="0x1" healthcheck="BusinessCritical_CloudApps" slatargetid=1
outbandwidthused="142kbps" inbandwidthused="23kbps" bibandwidthused="165kbps"
devid="FGVM02TM20000000" vd="root" devname="Branch_Office_01" csf="fabric"
```

In the FortiAnalyzer GUI:



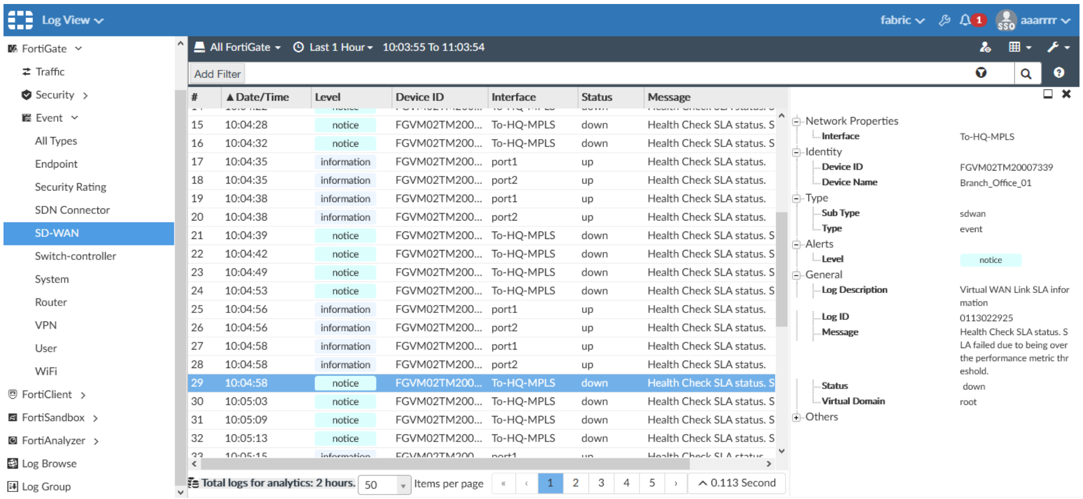
SLA fail logs

The FortiGate generates Performance SLA logs at the specified fail log interval (`sla-fail-log-period`) when SLA fails.

```
date="2021-04-15" time="10:04:59" id=6951431618280030243 bid=52507 dvid=1047
itime=1618506298 euid=3 epid=3 dsteuid=3 dstepid=3 logver=700000066 logid="0113022925"
type="event" subtype="sdwan" level="notice" msg="Health Check SLA status. SLA failed due to
being over the performance metric threshold." logdesc="SDWAN SLA information" status="down"
interface="To-HQ-MPLS" eventtime=1618506299718862835 tz="-0700" eventtype="SLA"
jitter="0.000" inbandwidthavailable="10.00Gbps" outbandwidthavailable="10.00Gbps"
bibandwidthavailable="20.00Gbps" packetloss="100.000%" latency="0.000" slamap="0x0"
healthcheck="BusinessCritical_CloudApps" slatargetid=1 metric="packetloss"
outbandwidthused="0kbps" inbandwidthused="0kbps" bibandwidthused="0kbps"
devid="FGVM02TM20000000" vd="root" devname="Branch_Office_01" csf="fabric"
```

```
date="2021-04-15" time="10:05:03" id=6951431639754866704 bid=52514 dvid=1046
itime=1618506303 euid=3 epid=3 dsteuid=3 dstepid=3 logver=700000066 logid="0113022925"
type="event" subtype="sdwan" level="notice" msg="Health Check SLA status. SLA failed due to
being over the performance metric threshold." logdesc="SDWAN SLA information" status="down"
interface="To-HQ-MPLS" eventtime=1618506304085863643 tz="-0700" eventtype="SLA"
jitter="0.000" inbandwidthavailable="10.00Gbps" outbandwidthavailable="10.00Gbps"
bibandwidthavailable="20.00Gbps" packetloss="100.000%" latency="0.000" slamap="0x0"
healthcheck="BusinessCritical_CloudApps" slatargetid=1 metric="packetloss"
outbandwidthused="6kbps" inbandwidthused="3kbps" bibandwidthused="9kbps"
devid="FGVM02TM20000000" vd="root" devname="Branch_Office_02" csf="fabric"
```

In the FortiAnalyzer GUI:



SLA monitoring using the REST API

SLA log information and interface SLA information can be monitored using the REST API. This feature is also be used by FortiManager as part of its detailed SLA monitoring and drill-down features.

Interface log command example:

```
https://172.172.172.9/api/v2/monitor/virtual-wan/interface-log
{
  "http_method":"GET",
  "results":[
    {
      "interface":"port13",
      "logs":[
        {
          "timestamp":1547087168,
          "tx_bandwidth":3447,
          "rx_bandwidth":3457,
          "bi_bandwidth":6904,
          "tx_bytes":748875,
          "rx_bytes":708799,
          "egress_queue":[
          ]
        },
        {
          "timestamp":1547087178,
          "tx_bandwidth":3364,
          "rx_bandwidth":3400,
          "bi_bandwidth":6764,
          "tx_bytes":753789,
          "rx_bytes":712835,
          "egress_queue":[
          ]
        },
....
....
```

SLA log command example:

```
https://172.172.172.9/api/v2/monitor/virtual-wan/sla-log
{
  "http_method":"GET",
  "results":[
    {
      "name":"ping",
      "interface":"spoke11-p1",
      "logs":[
        {
          "timestamp":1614813142,
          "link":"up",
          "latency":0.13763333857059479,
          "jitter":0.02996666356921196,
          "packetloss":0
        },

      "child_intfs":{
        "spoke11-p1_0":[
          {
            "timestamp":1614813142,
            "link":"up",
            "latency":0.12413334846496582,
            "jitter":0.028366668149828911,
            "packetloss":0
          },

    {
      "name":"ping",
      "interface":"spoke12-p1",
      "logs":[
        {
          "timestamp":1614813143,
          "link":"up",
          "latency":0.11373332887887955,
          "jitter":0.023099998012185097,
          "packetloss":0
        },

     "child_intfs":{
        "spoke12-p1_0":[
          {
            "timestamp":1614813143,
            "link":"up",
            "latency":0.0930333212018013,
            "jitter":0.011033335700631142,
            "packetloss":0
          },
....
....
```

Health check command example:

```
https://172.172.172.9/api/v2/monitor/virtual-wan/health-check
{
```

```
  "http_method":"GET",
  "results":{
    "ping":{
      "spoke11-p1":{
        "status":"up",
        "latency":0.13406667113304138,
        "jitter":0.023000005632638931,
        "packet_loss":0,
        "packet_sent":29722,
        "packet_received":29718,
        "sla_targets_met":[
          1
        ],
        "session":2,
        "tx_bandwidth":1353,
        "rx_bandwidth":1536,
        "state_changed":1614798274,
        "child_intfs":{
          "spoke11-p1_0":{
            "status":"up",
            "latency":0.12929999828338623,
            "jitter":0.028200000524520874,
            "packet_loss":0,
            "packet_sent":29626,
            "packet_received":29625,
            "sla_targets_met":[
              1
            ],
            "session":0,
            "tx_bandwidth":2608,
            "rx_bandwidth":1491,
            "state_changed":0
          }
        }
      },
      "spoke12-p1":{
        "status":"up",
        "latency":0.11356667429208755,
        "jitter":0.015699999406933784,
        "packet_loss":0,
        "packet_sent":29722,
        "packet_received":29717,
        "sla_targets_met":[
          1
        ],
        "session":2,
        "tx_bandwidth":1353,
        "rx_bandwidth":1536,
        "state_changed":1614798274,
        "child_intfs":{
          "spoke12-p1_0":{
            "status":"up",
            "latency":0.095466658473014832,
            "jitter":0.0092999991029500961,
            "packet_loss":0,
            "packet_sent":29687,
```

```
            "packet_received":29686,
            "sla_targets_met":[
              1
            ],
            "session":0,
            "tx_bandwidth":1309,
            "rx_bandwidth":2553,
            "state_changed":0
          }
        }
      }
    }
  },
....
....
```


CLI diagnose commands:

```
# diagnose sys sdwan intf-sla-log port13
    Timestamp: Wed Jan 9 18:33:49 2019, used inbandwidth: 3208bps, used outbandwidth:
3453bps, used bibandwidth: 6661bps, tx bytes: 947234bytes, rx bytes: 898622bytes.
    Timestamp: Wed Jan 9 18:33:59 2019, used inbandwidth: 3317bps, used outbandwidth:
3450bps, used bibandwidth: 6767bps, tx bytes: 951284bytes, rx bytes: 902937bytes.
    Timestamp: Wed Jan 9 18:34:09 2019, used inbandwidth: 3302bps, used outbandwidth:
3389bps, used bibandwidth: 6691bps, tx bytes: 956268bytes, rx bytes: 907114bytes.
    Timestamp: Wed Jan 9 18:34:19 2019, used inbandwidth: 3279bps, used outbandwidth:
3352bps, used bibandwidth: 6631bps, tx bytes: 958920bytes, rx bytes: 910793bytes.
    Timestamp: Wed Jan 9 18:34:29 2019, used inbandwidth: 3233bps, used outbandwidth:
3371bps, used bibandwidth: 6604bps, tx bytes: 964374bytes, rx bytes: 914854bytes.
    Timestamp: Wed Jan 9 18:34:39 2019, used inbandwidth: 3235bps, used outbandwidth:
3362bps, used bibandwidth: 6597bps, tx bytes: 968250bytes, rx bytes: 918846bytes.
    Timestamp: Wed Jan 9 18:34:49 2019, used inbandwidth: 3165bps, used outbandwidth:
3362bps, used bibandwidth: 6527bps, tx bytes: 972298bytes, rx bytes: 922724bytes.
    Timestamp: Wed Jan 9 18:34:59 2019, used inbandwidth: 3184bps, used outbandwidth:
3362bps, used bibandwidth: 6546bps, tx bytes: 977282bytes, rx bytes: 927019bytes.
```

```
# diagnose sys sdwan sla-log ping 1 spoke11-p1_0
    Timestamp: Wed Mar  3 15:35:20 2021, vdom root, health-check ping, interface: spoke11-
p1_0, status: up, latency: 0.135, jitter: 0.029, packet loss: 0.000%.

# diagnose sys sdwan sla-log ping 2 spoke12-p1_0
    Timestamp: Wed Mar  3 15:36:08 2021, vdom root, health-check ping, interface: spoke12-
p1_0, status: up, latency: 0.095, jitter: 0.010, packet loss: 0.000%.
```

```
# diagnose sys sdwan health-check
    Health Check(ping):
    Seq(1 spoke11-p1): state(alive), packet-loss(0.000%) latency(0.156), jitter(0.043) sla_
map=0x1
    Seq(1 spoke11-p1_0): state(alive), packet-loss(0.000%) latency(0.128), jitter(0.024)
sla_map=0x1
    Seq(2 spoke12-p1): state(alive), packet-loss(0.000%) latency(0.125), jitter(0.028) sla_
map=0x1
    Seq(2 spoke12-p1_0): state(alive), packet-loss(0.000%) latency(0.093), jitter(0.008)
sla_map=0x1
```

Mean opinion score calculation and logging in performance SLA health checks

The mean opinion score (MOS) is a method of measuring voice quality using a formula that takes latency, jitter, packet loss, and the codec into account to produce a score from zero to five (0 - 5). The G.711, G.729, and G.722 codecs can be selected in the health check configurations, and an MOS threshold can be entered to indicate the minimum MOS score for the SLA to pass. The maximum MOS score will depend on which codec is used, since each codec has a theoretical maximum limit.

```
config system sdwan
    config health-check
          edit <name>
            set mos-codec {g711 | g729 | g722}
            config sla
                edit <id>
                    set link-cost-factor {latency jitter packet-loss mos}
                    set mos-threshold <value>
                next
            end
        next
    end
end
```

`mos-codec {g711 | g729 | g722}`	Set the VoIP codec to use for the MOS calculation (default = g711).
`link-cost-factor {latency jitter packet-loss mos}`	Set the criteria to base the link selection on.
`mos-threshold <value>`	Set the minimum MOS for the SLA to be marked as pass (1.0 - 5.0, default = 3.6).

To configure a health check to calculate the MOS:

```
config system sdwan
    set status enable
    config zone
        edit "virtual-wan-link"
        next
    end
    config members
        edit 1
            set interface "dmz"
            set gateway 172.16.208.2
        next
        edit 2
            set interface "port15"
            set gateway 172.16.209.2
        next
    end
    config health-check
        edit "Test_MOS"
            set server "2.2.2.2"
            set sla-fail-log-period 30
            set sla-pass-log-period 30
            set members 0
            set mos-codec g729
```

```
            config sla
                edit 1
                    set link-cost-factor mos
                    set mos-threshold "4.0"
                next
            end
        next
    end
end
```

To use an MOS SLA to steer traffic in an SD-WAN rule:

```
config system sdwan
    config service
        edit 1
            set name "MOS_traffic_steering"
            set mode sla
            set dst "HQ_LAN"
            set src "Branch_LAN"
            config sla
                edit "Test_MOS"
                    set id 1
                next
            end
            set priority-members 0
        next
    end
end
```

The MOS currently cannot be used to steer traffic when the `mode` is set to `priority`.

To verify the MOS calculation results:

1. Verify the health check diagnostics:

```
# diagnose sys sdwan health-check
Health Check(Test_MOS):
Seq(1 dmz): state(alive), packet-loss(0.000%) latency(0.114), jitter(0.026), mos(4.123),
bandwidth-up(999999), bandwidth-dw(999997), bandwidth-bi(1999996) sla_map=0x1
Seq(2 port15): state(alive), packet-loss(0.000%) latency(0.100), jitter(0.008), mos
(4.123), bandwidth-up(999999), bandwidth-dw(999999), bandwidth-bi(1999998) sla_map=0x1
```

```
# diagnose sys sdwan sla-log Test_MOS 1
Timestamp: Tue Jan  4 11:23:06 2022, vdom root, health-check Test_MOS, interface: dmz,
status: up, latency: 0.151, jitter: 0.040, packet loss: 0.000%, mos: 4.123.
Timestamp: Tue Jan  4 11:23:07 2022, vdom root, health-check Test_MOS, interface: dmz,
status: up, latency: 0.149, jitter: 0.041, packet loss: 0.000%, mos: 4.123.
```

```
# diagnose sys sdwan sla-log Test_MOS 2
Timestamp: Tue Jan  4 11:25:09 2022, vdom root, health-check Test_MOS, interface:
port15, status: up, latency: 0.097, jitter: 0.009, packet loss: 0.000%, mos: 4.123.
Timestamp: Tue Jan  4 11:25:10 2022, vdom root, health-check Test_MOS, interface:
port15, status: up, latency: 0.097, jitter: 0.008, packet loss: 0.000%, mos: 4.123.
```

2. Change the `mos-codec` to `g722`. The diagnostics will now display different MOS values:

```
# diagnose sys sdwan health-check
Health Check(Test_MOS):
Seq(1 dmz): state(alive), packet-loss(0.000%) latency(0.150), jitter(0.031), mos(4.453),
bandwidth-up(999999), bandwidth-dw(999997), bandwidth-bi(1999996) sla_map=0x1
Seq(2 port15): state(alive), packet-loss(0.000%) latency(0.104), jitter(0.008), mos
(4.453), bandwidth-up(999999), bandwidth-dw(999999), bandwidth-bi(1999998) sla_map=0x1
```

3. Increase the latency on the link in port15. The calculated MOS value will decrease accordingly. In this example, port15 is out of SLA since its MOS value is now less than the 4.0 minimum:

```
# diagnose sys sdwan  health-check
Health Check(Test_MOS):
Seq(1 dmz): state(alive), packet-loss(0.000%) latency(0.106), jitter(0.022), mos(4.453),
bandwidth-up(999999), bandwidth-dw(999997), bandwidth-bi(1999996) sla_map=0x1
Seq(2 port15): state(alive), packet-loss(0.000%) latency(300.119), jitter(0.012), mos
(3.905), bandwidth-up(999999), bandwidth-dw(999999), bandwidth-bi(1999998) sla_map=0x0
```

Sample logs

```
date=2022-01-04 time=11:57:54 eventtime=1641326274876828300 tz="-0800" logid="0113022933"
type="event" subtype="sdwan" level="notice" vd="root" logdesc="SDWAN SLA notification"
eventtype="SLA" healthcheck="Test_MOS" slatargetid=1 interface="port15" status="up"
latency="300.118" jitter="0.013" packetloss="0.000" mos="3.905"
inbandwidthavailable="1000.00Mbps" outbandwidthavailable="1000.00Mbps"
bibandwidthavailable="2.00Gbps" inbandwidthused="0kbps" outbandwidthused="0kbps"
bibandwidthused="0kbps" slamap="0x0" metric="mos" msg="Health Check SLA status. SLA failed
due to being over the performance metric threshold."

date=2022-01-04 time=11:57:24 eventtime=1641326244286635920 tz="-0800" logid="0113022923"
type="event" subtype="sdwan" level="notice" vd="root" logdesc="SDWAN status"
eventtype="Health Check" healthcheck="Test_MOS" slatargetid=1 oldvalue="2" newvalue="1"
msg="Number of pass member changed."

date=2022-01-04 time=11:57:24 eventtime=1641326244286627260 tz="-0800" logid="0113022923"
type="event" subtype="sdwan" level="notice" vd="root" logdesc="SDWAN status"
eventtype="Health Check" healthcheck="Test_MOS" slatargetid=1 member="2" msg="Member status
changed. Member out-of-sla."

date=2022-01-04 time=11:57:02 eventtime=1641326222516756500 tz="-0800" logid="0113022925"
type="event" subtype="sdwan" level="information" vd="root" logdesc="SDWAN SLA information"
eventtype="SLA" healthcheck="Test_MOS" slatargetid=1 interface="port15" status="up"
latency="0.106" jitter="0.007" packetloss="0.000" mos="4.453"
inbandwidthavailable="1000.00Mbps" outbandwidthavailable="1000.00Mbps"
bibandwidthavailable="2.00Gbps" inbandwidthused="0kbps" outbandwidthused="0kbps"
bibandwidthused="0kbps" slamap="0x1" msg="Health Check SLA status."
```

SD-WAN rules

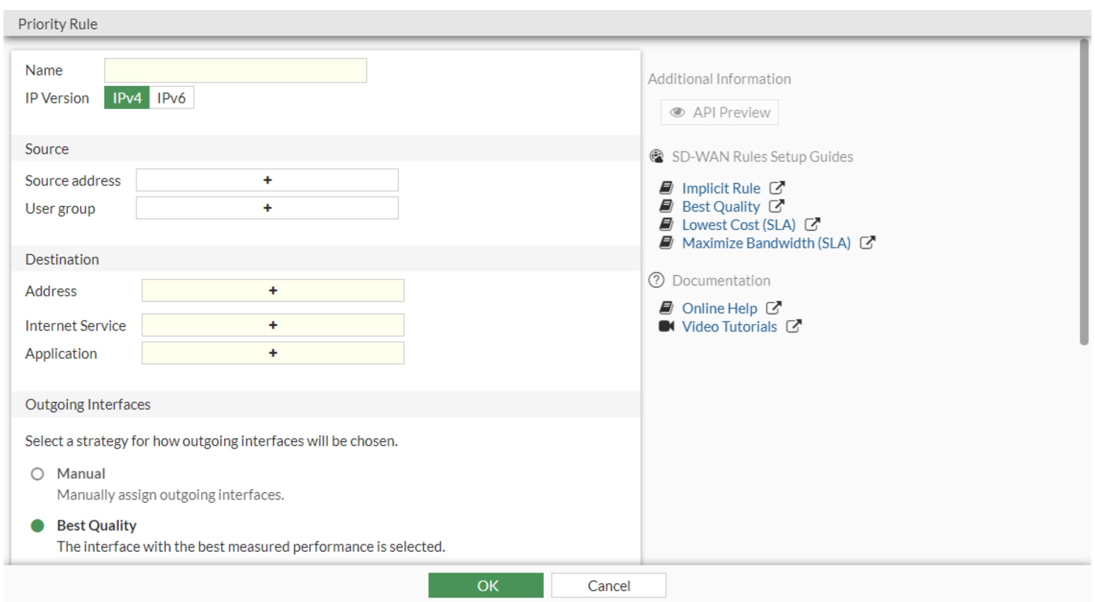
SD-WAN rules, which are sometimes called *service rules*, identify traffic of interest, and then route the traffic based on a strategy and the condition of the route or *link* between two devices. You can use many strategies to select the outgoing interface and many performance service level agreements (SLAs) to evaluate the link conditions.

Use the following topics to learn about and create SD-WAN rules for your needs:

Overview

SD-WAN rules control how sessions are distributed to SD-WAN members. You can configure SD-WAN rules from the GUI and CLI.

From the GUI, go to *Network* > *SD-WAN* > *SD-WAN Rules*. When creating a new SD-WAN rule, or editing an existing SD-WAN rule, use the *Source* and *Destination* sections to identify traffic, and use the *Outgoing interfaces* section to configure WAN intelligence for routing traffic.



From the CLI, use the following command to configure SD-WAN rules:

```
config system sdwan
    config service
        edit <ID>
        next
```

```
    end
end
```

The following topics describe the fields used to configure SD-WAN rules:

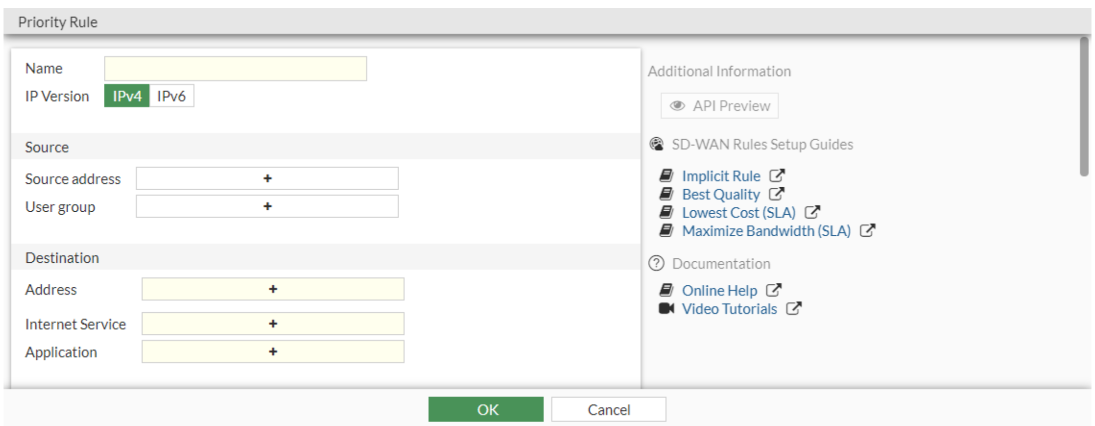
- Fields for identifying traffic on page 540
- Fields for configuring WAN intelligence on page 543
- Additional fields for configuring WAN intelligence on page 545

Fields for identifying traffic

This topic describes the fields in an SD-WAN rule used for defining the traffic to which the rule applies. Some fields are available only in the CLI.

SD-WAN rules can identify traffic by source address, destination address, service, and individual or user group matches. SD-WAN rules can also identify traffic by application control (application-aware routing), internet service database (ISDB), BGP route tags, and Differentiated Services Code Point (DSCP) tags.

In the GUI, go to *Network > SD-WAN > SD-WAN Rules*. Click *Create New*, or double-click an existing rule to open it for editing. The *Source* and *Destination* sections are used to identify traffic for the rule:



In the CLI, edit the service definition ID number to identify traffic for the rule:

```
config system sdwan
    config service
        edit <ID>
        <CLI commands from the following tables>
        ...
    end
end
```

The following table describes the fields used for the name, ID, and IP version of the SD-WAN rule:

Name, ID, and IP version		
Field	**CLI**	**Description**
Name	`set name <string>`	The name does not need to relate to the traffic being matched, but it is good practice to have intuitive rule names.

Name, ID, and IP version		
Field	**CLI**	**Description**

ID	`config system sdwan`
	`    config service`
	`        edit <ID>`
	`        next`
	`    end`
	`end`
		ID is generated when the rule is created. You can only specify the ID from the CLI.

IP version	`set addr-mode <ipv4 | ipv6>`	The addressing mode can be IPv4 or IPv6. To configure in the GUI, IPv6 must be enabled from *System > Feature Visibility* page.

The following table describes the fields used for source section of the SD-WAN rule:

Source		
Field	**CLI**	**Description**
Source address	`set src <object>` May be negated from the CLI with `set src-negate`.	One or more address objects.
User group	`set users <user object>` `set groups <group object>`	Individual users or user groups
Source interface (`input-device`)	`set input-device <interface name>` May be negated with `set input-device-negate enable`.	CLI only. Select one or more source interfaces.

The following table describes the fields used for the destination section of the SD-WAN rule:

Destination		
Field	**CLI**	**Description**

Address	`set dst <object>`
	`set protocol <integer>`
	`set start-port <integer>`, `set end-port <integer>`
	Use `set dst-negate enable` to negate the address object.
		One or more address objects.
		One protocol and one port range can be combined with the address object.
		If it is necessary for an SD-WAN rule to match multiple protocols or multiple port ranges, you can create a custom Internet Service.

Internet Service	`set internet-service enable`	One or more internet services or service groups.

Destination		
Field	**CLI**	**Description**

	`set internet-service-custom <name_1> <name_2> ... <name_n>`
	`set internet-service-custom-group <name_1> <name_2> ... <name_n>`
	`set internet-service-name <name_1> <name_2> ... <name_n>`
	`set internet-service-group <name_1> <name_2> ... <name_n>`
		This applies only to IPv4 rules, and cannot be used in conjunction with an address object.

Application	`set internet-service-app-ctrl <id_1> <id_2> ... <id_n>`
	`set internet-service-app-ctrl-group <name_1> <name_2> ... <name_n>`
	`set internet-service-app-ctrl-category <id_1> <id_2> ... <id_n>`
		One or more applications or application groups.
		This applies only to IPv4 rules, and cannot be used in conjunction with an address object.
		May be used with internet services or service group.

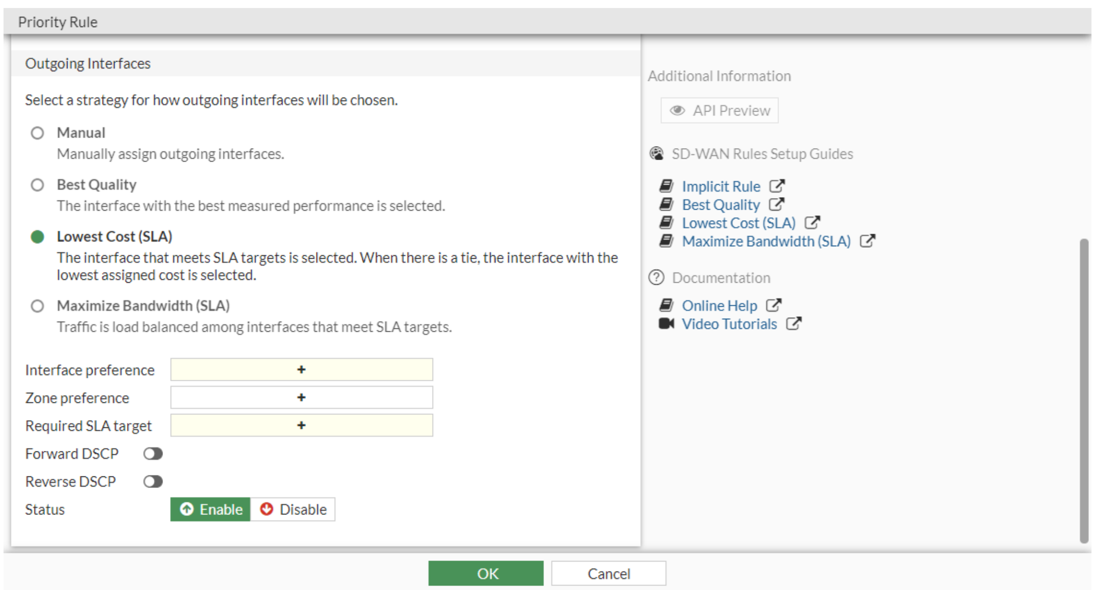
Route tag (`route-tag`)	`set route-tag <integer>`	CLI only. This replaces the `dst` field (if previously configured) and matches a BGP route tag configured in a route map. See Using BGP tags with SD-WAN rules on page 608.
TOS mask (`tos-mask`)	`set tos-mask <8-bit hex value>`	CLI only. In order to leverage type of service (TOS) matching or DSCP matching on the IP header, the SD-WAN rule must specify the bit mask of the byte holding the TOS value. For example, a TOS mask of 0xe0 (11100000) matches the upper 3 bits.
TOS (`tos`)	`set tos <8 bit hex value>`	CLI only. The value specified here is matched after the `tos-mask` is applied.

Destination		
Field	**CLI**	**Description**
		For example, the FortiGate receives DSCP values 110000 and 111011. (DSCP is the upper 6 bits of the TOS field – 11000000 and 11101100 respectively). Using the TOS value 0xe0 (11100000), only the second DSCP value is matched.

Fields for configuring WAN intelligence

This topic describes the fields in an SD-WAN rule used for configuring WAN intelligence, which processes and routes traffic that matches the SD-WAN rule.

In the GUI, go to *Network* > *SD-WAN* > *SD-WAN Rules*. Click *Create New*, or double-click an existing rule to open it for editing. The *Outgoing Interfaces* section is used to configure WAN intelligence for the rule:



WAN intelligence is comprised of the following parts:

- Interface or zone preference on page 543
- Strategy on page 544
- Performance SLA on page 544

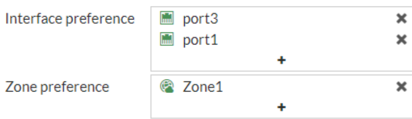
Interface or zone preference

By default, the configured order of interfaces and/or zones in a rule are used. Interfaces and zones that are selected first have precedence over interfaces selected second and so on.

You can specify both interfaces and zones. When a zone is specified in the *Zone preference* field, it is equivalent to selecting each of the contained interface members in the *Interface preference* section. Interface members in a zone have lower priority than interfaces configured in the *Interface preference* section.

For example:

- There are 3 interfaces: port1, port2 and port3.
  - Port2 is in Zone1
  - Port1 and port3 belong to the default *virtual-wan-link* zone.
- An SD-WAN rule is created with *Interface preference* set to *port3* and *port1*, and *Zone preference* set to *Zone1*.



The SD-WAN rule prefers the interfaces in the following order:

1. port3
2. port1
3. port2

You can configure the interface and zone preference in the CLI:

```
config system sdwan
    config service
        edit <ID>
            set priority-members <integer>
            set priority-zone <interface>
        next
    end
end
```

Strategy

Strategy dictates how the interface and/or zone order changes as link conditions change. You can use the following strategies:

- Automatic (`auto`): interfaces are assigned a priority based on quality. See Automatic strategy on page 551.
- Manual (`manual`): interfaces are manually assigned a priority. See Manual strategy on page 552.
- Best Quality (`priority`): interfaces are assigned a priority based on the `link-cost-factor` of the interface. See Best quality strategy on page 553.
- Lowest cost (SLA) (`sla`): interfaces are assigned a priority based on selected SLA settings. See Lowest cost (SLA) strategy on page 557.
- Maximize Bandwidth (SLA) (`load-balance`): traffic is distributed among all available links based on the selected load balancing algorithm. See Maximize bandwidth (SLA) strategy on page 560.

Performance SLA

The best quality, lowest cost, and maximize bandwidth strategies are the most intelligent modes, and they leverage SLA health checks to provide meaningful metrics for a given link. FortiGate uses the metrics to make intelligent decisions to route traffic.

Automatic and manual strategies have pre-configured logic that do not leverage SLA health checks.

The goal of the performance SLA is to measure the quality of each SD-WAN member link. The following methods can be used to measure the quality of a link:

- Active measurement
  - Health-check traffic is sent to a server with a variety of protocols options.
  - The following SLA metrics are measured on this probe traffic:
    - Latency
    - Jitter
    - Packet loss
- Passive measurement
  - SLA metrics are measured on real or live traffic, reducing the amount of probe traffic that is sent and received.
  - There is the option (prefer passive) to initiate probe traffic when no live traffic is present.

Performance SLA is utilized by `auto`, *Lowest Cost (SLA)*, *Maximize Bandwidth (SLA)*, and *Best Quality* strategies. *Lowest Cost (SLA)* and *Maximize Bandwidth SLA* use SLA targets in a pass or fail style to evaluate whether a link is considered for traffic. *Best Quality* compares a specific metric of the SLA to pick the best result.

Therefore it is integral to select or create an SLA target(s) that relates to the traffic targeted by the rule. It does not make sense to evaluate a public resource, such as YouTube, when the rule matches Azure traffic.

See Performance SLA on page 506 for more details.

Additional fields for configuring WAN intelligence

This topic describes the fields in an SD-WAN rule used for configuring WAN intelligence for egress traffic:

- Forward and/or reverse differentiated services code point (DSCP) on page 545
- Default and gateway options on page 546

For information about accessing fields for configuring WAN intelligence, see Fields for configuring WAN intelligence on page 543.

Forward and/or reverse differentiated services code point (DSCP)

The FortiGate differentiated services feature can be used to change the DSCP value for all packets accepted by a policy.

The packet's `DSCP` field for traffic initiating a session (forward) or for reply traffic (reverse) can be changed and enabled in each direction separately by configuring it in the firewall policy using the *Forward DSCP* and *Reverse DSCP* fields.

From the CLI:

```
config system sdwan
    config service
        edit <ID>
            ...
            set dscp-forward enable
            ...
        next
    end
end
```

`set dscp-forward enable`	Enable use of forward DSCP tag.

`set dscp-forward-tag 000000`	Forward traffic DSCP tag.
`set dscp-reverse enable`	Enable use of reverse DSCP tag.
`set dscp-reverse-tag 000000`	Reverse traffic DSCP tag.

Default and gateway options

Following are additional gateway options that can be set only in the CLI:

```
config system sdwan
    config service
        edit <ID>
            ...
            set default enable
            ...
        next
    end
end
```

`set default [enable|disable]`	Enable or disable use of SD-WAN as default service.
`set gateway [enable|disable]`	Enable or disable SD-WAN service gateway.

By default, these settings are set to `disable`.

These two commands help adjust FortiGate route selection by affecting how the FortiGate consults the Forward Information Base (FIB).

In order to decide whether an SD-WAN policy-route can be matched, FortiGate performs the following FIB lookups:

- FIB best match for the destination must return an SD-WAN member.
- FIB route to the destination must exist over the desired SD-WAN member.

When `set default enable` is used with `set gateway enable`, FortiGate bypasses the FIB checks, and instead routes any matching traffic of the SD-WAN rule to the chosen SD-WAN member using the member's configured gateway. SD-WAN members must have a gateway configured.

When `set default disable` is used with `set gateway enable`, FortiGate keeps the first rule in effect but causes the second rule to change to:

- FIB route to the gateway IP address must exist over any interface.

See also Fields for configuring WAN intelligence on page 543.

Implicit rule

SD-WAN rules define specific policy routing options to route traffic to an SD-WAN member. When no explicit SD-WAN rules are defined, or if none of the rules are matched, then the default implicit rule is used.

In an SD-WAN configuration, the default route usually points to the SD-WAN interface, so each active member's gateway is added to the routing table's default route. FortiOS uses equal-cost multipath (ECMP) to balance traffic between the interfaces. One of five load balancing algorithms can be selected:

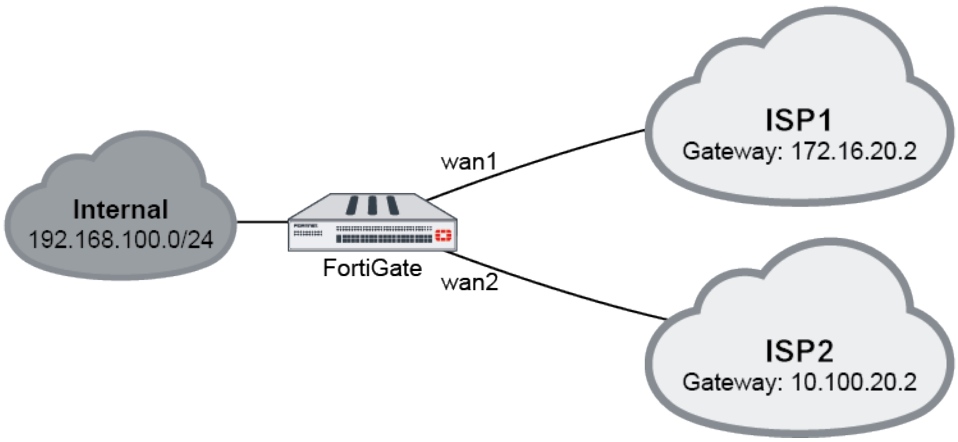
Source IP (`source-ip-based`)	Traffic is divided equally between the interfaces, including the SD-WAN interface. Sessions that start at the same source IP address use the same path. This is the default selection.
Sessions (`weight-based`)	The workload is distributing based on the number of sessions that are connected through the interface. The weight that you assign to each interface is used to calculate the percentage of the total sessions that are allowed to connect through an interface, and the sessions are distributed to the interfaces accordingly. Sessions with the same source and destination IP addresses (`src-ip` and `dst-ip`) are forwarded to the same path, but are still considered in later session ratio calculations. An interface's weight value cannot be zero.
Spillover (`usage-based`)	The interface is used until the traffic bandwidth exceeds the ingress and egress thresholds that you set for that interface. Additional traffic is then sent through the next SD-WAN interface member.
Source-Destination IP (`source-dest-ip-based`)	Traffic is divided equally between the interfaces. Sessions that start at the same source IP address and go to the same destination IP address use the same path.
Volume (`measured-volume-based`)	The workload is distributing based on the number of packets that are going through the interface. The volume weight that you assign to each interface is used to calculate the percentage of the total bandwidth that is allowed to go through an interface, and the bandwidth is distributed to the interfaces accordingly. An interface's volume value cannot be zero.

You cannot exclude an interface from participating in load balancing using the implicit rule. If the weight or volume were set to zero in a previous FortiOS version, the value is treated as a one.

Interfaces with static routes can be excluded from ECMP if they are configured with a lower priority than other static routes.

Examples

The following four examples demonstrate how to use the implicit rules (load-balance mode).



If no SD-WAN zone is specified, members are added to the default *virtual-wan-link* zone.

Example 1

Outgoing traffic is equally balanced between wan1 and wan2, using *source-ip-based* or *source-dest-ip-based* mode.

Using the GUI:

1. On the FortiGate, enable SD-WAN and add wan1 and wan2 as SD-WAN members, then add a policy and static route. See SD-WAN quick start on page 487 for details.
2. Go to *Network* > *SD-WAN* and select the *SD-WAN Rules* tab.
3. Edit the *sd-wan* rule (the last default rule).
4. For the *Load Balancing Algorithm*, select either *Source IP* or *Source-Destination IP*.
5. Click *OK*.

Using the CLI:

1. Enable SD-WAN and add wan1 and wan2 as SD-WAN members, then add a policy and static route. See SD-WAN quick start on page 487 for details.
2. Set the load balancing algorithm:
   Source IP based:
   ```
   config system sdwan
       set load-balance-mode source-ip-based
   end
   ```
   Source-Destination IP based:
   ```
   config system sdwan
       set load-balance-mode source-dest-ip-based
   end
   ```

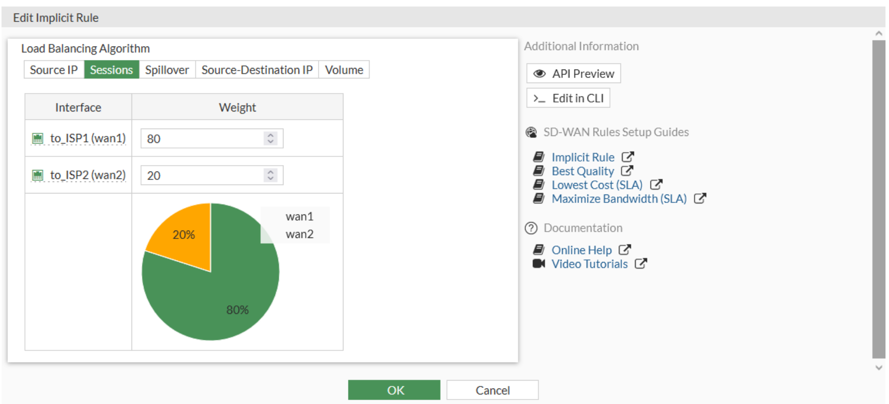
Example 2

Outgoing traffic is balanced between wan1 and wan2 with a customized ratio, using *weight-based* mode: wan1 runs 80% of the sessions, and wan2 runs 20% of the sessions.

Sessions with the same source and destination IP addresses (`src-ip` and `dst-ip`) will be forwarded to the same path, but will still be considered in later session ratio calculations.

Using the GUI:

1. Go to *Network > SD-WAN* and select the *SD-WAN Rules* tab.
2. Edit the *sd-wan* rule (the last default rule).
3. For the *Load Balancing Algorithm*, select *Sessions*.
4. Enter 80 in the *wan1* field, and 20 in the *wan2* field.



5. Click *OK*.

Using the CLI:

```
config system sdwan
    set load-balance-mode weight-based
    config members
        edit 1
            set interface "wan1"
            set weight 80
        next
        edit 2
            set interface "wan2"
            set weight 20
        next
    end
end
```

Example 3

Outgoing traffic is balanced between wan1 and wan2 with a customized ratio, using *measured-volume-based* mode: wan1 runs 80% of the volume, and wan2 runs 20% of the volume.

Using the GUI:

1. Go to *Network > SD-WAN* and select the *SD-WAN Rules* tab.
2. Edit the *sd-wan* rule (the last default rule).
3. For the *Load Balancing Algorithm*, select *Volume*.
4. Enter 80 in the *wan1* field, and 20 in the *wan2* field.
5. Click *OK*.

Using the CLI:

```
config system sdwan
    set load-balance-mode measured-volume-based
    config members
        edit 1
            set interface "wan1"
            set volume-ratio 80
        next
        edit 2
            set interface "wan2"
            set volume-ratio 20
        next
    end
end
```

Example 4

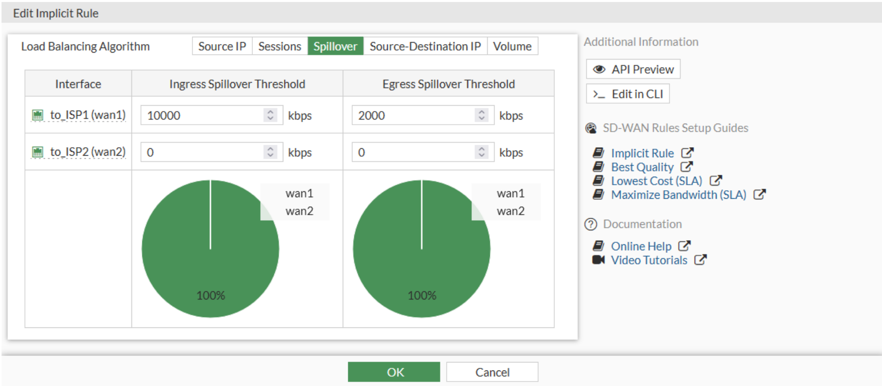
Load balancing can be used to reduce costs when internet connections are charged at different rates. For example, if wan2 charges based on volume usage and wan1 charges a fixed monthly fee, we can use wan1 at its maximum bandwidth, and use wan2 for overflow.

In this example, wan1's bandwidth is 10Mbps down and 2Mbps up. Traffic will use wan1 until it reaches its spillover limit, then it will start to use wan2. Note that `auto-asic-offload` must be disabled in the firewall policy.

Using the GUI:

1. On the FortiGate, enable SD-WAN and add wan1 and wan2 as SD-WAN members, then add a policy and static route. See SD-WAN quick start on page 487 for details.
2. Go to *Network > SD-WAN* and select the *SD-WAN Rules* tab.
3. Edit the *sd-wan* rule (the last default rule).
4. For the *Load Balancing Algorithm*, select *Spillover*.

5. Enter 10000 in the *wan1 Ingress Spillover Threshold* field, and 2000 in the *wan1 Egress Spillover Threshold* field.



6. Click *OK*.

Using the CLI:

```
config system sdwan
    set load-balance-mode usage-based
    config members
        edit 1
            set interface "wan1"
            set spillover-threshold 2000
            set ingress-spillover-threshold 10000
        next
    end
end
```

Automatic strategy

The automatic strategy is a legacy rule that lets you select an outgoing interface based on its performance ranking compared to the other SD-WAN interfaces. This is achieved by applying a performance SLA to rank the interfaces, and then selecting the desired rank.

In this example, you have three SD-WAN interfaces to three different ISPs that all go to the public internet. WAN1 is your highest quality link and should be reserved for business critical traffic. WAN2 and WAN3 are redundant backup links. You noticed one non-critical application is taking up a lot of bandwidth and want to prioritize it to the lowest quality link at any given time.

To configure automatic SD-WAN rules from the CLI:

```
config system sdwan
    config members
        edit 1
            set interface "wan1"
        next
        edit 2
            set interface "wan2"
        next
        edit 3
            set interface "wan3"
        next
```

```
    end
    config health-check
        edit "non-critical application"
            set server "noncritical.application.com"
            set members 1 2 3
            config sla
                edit 1
                    set latency-threshold 250
                    set jitter-threshold 50
                    set packletloss-threshold 3
                next
            end
        next
    end
    config service
        edit 1
            set name "non-critical application"
            set mode auto
            set quality-link 3
            set dst "non-critical-app-address-object"
            set health-check "non-critical application"
        next
    end
end
```

The `auto` option is only available in the CLI. If you use the GUI to edit the rule, the `auto` option will be overwritten because you cannot select *auto* in the GUI.

Manual strategy

In manual mode, no health checks are used. As a result, the decision making closer resembles logic than intelligence. SD-WAN manual rules are similar to regular policy-based routes, but have the added features of application-aware routing and BGP-tag routing. A manual strategy rule is comprised of the following parts:

- Defining the interfaces to be used
- Ordering the interfaces based on preference

To configure manual SD-WAN rules from the GUI:

1. Go to *Network* > *SD-WAN*.
2. Select the *SD-WAN Rules* tab, and click *Create New*.

3. Set the following options to create a manual rule:

Name	Type a name for the rule.
Source	(Optional) Specify a *Source address* and/or *User group*.
Destination	Specify the destination using an *Address* object or an *Internet Service* or an *Application*.
Zone preference	Specify one or more SD-WAN interfaces or zones. The order in which the interfaces or zones are specified determines their priority when the rule is matched.

4. Set the remaining options as desired, and click *OK* to create the rule.

To configure manual SD-WAN rules from the CLI:

```
config system sdwan
    config members
        edit 1
            set interface "wan1"
        next
        edit 2
            set interface "wan2"
        next
    end
    config service
        edit 1
            set name "manual"
            set mode manual
            set priority-members 2 1
            set dst "DC_net"
            set hold-down-time 60
        next
    end
end
```

- The command `set mode manual` will not appear in the configuration because it is the default mode.
- The command `set hold-down-time <integer>` is an optional command that controls how long to wait before switching back to the primary interface in the event of a failover.

Best quality strategy

When using *Best Quality* mode, SD-WAN will choose the best link to forward traffic by comparing the *link-cost-factor*. A link-cost factor is a specific metric of participating link(s) (such as, latency, packet loss, and so on) evaluated against a target that you define (such as a health-check server), for example, the latency of WAN1 and WAN2 to your datacenter. Below is a list of link-cost factors available to you:

GUI	CLI	Description
Latency	`latency`	Select a link based on latency.
Jitter	`jitter`	Select a link based on jitter.
Packet Loss	`packet-loss`	Select a link based on packet loss.
Downstream	`inbandwidth`	Select a link based on available bandwidth of incoming traffic.
Upstream	`outbandwidth`	Select a link based on available bandwidth of outgoing traffic.
Bandwidth	`bibandwidth`	Select a link based on available bandwidth of bidirectional traffic.
Customized profile	`custom-profile-1`	Select link based on customized profile. If selected, set the following weights: • packet-loss-weight: Coefficient of packet-loss. • latency-weight: Coefficient of latency. • jitter-weight: Coefficient of jitter. • bandwidth-weight: Coefficient of reciprocal of available bidirectional bandwidth.

Although SD-WAN intelligence selects the best quality link according to the selected metric, by default a preference or advantage is given to the first configured SD-WAN member. This default is 10% and may be configured with the CLI command `set link-cost-threshold 10`.

Example of how `link-cost-threshold` works:

```
config system sdwan
    config members
        edit 1
            set interface "wan1"
        next
        edit 2
            set interface "wan2"
        next
    end
    config service
        edit 1
            set name "Best_Quality"
            set mode priority
            set priority-members 2 1
            set dst "DC_net"
            set health-check “DC_HealthCheck”
            set link-cost-factor latency
            set link-cost-threshold 10
        next
    end
end
```

In this example both WAN1 and WAN2 are assumed to have 200ms latency to the health-check server named `DC_HealthCheck`. Because WAN2 is specified before WAN1 in priority-members, SD-WAN parses the two interfaces metric as follows:

- WAN1: 200ms
- WAN2: 200ms / (1+10%) = ~182ms

As a result, WAN2 is selected because the latency is lower.

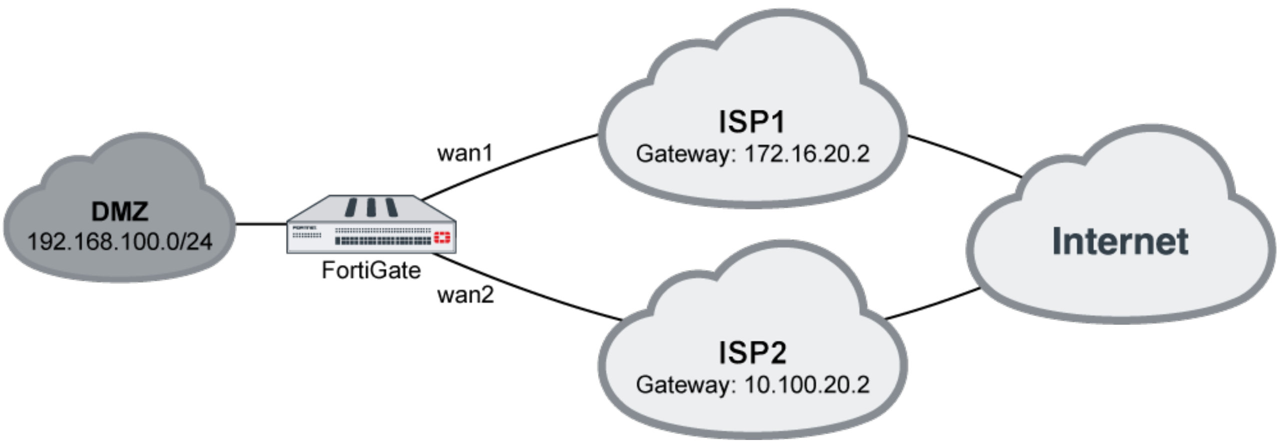
If the *Downstream* (`inbandwidth`), *Upstream* (`outbandwidth`), or *Bandwidth* (`bibandwidth`) quality criteria is used, the FortiGate uses the upstream and downstream bandwidth values configured on the member interfaces to calculate bandwidth.

The interface bandwidth configuration can be done manually, or the interface speedtest can be used to populate the bandwidth values based on the speedtest results. See Manual interface speedtest on page 526 for details.

To manually configure the upstream and downstream interface bandwidth values:

```
config system interface
    edit <interface>
        set estimated-upstream-bandwidth <speed in kbps>
        set estimated-downstream-bandwidth <speed in kbps>
    next
end
```

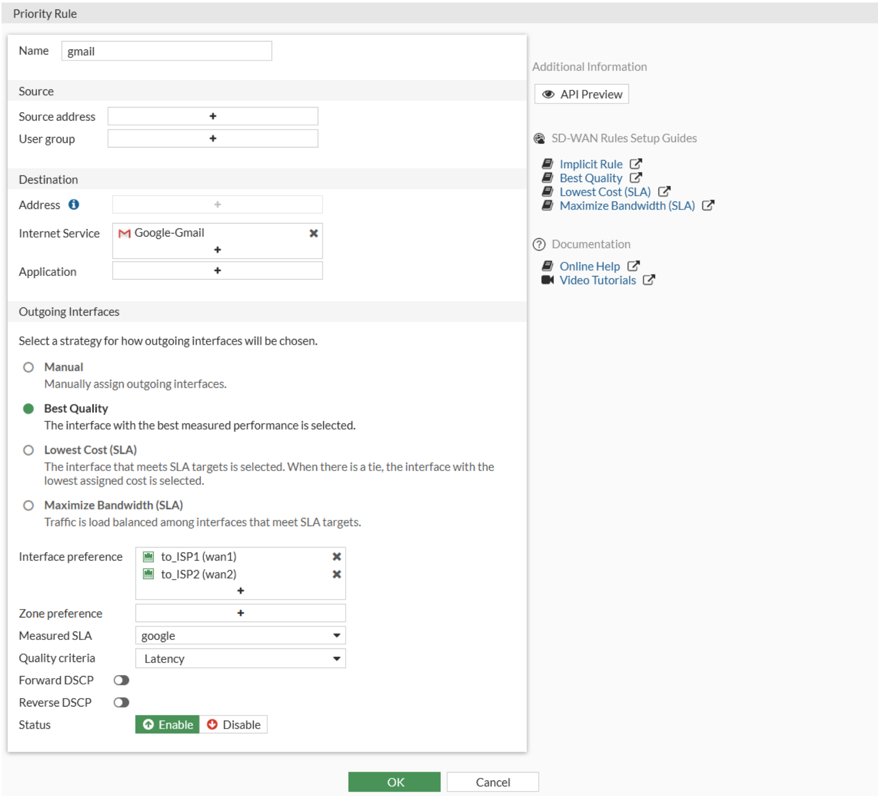
Example



In this example, your wan1 and wan2 SD-WAN interfaces connect to two ISPs that both go to the public internet, and you want Gmail services to use the link with the least latency.

To configure an SD-WAN rule to use Best Quality:

1. On the FortiGate, add wan1 and wan2 as SD-WAN members, then add a policy and static route. See SD-WAN quick start on page 487 for details.
2. Create a new Performance SLA named *google*. See Link monitoring example on page 514.
3. Go to *Network > SD-WAN*, select the *SD-WAN Rules* tab, and click *Create New*.
4. Enter a name for the rule, such as *gmail*.
5. Configure the following settings:



Internet Service	Google-Gmail
Strategy	Best Quality
Interface preference	wan1 and wan2
Measured SLA	google (created in step 2).
Quality criteria	Latency

6. Click *OK* to create the rule.

To configure an SD-WAN rule to use priority:

```
config system sdwan
    config health-check
        edit "google"
            set server "google.com"
            set members 1 2
        next
    end
    config service
        edit 1
            set name "gmail"
            set mode priority
            set internet-service enable
            set internet-service-id 65646
            set health-check "google"
```

```
            set link-cost-factor latency
            set priority-members 1 2
        next
    end
end
```

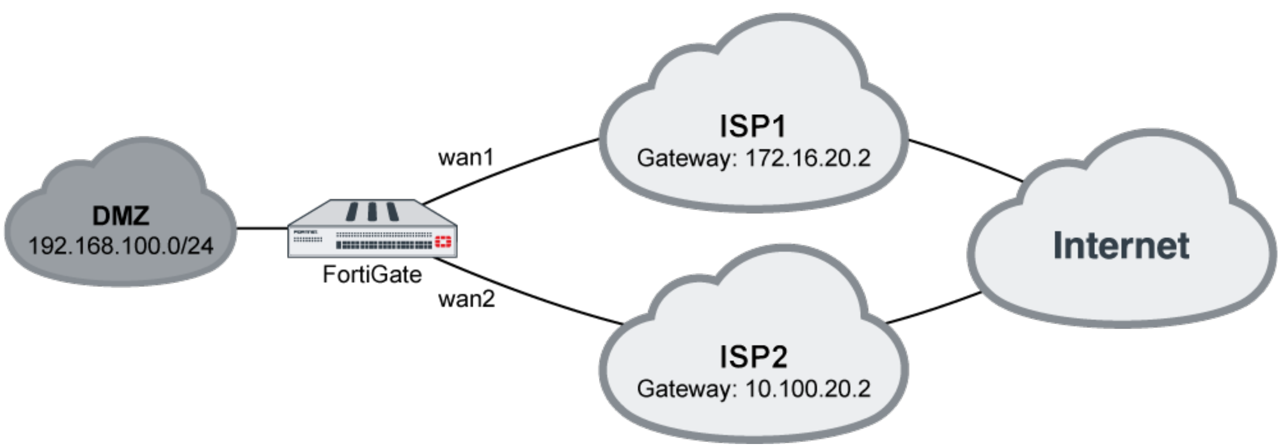
To diagnose the Performance SLA status:

```
FGT # diagnose sys sdwan health-check google
Health Check(google):
Seq(1): state(alive), packet-loss(0.000%) latency(14.563), jitter(4.334) sla_map=0x0
Seq(2): state(alive), packet-loss(0.000%) latency(12.633), jitter(6.265) sla_map=0x0

FGT # diagnose sys sdwan service 1
Service(1):

    TOS(0x0/0x0), protocol(0: 1->65535), Mode(priority), link-cost-facotr(latency), link-
cost-threshold(10), health-check(google) Members:

        1: Seq_num(2), alive, latency: 12.633, selected
        2: Seq_num(1), alive, latency: 14.563, selected

    Internet Service: Google-Gmail(65646)
```

As wan2 has a smaller latency, SD-WAN will put Seq_num(2) on top of Seq_num(1) and wan2 will be used to forward Gmail traffic.

Lowest cost (SLA) strategy

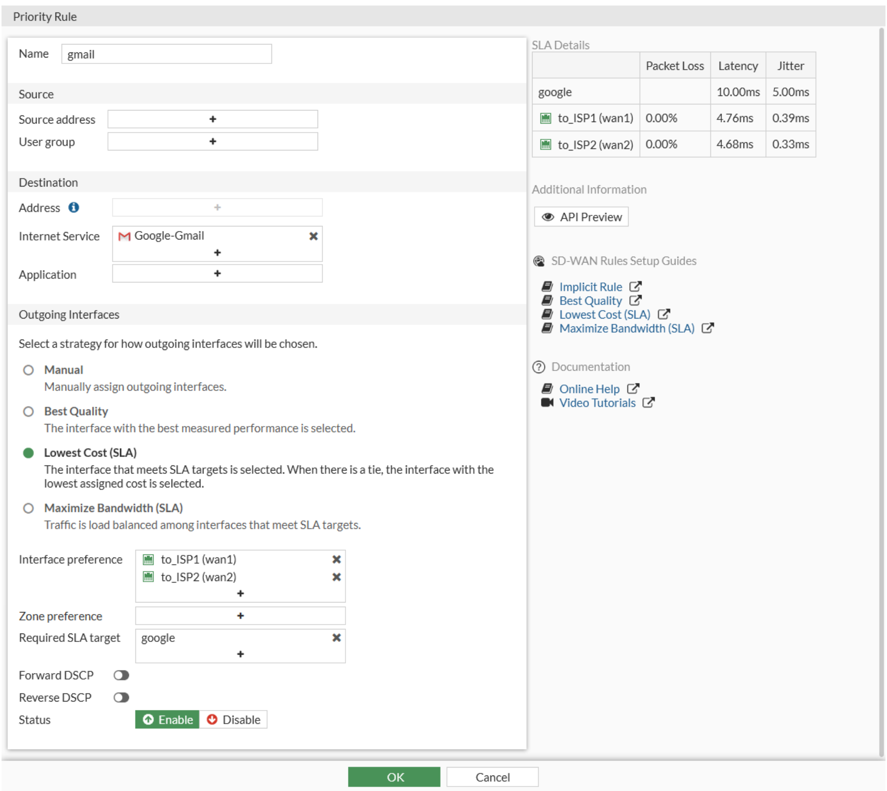
When using *Lowest Cost (SLA)* mode (`sla` in the CLI), SD-WAN will choose the lowest cost link that satisfies SLA to forward traffic. The lowest possible cost is `0`. If multiple eligible links have the same cost, the *Interface preference* order will be used to select a link.



In this example, your wan1 and wan2 SD-WAN interfaces connect to two ISPs that both go to the public internet. The cost of wan2 is less than that of wan1. You want to configure Gmail services to use the lowest cost interface, but the link quality must meet a standard of latency: 10ms, and jitter: 5ms.

To configure an SD-WAN rule to use Lowest Cost (SLA):

1. On the FortiGate, add wan1 and wan2 as SD-WAN members, then add a policy and static route. See SD-WAN quick start on page 487 for details.
2. Create a new Performance SLA named *google* that includes an SLA Target with *Latency threshold* = 10ms and *Jitter threshold* = 5ms. See Link monitoring example on page 514.
3. Go to *Network > SD-WAN*, select the *SD-WAN Rules* tab, and click *Create New*.
4. Enter a name for the rule, such as *gmail*.
5. Configure the following settings:



Internet Service	Google-Gmail
Strategy	Lowest Cost (SLA)
Interface preference	wan1 and wan2
Required SLA target	google (created in step 2).

6. Click *OK* to create the rule.

To configure an SD-WAN rule to use SLA:

```
config system sdwan
    config members
        edit 1
            set interface "wan1"
            set cost 10
```

```
        next
        edit 2
            set interface "wan2"
            set cost 5
        next
    end
    config health-check
        edit "google"
            set server "google.com"
            set members 1 2
            config sla
                edit 1
                    set latency-threshold 10
                    set jitter-threshold 5
                next
            end
        next
    end
    config service
        edit 1
            set name "gmail"
            set mode sla
            set internet-service enable
            set internet-service-id 65646
            config sla
                edit "google"
                    set id 1
                next
            end
            set priority-members 1 2
        next
    end
end
```

If no SD-WAN zone is specified, members are added to the default *virtual-wan-link* zone.

The CLI command `set minimum-sla-meet-members` allows you to specify the number of links that must meet SLA for the rule to take effect. If the number of members is less than the minimum set with this command, the rule will not take effect.

To diagnose the Performance SLA status:

```
FGT # diagnose sys sdwan health-check google
Health Check(google):
Seq(1): state(alive), packet-loss(0.000%) latency(14.563), jitter(4.334) sla_map=0x0
Seq(2): state(alive), packet-loss(0.000%) latency(12.633), jitter(6.265) sla_map=0x0

FGT # diagnose sys sdwan service 1
Service(1): Address Mode(IPV4) flags=0x0

    TOS(0x0/0x0), Protocol(0: 1->65535), Mode(sla)
```

```
    Members:<<BR>>

        1: Seq_num(2), alive, sla(0x1), cfg_order(1), selected
        2: Seq_num(1), alive, sla(0x1), cfg_order(0), selected

    Internet Service: Google.Gmail(65646)
```

When both wan1 and wan2 meet the SLA requirements, Gmail traffic will only use wan2. If only wan1 meets the SLA requirements, Gmail traffic will only use wan1, even though it has a higher cost. If neither interface meets the requirements, wan2 will be used.

If both interface had the same cost and both met the SLA requirements, the first link configured in `set priority-members` would be used.

Maximize bandwidth (SLA) strategy

When using *Maximize Bandwidth* mode (`load-balance` in the CLI), SD-WAN will choose all of the links that satisfies SLA to forward traffic based on a load balancing algorithm. The load balancing algorithm, or hash method, can be one of the following:

round-robin	All traffic are distributed to selected interfaces in equal portions and circular order. This is the default method, and the only option available when using the GUI.
source-ip-based	All traffic from a source IP is sent to the same interface.
source-dest-ip-based	All traffic from a source IP to a destination IP is sent to the same interface.
inbandwidth	All traffic are distributed to a selected interface with most available bandwidth for incoming traffic.
outbandwidth	All traffic are distributed to a selected interface with most available bandwidth for outgoing traffic.
bibandwidth	All traffic are distributed to a selected interface with most available bandwidth for both incoming and outgoing traffic.

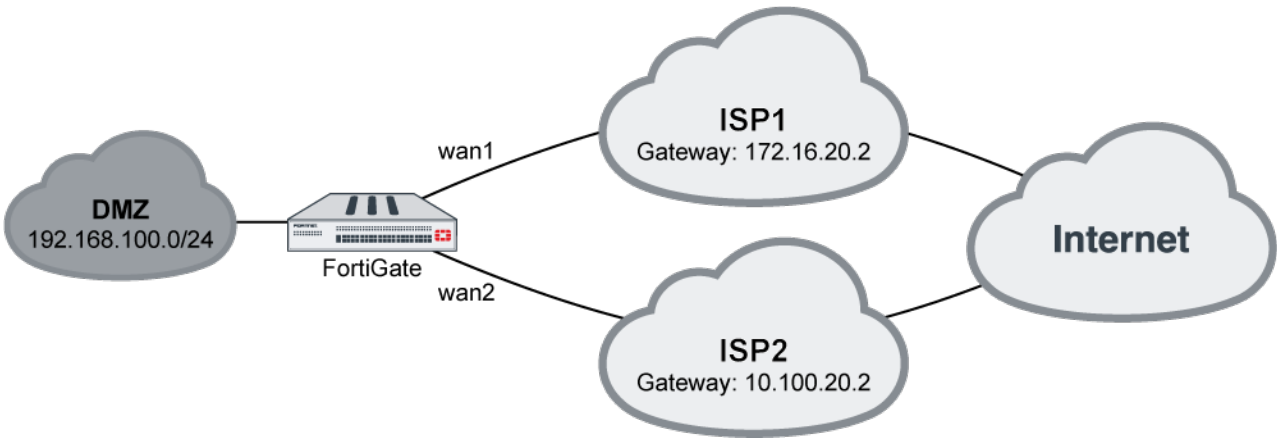
When the `inbandwidth`, `outbandwidth`), or `bibandwidth` load balancing algorithm is used, the FortiGate will compare the bandwidth based on the configured upstream and downstream bandwidth values.

The interface speedtest can be used to populate the bandwidth values based on the speedtest results. See Manual interface speedtest on page 526 for details.

To manually configure the upstream and downstream bandwidth values:

```
config system interface
    edit <interface>
        set estimated-upstream-bandwidth <speed in kbps>
        set estimated-downstream-bandwidth <speed in kbps>
    next
end
```

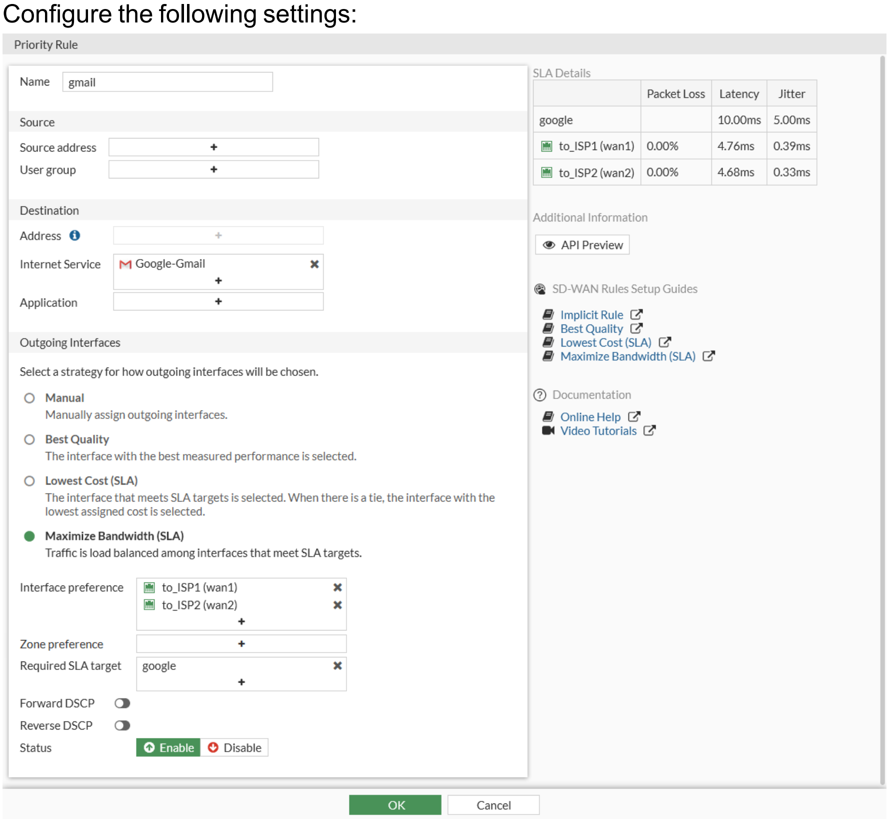
ADVPN is not supported in this mode.



In this example, your wan1 and wan2 SD-WAN interfaces connect to two ISPs that both go to the public internet. You want to configure Gmail services to use both of the interface, but the link quality must meet a standard of latency: 10ms, and jitter: 5ms. This can maximize the bandwidth usage.

To configure an SD-WAN rule to use Maximize Bandwidth (SLA):

1. On the FortiGate, add wan1 and wan2 as SD-WAN members, then add a policy and static route. See SD-WAN quick start on page 487 for details.
2. Create a new Performance SLA named *google* that includes an SLA Target 1 with *Latency threshold* = 10ms and *Jitter threshold* = 5ms. See Link monitoring example on page 514.
3. Go to *Network > SD-WAN*, select the *SD-WAN Rules* tab, and click *Create New*.
4. Enter a name for the rule, such as *gmail*.
5. Configure the following settings:



Field	Setting
Internet Service	Google-Gmail
Strategy	Maximize Bandwidth (SLA)
Interface preference	wan1 and wan2
Required SLA target	google (created in step 2).

6. Click *OK* to create the rule.

To configure an SD-WAN rule to use SLA:

```
config system sdwan
    config health-check
        edit "google"
            set server "google.com"
            set members 1 2
            config sla
                edit 1
                    set latency-threshold 10
                    set jitter-threshold 5
                next
            end
        next
    end
    config service
        edit 1
            set name "gmail"
            set addr-mode ipv4
            set mode load-balance
            set hash-mode round-robin
            set internet-service enable
            set internet-service-name Google-Gmail
            config sla
                edit "google"
                    set id 1
                next
            end
            set priority-members 1 2
        next
    end
end
```

The CLI command `set minimum-sla-meet-members` allows you to specify the number of links that must meet SLA for the rule to take effect. If the number of members is less than the minimum set with this command, the rule will not take effect.

To diagnose the performance SLA status:

```
FGT # diagnose sys sdwan health-check google
Health Check(google):
Seq(1): state(alive), packet-loss(0.000%) latency(14.563), jitter(4.334) sla_map=0x0
Seq(2): state(alive), packet-loss(0.000%) latency(12.633), jitter(6.265) sla_map=0x0
```

```
FGT # diagnose sys sdwan service 1
Service(1): Address Mode(IPV4) flags=0x0

    TOS(0x0/0x0), Protocol(0: 1->65535), Mode(load-balance)
    Members:<<BR>>

        1: Seq_num(1), alive, sla(0x1), num of pass(1), selected
        2: Seq_num(2), alive, sla(0x1), num of pass(1), selected

    Internet Service: Google.Gmail(65646)
```

When both wan1 and wan2 meet the SLA requirements, Gmail traffic will use both wan1 and wan2. If only one of the interfaces meets the SLA requirements, Gmail traffic will only use that interface.

If neither interface meets the requirements but health-check is still alive, then wan1 and wan2 tie. The traffic will try to balance between wan1 and wan2, using both interfaces to forward traffic.

Use MAC addresses in SD-WAN rules and policy routes

You can use MAC addresses as the source in SD-WAN rules and policy routes.

The FABRIC_DEVICE address object (a dynamic object that includes the IPs of Security Fabric devices) can be used as a source or destination in SD-WAN rules and policy routes.

The `diagnose ip proute match` command accepts either the IP or MAC address format for the source:

```
diagnose ip proute match <destination> <source> <interface> <protocol> <port>
```

To configure a MAC address as a source for SD-WAN and a policy route:

1. Configure the MAC address:

```
config firewall address
    edit "mac-add"
        set type mac
        set macaddr 70:4c:a5:86:de:56
    next
end
```

2. Configure the policy route:

```
config router policy
    edit 3
        set srcaddr "mac-add"
        set gateway 15.1.1.34
        set output-device ha
    next
end
```

3. Configure the SD-WAN rule:

```
config system sdwan
    config service
        edit 1
            set dst "all"
            set src "mac-add"
```

```
            set priority-members 1
        next
        edit 2
            set dst "FABRIC_DEVICE"
            set priority-members 2
        next
    end
end
```

To verify the policy route matching for a MAC address:

```
# diagnose ip proute match 3.1.1.34 70:4c:a5:86:de:56 port3 22 6
dst=3.1.1.34 src=0.0.0.0 smac=70:4c:a5:86:de:56 iif=11 protocol=22 dport=6
id=00000003 type=Policy Route
seq-num=3
```

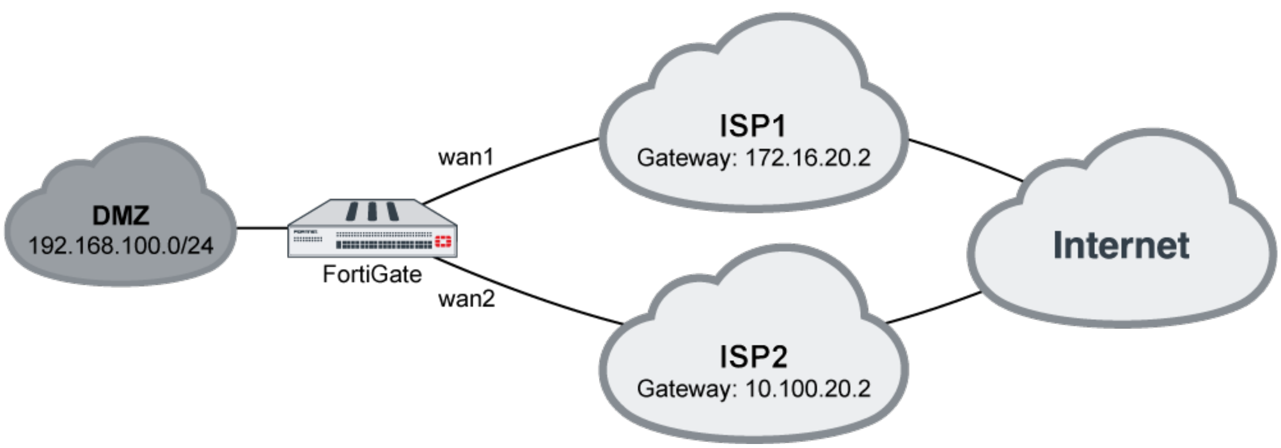
SD-WAN traffic shaping and QoS

Use a traffic shaper in a firewall shaping policy to control traffic flow. You can use it to control maximum and guaranteed bandwidth, or put certain traffic to one of the three different traffic priorities: high, medium, or low.

An advanced shaping policy can classify traffic into 30 groups. Use a shaping profile to define the percentage of the interface bandwidth that is allocated to each group. Each group of traffic is shaped to the assigned speed limit based on the outgoing bandwidth limit configured on the interface.

For more information, see Traffic shaping on page 917.

Sample topology



Sample configuration

This example shows a typical customer usage where the customer's SD-WAN uses the default zone, and has two member: wan1 and wan2, each set to 10Mb/s.

An overview of the procedures to configure SD-WAN traffic shaping and QoS with SD-WAN includes:

1. Give HTTP/HTTPS traffic high priority and give FTP low priority so that if there are conflicts, FortiGate will forward HTTP/HTTPS traffic first.

2. Even though FTP has low priority, configure FortiGate to give it a 1Mb/s guaranteed bandwidth on each SD-WAN member so that if there is no FTP traffic, other traffic can use all the bandwidth. If there is heavy FTP traffic, it can still be guaranteed a 1Mb/s bandwidth.
3. Traffic going to specific destinations such as a VOIP server uses wan1 to forward, and SD-WAN forwards with an Expedited Forwarding (EF) DSCP tag 101110.

To configure SD-WAN traffic shaping and QoS with SD-WAN in the GUI:

1. On the FortiGate, add wan1 and wan2 as SD-WAN members, then add a policy and static route. See SD-WAN quick start on page 487.
2. Add a firewall policy with *Application Control* enabled. See Configuring firewall policies for SD-WAN on page 490.
3. Go to *Policy & Objects > Traffic Shaping*, select the *Traffic Shapers* tab, and edit *low-priority*.
   a. Enable *Guaranteed Bandwidth* and set it to *1000* kbps.
4. Go to *Policy & Objects > Traffic Shaping*, select the *Traffic Shaping Policies* tab, and click *Create New*.
   a. Name the traffic shaping policy, for example, *HTTP-HTTPS*.
   b. Set the following:

Source	*all*
Destination	*all*
Service	*HTTP* and *HTTPS*
Outgoing interface	*virtual-wan-link*
Shared Shaper	Enable and set to *high-priority*
Reverse Shaper	Enable and set to *high-priority*

   c. Click *OK*.
5. Go to *Policy & Objects > Traffic Shaping*, select the *Traffic Shaping Policies* tab, and click *Create New*.
   a. Name the traffic shaping policy, for example, *FTP*.
   b. Set the following:

Source	*all*
Destination	*all*
Service	*FTP*, *FTP_GET*, and *FTP_PUT*
Outgoing interface	*virtual-wan-link*
Shared Shaper	Enable and set to *low-priority*
Reverse Shaper	Enable and set to *low-priority*

   c. Click *OK*
6. Go to *Network > SD-WAN*, select the *SD-WAN Rules* tab, and click *Create New*.
   a. Enter a name for the rule, such as *Internet*.
   b. In the *Destination* section, click *Address* and select the VoIP server that you created in the firewall address.
   c. Under *Outgoing Interfaces* select *Manual*.
   d. For *Interface preference* select *wan1*.
   e. Click *OK*.
7. Use CLI commands to modify DSCP settings. See the DSCP CLI commands below.

To configure the firewall policy using the CLI:

```
config firewall policy
    edit 1
        set name "1"
        set srcintf "dmz"
        set dstintf "virtual-wan-link"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set ssl-ssh-profile "certificate-inspection"
        set application-list "default"
        set nat enable
    next
end
```

To configure the firewall traffic shaper priority using the CLI:

```
config firewall shaper traffic-shaper
    edit "high-priority"
        set maximum-bandwidth 1048576
        set per-policy enable
    next
    edit "low-priority"
        set guaranteed-bandwidth 1000
        set maximum-bandwidth 1048576
        set priority low
        set per-policy enable
    next
end
```

To configure the firewall traffic shaping policy using the CLI:

```
config firewall shaping-policy
    edit 1
        set name "http-https"
        set service "HTTP" "HTTPS"
        set dstintf "virtual-wan-link"
        set traffic-shaper "high-priority"
        set traffic-shaper-reverse "high-priority"
        set srcaddr "all"
        set dstaddr "all"
    next
    edit 2
        set name "FTP"
        set service "FTP" "FTP_GET" "FTP_PUT"
        set dstintf "virtual-wan-link"
        set traffic-shaper "low-priority"
        set traffic-shaper-reverse "low-priority"
        set srcaddr "all"
        set dstaddr "all"
    next
end
```

To configure SD-WAN traffic shaping and QoS with SD-WAN in the CLI:

```
config system sdwan
    set status enable
    config members
        edit 1
            set interface "wan1"
            set gateway 172.16.20.2
        next
        edit 2
            set interface "wan2"
            set gateway 10.100.20.2
        next
    end
    config service
        edit 1
            set name "SIP"
            set priority-members 1
            set dst "voip-server"
            set dscp-forward enable
            set dscp-forward-tag 101110
        next
    end
end
```

If no SD-WAN zone is specified, members are added to the default *virtual-wan-link* zone.

To use the diagnose command to check if specific traffic is attached to the correct traffic shaper:

```
# diagnose firewall iprope list 100015

policy index=1 uuid_idx=0 action=accept
flag (0):
shapers: orig=high-priority(2/0/134217728) reply=high-priority(2/0/134217728)
cos_fwd=0  cos_rev=0
group=00100015 av=00000000 au=00000000 split=00000000
host=0 chk_client_info=0x0 app_list=0 ips_view=0
misc=0 dd_type=0 dd_mode=0
zone(1): 0 -> zone(2): 36 38
source(1): 0.0.0.0-255.255.255.255, uuid_idx=6,
dest(1): 0.0.0.0-255.255.255.255, uuid_idx=6,
service(2):
        [6:0x0:0/(1,65535)->(80,80)] helper:auto
        [6:0x0:0/(1,65535)->(443,443)] helper:auto

policy index=2 uuid_idx=0 action=accept
flag (0):
shapers: orig=low-priority(4/128000/134217728) reply=low-priority(4/128000/134217728)
cos_fwd=0  cos_rev=0
group=00100015 av=00000000 au=00000000 split=00000000
host=0 chk_client_info=0x0 app_list=0 ips_view=0
misc=0 dd_type=0 dd_mode=0
```

```
zone(1): 0 -> zone(2): 36 38
source(1): 0.0.0.0-255.255.255.255, uuid_idx=6,
dest(1): 0.0.0.0-255.255.255.255, uuid_idx=6,
service(3):
        [6:0x0:0/(1,65535)->(21,21)] helper:auto
        [6:0x0:0/(1,65535)->(21,21)] helper:auto
        [6:0x0:0/(1,65535)->(21,21)] helper:auto
```

To use the diagnose command to check if the correct traffic shaper is applied to the session:

```
# diagnose sys session list
session info: proto=6 proto_state=01 duration=11 expire=3599 timeout=3600 flags=00000000
sockflag=00000000 sockport=0 av_idx=0 use=5
origin-shaper=low-priority prio=4 guarantee 128000Bps max 1280000Bps traffic 1050Bps drops
0B
reply-shaper=
per_ip_shaper=
class_id=0 shaping_policy_id=2 ha_id=0 policy_dir=0 tunnel=/ helper=ftp vlan_cos=0/255
state=may_dirty npu npd os mif route_preserve
statistic(bytes/packets/allow_err): org=868/15/1 reply=752/10/1 tuples=2
tx speed(Bps/kbps): 76/0 rx speed(Bps/kbps): 66/0
orgin->sink: org pre->post, reply pre->post dev=39->38/38->39 gwy=172.16.200.55/0.0.0.0
hook=post dir=org act=snat 10.1.100.11:58241->172.16.200.55:21(172.16.200.1:58241)
hook=pre dir=reply act=dnat 172.16.200.55:21->172.16.200.1:58241(10.1.100.11:58241)
pos/(before,after) 0/(0,0), 0/(0,0)
misc=0 policy_id=1 auth_info=0 chk_client_info=0 vd=4
serial=0003255f tos=ff/ff app_list=0 app=0 url_cat=0
sdwan_mbr_seq=0 sdwan_service_id=0
rpdb_link_id = 00000000
dd_type=0 dd_mode=0
npu_state=0x100000
npu info: flag=0x00/0x00, offload=0/0, ips_offload=0/0, epid=0/0, ipid=0/0,
vlan=0x0000/0x0000
vlifid=0/0, vtag_in=0x0000/0x0000 in_npu=0/0, out_npu=0/0, fwd_en=0/0, qid=0/0
no_ofld_reason:  offload-denied helper
total session 1
```

To use the diagnose command to check the status of a shared traffic shaper:

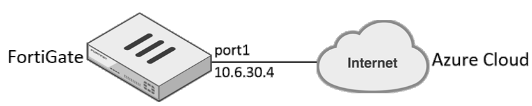
```
# diagnose firewall shaper traffic-shaper list

name high-priority
maximum-bandwidth 131072 KB/sec
guaranteed-bandwidth 0 KB/sec
current-bandwidth 0 B/sec
priority 2
tos ff
packets dropped 0
bytes dropped 0

name low-priority
maximum-bandwidth 131072 KB/sec
guaranteed-bandwidth 125 KB/sec
current-bandwidth 0 B/sec
priority 4
tos ff
```

```
packets dropped 0
bytes dropped 0

name high-priority
maximum-bandwidth 131072 KB/sec
guaranteed-bandwidth 0 KB/sec
current-bandwidth 0 B/sec
priority 2
policy 1
tos ff
packets dropped 0
bytes dropped 0

name low-priority
maximum-bandwidth 131072 KB/sec
guaranteed-bandwidth 125 KB/sec
current-bandwidth 0 B/sec
priority 4
policy 2
tos ff
packets dropped 0
bytes dropped 0
```

SDN dynamic connector addresses in SD-WAN rules

SDN dynamic connector addresses can be used in SD-WAN rules. FortiGate supports both public (AWS, Azure, GCP, OCI, AliCloud) and private (Kubernetes, VMware ESXi and NSX, OpenStack, ACI, Nuage) SDN connectors.

The configuration procedure for all of the supported SDN connector types is the same. This example uses an Azure public SDN connector.



There are four steps to create and use an SDN connector address in an SD-WAN rule:

1. Configure the FortiGate IP address and network gateway so that it can reach the Internet.
2. Create an Azure SDN connector.
3. Create a firewall address to associate with the configured SDN connector.
4. Use the firewall address in an SD-WAN service rule.

To create an Azure SDN connector:

1. Go to *Security Fabric > External Connectors*.
2. Click *Create New*.
3. In the *Public SDN* section, click *Microsoft Azure*.

4. Enter the following:

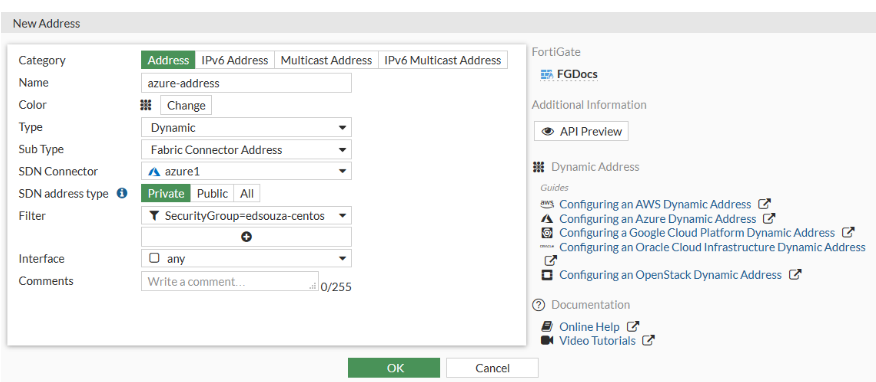
Name	azure1
Status	Enabled
Update Interval	Use Default
Server region	Global
Directory ID	942b80cd-1b14-42a1-8dcf-4b21dece61ba
Application ID	14dbd5c5-307e-4ea4-8133-68738141feb1
Client secret	xxxxxx
Resource path	disabled

5. Click *OK*.

To create a firewall address to associate with the configured SDN connector:

1. Go to *Policy & Objects > Addresses*.
2. Click *Create New > Address*.
3. Enter the following:

Category	Address
Name	azure-address
Type	Dynamic
Sub Type	Fabric Connector Address
SDN Connector	azure1
SDN address type	Private
Filter	SecurityGroup=edsouza-centos
Interface	Any



4. Click *OK*.

To use the firewall address in an SD-WAN service rule:

1. Go to *Network > SD-WAN*, select the *SD-WAN Rules* tab, and click *Create New*.
2. Set the *Name* to *Azure1*.
3. For the *Destination Address* select *azure-address*.
4. Configure the remaining settings as needed. See SD-WAN rules on page 538 for details.
5. Click *OK*.

Diagnostics

Use the following CLI commands to check the status of and troubleshoot the connector.

To see the status of the SDN connector:

```
# diagnose sys sdn status
SDN Connector           Type        Status          Updating        Last update
-----------------------------------------------------------------------------------------
azure1                  azure       connected       no              n/a
```

To debug the SDN connector to resolve the firewall address:

```
# diagnose debug application azd -1
    Debug messages will be on for 30 minutes.

    ...
    azd sdn connector azure1 start updating IP addresses
    azd checking firewall address object azure-address-1, vd 0
     IP address change, new list:
      10.18.0.4
      10.18.0.12
      ...
      ...

# diagnose sys sdwan service

    Service(2): Address Mode(IPV4) flags=0x0
      TOS(0x0/0x0), Protocol(0: 1->65535), Mode(manual)
      Service role: standalone
      Member sub interface:
      Members:
        1: Seq_num(1), alive, selected
      Dst address:
            10.18.0.4 - 10.18.0.4
            10.18.0.12 - 10.18.0.12
             ... ...
             ... ...
             ... ...
```

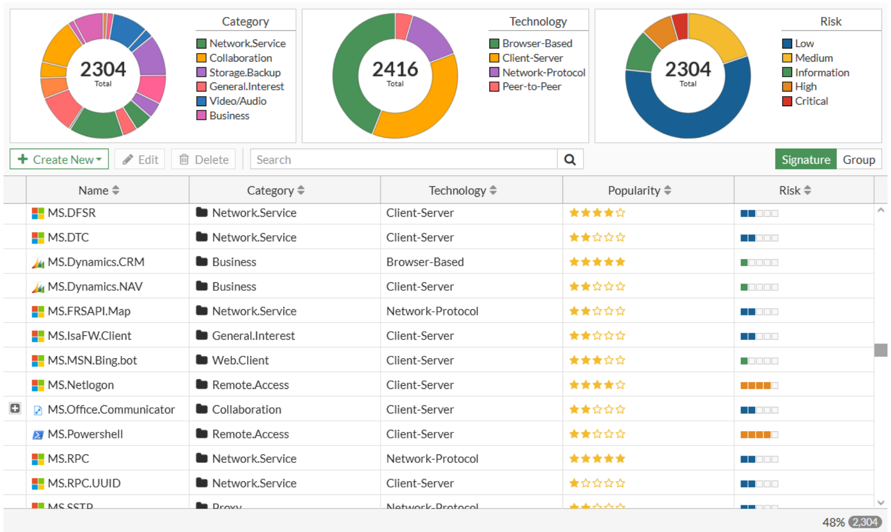
Application steering using SD-WAN rules

This topic covers how to use application steering in a topology with multiple WAN links. The following examples illustrate how to use different strategies to perform application steering to accommodate different business needs:

- Static application steering with a manual strategy on page 572
- Dynamic application steering with lowest cost and best quality strategies on page 575

Application matching

To apply application steering, SD-WAN service rules match traffic based on the applications that are in the application signature database. To view the signatures, go to *Security Profiles > Application Signatures* and select *Signature*.



On the first session that passes through, the IPS engine processes the traffic in the application layer to match it to a signature in the application signature database. The first session does not match any SD-WAN rules because the signature has not been recognized yet. When the IPS engine recognizes the application, it records the 3-tuple IP address, protocol, and port in the application control Internet Service ID list. To view the application and corresponding 3-tuple:

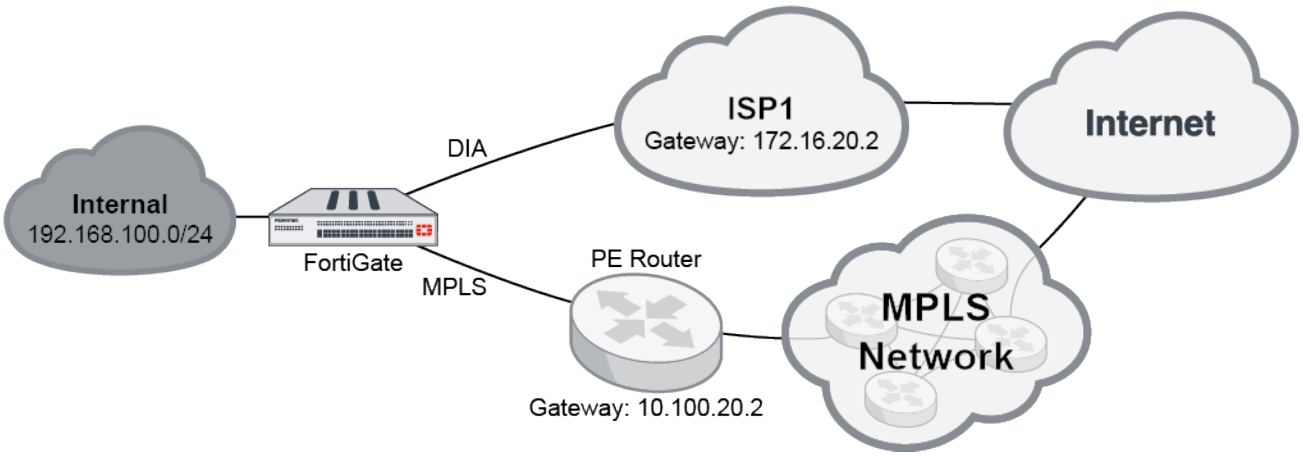
```
# diagnose sys sdwan internet-service-app-ctrl-list [app ID]
52.114.142.254
Microsoft.Teams(43541 4294837333): 52.114.142.254 6 443 Fri Jun 18 13:52:18 2021
```

The recognized application and 3-tuple stay in the application control list for future matches to occur. If there are no hits on the entry for eight hours, the entry is deleted.

For services with multiple IP addresses, traffic might not match the expected SD-WAN rule because the traffic is destined for an IP address that hat no previously been recognized by the FortiGate. The `diagnose sys sdwan internet-service-app-ctrl-list` command can be used to help troubleshoot such situations.

Static application steering with a manual strategy

This example covers a typical usage scenario where the SD-WAN has two members: MPLS and DIA. DIA is primarily used for direct internet access to internet applications, such as Office365, Google applications, Amazon, and Dropbox. MPLS is primarily used for SIP, and works as a backup when DIA is not working.



This example configures all SIP traffic to use MPLS while all other traffic uses DIA. If DIA is not working, the traffic will use MPLS.

To configure an SD-WAN rule to use SIP and DIA in the GUI:

1. Add port1 (DIA) and port2 (MPLS) as SD-WAN members, and configure a static route. See Configuring the SD-WAN interface on page 488 for details.
2. Create a firewall policy with an *Application Control* profile configured. See Configuring firewall policies for SD-WAN on page 490 for details.
3. Go to *Network > SD-WAN*, select the *SD-WAN Rules* tab, and click *Create New*.
4. Enter a name for the rule, such as *SIP*.
5. Click the *Application* field and select the applicable SIP applications from the *Select Entries* panel.
6. Under *Outgoing Interfaces*, select *Manual*.
7. For *Interface preference*, select *MPLS*.
8. Click *OK*.
9. Click *Create New* to create another rule.
10. Enter a name for the rule, such as *Internet*.
11. Click the *Address* field and select *all* from the panel.
12. Under *Outgoing Interfaces*, select *Manual*.
13. For *Interface preference*, select *DIA*.
14. Click *OK*.

To configure the firewall policy using the CLI:

```
config firewall policy
    edit 1
        set name "1"
        set srcintf "dmz"
        set dstintf "virtual-wan-link"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set fsso disable
```

```
        set application-list "default"
        set ssl-ssh-profile "certificate-inspection"
        set nat enable
    next
end
```

To configure an SD-WAN rule to use SIP and DIA using the CLI:

```
config system sdwan
    set status enable
    config members
        edit 1
            set interface "MPLS"
        next
        edit 2
            set interface "DIA"
        next
    end
    config service
        edit 1
            set name "SIP"
            set internet-service enable
            set internet-service-app-ctrl 34640 152305677 38938 26180 26179 30251
            set priority-members 2
        next
        edit 2
            set name "Internet"
            set dst "all"
            set priority-members 1
        next
    end
end
```

All SIP traffic uses MPLS. All other traffic goes to DIA. If DIA is broken, the traffic uses MPLS. If you use VPN instead of MPLS to run SIP traffic, you must configure a VPN interface, for example vpn1, and then replace member 1 from MPLS to vpn1 for SD-WAN member.

If no SD-WAN zone is specified, members are added to the default *virtual-wan-link* zone.

To use the diagnose command to check performance SLA status using the CLI:

```
# diagnose sys sdwan service 1

Service(1): Address Mode(IPV4) flags=0x0

TOS(0x0/0x0), Protocol(0: 1->65535), Mode(manual)
Members:<<BR>>

1: Seq_num(1), alive, selected

Internet Service: SIP(4294836224 34640) SIP.Method(4294836225 152305677) SIP.Via.NAT
```

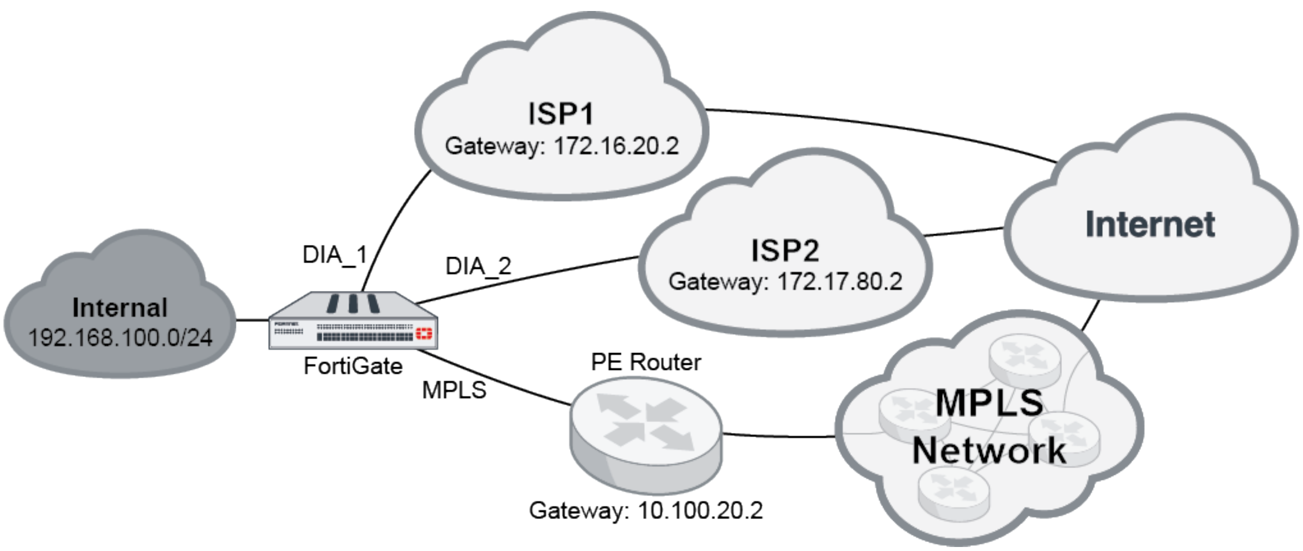
```
(4294836226 38938) SIP_Media.Type.Application(4294836227 26180) SIP_Message(4294836228
26179) SIP_Voice(4294836229 30251)
```

```
# diagnose sys sdwan service 2

Service(2): Address Mode(IPV4) flags=0x0

TOS(0x0/0x0), Protocol(0: 1->65535), Mode(manual)
Members:<<BR>>

1: Seq_num(2), alive, selected

Dst address: 0.0.0.0-255.255.255.255
```

```
# diagnose sys sdwan internet-service-app-ctrl-list
Ctrl application(SIP 34640):Internet Service ID(4294836224)
Ctrl application(SIP.Method 152305677):Internet Service ID(4294836225)
Ctrl application(SIP.Via.NAT 38938):Internet Service ID(4294836226)
Ctrl application(SIP_Media.Type.Application 26180):Internet Service ID(4294836227)
Ctrl application(SIP_Message 26179):Internet Service ID(4294836228)
Ctrl application(SIP_Voice 30251):Internet Service ID(4294836229)
```

Dynamic application steering with lowest cost and best quality strategies

In this example, the SD-WAN has three members: two ISPs (DIA_1 and DIA_2) that are used for access to internet applications, and an MPLS link that is used exclusively as a backup for business critical applications.



Business applications, such as Office365, Google, Dropbox, and SIP, use the *Lowest Cost (SLA)* strategy to provide application steering, and traffic falls back to MPLS only if both ISP1 and ISP2 are down. Non-business applications, such as Facebook and Youtube, use the *Best Quality* strategy to choose between the ISPs.

To configure the SD-WAN members, static route, and firewall policy in the GUI:

1. Add port1 (DIA_1), port2 (DIA_2), and port3 (MPLS) as SD-WAN members. Set the cost of DIA_1 and DIA_2 to *0*, and MPLS to *20*. See Configuring the SD-WAN interface on page 488 for details.
2. Configure a static route. See Adding a static route on page 489 for details.

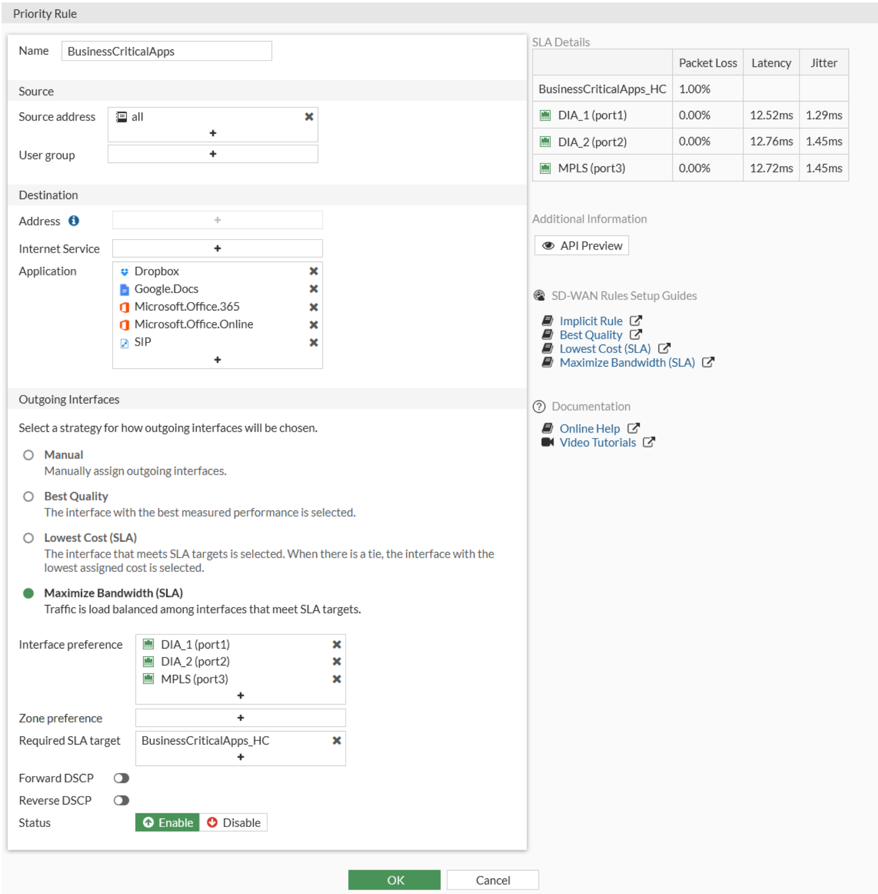
3. Create a firewall policy to allow traffic out on SD-WAN, with an *Application Control* profile configured. See Configuring firewall policies for SD-WAN on page 490 for details.

To configure the SD-WAN rule and performance SLA checks for business critical application in the GUI:

1. Go to *Network > SD-WAN*, select the *SD-WAN Rules* tab, and click *Create New*.
2. Set the name to *BusinessCriticalApps*.
   This rule will steer your business critical traffic to the appropriate link based on the *Lowest Cost (SLA)*.
3. Set *Source address* to *all*.
4. Under *Destination*, set *Application* to your required applications. In this example: *Microsoft.Office.365*, *Microsoft.Office.Online*, *Google.Docs*, *Dropbox*, and *SIP*.
5. Under *Outgoing Interfaces*, select *Lowest Cost (SLA)*.
   The lowest cost is defined in the SD-WAN member interface settings (see Configuring the SD-WAN interface on page 488). The lowest possible cost is *0*, which represents the most preferred link. In this example, DIA_1 and DIA_2 both have a cost of *0*, while MPLS has a cost of *20* because it is used for backup.
6. In *Interface preference*, add the interfaces in order of preference when the cost of the links is tied. In this example, DIA_1, DIA_2, then MPLS.
   MPLS will always be chosen last, because it has the highest cost. DIA_1 and DIA_2 have the same cost, so an interface is selected based on their order in the *Interface preference* list.
7. Set *Required SLA target* to ensure that only links that pass your SLA target are chosen in this SD-WAN rule:
   a. Click in the *Required SLA target* field.
   b. In the *Select Entries* pane, click *Create*. The *New Performace SLA* pane opens.
   c. Set *Name* to *BusinessCriticalApps_HC*.
      This health check is used for business critical applications in your SD-WAN rule.
   d. Leave *Protocol* set to *Ping*, and add up to two servers, such as *office.com* and *google.com*.
   e. Set *Participants* to *Specify*, and add all three interfaces: DIA_1, DIA_2, and MPLS.
   f. Enable *SLA Target*.
      The attributes in your target determine the quality of your link. The SLA target of each link is compared when determining which link to use based on the lowest cost. Links that meet the SLA target are preferred over links that fail, and move to the next step of selection based on cost. If no links meet the SLA target, then they all move to the next step.

      In this example, disable *Latency threshold* and *Jitter threshold*, and set *Packet loss threshold* to *1*.
   g. Click *OK*.
   h. Select the new performance SLA to set it as the *Required SLA target*.

   When multiple SLA targets are added, you can choose which target to use in the SD-WAN rule.



8. Click *OK* to create the SD-WAN rule.

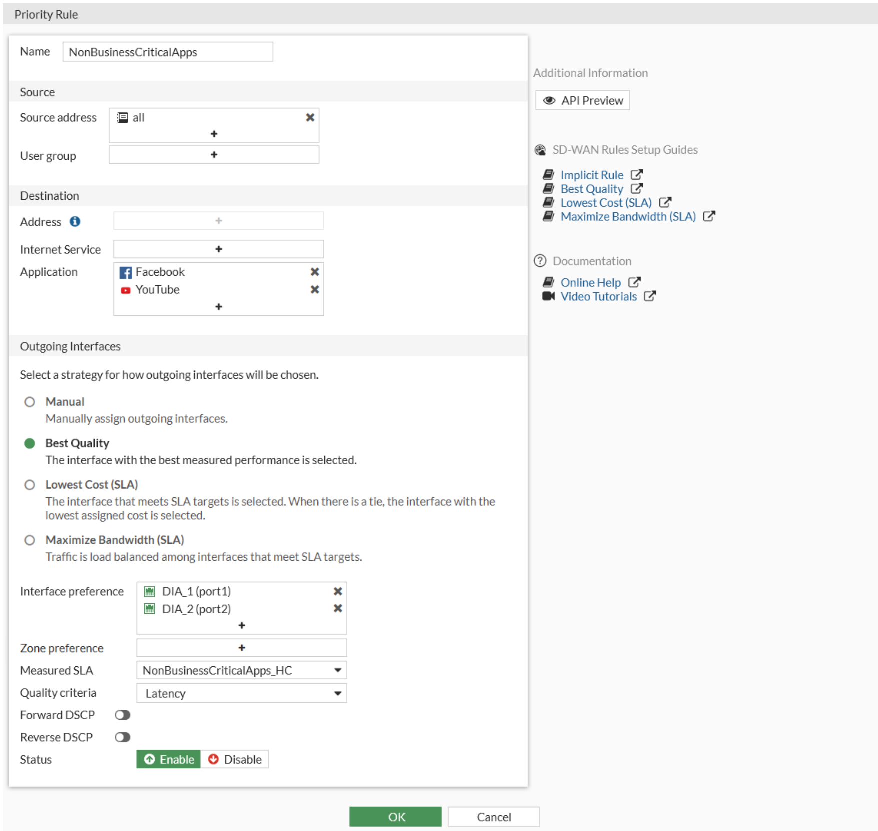
To configure the SD-WAN rule and performance SLA checks for non-business critical application in the GUI:

1. Go to *Network > SD-WAN*, select the *SD-WAN Rules* tab, and click *Create New*.
2. Set the name to *NonBusinessCriticalApps*.
   This rule will steer your non-business critical traffic to the appropriate link based on the *Best Quality*. No SLA target must be met, as the best link is selected based on the configured quality criteria and interface preference order.
3. Set *Source address* to *all*.
4. Under *Destination*, set *Application* to your required applications. In this example: *Facebook*, and *Youtube*.
5. Under *Outgoing Interfaces*, select *Best Quality*.
6. In *Interface preference*, add the interfaces in order of preference.
   By default, a more preferred link has an advantage of 10% over a less preferred link. For example, when latency is used, the preferred link's calculated latency = real latency / (1+10%).

The preferred link advantage can be customized in the CLI when the mode is `priority` (*Best Quality*) or `auto`:

```
config system sdwan
    config service
        edit <id>
            set link-cost-threshold <integer>
        next
    end
end
```

7. Create and apply a new performance SLA profile:
   a. Click in the *Measured SLA* field.
   b. In the drop-down list, click *Create*. The *New Performace SLA* pane opens.
   c. Set *Name* to *NonBusinessCritical_HC*.
      This health check is used for non-business critical applications in your SD-WAN rule.
   d. Leave *Protocol* set to *Ping*, and add up to two servers, such as *youtube.com* and *facebook.com*.
   e. Set *Participants* to *Specify*, and add the DIA_1 and DIA_2 interfaces. In this example, MPLS is not used for non-business critical applications.
   f. Leave *SLA Target* disabled.
   g. Click *OK*.
   h. Select the new performance SLA from the list to set it as the *Measured SLA*.
8. Set *Quality criteria* as required. In this example, *Latency* is selected.
   For bandwidth related criteria, such as *Downstream*, *Upstream*, and *Bandwidth* (bi-directional), the selection is based on available bandwidth. An estimated bandwidth should be configured on the interface to provide a baseline, maximum available bandwidth.



9. Click *OK* to create the SD-WAN rule.

To configure the SD-WAN members, static route, and firewall policy in the CLI:

1. Configure the interfaces:

```
config system interface
    edit "port1"
        set ip <class_ip&net_netmask>
        set alias "DIA_1"
        set role wan
    next
    edit "port2"
        set ip <class_ip&net_netmask>
        set alias "DIA_2"
        set role wan
    next
    edit "port3"
        set ip <class_ip&net_netmask>
        set alias "MPLS"
        set role wan
    next
end
```

2. Configure the SD-WAN members:

```
config system sdwan
    set status enable
    config members
```

```
        edit 1
            set interface "port1"
            set gateway 172.16.20.2
        next
        edit 2
            set interface "port2"
            set gateway 172.17.80.2
        next
        edit 3
            set interface "port3"
            set gateway 10.100.20.2
            set cost 20
        next
    end
end
```

If no SD-WAN zone is specified, members are added to the default *virtual-wan-link* zone.

3. Configure a static route. See Adding a static route on page 489 for details.
4. Create a firewall policy to allow traffic out on SD-WAN, with an *Application Control* profile configured. See Configuring firewall policies for SD-WAN on page 490 for details.

To configure the SD-WAN rule and performance SLA checks for business critical application in the CLI:

1. Configure the *BusinessCriticalApps_HC* health-check:

```
config system sdwan
    config health-check
        edit "BusinessCriticalApps_HC"
            set server "office.com" "google.com"
            set members 1 2 3
            config sla
                edit 1
                    set link-cost-factor packet-loss
                    set packetloss-threshold 1
                next
            end
        next
    end
end
```

2. Configure the *BusinessCriticalApps* service to use *Lowest Cost (SLA)*:

```
config system sdwan
    config service
        edit 1
            set name "BusinessCriticalApps"
            set mode sla
            set src "all"
            set internet-service enable
            set internet-service-app-ctrl 17459 16541 33182 16177 34640
            config sla
                edit "BusinessCriticalApps_HC"
                    set id 1
```

```
                next
            end
            set priority-members 1 2 3
        next
    end
end
```

To configure the SD-WAN rule and performance SLA checks for non-business critical application in the CLI:

1. Configure the *nonBusinessCriticalApps_HC* health-check:

```
config system sdwan
    config health-check
        edit "NonBusinessCriticalApps_HC"
            set server "youtube.com" "facebook.com"
            set members 1 2
        next
    end
end
```

2. Configure the *NonBusinessCriticalApps* service to use *Lowest Cost (SLA)*:

```
config system sdwan
    config service
        edit 4
            set name "NonBusinessCriticalApps"
            set mode priority
            set src "all"
            set internet-service enable
            set internet-service-app-ctrl 15832 31077
            set health-check "NonBusinessCriticalApps_HC"
            set priority-members 1 2
        next
    end
end
```

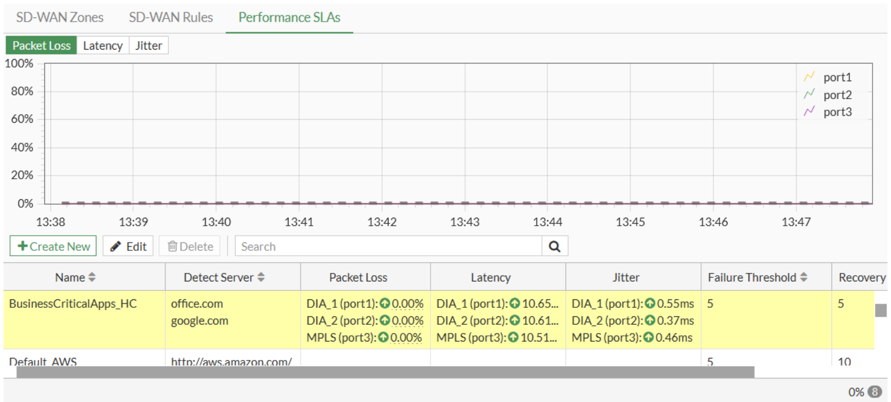
Verification

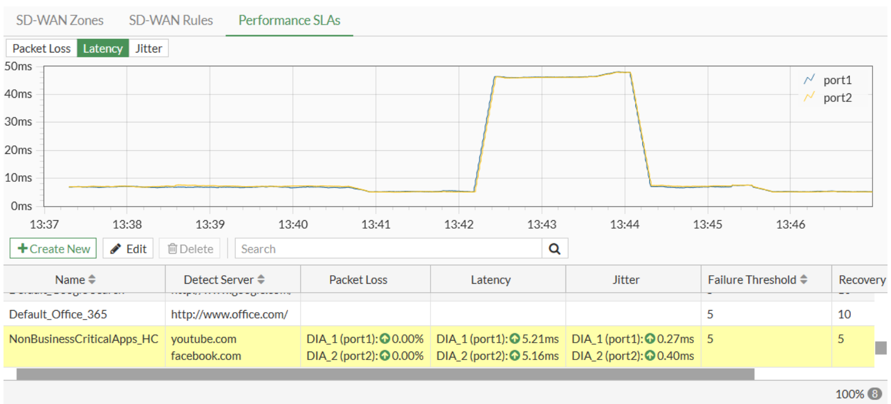
Check the following GUI pages, and run the following CLI commands to confirm that your traffic is being steered by the SD-WAN rules.

Health checks

To verify the status of each of the health checks in the GUI:

1. Go to *Network* > *SD-WAN*, select the *Performance SLAs* tab, and select each of the health checks from the list.





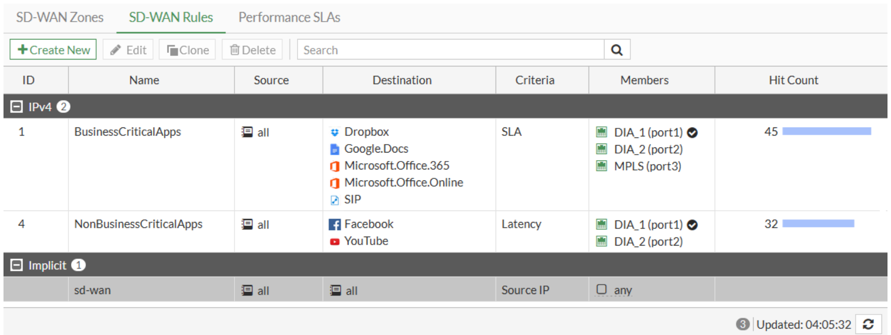
To verify the status of each of the health checks in the CLI:

```
# diagnose sys sdwan health-check
Health Check(BusinessCritical_HC):
Seq(1 port1): state(alive), packet-loss(0.000%) latency(12.884), jitter(0.919) sla_map=0x1
Seq(2 port2): state(alive), packet-loss(0.000%) latency(13.018), jitter(0.723) sla_map=0x1
Seq(3 port3): state(alive), packet-loss(0.000%) latency(13.018), jitter(0.923) sla_map=0x1
Health Check(NonBusinessCritical_HC):
Seq(1 port1): state(alive), packet-loss(0.000%) latency(6.888), jitter(0.953) sla_map=0x0
Seq(2 port2): state(alive), packet-loss(0.000%) latency(6.805), jitter(0.830) sla_map=0x0
```

Rule members and hit count

To verify the active members and hit count of the SD-WAN rule in the GUI:

1. Go to *Network > SD-WAN* and select the *SD-WAN Rules* tab.



The interface that is currently selected by the rule has a checkmark next to its name in the *Members* column. Hover the cursor over the checkmark to open a tooltip that gives the reason why that member is selected. If multiple members are selected, only the highest ranked member is highlighted (unless the mode is *Maximize Bandwidth (SLA)*).

To verify the active members and hit count of the SD-WAN rule in the CLI:

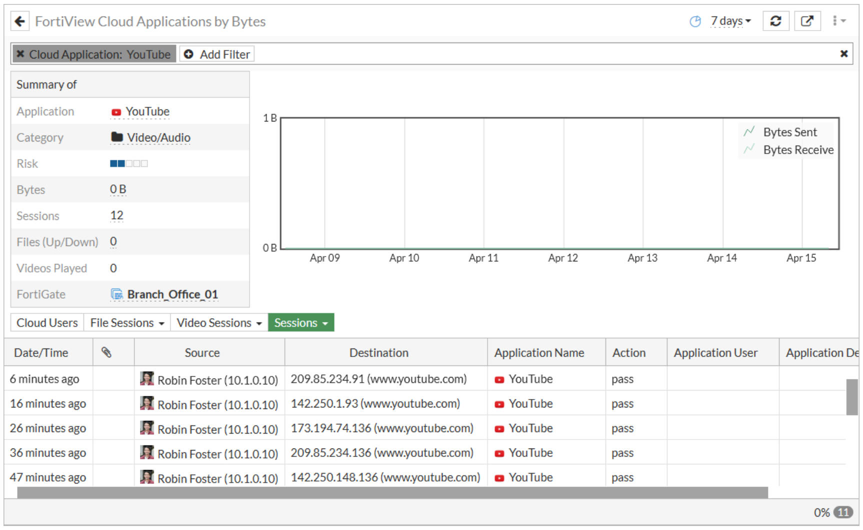
```
# diagnose sys sdwan service

Service(3): Address Mode(IPV4) flags=0x0
  Gen(13), TOS(0x0/0x0), Protocol(0: 1->65535), Mode(sla), sla-compare-order
  Members:
    1: Seq_num(1 port1), alive, sla(0x1), cfg_order(0), cost(0), selected
    2: Seq_num(2 port2), alive, sla(0x1), cfg_order(1), cost(0), selected
    3: Seq_num(3 port3), alive, sla(0x1), cfg_order(2), cost(20), selected
  Internet Service: Dropbox(4294836727,0,0,0 17459) Google.Docs(4294836992,0,0,0 16541)
Microsoft.Office.365(4294837472,0,0,0 33182) Microsoft.Office.Online(4294837475,0,0,0 16177)
SIP(4294837918,0,0,0 34640)
  Src address:
        0.0.0.0-255.255.255.255

Service(4): Address Mode(IPV4) flags=0x0
  Gen(211), TOS(0x0/0x0), Protocol(0: 1->65535), Mode(priority), link-cost-factor(latency),
link-cost-threshold(10), heath-check(NonBusinessCritical_HC)
  Members:
    1: Seq_num(1 port1), alive, latency: 5.712, selected
    2: Seq_num(2 port2), alive, latency: 5.511, selected
  Internet Service: Facebook(4294836806,0,0,0 15832) YouTube(4294838537,0,0,0 31077)
  Src address:
        0.0.0.0-255.255.255.255
```

Applications and sessions

To verify sessions in FortiView:

1. Go to a dashboard and add the *FortiView Cloud Applications* widget sorted by bytes. See Cloud application view on page 124 for details.

2. Drill down on an application, such as *YouTube*, then select the *Sessions* tab.



To verify applications identified by Application Control in SD-WAN:

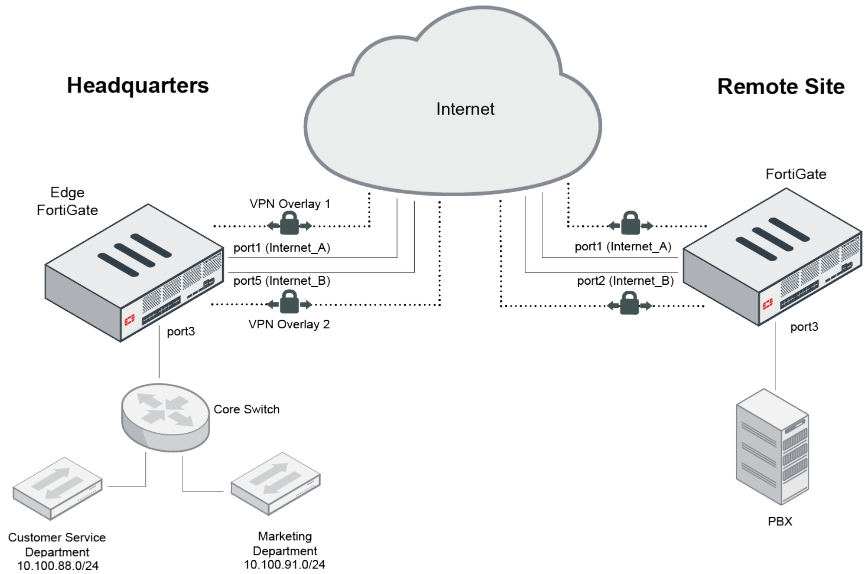
```
# diagnose sys sdwan internet-service-app-ctrl-list

Steam(16518 4294838108): 23.6.148.10 6 443 Thu Apr 15 08:51:54 2021
Netflix(18155 4294837589): 54.160.93.182 6 443 Thu Apr 15 09:13:25 2021
Netflix(18155 4294837589): 54.237.226.164 6 443 Thu Apr 15 10:04:37 2021
Minecraft(27922 4294837491): 65.8.232.41 6 443 Thu Apr 15 09:12:19 2021
Minecraft(27922 4294837491): 65.8.232.46 6 443 Thu Apr 15 09:02:07 2021
Minecraft(27922 4294837491): 99.84.244.51 6 443 Thu Apr 15 10:23:57 2021
Minecraft(27922 4294837491): 99.84.244.63 6 443 Thu Apr 15 10:03:30 2021
YouTube(31077 4294838537): 74.125.69.93 6 443 Thu Apr 15 08:52:59 2021
YouTube(31077 4294838537): 108.177.112.136 6 443 Thu Apr 15 09:33:53 2021
YouTube(31077 4294838537): 142.250.1.93 6 443 Thu Apr 15 10:35:13 2021
...
```

DSCP tag-based traffic steering in SD-WAN

This document demonstrates the Differentiated Services Code Point (DSCP) tag-based traffic steering in Fortinet secure SD-WAN. You can use this guide as an example to deploy DSCP tag-based traffic steering in Fortinet secure SD-WAN.

DSCP tags are often used to categorize traffic to provide quality of service (QoS). Based on DSCP tags, you can provide SD-WAN traffic steering on an edge device.

In this example, we have two different departments at the Headquarters site - Customer Service and Marketing. Traffic from each of these departments is marked with separate DSCP tags by the core switch, and passes through the core switch to the edge FortiGate. The edge FortiGate reads the DSCP tags and steers traffic to the preferred interface based on the defined SD-WAN rules.



In our example, we consider two types of traffic - social media traffic and VoIP traffic. VoIP traffic from Customer Service is considered to be more important than social media traffic. Each of these traffic types is marked with a DSCP tag by the core switch - VoIP traffic is marked with the DSCP tag of `011100`, and social media traffic is marked with the DSCP tag of `001100`. The DSCP tagged traffic is then passed on to the edge FortiGate. The edge FortiGate identifies the DSCP tagged traffic and based on the defined SD-WAN rules, the edge FortiGate steers:

- VoIP traffic to the preferred VPN overlay with the least jitter in order to provide the best quality of voice communication with the remote VoIP server (PBX)
- Social media traffic to the preferred Internet link with a lower cost (less expensive and less reliable)

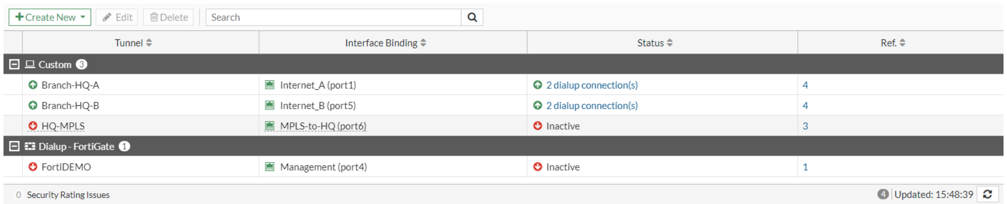
If you are familiar with SD-WAN configurations in FortiOS, you can directly jump to the Configuring SD-WAN rules on page 587 section to learn how to configure the SD-WAN rules to perform traffic steering. Otherwise, you can proceed with all of the following topics to configure the edge FortiGate:

- Configuring IPsec tunnels on page 585
- Configuring SD-WAN zones on page 586
- Configuring firewall policies on page 586
- Configuring Performance SLA test on page 587
- Configuring SD-WAN rules on page 587
- Results on page 590

Configuring IPsec tunnels

In our example, we have two interfaces `Internet_A (port1)` and `Internet_B(port5)` on which we have configured IPsec tunnels `Branch-HQ-A` and `Branch-HQ-B` respectively. To learn how to configure IPsec tunnels, refer to the IPsec VPNs on page 1328 section.

After you have configured the IPsec tunnels, go to *VPN > IPsec Tunnels* to verify the IPsec tunnels.

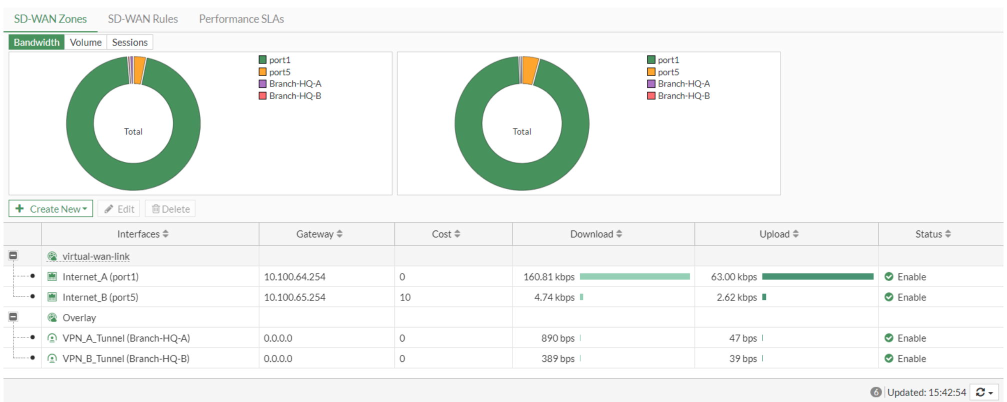


Configuring SD-WAN zones

In order for us to steer traffic based on SD-WAN rules, first we need to configure SD-WAN interface members and assign them to SD-WAN zones. For more information about SD-WAN zones, see SD-WAN zones on page 497.

In our example, we created two SD-WAN zones. The `virtual-wan-link` SD-WAN zone for the underlay traffic passing through the `Internet_A(port1)` and `Internet_B(port5)` interfaces, and the `Overlay` SD-WAN zone for the overlay traffic passing through the `Branch-HQ-A` and `Branch-HQ-B` interfaces.

Go to *Network > SD-WAN* and select the *SD-WAN Zones* tab to verify the configurations.



In this screenshot, we have configured the `Internet_A(port1)` and `Internet_B(port5)` SD-WAN interface members with their *Cost* values being `0` and `10` respectively. A lower *Cost* value indicates that this member is the primary interface member, and is preferred more than a member with a higher *Cost* value when using the *Lowest Cost (SLA)* strategy.

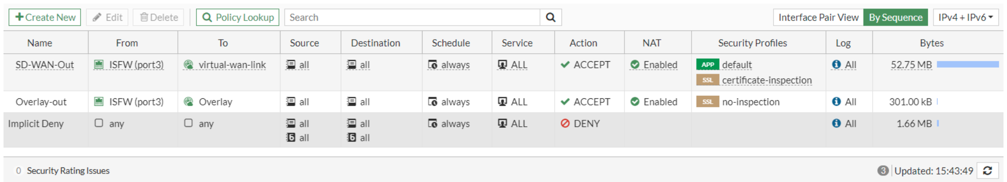
We also need to configure a static route that points to the *SD-WAN* interface. For more information static routes, see Adding a static route on page 489.

Configuring firewall policies

Configure firewall policies for both the overlay and underlay traffic. For more information about firewall policies, see Policies on page 758.

In this example, the *Overlay-out* policy governs the overlay traffic and the *SD-WAN-Out* policy governs the underlay traffic. The firewall policies are configured accordingly.

Once created, verify the firewall policies by navigating to *Policy & Objects > Firewall Policy*:

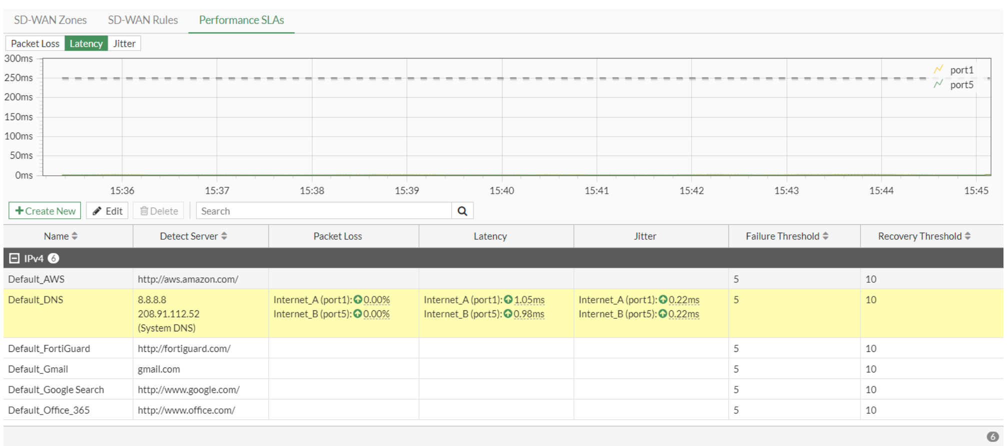


The *Security Profiles* column indicates that the *Overlay-out* firewall policy for the overlay traffic is set up to not scan any traffic, while the *SD-WAN-Out* firewall policy is set to scan all web traffic to identify and govern social media traffic as *Application Control* profile is active.

Configuring Performance SLA test

Configure a performance SLA test that will be tied to the SD-WAN interface members we created and assigned to SD-WAN zones. For more information about Performance SLA, see SLA targets example on page 515.

In this example, we created a *Performance SLA* test *Default_DNS* with `Internet_A(port1)` and `Internet_B (port5)` interface members as participants. We will use the created *Performance SLA* test to steer all web traffic passing through the underlays other than social media traffic based on the *Lowest Cost (SLA)* strategy.



Configuring SD-WAN rules

Configure SD-WAN rules to govern the steering of DSCP tag-based traffic to the appropriate interfaces. Traffic will be steered based on the *Criteria* configured as part of the SD-WAN rules configuration.

In our example, we configured three different SD-WAN rules to govern DSCP tagged traffic. We have one SD-WAN rule each for VoIP traffic, social media traffic (Facebook in this case), and all other web traffic. VoIP traffic is always steered to either of the two overlay SD-WAN zones - `VPN_A_tunnel(Branch-HQ-A)` or `VPN_B_tunnel(Branch-HQ-B)`. Similarly, social media traffic and other web traffic is always steered to either of the two underlay SD-WAN zones - `Internet_A(port1)` or `Internet_B(port5)`. The interface that is preferred by the system over another depends upon the *Criteria* configured in the SD-WAN rule definition.

We configured the following SD-WAN rules:

- SD-WAN rule for VoIP traffic on page 588
- SD-WAN rule for social media traffic on page 588
- SD-WAN rule for other web traffic on page 589

SD-WAN rule for VoIP traffic

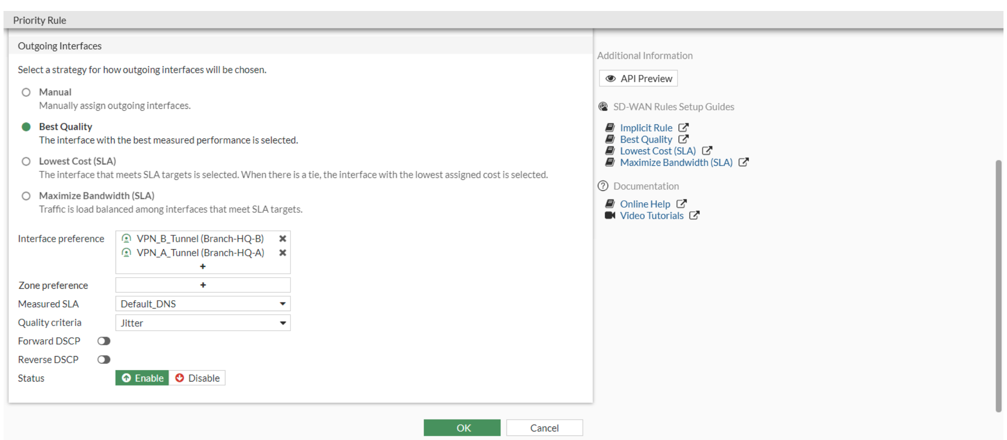
To configure SD-WAN rule for DSCP tagged VoIP traffic using the CLI:

```
config sys sdwan
    config service
        edit 5
            set name "VoIP-Steer"
            set mode priority
            set tos 0x70
            set tos-mask 0xf0
            set dst "all"
            set health-check "Default_DNS"
            set link-cost-factor jitter
            set priority-members 4 3
        next
    end
end
```

The `VoIP-Steer` SD-WAN rule configured above governs the DSCP tagged VoIP traffic.

DSCP values commonly are 6-bit binary numbers that are padded with zeros at the end. Therefore, in this example, VoIP traffic with DSCP tag `011100` will become `01110000`. This 8-bit binary number `01110000` is represented in its hexadecimal form `0x70` as the `tos` (Type of Service bit pattern) value. The `tos-mask` (Type of Service evaluated bits) hexadecimal value of `0xf0` (binary `11110000`) is used to check the four most significant bits from the `tos` value in this case. Hence, the first four bits of the `tos` (`0111`) will be used to match the first four bits of the DSCP tag in our policy above. Only the non-zero bit positions are used for comparison and the zero bit positions are ignored from the `tos-mask`.

We used the *Best Quality* strategy to define the *Criteria* to select the preferred interface from the overlay SD-WAN zone. With the *Best Quality* strategy selected, the interface with the best measured performance is selected. The system prefers the interface with the least *Jitter*.



For more information about configuring SD-WAN rules with the *Best Quality* strategy, see Best quality strategy on page 553.

SD-WAN rule for social media traffic

To configure SD-WAN rule for DSCP tagged social media traffic using the CLI:

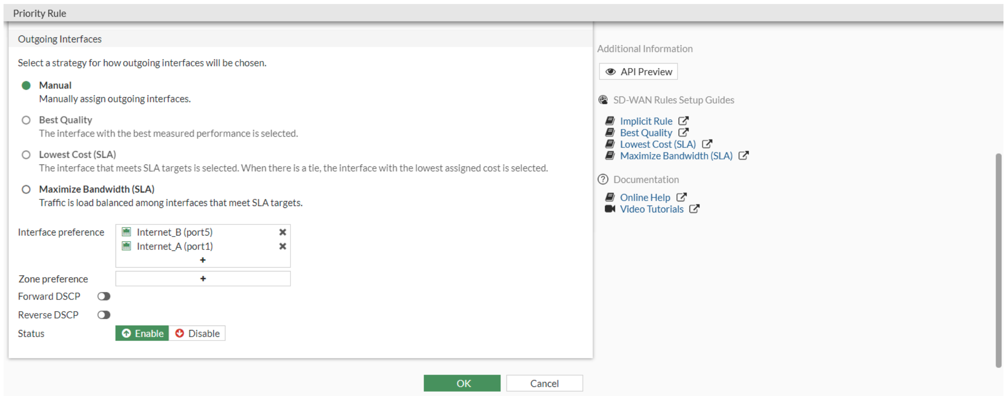
```
FortiGate # config sys sdwan
```

```
config service
   edit 3
      set name "Facebook-DSCP-steer"
      set tos 0x30
      set tos-mask 0xf0
      set dst "all"
      set priority-members 2 1
end
```

The `Facebook-DSCP-steer` SD-WAN rule configured above governs the DSCP tagged social media traffic.

DSCP values commonly are 6-bit binary numbers that are padded with zeros at the end. Therefore, in this example, social media traffic with DSCP tag `001100` will become `00110000`. This 8-bit binary number `00110000` is represented in its hexadecimal form `0x30` as the `tos` (Type of Service bit pattern) value. The `tos-mask` (Type of Service evaluated bits) hexadecimal value of `0xf0` (binary `11110000`) is used to check the four most significant bits from the `tos` value in this case. Hence, the first four bits of the `tos` (`0011`) will be used to match the first four bits of the DSCP tag in our policy above. Only the non-zero bit positions are used for comparison and the zero bit positions are ignored from the `tos-mask`.

We used a manual strategy to select the preferred interface from the underlay SD-WAN zone. We manually select the preferred interface as *Internet_B(port5)* to steer all social media traffic to.



For more information about configuring SD-WAN rules with static application steering with a manual strategy, see Static application steering with a manual strategy on page 572.

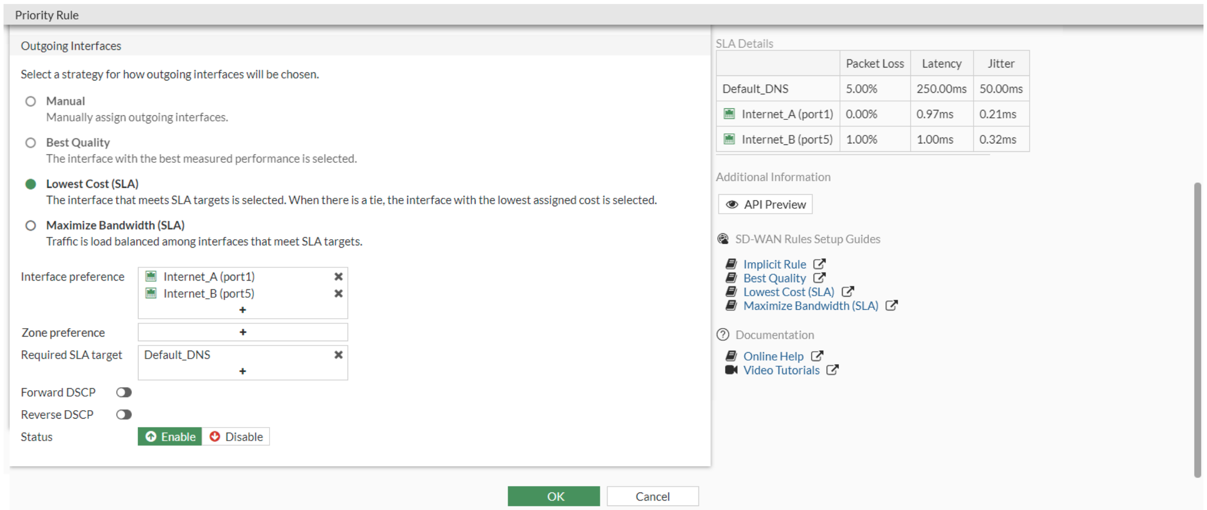
SD-WAN rule for other web traffic

To configure SD-WAN rule for all other web traffic using the CLI:

```
FortiGate # config sys sdwan
   config service
      edit 2
         set name "All-traffic"
         set mode sla
         set dst "all"
         config sla
            edit "Default_DNS"
               set id 1
            next
         end
         set priority-members 1 2
   end
```

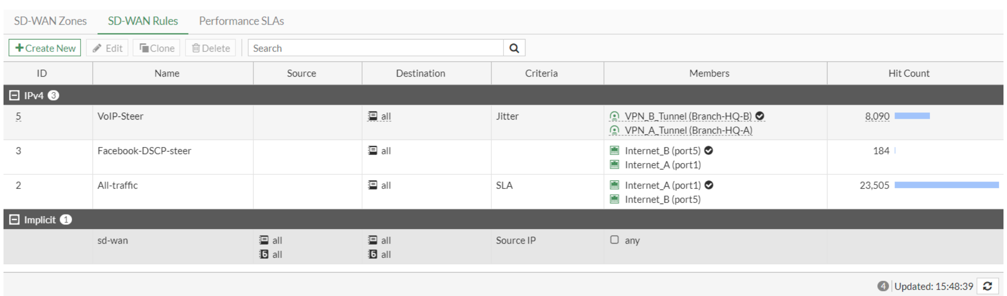
The `All-traffic` SD-WAN rule configured above governs all other web traffic.

We used the *Lowest Cost (SLA)* strategy to define the *Criteria* to select the preferred interface from the underlay SD-WAN zone. With the *Lowest Cost (SLA)* strategy selected, the interface that meets the defined *Performance SLA* targets (*Default_DNS* in our case) is selected. When there is a tie, the interface with the lowest assigned *Cost* (`Internet_A (port1)` in our case) is selected.



For more information about configuring SD-WAN rules with the *Lowest Cost (SLA)* strategy, see Lowest cost (SLA) strategy on page 557.

Once configured, verify your SD-WAN rules by navigating to *Network* > *SD-WAN* and selecting the *SD-WAN Rules* tab.



Results

The following sections show the function of the FortiGate and specifically of secure SD-WAN with respect to DSCP tagged traffic steering, and can be used to confirm that it is setup and running correctly:

- Verifying the DSCP tagged traffic on FortiGate on page 590
- Verifying service rules on page 592
- Verifying traffic steering as per the defined SD-WAN rules on page 593
- Verifying steered traffic leaving the required interface on page 593

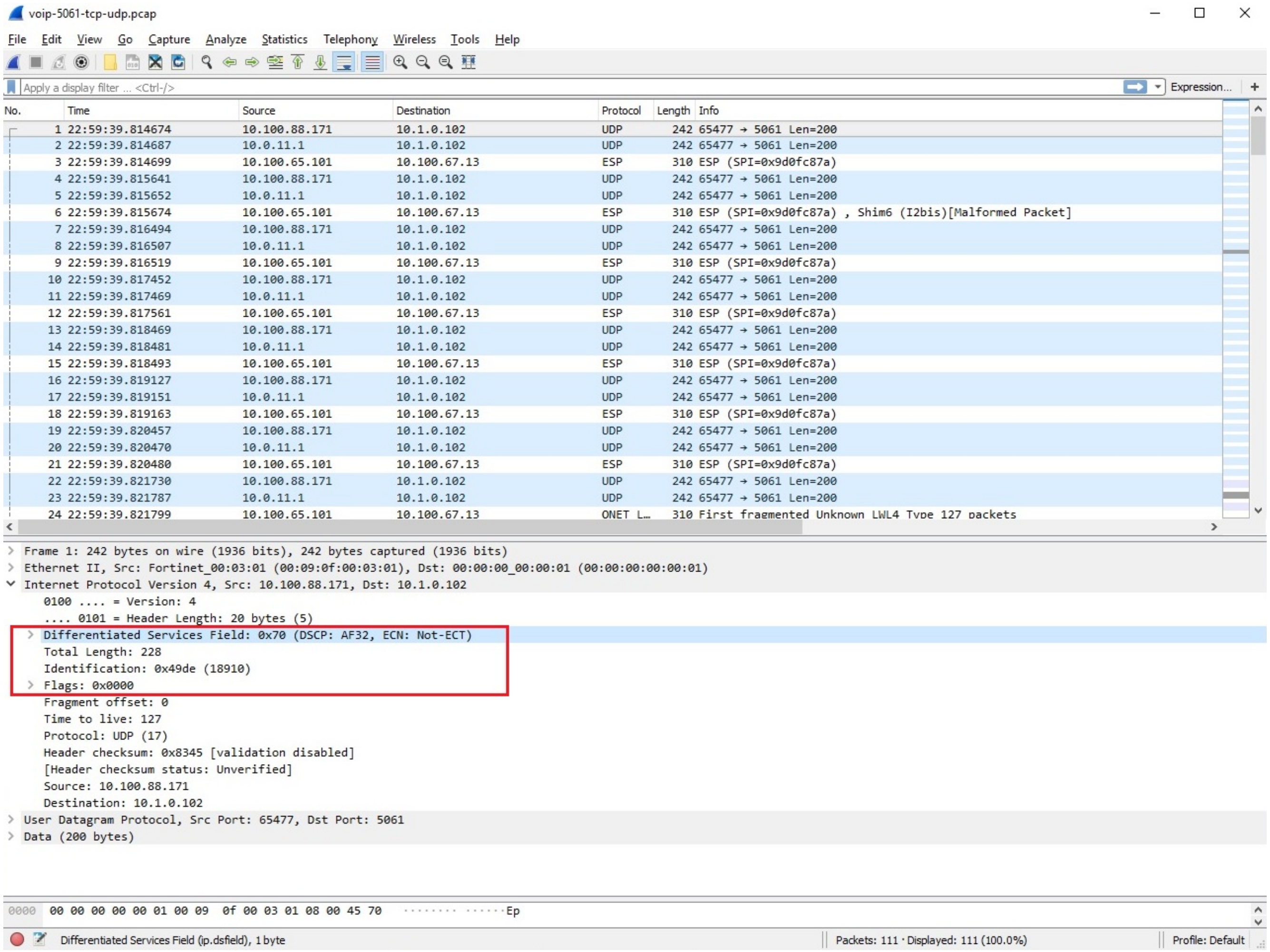
Verifying the DSCP tagged traffic on FortiGate

To verify the incoming DSCP tagged traffic, we used packet sniffing and converting the sniffed traffic to a desired format. For more information about packet sniffing, see Using the FortiOS built-in packet sniffer on the *Fortinet Knowledge Base*.

For VoIP traffic that is marked with DSCP tag `0x70`:

```
# diagnose sniffer packet any '(ip and ip[1] & 0xfc == 0x70)' 6 0 l
```

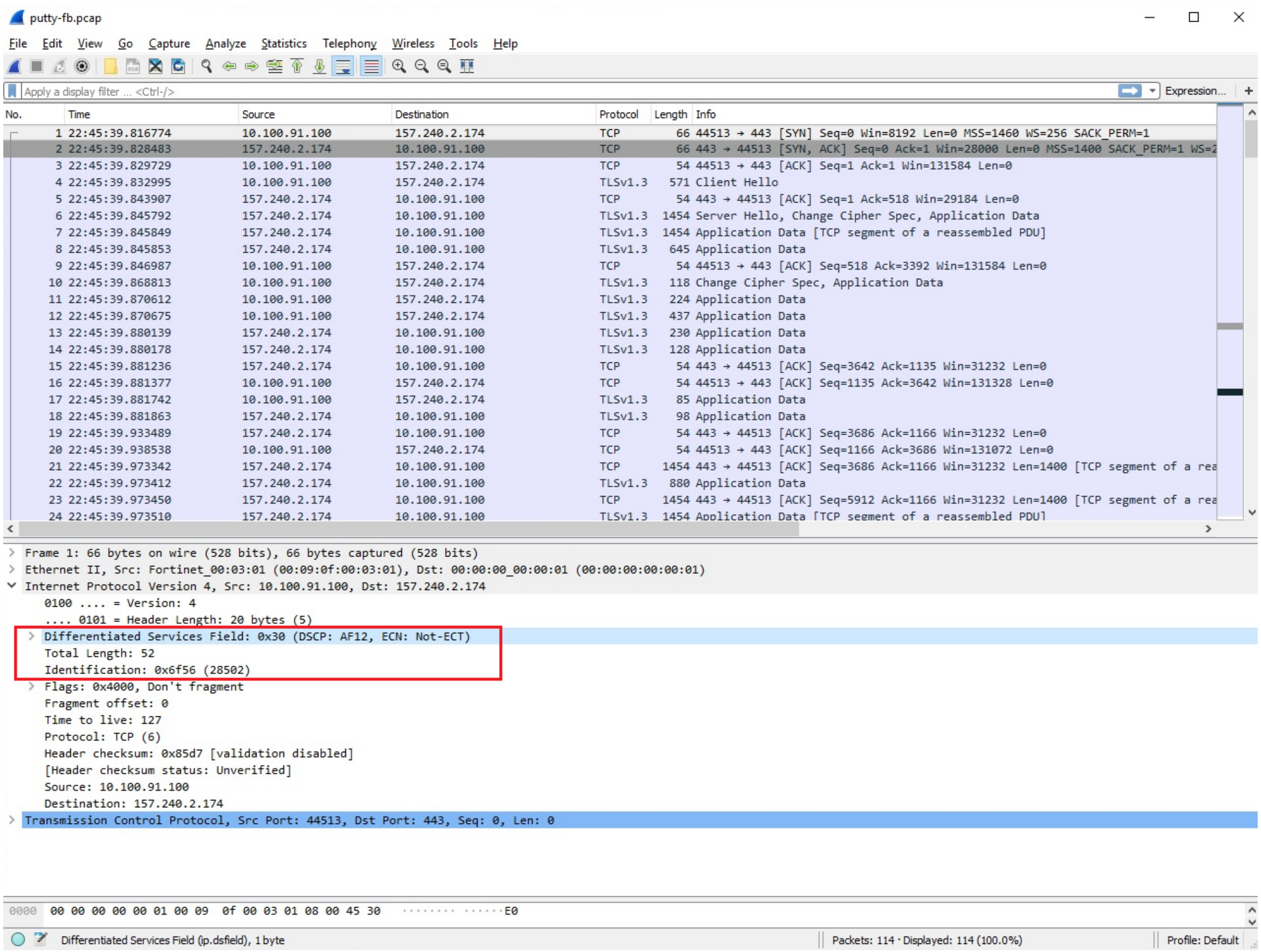
We used the open-source packet analyzer *Wireshark* to verify that VoIP traffic is tagged with the `0x70` DSCP tag.



For web traffic marked with DSCP tag `0x30`:

```
# diagnose sniffer packet any '(ip and ip[1] & 0xfc == 0x30)' 6 0 l
```

We used the open-source packet analyzer *Wireshark* to verify that web traffic is tagged with the `0x30` DSCP tag.



Verifying service rules

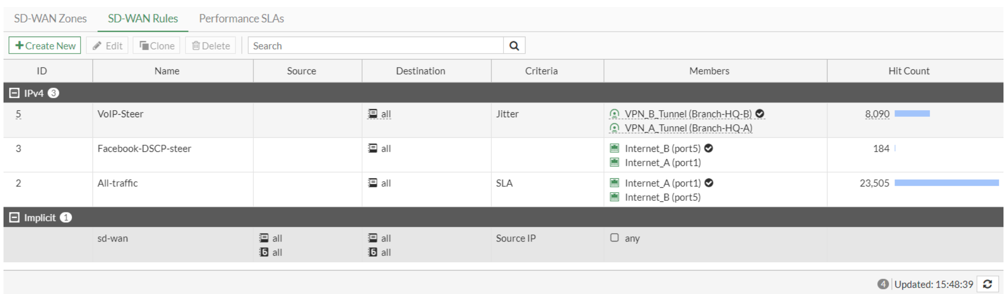
The following CLI commands show the appropriate DSCP tags and the corresponding interfaces selected by the SD-WAN rules to steer traffic:

```
# diagnose sys sdwan service

Service(5): Address Mode(IPV4) flags=0x0
   Gen(1), TOS(0x70/0xf0), Protocol(0: 1->65535), Mode(manual)
   Members:
      1: Seq_num(4 Branch-HQ-B), alive, selected
   Dst address:
         0.0.0.0-255.255.255.255

Service(3): Address Mode(IPV4) flags=0x0
   Gen(1), TOS(0x30/0xf0), Protocol(0: 1->65535), Mode(manual)
   Members:
      1: Seq_num(2 port5), alive, selected
   Dst address:
         0.0.0.0-255.255.255.255

Service(2): Address Mode(IPV4) flags=0x0
```

```
Gen(1), TOS(0x0/0x0), Protocol(0: 1->65535), Mode(sla), sla-compare-order
Members:
  1: Seq_num(1 port1), alive, sla(0x1), cfg_order(0), cost(0), selected
  2: Seq_num(2 port5), alive, sla(0x1), cfg_order(1), cost(10), selected
Dst address:
      0.0.0.0-255.255.255.255
```

Verifying traffic steering as per the defined SD-WAN rules

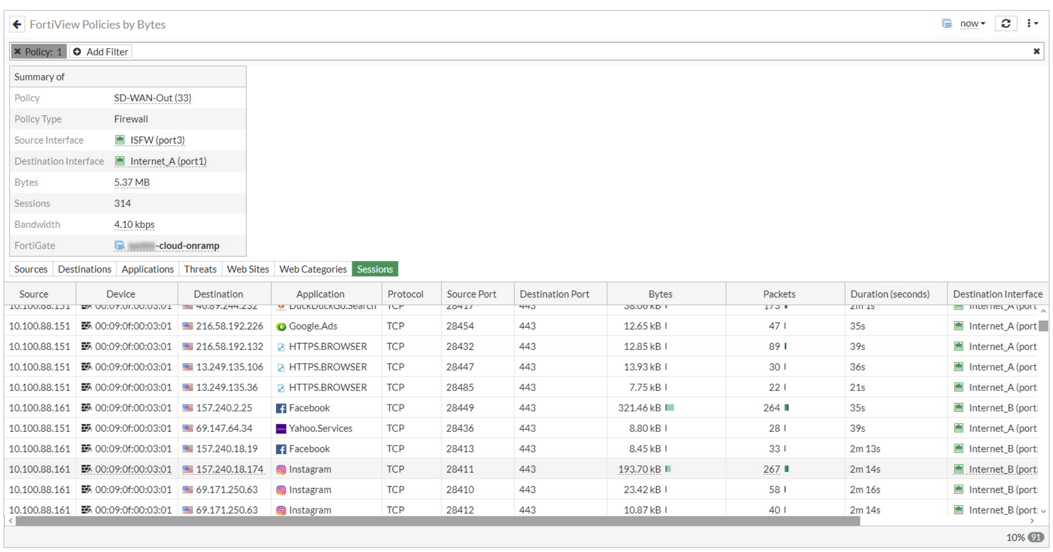
Go to *Network > SD-WAN* and select the *SD-WAN Rules* tab to review the *Hit Count* on the appropriate SD-WAN interfaces.



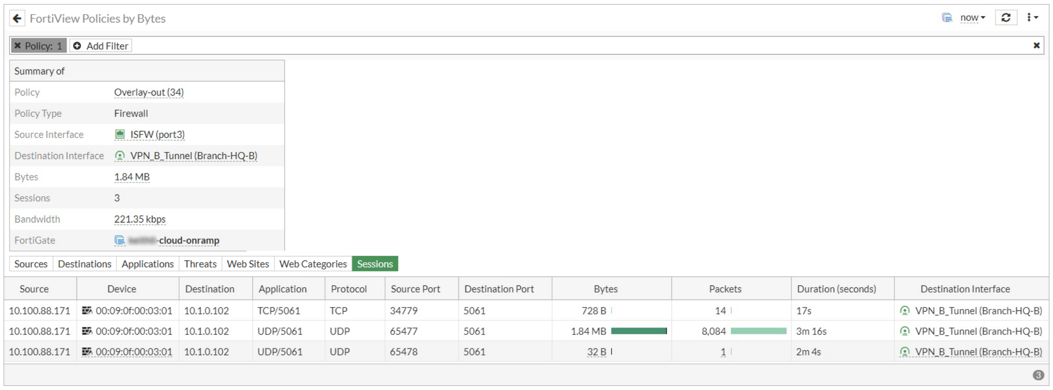
Verifying steered traffic leaving the required interface

Go to *Dashboard > Top Policies* to confirm that web traffic (port 443) flows through the right underlay interface members, and VoIP traffic flows through the right overlay interface member.

Web traffic leaves either `Interface_A(port1)` or `Interface_B(port5)`:



VoIP traffic leaves the preferred `VPN_B_Tunnel(Branch-HQ-B)` interface:



ECMP support for the longest match in SD-WAN rule matching

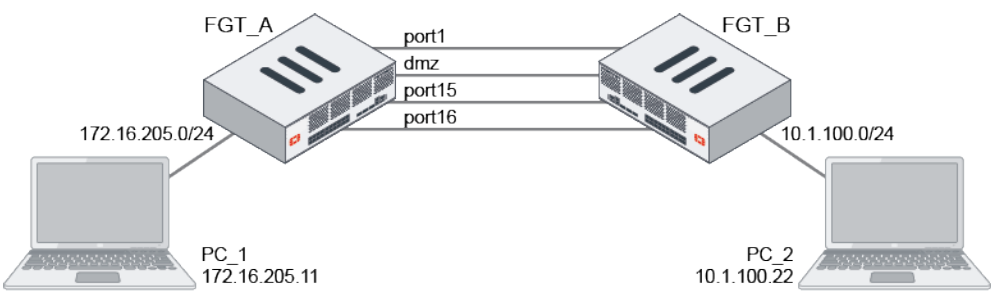
The longest match SD-WAN rule can match ECMP best routes. The rule will select the egress ports on ECMP specific routes, and not the less specific routes, to transport traffic.

The service mode determines which egress port on the ECMP specific routes is selected to forward traffic:

- Manual (`manual`): The first configured alive port is selected.
- Best Quality (`priority`): The best quality port is selected.
- Lowest Cost (`sla`): The first configured or lower cost port in SLA is selected.

Example

By default, SD-WAN selects the outgoing interface from all of the links that have valid routes to the destination. In some cases, it is required that only the links that have the best (or longest match) routes (single or ECMP) to the destination are considered.



In this example, four SD-WAN members in two zones are configured. The remote PC (PC_2 - 10.1.100.22) is accessible on port15 and port16, even though there are valid routes for all of the SD-WAN members. A single SD-WAN service rule is configured that allows traffic to balanced between all four of the members, but only chooses between port15 and port16 for the specific 10.1.100.22 address.

A performance SLA health check is configured to monitor 10.1.100.2. An SD-WAN service rule in Lowest Cost (SLA) mode is configured to select the best interface to steer the traffic. In the rule, the method of selecting a member if more than one meets the SLA (`tie-break`) is configured to select members that meet the SLA and match the longest prefix in the routing table (`fib-best-match`). If there are multiple ECMP routes with the same destination, the FortiGate will take the longest (or best) match in the routing table, and choose from those interface members.

To configure the SD-WAN:

```
config system sdwan
    config zone
        edit "virtual-wan-link"
        next
        edit "z1"
        next
    end
    config members
        edit 1
            set interface "port1"
            set gateway 172.16.200.2
        next
        edit 2
            set interface "dmz"
            set gateway 172.16.208.2
        next
        edit 3
            set interface "port15"
            set zone "z1"
            set gateway 172.16.209.2
        next
        edit 4
            set interface "port16"
            set zone "z1"
            set gateway 172.16.210.2
        next
    end
    config health-check
        edit "1"
            set server "10.1.100.2"
            set members 0
            config sla
                edit 1
                next
            end
        next
    end
    config service
        edit 1
            set name "1"
            set mode sla
            set dst "all"
            set src "172.16.205.0"
            config sla
                edit "1"
                    set id 1
                next
            end
            set priority-members 1 2 3 4
            set tie-break fib-best-match
        next
    end
end
```

To check the results:

1. The debug shows the SD-WAN service rule. All of the members meet SLA, and because no specific costs are attached to the members, the egress interface is selected based on the interface priority order that is configured in the rule:

```
FGT_A (root) # diagnose sys sdwan service

Service(1): Address Mode(IPV4) flags=0x200 use-shortcut-sla
  Gen(4), TOS(0x0/0x0), Protocol(0: 1->65535), Mode(sla), sla-compare-order
  Members(4):
    1: Seq_num(1 port1), alive, sla(0x1), gid(0), cfg_order(0), cost(0), selected
    2: Seq_num(2 dmz), alive, sla(0x1), gid(0), cfg_order(1), cost(0), selected
    3: Seq_num(3 port15), alive, sla(0x1), gid(0), cfg_order(2), cost(0), selected
    4: Seq_num(4 port16), alive, sla(0x1), gid(0), cfg_order(3), cost(0), selected
  Src address(1):
        172.16.205.0-172.16.205.255
  Dst address(1):
        0.0.0.0-255.255.255.255
```

2. The routing table shows that there are ECMP default routes on all of the members, and ECMP specific (or best) routes only on port15 and port16:

```
FGT_A (root) # get router info routing-table static
Routing table for VRF=0
S*      0.0.0.0/0 [1/0] via 172.16.200.2, port1
                          [1/0] via 172.16.208.2, dmz
                          [1/0] via 172.16.209.2, port15
                          [1/0] via 172.16.210.2, port16
S       10.1.100.22/32 [10/0] via 172.16.209.2, port15
                                 [10/0] via 172.16.210.2, port16
```

Because `tie-break` is set to `fib-best-match`, the first configured member from port15 and port16 is selected to forward traffic to PC_2. For all other traffic, the first configured member from all four of the interfaces is selected to forward traffic.

3. On PC-1, generate traffic to PC-2:

```
ping 10.1.100.22
```

4. On FGT_A, sniff for traffic sent to PC_2:

```
# diagnose sniffer packet any 'host 10.1.100.22' 4
interfaces=[any]
filters=[host 10.1.100.22]
2.831299 port5 in 172.16.205.11 -> 10.1.100.22: icmp: echo request
2.831400 port15 out 172.16.205.11 -> 10.1.100.22: icmp: echo request
```

Traffic is leaving on port15, the first configured member from port15 and port16.

Override quality comparisons in SD-WAN longest match rule matching

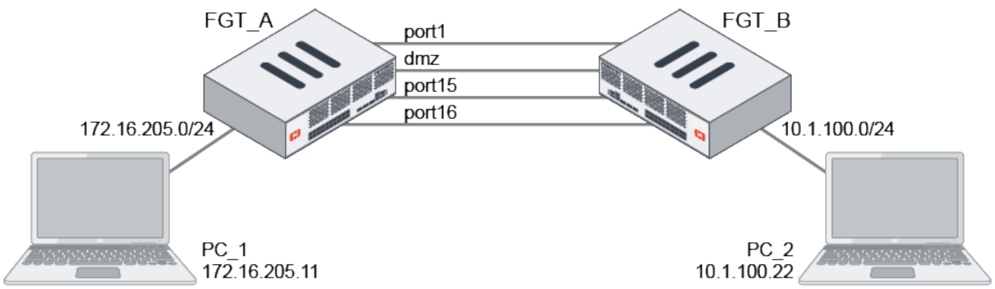
In SD-WAN rules, the longest match routes will override the quality comparisons when all of the specific routes are out of SLA.

With this feature in an SD-WAN rule:

- Lowest Cost (`sla`): Even though all of the egress ports on specific routes (longest matched routes) are out of SLA, the SD-WAN rule still selects the first configured or lower-cost port from the egress ports to forward traffic.
- Best Quality (`priority`): Even though the egress ports on specific routes (longest matched routes) have worse quality that all other ports on less specific routes, the SD-WAN rule still selects the best quality port from the ports on specific routes to forward traffic.

This features avoids a situation where, if the members on specific routes (longest matched routes) are out of SLA or have worse quality, the traffic might be forwarded to the wrong members in SLA (higher quality) on the default or aggregate routes.

Example



In this example, four SD-WAN members in two zones are configured. The remote PC (PC_2 - 10.1.100.22) is accessible on port15 and port16, even though there are valid routes for all of the SD-WAN members. A single SD-WAN service rule is configured that allows traffic to balanced between all four of the members, but only chooses between port15 and port16 for the specific 10.1.100.22 address. If neither port15 nor port16 meet the SLAs, traffic will be forwarded on one of these interfaces, instead of on port1 or dmz.

A performance SLA health check is configured to monitor 10.1.100.2. An SD-WAN service rule in Lowest Cost (SLA) mode is configured to select the best interface to steer the traffic. In the rule, the method of selecting a member if more than one meets the SLA (`tie-break`) is configured to select members that meet the SLA and match the longest prefix in the routing table (`fib-best-match`). If there are multiple ECMP routes with the same destination, the FortiGate will take the longest (or best) match in the routing table, and choose from those interface members.

To configure the SD-WAN:

```
config system sdwan
    config zone
        edit "virtual-wan-link"
        next
        edit "z1"
        next
    end
    config members
        edit 1
            set interface "port1"
            set gateway 172.16.200.2
        next
        edit 2
            set interface "dmz"
            set gateway 172.16.208.2
        next
        edit 3
```

```
            set interface "port15"
            set zone "z1"
            set gateway 172.16.209.2
        next
        edit 4
            set interface "port16"
            set zone "z1"
            set gateway 172.16.210.2
        next
    end
    config health-check
        edit "1"
            set server "10.1.100.2"
            set members 0
            config sla
                edit 1
                next
            end
        next
    end
    config service
        edit 1
            set name "1"
            set mode sla
            set dst "all"
            set src "172.16.205.0"
            config sla
                edit "1"
                    set id 1
                next
            end
            set priority-members 1 2 3 4
            set tie-break fib-best-match
        next
    end
end
```

To check the results:

1. The debug shows the SD-WAN service rule. Both port15 and port16 are up, but out of SLA:

```
FGT_A (root) # diagnose sys sdwan service
Service(1): Address Mode(IPV4) flags=0x200 use-shortcut-sla
  Gen(3), TOS(0x0/0x0), Protocol(0: 1->65535), Mode(sla), sla-compare-order
  Members(4):
    1: Seq_num(1 port1), alive, sla(0x1), gid(0), cfg_order(0), cost(0), selected
    2: Seq_num(2 dmz), alive, sla(0x1), gid(0), cfg_order(1), cost(0), selected
    3: Seq_num(3 port15), alive, sla(0x0), gid(0), cfg_order(2), cost(0), selected
    4: Seq_num(4 port16), alive, sla(0x0), gid(0), cfg_order(3), cost(0), selected
  Src address(1):
        172.16.205.0-172.16.205.255

  Dst address(1):
        0.0.0.0-255.255.255.255
```

2. The routing table shows that there are ECMP default routes on all of the members, and ECMP specific (or best) routes only on port15 and port16:

```
FGT_A (root) # get router info routing-table static
Routing table for VRF=0
S*      0.0.0.0/0 [1/0] via 172.16.200.2, port1
                         [1/0] via 172.16.208.2, dmz
                         [1/0] via 172.16.209.2, port15
                         [1/0] via 172.16.210.2, port16
S       10.1.100.22/32 [10/0] via 172.16.209.2, port15
                                  [10/0] via 172.16.210.2, port16
```

Because `tie-break` is set to `fib-best-match`, even though both port15 and port16 are out of SLA, the first configured member of the two (port15) is selected to forward traffic to PC_2. For all other traffic, the first configured member from all of the interfaces that are in SLA is selected to forward traffic (port1).

3. On PC-1, generate traffic to PC-2:

```
ping 10.1.100.22
```

4. On FGT_A, sniff for traffic sent to PC_2:

```
# diagnose sniffer packet any 'host 10.1.100.22' 4
interfaces=[any]
filters=[host 10.1.100.22]
2.831299 port5 in 172.16.205.11 -> 10.1.100.22: icmp: echo request
2.831400 port15 out 172.16.205.11 -> 10.1.100.22: icmp: echo request
```

Traffic is leaving on port15, the first configured member from port15 and port16, even though both are out of SLA.

Use an application category as an SD-WAN rule destination

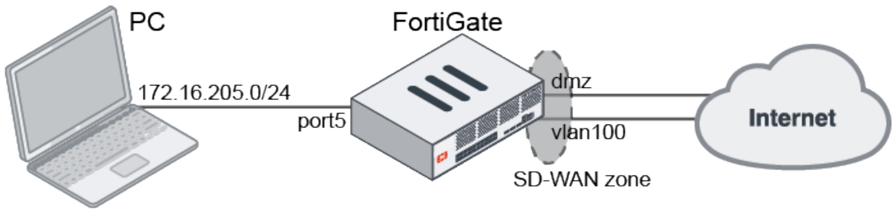
An application category can be selected as an SD-WAN service rule destination criterion. Previously, only application groups or individual applications could be selected.

```
config system sdwan
    config service
        edit <id>
            set internet-service enable
            set internet-service-app-ctrl-category <id_1> <id_2> ... <id_n>
        next
    end
end
```

To view the detected application categories details based on category ID, use `diagnose sys sdwan internet-service-app-ctrl-category-list <id>`.

Example

In this example, traffic steering is applied to traffic detected as video/audio (category ID 5) or email (category ID 21) and applies the lowest cost (SLA) strategy to this traffic. When costs are tied, the priority goes to member 1, dmz.



To configure application categories as an SD-WAN rule destination:

1. Configure the SD-WAN settings:

```
config system sdwan
    set status enable
    config zone
        edit "virtual-wan-link"
        next
    end
    config members
        edit 1
            set interface "dmz"
            set gateway 172.16.208.2
        next
        edit 2
            set interface "vlan100"
            set gateway 172.16.206.2
        next
    end
    config health-check
        edit "1"
            set server "8.8.8.8"
            set protocol dns
            set members 0
            config sla
                edit 1
                next
            end
        next
    end
end
```

2. Configure the SD-WAN rule to use application categories 5 and 21:

```
config system sdwan
    config service
        edit 1
            set name "1"
            set mode sla
            set src "172.16.205.0"
            set internet-service enable
            set internet-service-app-ctrl-category 5 21
            config sla
                edit "1"
                    set id 1
                next
            end
            set priority-members 1 2
```

```
        next
    end
end
```

3. Configure the firewall policy:

```
config firewall policy
    edit 1
        set srcintf "port5"
        set dstintf "virtual-wan-link"
        set action accept
        set srcaddr 172.16.205.0
        set dstaddr "all"
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set ssl-ssh-profile "certificate-inspection"
        set application-list "g-default"
    next
end
```

4. Verify that the traffic is sent over dmz:

```
# diagnose firewall proute list
list route policy info(vf=root):
id=2133590017(0x7f2c0001) vwl_service=1(1) vwl_mbr_seq=1 2 dscp_tag=0xff 0xff flags=0x0
tos=0x00 tos_mask=0x00 protocol=0 sport=0-65535 iif=0 dport=1-65535 path(2) oif=5(dmz)
oif=95(vlan100)
source(1): 172.16.205.0-172.16.205.255
destination wildcard(1): 0.0.0.0/0.0.0.0
internet service(2): (null)(0,5,0,0,0) (null)(0,21,0,0,0)
hit_count=469 last_used=2021-12-15 15:06:05
```

5. View some videos and emails on the PC, then verify the detected application details for each category:

```
# diagnose sys sdwan internet-service-app-ctrl-category-list 5
YouTube(31077 4294838537): 142.250.217.110 6 443 Wed Dec 15 15:39:50 2021
YouTube(31077 4294838537): 173.194.152.89 6 443 Wed Dec 15 15:37:20 2021
YouTube(31077 4294838537): 173.194.152.170 6 443 Wed Dec 15 15:37:37 2021
YouTube(31077 4294838537): 209.52.146.205 6 443 Wed Dec 15 15:37:19 2021
```

```
# diagnose sys sdwan internet-service-app-ctrl-category-list 21
Gmail(15817 4294836957): 172.217.14.197 6 443 Wed Dec 15 15:39:47 2021
```

6. Verify that the captured email traffic is sent over dmz:

```
# diagnose sniffer packet any 'host 172.217.14.197' 4
interfaces=[any]
filters=[host 172.217.14.197]
5.079814 dmz out 172.16.205.100.60592 -> 172.217.14.197.443: psh 2961561240 ack
2277134591
```

7. Edit the SD-WAN rule so that dmz has a higher cost and vlan100 is preferred.
8. Verify that the traffic is now sent over vlan100:

```
# diagnose firewall proute list
list route policy info(vf=root):
id=2134048769(0x7f330001) vwl_service=1(1) vwl_mbr_seq=2 1 dscp_tag=0xff 0xff flags=0x0
tos=0x00 tos_mask=0x00 protocol=0 sport=0-65535 iif=0 dport=1-65535 path(2) oif=95
(vlan100) oif=5(dmz)
```

```
source(1): 172.16.205.0-172.16.205.255
destination wildcard(1): 0.0.0.0/0.0.0.0
internet service(2): (null)(0,5,0,0,0) (null)(0,21,0,0,0)
hit_count=635 last_used=2021-12-15 15:55:43
```

```
# diagnose sniffer packet any 'host 172.217.14.197' 4
interfaces=[any]
filters=[host 172.217.14.197]
304.625168 vlan100 in 172.16.205.100.60592 -> 172.217.14.197.443: psh 2961572711 ack
2277139565
```

Advanced routing

The following topics provide instructions on SD-WAN advanced routing:

- Local out traffic on page 602
- Using BGP tags with SD-WAN rules on page 608
- BGP multiple path support on page 611
- Controlling traffic with BGP route mapping and service rules on page 613
- Applying BGP route-map to multiple BGP neighbors on page 620
- Using multiple members per SD-WAN neighbor configuration on page 626

Local out traffic

Local out, or self-originating, traffic is traffic that originates from the FortiGate going to external servers and services. The traffic can be from Syslog, FortiAnalyzer logging, FortiGuard services, remote authentication, and others.

By default, local out traffic relies on routing table lookups to determine the egress interface that is used to initiate the connection. However, many types of local out traffic support selecting the egress interface based on SD-WAN or manually specified interfaces. When manually specifying the egress interface, the source IP address can also be manually configured.

Go to *Network > Local Out Routing* to configure the available types of local out traffic. Some types of traffic can only be configured in the CLI.

By default *Local Out Routing* is not visible in the GUI. Go to *System > Feature Visibility* to enable it. See Feature visibility on page 2163 for more information.

When VDOMs are enabled, the following entries are available on the local out routing page:

Global view
External Resources
AWS_IP_Blacklist
AWS_Malware_Hash

VDOM view
LDAP Servers
ldap
Log

Global view	VDOM view
Log	Log FortiAnalyzer Override Settings
Log FortiAnalyzer Setting	Log Syslogd Override Settings
Log FortiAnalyzer Cloud Setting	**RADIUS Servers**
FortiGate Cloud Log Settings	fac_radius_server
Log Syslogd Setting	**TACACS+**
System	TACACS
System DNS	
System FortiGuard	
System FortiSandbox	

If a service is disabled, it is grayed out. To enable it, select the service and click *Enable Service*. If a service is enabled, there is a *Local Out Setting* button in the gutter of that service's edit page to directly configure the local-out settings.

Examples

To configure DNS local-out routing:

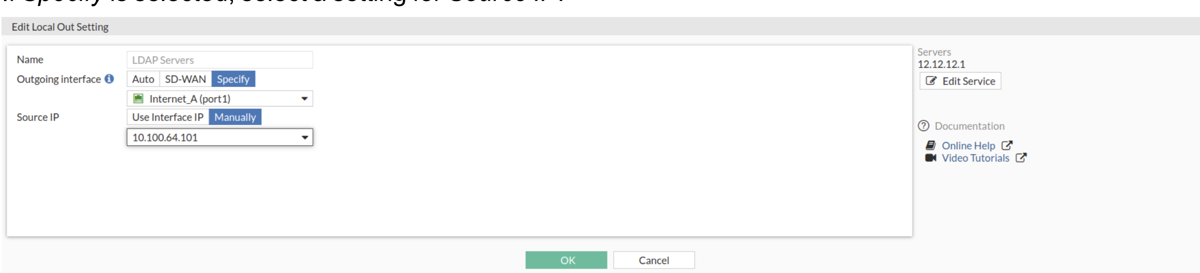
1. Go to *Network > Local Out Routing* and double-click *System DNS*.
2. For *Outgoing interface*, select one of the following:

Auto	Select the outgoing interface automatically based on the routing table.
SD-WAN	Select the outgoing interface using the configured SD-WAN interfaces and rules.
Specify	Select the outgoing interface from the dropdown.

3.

Use Interface IP	Use the primary IP, which cannot be configured by the user.
Manually	Selected an IP from the list, if the selected interface has multiple IPs configured.

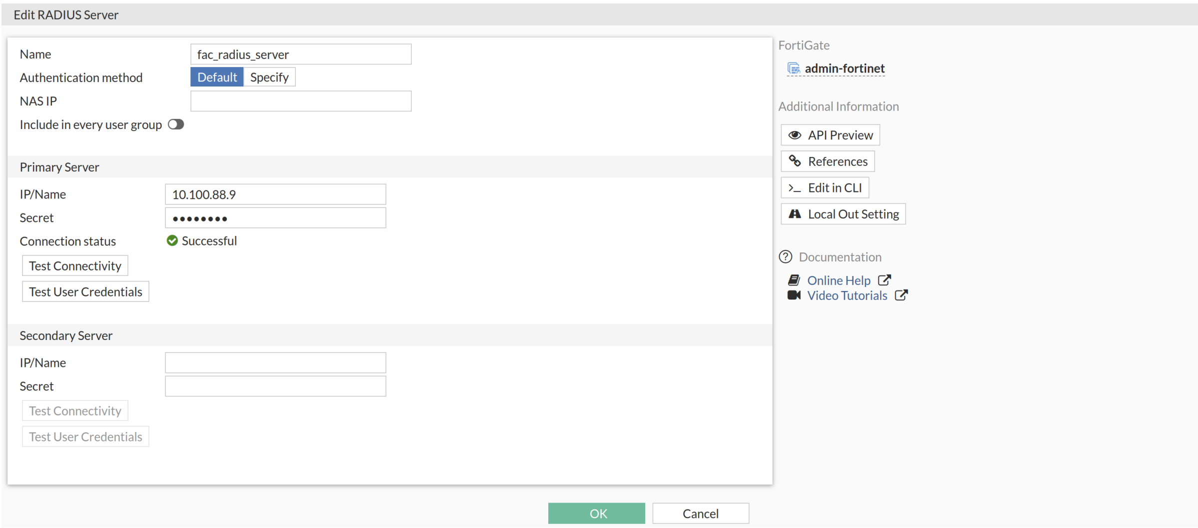
If *Specify* is selected, select a setting for *Source IP*:



4. Click *OK*.

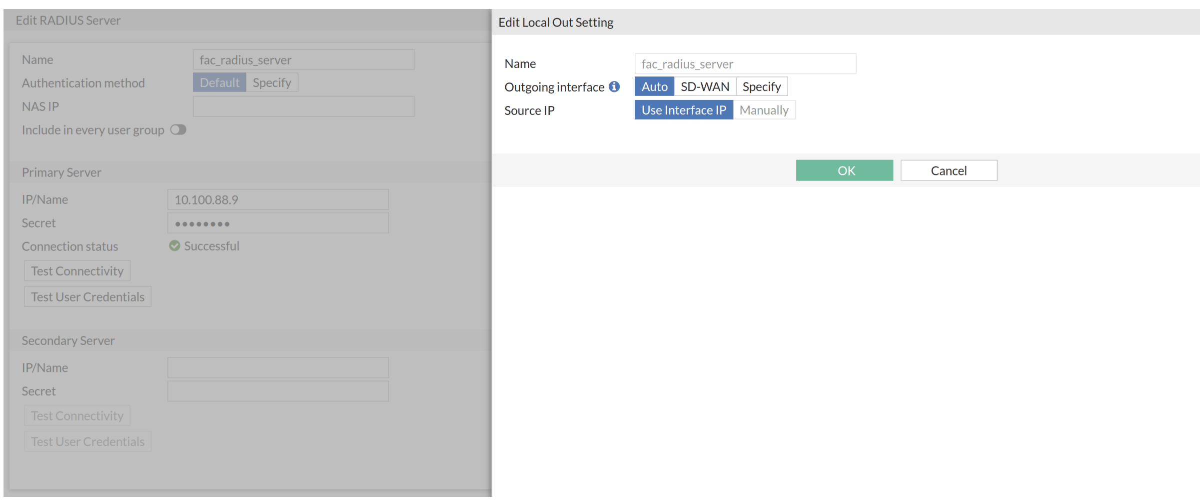
To edit local-out settings from a RADIUS server entry:

1. Go to *User & Authentication > RADIUS Servers* and double-click an entry to edit it.
2. Click *Local Out Setting*.



The *Edit Local Out Setting* pane opens.

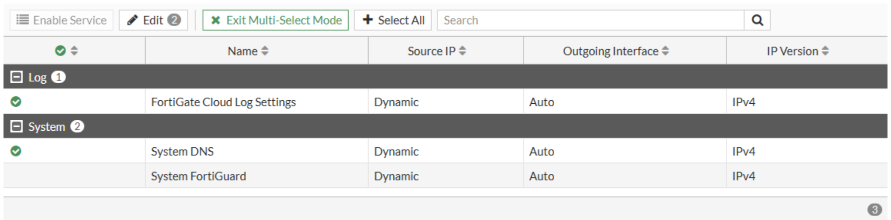
3. Configure the settings for *Outgoing interface* and *Source IP*.



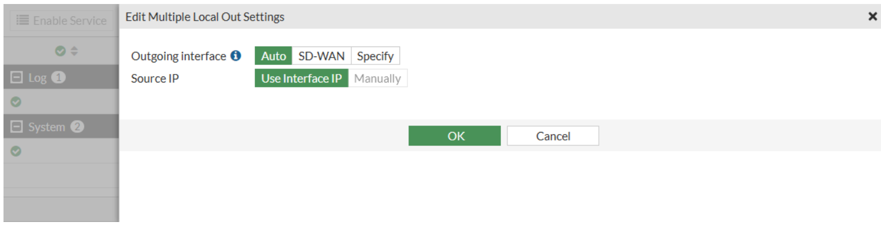
4. Click *OK*.

To edit multiple entries concurrently:

1. Go to *Network > Local Out Routing*.
2. If applicable, select *IPv4* or *IPv6*. *IPv4+IPv6* does not support multi-select.
3. Click *Multi-Select Mode*. All of the local out settings that can be edited concurrently are shown.
4. Select the specific entries, or click *Select All* to select all of the entries.



5. Click *Edit* and configure the local out settings as required.



6. Click *OK*.
7. Click *Exit Multi-Select Mode* to return to the normal view.

Configuring local out routing in the CLI

Some local out routing settings can only be configured using the CLI.

PING

IPv4 and IPv6 pings can be configured to use SD-WAN rules:

```
execute ping-options use-sdwan {yes | no}
execute ping6-options use-sd-wan {yes | no}
```

Traceroute

IPv4 traceroute can be configured to use SD-WAN rules:

```
execute traceroute-options use-sdwan {yes | no}
```

Central management

Central management traffic can use SD-WAN rules or a specific interface:

```
config system central-management
    set interface-select-method {auto | sdwan | specify}
    set interface <interface>
end
```

NTP server

NTP server traffic can use SD-WAN rules or a specific interface:

```
config system ntp
    config ntpserver
        edit <id>
            set interface-select-method {auto | sdwan | specify}
```

```
            set interface <interface>
        next
    end
end
```

DHCP proxy

DHCP proxy traffic can use SD-WAN rules or a specific interface:

```
config system settings
    set dhcp-proxy-interface-select-method {auto | sdwan | specify}
    set dhcp-proxy-interface <interface>
end
```

dhcp-proxy-interface-select-method {auto | sdwan | specify}	Select the interface selection method: • `auto`: Set the outgoing interface automatically (default). • `sdwan`: Set the interface by SD-WAN or policy routing rules. • `specify`: Set the interface manually.
dhcp-proxy-interface <interface>	Specify the outgoing interface. This option is only available and must be configured when `interface-select-method` is `specify`.

DHCP relay

DHCP relay traffic can use SD-WAN rules or a specific interface:

```
config system interface
    edit <interface>
        set dhcp-relay-interface-select-method {auto | sdwan | specify}
        set dhcp-relay-interface <interface>
    next
end
```

dhcp-relay-interface-select-method {auto | sdwan | specify}	Select the interface selection method: • `auto`: Set the outgoing interface automatically (default). • `sdwan`: Set the interface by SD-WAN or policy routing rules. • `specify`: Set the interface manually.
dhcp-relay-interface <interface>	Specify the outgoing interface. This option is only available and must be configured when `interface-select-method` is `specify`.

CA and local certificate renewal with SCEP

Certificate renewal with SCEP traffic can use SD-WAN rules or a specific interface:

```
config vpn certificate setting
    set interface-select-method {auto | sdwan | specify}
    set interface <interface>
end
```

IPS TLS protocol active probing

TLS active probing can use SD-WAN rules or a specific interface:

```
config ips global
    config tls-active-probe
        set interface-selection-method {auto | sdwan | specify}
        set interface <interface>
        set vdom <VDOM>
        set source-ip <IPv4 address>
        set source-ip6 <IPv6 address>
    end
end
```

interface-select-method {auto | sdwan | specify}	Select the interface selection method: • `auto`: Set the outgoing interface automatically (default). • `sdwan`: Set the interface by SD-WAN or policy routing rules. • `specify`: Set the interface manually.
interface <interface>	Specify the outgoing interface. This option is only available and must be configured when `interface-select-method` is `specify`.
vdom <VDOM>	Specify the VDOM. This option is only available and must be configured when `interface-select-method` is `sdwan` or `specify`.
source-ip <IPv4 address>	Specify the source IPv4 address. This option is only available and must be configured when `interface-select-method` is `sdwan` or `specify`.
source-ip6 <IPv6 address>	Specify the source IPv6 address. This option is only available and must be configured when `interface-select-method` is `sdwan` or `specify`.

Netflow and sflow

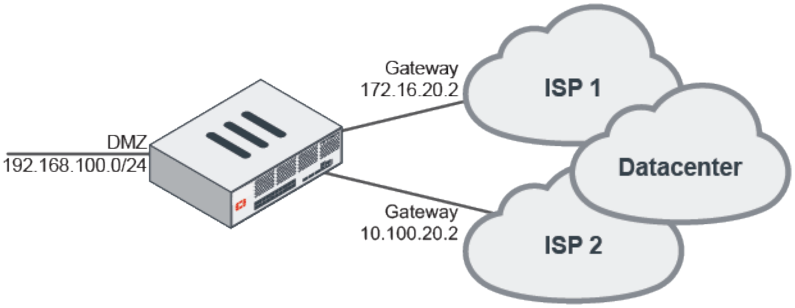
Netflow and sflow can use SD-WAN rules or a specific interface:

```
config system {netflow | sflow | vdom-netflow | vdom-sflow}
    set interface-select-method {auto | sdwan | specify}
    set interface <interface>
end
```

interface-select-method {auto | sdwan | specify}	Select the interface selection method: • `auto`: Set the outgoing interface automatically (default). • `sdwan`: Set the interface by SD-WAN or policy routing rules. • `specify`: Set the interface manually.
interface <interface>	Specify the outgoing interface. This option is only available and must be configured when `interface-select-method` is `specify`.

Using BGP tags with SD-WAN rules

SD-WAN rules can use Border Gateway Protocol (BGP) learned routes as dynamic destinations.



In this example, a customer has two ISP connections, wan1 and wan2. wan1 is used primarily for direct access to internet applications, and wan2 is used primarily for traffic to the customer's data center.

The customer could create an SD-WAN rule using the data center's IP address range as the destination to force that traffic to use wan2, but the data center's IP range is not static. Instead, a BGP tag can be used.

For this example, wan2's BGP neighbor advertises the data center's network range with a community number of 30:5.

This example assumes that SD-WAN is enabled on the FortiGate, wan1 and wan2 are added as SD-WAN members in the *virtual-wan-link* SD-WAN zone, and a policy and static route have been created. See SD-WAN quick start on page 487 for details.

FortiOS supports IPv4 and IPv6 route tags.

To configure BGP tags with SD-WAN rules:

1. Configure the community list:

```
config router community-list
    edit "30:5"
        config rule
            edit 1
                set action permit
                set match "30:5"
            next
        end
    next
end
```

2. Configure the route map:

```
config router route-map
    edit "comm1"
        config rule
            edit 1
                set match-community "30:5"
                set set-route-tag 15
            next
```

```
        end
    next
end
```

3. Configure BGP:

```
config router bgp
    set as xxxxx
    set router-id xxxx
    config neighbor
        edit "10.100.20.2"
            set soft-reconfiguration enable
            set remote-as xxxxx
            set route-map-in "comm1"
        next
    end
end
```

4. Configure a firewall policy:

```
config firewall policy
    edit 1
        set name "1"
        set srcintf "dmz"
        set dstintf "virtual-wan-link"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set nat enable
    next
end
```

5. Edit the SD-WAN configuration:

```
config system sdwan
    set status enable
    config members
        edit 1
            set interface "wan1"
            set gateway 172.16.20.2
        next
        edit 2
            set interface "wan2"
        next
    end
    config service
        edit 1
            set name "DataCenter"
            set mode manual
            set route-tag 15
            set priority-members 2
        next
    end
end
```

Troubleshooting BGP tags with SD-WAN rules

Check the network community

Use the `get router info bgp network` command to check the network community:

```
# get router info bgp network
BGP table version is 5, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network Next Hop Metric LocPrf Weight RouteTag Path
*> 0.0.0.0/0 10.100.1.5 32768 0 ?
*> 1.1.1.1/32 0.0.0.0 32768 0 ?
*> 10.1.100.0/24 172.16.203.2 32768 0 ?
*> 10.100.1.0/30 0.0.0.0 32768 0 ?
*> 10.100.1.4/30 0.0.0.0 32768 0 ?
*> 10.100.1.248/29 0.0.0.0 32768 0 ?
*> 10.100.10.0/24 10.100.1.5 202 10000 15 20 e
*> 172.16.200.0/24 0.0.0.0 32768 0 ?
*> 172.16.200.200/32
                    0.0.0.0 32768 0 ?
*> 172.16.201.0/24 172.16.200.4 32768 0 ?
*> 172.16.203.0/24 0.0.0.0 32768 0 ?
*> 172.16.204.0/24 172.16.200.4 32768 0 ?
*> 172.16.205.0/24 0.0.0.0 32768 0 ?
*> 172.16.206.0/24 0.0.0.0 32768 0 ?
*> 172.16.207.1/32 0.0.0.0 32768 0 ?
*> 172.16.207.2/32 0.0.0.0 32768 0 ?
*> 172.16.212.1/32 0.0.0.0 32768 0 ?
*> 172.16.212.2/32 0.0.0.0 32768 0 ?
*> 172.17.200.200/32
                    0.0.0.0 32768 0 ?
*> 172.27.1.0/24 0.0.0.0 32768 0 ?
*> 172.27.2.0/24 0.0.0.0 32768 0 ?
*> 172.27.5.0/24 0.0.0.0 32768 0 ?
*> 172.27.6.0/24 0.0.0.0 32768 0 ?
*> 172.27.7.0/24 0.0.0.0 32768 0 ?
*> 172.27.8.0/24 0.0.0.0 32768 0 ?
*> 172.29.1.0/24 0.0.0.0 32768 0 ?
*> 172.29.2.0/24 0.0.0.0 32768 0 ?
*> 192.168.1.0 0.0.0.0 32768 0 ?

Total number of prefixes 28

# get router info bgp network 10.100.11.0
BGP routing table entry for 10.100.10.0/24
Paths: (2 available, best 1, table Default-IP-Routing-Table)
  Advertised to non peer-group peers:
   172.10.22.2
  20
    10.100.20.2 from 10.100.20.2 (6.6.6.6)
      Origin EGP metric 200, localpref 100, weight 10000, valid, external, best
      Community: 30:5 <<<<===========================
      Last update: Wen Mar 20 18:45:17 2019
```

Check dynamic BGP addresses

Use the `get router info route-map-address` command to check dynamic BGP addresses:

```
# get router info route-map-address
Extend-tag: 15, interface(wan2:16)
        10.100.11.0/255.255.255.0
```

Check dynamic BGP addresses used in policy routes

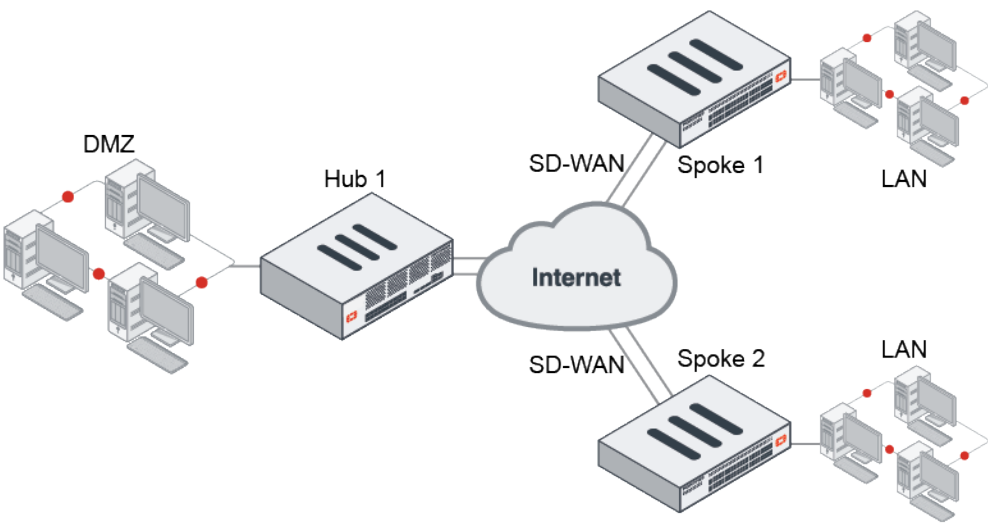
Use the `diagnose firewall proute list` command to check dynamic BGP addresses used in policy routes:

```
# diagnose firewall proute list
list route policy info(vf=root):

id=4278779905 vwl_service=1(DataCenter) flags=0x0 tos=0x00 tos_mask=0x00 protocol=0
sport=0:65535 iif=0 dport=1-65535 oif=16
source wildcard(1): 0.0.0.0/0.0.0.0
destination wildcard(1): 10.100.11.0/255.255.255.0
```

BGP multiple path support

BGP supports multiple paths, allowing an ADVPN to advertise multiple paths. This allows BGP to extend and keep additional network paths according to RFC 7911.

In this example, Spoke1 and Spoke2 each have four VPN tunnels that are connected to the Hub with ADVPN. The Spoke-Hub has established four BGP neighbors on all four tunnels.



Spoke 1 and Spoke 2 can learn four different routes from each other.

To configure the hub:

```
config router bgp
    set as 65505
    set router-id 11.11.11.11
    set ibgp-multipath enable
    set additional-path enable
```

```
    set additional-path-select 4
    config neighbor-group
        edit "gr1"
            set capability-default-originate enable
            set remote-as 65505
            set additional-path both
            set adv-additional-path 4
            set route-reflector-client enable
        next
    end
    config neighbor-range
        edit 1
            set prefix 10.10.0.0 255.255.0.0
            set neighbor-group "gr1"
        next
    end
    config network
        edit 12
            set prefix 11.11.11.11 255.255.255.255
        next
    end
end
```

To configure a spoke:

```
config router bgp
    set as 65505
    set router-id 2.2.2.2
    set ibgp-multipath enable
    set additional-path enable
    set additional-path-select 4
    config neighbor
        edit "10.10.100.254"
            set soft-reconfiguration enable
            set remote-as 65505
            set additional-path both
            set adv-additional-path 4
        next
        edit "10.10.200.254"
            set soft-reconfiguration enable
            set remote-as 65505
            set additional-path both
            set adv-additional-path 4
        next
        edit "10.10.203.254"
            set soft-reconfiguration enable
            set remote-as 65505
            set additional-path both
            set adv-additional-path 4
        next
        edit "10.10.204.254"
            set soft-reconfiguration enable
            set remote-as 65505
            set additional-path both
            set adv-additional-path 4
        next
```

```
    end
    config network
        edit 3
            set prefix 22.1.1.0 255.255.255.0
        next
    end
end
```

To view the BGP routing table on a spoke:

```
Spoke1 # get router info routing-table bgp
    Routing table for VRF=0
    B*      0.0.0.0/0 [200/0] via 10.10.200.254, vd2-2, 03:57:26
        [200/0] via 10.10.203.254, vd2-3, 03:57:26
        [200/0] via 10.10.204.254, vd2-4, 03:57:26
        [200/0] via 10.10.100.254, vd2-1, 03:57:26
    B       1.1.1.1/32 [200/0] via 11.1.1.1 (recursive via 12.1.1.1), 03:57:51
        [200/0] via 11.1.1.1 (recursive via 12.1.1.1), 03:57:51
        [200/0] via 11.1.1.1 (recursive via 12.1.1.1), 03:57:51
        [200/0] via 11.1.1.1 (recursive via 12.1.1.1), 03:57:51
    B       11.11.11.11/32 [200/0] via 10.10.200.254, vd2-2, 03:57:51
        [200/0] via 10.10.203.254, vd2-3, 03:57:51
        [200/0] via 10.10.204.254, vd2-4, 03:57:51
        [200/0] via 10.10.100.254, vd2-1, 03:57:51
    B       33.1.1.0/24 [200/0] via 10.10.204.3, vd2-4, 03:57:26
        [200/0] via 10.10.203.3, vd2-3, 03:57:26
        [200/0] via 10.10.200.3, vd2-2, 03:57:26
        [200/0] via 10.10.100.3, vd2-1, 03:57:26
        [200/0] via 10.10.204.3, vd2-4, 03:57:26
        [200/0] via 10.10.203.3, vd2-3, 03:57:26
        [200/0] via 10.10.200.3, vd2-2, 03:57:26
        [200/0] via 10.10.100.3, vd2-1, 03:57:26
        [200/0] via 10.10.204.3, vd2-4, 03:57:26
        [200/0] via 10.10.203.3, vd2-3, 03:57:26
        [200/0] via 10.10.200.3, vd2-2, 03:57:26
        [200/0] via 10.10.100.3, vd2-1, 03:57:26
        [200/0] via 10.10.204.3, vd2-4, 03:57:26
        [200/0] via 10.10.203.3, vd2-3, 03:57:26
        [200/0] via 10.10.200.3, vd2-2, 03:57:26
        [200/0] via 10.10.100.3, vd2-1, 03:57:26
```

Controlling traffic with BGP route mapping and service rules

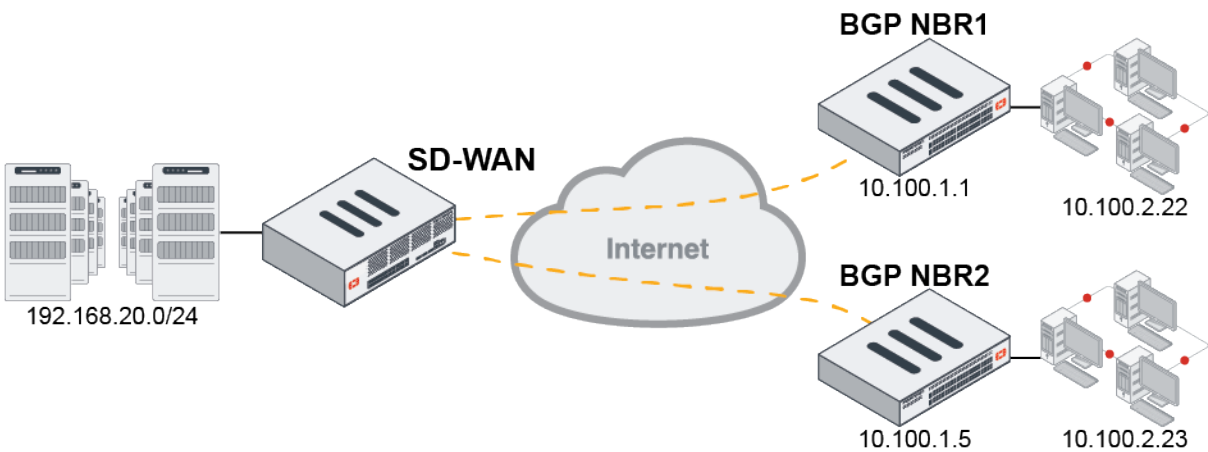
SD-WAN allows you to select different outbound WAN links based on performance SLAs. It is important that BGP neighbors are aware of these settings, and changes to them.

BGP can adapt to changes in SD-WAN link SLAs in the following ways:

- Applying different route-maps based on the SD-WAN's health checks. For example, different BGP community strings can be advertised to BGP neighbors when SLAs are not met.
- Traffic can be selectively forwarded based on the active BGP neighbor. If the SD-WAN service's role matches the active SD-WAN neighbor, the service is enabled. If there is no match, then the service is disabled.

Example

In this topology, a branch FortiGate has two SD-WAN gateways serving as the primary and secondary gateways. The gateways reside in different datacenters, but have a full mesh network between them.



This example shows how route-maps and service rules are selected based on performance SLAs and the member that is currently active. Traffic flows through the primary gateway unless the neighbor's health check is outside of its SLA. If that happens, traffic routes to the secondary gateway.

BGP NBR1 is the primary neighbor and BGP NBR2 is the secondary neighbor.

The branch FortiGate's wan1 and wan2 interfaces are members of the SD-WAN. When the SD-WAN neighbor status is primary, it will advertise community *20:1* to BGP NBR1 and *20:5* to BGP NBR2. When the SD-WAN neighbor status is secondary, it will advertise *20:5* to BGP NBR1 and *20:2* to BGP NBR2.

Only one of the primary or secondary neighbors can be active at one time. The SD-WAN neighbor status is used to decide which neighbor is selected:

- **Primary**: The primary neighbor takes precedence if its SLAs are met.
- **Secondary**: If the primary neighbor's SLAs are not met, the secondary neighbor becomes active if its SLAs are met.
- **Standalone**: If neither the primary or secondary neighbor's SLAs are met, the SD-WAN neighbor status becomes standalone.

Route map

SD-WAN is configured to let BGP advertise different communities when the SLA status changes. When the SLA is missed, it triggers BGP to advertise a different community to its BGP neighbor based on its route-map. The BGP neighbors can use the received community string to select the best path to reach the branch.

To configure BGP route-maps and neighbors:

1. Configure an access for the routes to be matched:

```
config router access-list
    edit "net192"
        config rule
            edit 1
                set prefix 192.168.20.0 255.255.255.0
            next
```

```
        end
    next
end
```

2. Configure the primary neighbor's preferred route-map:

```
config router route-map
    edit "comm1"
        config rule
            edit 1
                set match-ip-address "net192"
                set set-community "20:1"
            next
        end
    next
end
```

3. Configure the secondary neighbor's preferred route-map:

```
config router route-map
    edit "comm2"
        config rule
            edit 1
                set match-ip-address "net192"
                set set-community "20:2"
            next
        end
    next
end
```

4. Configure the failed route-map:

```
config router route-map
    edit "comm5"
        config rule
            edit 1
                set match-ip-address "net192"
                set set-community "20:5"
            next
        end
    next
end
```

5. Configure BGP neighbors:

```
config router bgp
    set as 65412
    set router-id 1.1.1.1
    set ibgp-multipath enable
    config neighbor
        edit "10.100.1.1"
            set soft-reconfiguration enable
            set remote-as 20
            set route-map-out "comm5"
            set route-map-out-preferable "comm1"
        next
        edit "10.100.1.5"
            set soft-reconfiguration enable
            set remote-as 20
```

```
            set route-map-out "comm5"
            set route-map-out-preferable "comm2"
        next
    end
end
```

When SLAs are met, `route-map-out-preferable` is used. When SLAs are missed, `route-map-out` is used.

To configure SD-WAN:

1. Configure the SD-WAN members:

```
config system sdwan
    set status enable
    config members
        edit 1
            set interface "port1"
        next
        edit 2
            set interface "port2"
        next
    end
end
```

2. Configure health checks for each member:

```
config system sdwan
    config health-check
        edit "ping"
            set server "10.100.2.22"
            set members 1
            config sla
                edit 1
                    set link-cost-factor packet-loss
                    set packetloss-threshold 1
                next
            end
        next
        edit "ping2"
            set server "10.100.2.23"
            set members 2
            config sla
                edit 1
                    set link-cost-factor packet-loss
                    set packetloss-threshold 1
                next
            end
        next
    end
end
```

3. Configure the SD-WAN neighbors and assign them a role and the health checks used to determine if the neighbor meets the SLA:
SD-WAN neighbors can only be configured in the CLI.

```
config system sdwan
    config neighbor
        edit "10.100.1.1"
```

```
            set member 1
            set role primary
            set health-check "ping"
            set sla-id 1
        next
        edit "10.100.1.5"
            set member 2
            set role secondary
            set health-check "ping2"
            set sla-id 1
        next
    end
end
```

Service rules

Create SD-WAN service rules to direct traffic to the primary neighbor when its SLAs are met, and to the secondary neighbor when the primary neighbor's SLAs are missed.

To configure the SD-WAN service rules:

```
config system sdwan
    config service
        edit 1
            set name "Primary-Out"
            set role primary
            set dst "all"
            set src "all"
            set priority-members 1
        next
        edit 2
            set name "Secondary-Out"
            set role secondary
            set dst "all"
            set src "all"
            set priority-members 2
        next
    end
end
```

If neither the primary nor secondary neighbors are active, the SD-WAN neighbor status becomes standalone. Only service rules with `standalone-action` enabled will continue to pass traffic. This option is disabled by default.

Verification

To verify when the primary neighbor is passing traffic:

1. Verify the health check status:

```
FortiGate-Branch # diagnose sys sdwan health-check
Health Check(ping):
Seq(1 port1): state(alive), packet-loss(0.000%) latency(0.569), jitter(0.061) sla_
```

```
map=0x1
Health Check(ping2):
Seq(2 port2): state(alive), packet-loss(0.000%) latency(3.916), jitter(2.373) sla_
map=0x1
```

2. Verify SD-WAN neighbor status:

```
FortiGate-Branch # diagnose sys sdwan neighbor
SD-WAN neighbor status: hold-down(disable), hold-down-time(0), hold_boot_time(0)
        Selected role(primary) last_secondary_select_time/current_time in seconds 0/572
Neighbor(10.100.1.1): member(1) role(primary)
        Health-check(ping:1)  sla-pass selected alive
Neighbor(10.100.1.5): member(2) role(secondary)
        Health-check(ping2:1)  sla-pass alive
```

3. Verify service rules status:

```
FortiGate-Branch # diagnose sys sdwan service

Service(1): Address Mode(IPV4) flags=0x0
  Gen(3), TOS(0x0/0x0), Protocol(0: 1->65535), Mode(manual)
  Service role: primary
  Members:
    1: Seq_num(1 port1), alive, selected
  Src address:
        0.0.0.0-255.255.255.255

  Dst address:
        0.0.0.0-255.255.255.255

Service(2): Address Mode(IPV4) flags=0x0
  Gen(6), TOS(0x0/0x0), Protocol(0: 1->65535), Mode(manual)
  Service role: secondary, disabled by unselected.
  Members:
    1: Seq_num(2 port2), alive, selected
  Src address:
        0.0.0.0-255.255.255.255

  Dst address:
        0.0.0.0-255.255.255.255
```

4. Verify neighbor routers:
   a. Primary neighbor router:

```
FGT-NBR1 # get router info bgp network 192.168.20.0
BGP routing table entry for 192.168.20.0/24
Paths: (1 available, best #1, table Default-IP-Routing-Table)
  Not advertised to any peer
  64512
    10.100.1.2 from 10.100.1.2 (192.168.122.98)
      Origin IGP metric 0, localpref 100, valid, external, best
      Community: 20:1
      Last update: Thu Apr 30 13:41:40 2020
```

   b. Secondary neighbor router:

```
FGT-NBR2 # get router info bgp network 192.168.20.0
VRF 0 BGP routing table entry for 192.168.20.0/24
Paths: (1 available, best #1, table Default-IP-Routing-Table)
```

```
    Not advertised to any peer
    Original VRF 0
    64512
      10.100.1.6 from 10.100.1.6 (192.168.122.98)
        Origin IGP metric 0, localpref 100, valid, external, best
        Community: 20:5
        Last update: Thu Apr 30 13:41:39 2020
```

To verify when the secondary neighbor is passing traffic:

1. Verify the health check status:

```
FortiGate-Branch # diagnose sys sdwan health-check
Health Check(ping):
Seq(1 port1): state(dead), packet-loss(54.000%) sla_map=0x0
Health Check(ping2):
Seq(2 port2): state(alive), packet-loss(0.000%) latency(4.339), jitter(3.701) sla_
map=0x1
```

2. Verify SD-WAN neighbor status:

```
FortiGate-Branch # diagnose sys sdwan neighbor
SD-WAN neighbor status: hold-down(disable), hold-down-time(0), hold_boot_time(0)
        Selected role(secondary) last_secondary_select_time/current_time in seconds
936/936
Neighbor(10.100.1.1): member(1) role(primary)
        Health-check(ping:1)  sla-fail dead
Neighbor(10.100.1.5): member(2) role(secondary)
        Health-check(ping2:1)  sla-pass selected alive
```

3. Verify service rules status:

```
FortiGate-Branch # diagnose sys sdwan service

Service(1): Address Mode(IPV4) flags=0x0
  Gen(4), TOS(0x0/0x0), Protocol(0: 1->65535), Mode(manual)
  Service role: primary, disabled by unselected.
  Members:
    1: Seq_num(1 port1), alive, selected
  Src address:
        0.0.0.0-255.255.255.255

  Dst address:
        0.0.0.0-255.255.255.255

Service(2): Address Mode(IPV4) flags=0x0
  Gen(7), TOS(0x0/0x0), Protocol(0: 1->65535), Mode(manual)
  Service role: secondary
  Members:
    1: Seq_num(2 port2), alive, selected
  Src address:
        0.0.0.0-255.255.255.255

  Dst address:
        0.0.0.0-255.255.255.255
```

4. Verify neighbor routers:
   a. Primary neighbor router:

```
FGT-NBR1 # get router info bgp network 192.168.20.0
BGP routing table entry for 192.168.20.0/24
Paths: (1 available, best #1, table Default-IP-Routing-Table)
  Not advertised to any peer
  64512
    10.100.1.2 from 10.100.1.2 (192.168.122.98)
      Origin IGP metric 0, localpref 100, valid, external, best
      Community: 20:5
      Last update: Thu Apr 30 15:41:58 2020
```

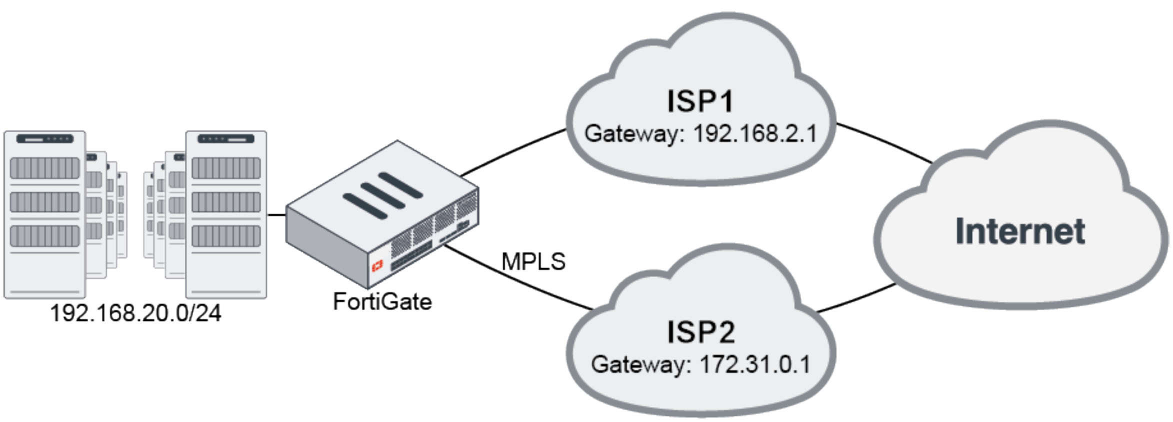
   b. Secondary neighbor router:

```
FGT-NBR2 # get router info bgp network 192.168.20.0
VRF 0 BGP routing table entry for 192.168.20.0/24
Paths: (1 available, best #1, table Default-IP-Routing-Table)
  Not advertised to any peer
  Original VRF 0
  64512
    10.100.1.6 from 10.100.1.6 (192.168.122.98)
      Origin IGP metric 0, localpref 100, valid, external, best
      Community: 20:2
      Last update: Thu Apr 30 15:42:07 2020
```

Applying BGP route-map to multiple BGP neighbors

Controlling traffic with BGP route mapping and service rules explained how BGP can apply different route-maps to the primary and secondary SD-WAN neighbors based on SLA health checks.

In this example, SD-WAN neighbors that are not bound to primary and secondary roles are configured.



The FortiGate has multiple SD-WAN links and has formed BGP neighbors with both ISPs.

ISP1 is used primarily for outbound traffic, and has an SD-WAN service rule using the lowest cost algorithm applied to it. When SLAs for ISP1 are not met, it will fail over to the MPLS line.

Inbound traffic is allowed by both WAN links, with each WAN advertising a community string when SLAs are met. When SLAs are not met, the WAN links advertise a different community string.

This example uses two SD-WAN links. The topology can be expanded to include more links as needed.

To configure BGP route-maps and neighbors:

1. Configure an access list for routes to be matched:

```
config router access-list
    edit "net192"
        config rule
            edit 1
                set prefix 192.168.20.0 255.255.255.0
            next
        end
    next
end
```

2. Configure route-maps for neighbor ISP1:

```
config router route-map
    edit "comm1"
        config rule
            edit 1
                set match-ip-address "net192"
                set set-community "64511:1"
            next
        end
    next
    edit "comm-fail1"
        config rule
            edit 1
                set match-ip-address "net192"
                set set-community "64511:5"
            next
        end
    next
end
```

3. Configure route-maps for neighbor ISP2:

```
config router route-map
    edit "comm2"
        config rule
            edit 1
                set match-ip-address "net192"
                set set-community "64522:1"
            next
        end
    next
    edit "comm-fail2"
        config rule
            edit 1
                set match-ip-address "net192"
                set set-community "64522:5"
            next
        end
    next
end
```

4. Configure the BGP neighbors:

```
config router bgp
    set as 64512
    set keepalive-timer 1
    set holdtime-timer 3
    config neighbor
        edit "192.168.2.1"
            set soft-reconfiguration enable
            set remote-as 64511
            set route-map-out "comm-fail1"
            set route-map-out-preferable "comm1"
        next
        edit "172.31.0.1"
            set soft-reconfiguration enable
            set remote-as 64522
            set route-map-out "comm-fail2"
            set route-map-out-preferable "comm2"
        next
    end
    config network
        edit 1
            set prefix 192.168.20.0 255.255.255.0
        next
    end
end
```

When SLAs are met, `route-map-out-preferable` is used. When SLAs are missed, `route-map-out` is used.

To configure SD-WAN:

1. Configure the SD-WAN members:

```
config system sdwan
    set status enable
    config members
        edit 1
            set interface "port1"
            set gateway 192.168.2.1
        next
        edit 2
            set interface "MPLS"
            set cost 20
        next
    end
end
```

2. Configure the health checks that must be met:

```
config system sdwan
    config health-check
        edit "pingserver"
            set server "8.8.8.8"
            set members 2 1
            config sla
                edit 1
                    set link-cost-factor packet-loss
                    set packetloss-threshold 2
```

```
                next
            end
        next
    end
end
```

3. Configure the SD-WAN neighbors and assign them a role and the health checks used to determine if the neighbor meets the SLA:
   When no role is defined, the default role, `standalone`, is used.

   ```
   config system sdwan
       config neighbor
           edit "192.168.2.1"
               set member 1
               set health-check "pingserver"
               set sla-id 1
           next
           edit "172.31.0.1"
               set member 2
               set health-check "pingserver"
               set sla-id 1
           next
       end
   end
   ```

Service rules

Create SD-WAN service rules to direct traffic to the SD-WAN links based on the lowest cost algorithm The same SLA health check and criteria that are used for the SD-WAN neighbor are used for this SD-WAN service rule.

When no roles are defined in the service rule, the default role, `standalone`, is used.

To configure the SD-WAN service rule:

```
config system sdwan
    config service
        edit 1
            set name "OutboundAll"
            set mode sla
            set dst "all"
            set src "all"
            config sla
                edit "pingserver"
                    set id 1
                next
            end
            set priority-members 1 2
        next
    end
end
```

Verification

To verify that when both SLAs are met, port1 is selected due to its lower cost:

1. Verify the health check status:

```
FortiGate-Branch # diagnose sys sdwan health-check
Health Check(pingserver):
Seq(2 MPLS): state(alive), packet-loss(0.000%) latency(24.709), jitter(14.996) sla_
map=0x1
Seq(1 port1): state(alive), packet-loss(0.000%) latency(28.771), jitter(14.840) sla_
map=0x1
```

2. Verify SD-WAN neighbor status:

```
FortiGate-Branch # diagnose sys sdwan neighbor
Neighbor(192.168.2.1): member(1) role(standalone)
        Health-check(pingserver:1)  sla-pass selected alive
Neighbor(172.31.0.1): member(2) role(standalone)
        Health-check(pingserver:1)  sla-pass selected alive
```

3. Verify service rules status:
   Because the service role is `standalone`, it matches both neighbors. The mode (SLA) determines that port1 is lower cost.

```
FortiGate-Branch # diagnose sys sdwan service

Service(1): Address Mode(IPV4) flags=0x0
  Gen(1), TOS(0x0/0x0), Protocol(0: 1->65535), Mode(sla), sla-compare-order
  Service role: standalone
  Members:
    1: Seq_num(1 port1), alive, sla(0x1), cfg_order(0), cost(0), selected
    2: Seq_num(2 MPLS), alive, sla(0x1), cfg_order(1), cost(20), selected
  Src address:
        0.0.0.0-255.255.255.255

  Dst address:
        0.0.0.0-255.255.255.255
```

4. Verify neighbor routers:
   a. Primary neighbor router:

```
FGT-NBR1 # get router info bgp network 192.168.20.0
BGP routing table entry for 192.168.20.0/24
Paths: (1 available, best #1, table Default-IP-Routing-Table)
  Not advertised to any peer
  64512
    192.168.2.5 from 192.168.2.5 (192.168.122.98)
      Origin IGP metric 0, localpref 100, valid, external, best
      Community: 64511:1
      Last update: Thu Apr 30 23:59:05 2020
```

   b. Secondary neighbor router:

```
FGT-NBR2 # get router info bgp network 192.168.20.0
VRF 0 BGP routing table entry for 192.168.20.0/24
Paths: (1 available, best #1, table Default-IP-Routing-Table)
  Not advertised to any peer
  Original VRF 0
```

```
      64512
        172.31.0.2 from 172.31.0.2 (192.168.122.98)
          Origin IGP metric 0, localpref 100, valid, external, best
          Community: 64522:1
          Last update: Fri May  1 00:11:28 2020
```

To verify that when neighbor ISP1 misses SLAs, MPLS is selected and BGP advertises a different community string for ISP1:

1. Verify the health check status:

```
FortiGate-Branch # diagnose sys sdwan health-check
Health Check(pingserver):
Seq(2 MPLS): state(alive), packet-loss(0.000%) latency(25.637), jitter(17.820) sla_
map=0x1
Seq(1 port1): state(dead), packet-loss(16.000%) sla_map=0x0
```

2. Verify SD-WAN neighbor status:

```
FortiGate-Branch # diagnose sys sdwan neighbor
Neighbor(192.168.2.1): member(1) role(standalone)
        Health-check(pingserver:1)  sla-fail dead
Neighbor(172.31.0.1): member(2) role(standalone)
        Health-check(pingserver:1)  sla-pass selected alive
```

3. Verify service rules status:
   As SLA failed for neighbor ISP1, MPLS is preferred.

```
FortiGate-Branch # diagnose sys sdwan service

Service(1): Address Mode(IPV4) flags=0x0
  Gen(3), TOS(0x0/0x0), Protocol(0: 1->65535), Mode(sla), sla-compare-order
  Service role: standalone
  Members:
    1: Seq_num(2 MPLS), alive, sla(0x1), cfg_order(1), cost(20), selected
    2: Seq_num(1 port1), dead, sla(0x0), cfg_order(0), cost(0)
  Src address:
        0.0.0.0-255.255.255.255

  Dst address:
        0.0.0.0-255.255.255.255
```

4. Verify neighbor routers:
   The community received on ISP1 is updated.
   a. Primary neighbor router:

```
FGT-NBR1 # get router info bgp network 192.168.20.0
BGP routing table entry for 192.168.20.0/24
Paths: (1 available, best #1, table Default-IP-Routing-Table)
  Not advertised to any peer
  64512
    192.168.2.5 from 192.168.2.5 (192.168.122.98)
      Origin IGP metric 0, localpref 100, valid, external, best
      Community: 64511:5
      Last update: Fri May  1 00:33:26 2020
```

b. Secondary neighbor router:

```
FGT-NBR2 # get router info bgp network 192.168.20.0
VRF 0 BGP routing table entry for 192.168.20.0/24
Paths: (1 available, best #1, table Default-IP-Routing-Table)
  Not advertised to any peer
  Original VRF 0
  64512
    172.31.0.2 from 172.31.0.2 (192.168.122.98)
      Origin IGP metric 0, localpref 100, valid, external, best
      Community: 64522:1
      Last update: Fri May  1 00:22:42 2020
```

Using multiple members per SD-WAN neighbor configuration

SD-WAN BGP neighbor configurations are used to define the SLA health check in which an SD-WAN member must meet to qualify as being up. When the SD-WAN member meets the SLA threshold, the FortiGate will apply the route map defined in the BGP neighbor's `route-map-out-preferable` option. If the SD-WAN member fails to meet the SLA, the FortiGate will apply the route map defined in the BGP neighbor's `route-map-out` option instead. This allows the FortiGate to advertise the health of the SD-WAN member to its BGP neighbor by advertising different community strings based on its SLA status.

For more information, refer to the following BGP examples: Controlling traffic with BGP route mapping and service rules on page 613 and Applying BGP route-map to multiple BGP neighbors on page 620.

Selecting multiple SD-WAN members allows the SD-WAN neighbor feature to support topologies where there are multiple SD-WAN overlays and/or underlays to a neighbor. The `minimum-sla-meet-members` option is used to configure the minimum number of members that must be in an SLA per neighbor for the preferable route map to be used.

```
config system sdwan
    config neighbor
        edit <ip>
            set member {<seq-num_1>} [<seq-num_2>] ... [<seq-num_n>]
            set minimum-sla-meet-members <integer>
        next
    end
end
```

`member {<seq-num_1>} [<seq-num_2>] ... [<seq-num_n>]`	Enter the member sequence number list. Multiple members can be defined.
`minimum-sla-meet-members <integer>`	Set the minimum number of members that meet SLA when the neighbor is preferred (1 - 255, default = 1). • If the number of in SLA members is less than the `minimum-sla-meet-members` value, the default route map will be used. • If the number of in SLA members is equal or larger than the `minimum-sla-meet-members` value, the preferable route map will be used.

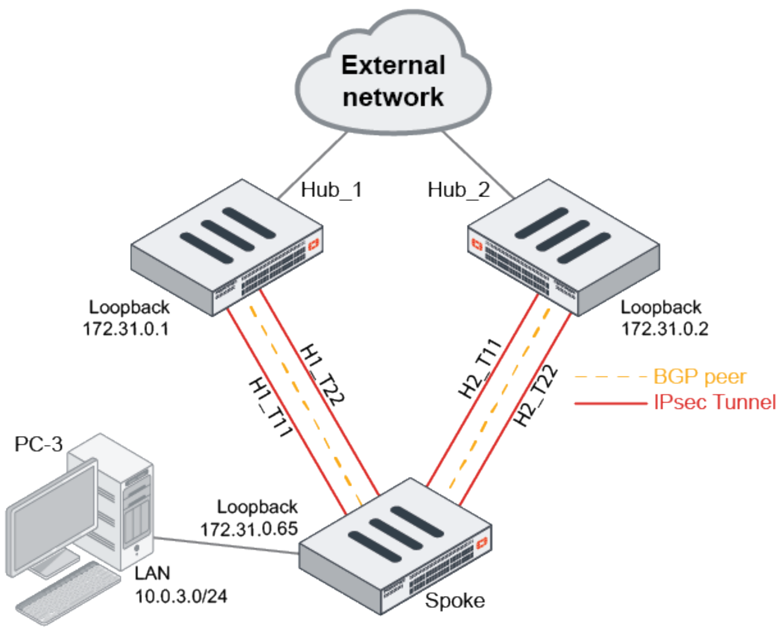
Example

In the following example, the spoke FortiGate has four tunnels: two tunnels to Hub_1 and two tunnels to Hub_2. The spoke has two BGP neighbors: one to Hub_1 and one to Hub-2. BGP neighbors are established on loopback IPs.

The SD-WAN neighbor plus `route-map-out-preferable`configuration is deployed on the spoke to achieve the following:

- If any tunnel to Hub_1 or Hub_2 is in SLA, the preferable route map will be applied on the BGP neighbor to Hub_1 or Hub_2.
- If both tunnels to Hub_1 or Hub_2 are out of SLA, the default route map will be applied on the BGP neighbor to Hub_1 or Hub_2.

The preferable route map and default route map are used to set different custom BGP communities as the spoke advertises its LAN routes to the hub. Each hub can translate communities into different BGP MED or AS prepends and signal them to the external peers to manipulate inbound traffic, thereby routing traffic to the spoke only when the SLAs are met on at least one of two VPN overlays. In this example, community string 10:1 signals to the neighbor that SLAs are met, and 10:2 signals that SLAs are not met.



To configure the BGP route maps and neighbors:

1. Configure an access list of prefixes to be matched:

```
config router access-list
    edit "net10"
        config rule
            edit 1
                set prefix 10.0.3.0 255.255.255.0
            next
        end
    next
end
```

2. Configure route maps for neighbors in SLA (preferable) and out of SLA (default):

```
config router route-map
    edit "in_sla"
        config rule
            edit 1
                set match-ip-address "net10"
                set set-community "10:1"
            next
        end
    next
    edit "out_sla"
        config rule
            edit 1
                set match-ip-address "net10"
                set set-community "10:2"
            next
        end
    next
end
```

3. Configure the BGP neighbors:

```
config router bgp
    set router-id 172.31.0.65
    config neighbor
        edit "172.31.0.1"
            set route-map-out "out_sla"
            set route-map-out-preferable "in_sla"
            set update-source "Loopback0"
        next
        edit "172.31.0.2"
            set route-map-out "out_sla"
            set route-map-out-preferable "in_sla"
            set update-source "Loopback0"
        next
    end
    config network
        edit 1
            set prefix 10.0.3.0 255.255.255.0
        next
    end
end
```

To configure SD-WAN:

1. Configure the SD-WAN members:

```
config system sdwan
    set status enable
    config members
        edit 1
            set interface "H1_T11"
            set source 172.31.0.65
        next
        edit 4
            set interface "H1_T22"
```

```
            set source 172.31.0.65
        next
        edit 6
            set interface "H2_T11"
            set source 172.31.0.65
        next
        edit 9
            set interface "H2_T22"
            set source 172.31.0.65
        next
    end
end
```

2. Configure the health check that must be met:

```
config system sdwan
    config health-check
        edit "HUB"
            set server "172.31.100.100"
            set members 0
            config sla
                edit 1
                    set link-cost-factor latency
                    set latency-threshold 100
                next
            end
        next
    end
end
```

3. Configure the SD-WAN neighbors:

```
config system sdwan
    config neighbor
        edit "172.31.0.1"
            set member 1 4
            set health-check "HUB"
            set sla-id 1
            set minimum-sla-meet-members 1
        next
        edit "172.31.0.2"
            set member 6 9
            set health-check "HUB"
            set sla-id 1
            set minimum-sla-meet-members 1
        next
    end
end
```

To verify that when two members to Hub_1/Hub_2 are in SLA, the preferable route map is be applied on BGP neighbors to Hub_1/Hub_2:

```
Branch1_A_FGT (root) # diagnose sys sdwan health-check
Health Check(HUB):
Seq(1 H1_T11): state(alive), packet-loss(0.000%) latency(0.209), jitter(0.017), mos(4.404),
bandwidth-up(999999), bandwidth-dw(999999), bandwidth-bi(1999998) sla_map=0x1
Seq(4 H1_T22): state(alive), packet-loss(0.000%) latency(0.171), jitter(0.004), mos(4.404),
bandwidth-up(999999), bandwidth-dw(999998), bandwidth-bi(1999997) sla_map=0x1
```

```
Seq(6 H2_T11): state(alive), packet-loss(0.000%) latency(0.175), jitter(0.014), mos(4.404),
bandwidth-up(999999), bandwidth-dw(999999), bandwidth-bi(1999998) sla_map=0x1
Seq(9 H2_T22): state(alive), packet-loss(0.000%) latency(0.176), jitter(0.019), mos(4.404),
bandwidth-up(999999), bandwidth-dw(999998), bandwidth-bi(1999997) sla_map=0x1
```

```
# diagnose sys sdwan neighbor
Neighbor(172.31.0.1): member(1 4 )role(standalone)
        Health-check(HUB:1)  sla-pass selected alive
Neighbor(172.31.0.2): member(6 9 )role(standalone)
        Health-check(HUB:1)  sla-pass selected alive
```

On Hub_1 and Hub_2, the expected communities have been attached into the spoke's LAN route:

```
Hub_1_FGT (root) # get router info bgp network 10.0.3.0/24
VRF 0 BGP routing table entry for 10.0.3.0/24
Paths: (1 available, best #1, table Default-IP-Routing-Table)
  Not advertised to any peer
  Original VRF 0
  Local, (Received from a RR-client)
    172.31.0.65 from 172.31.0.65 (172.31.0.65)
      Origin IGP metric 0, localpref 100, valid, internal, best
      Community: 10:1
      Last update: Wed Dec 29 22:38:29 2021
```

```
Hub_2_FGT (root) # get router info bgp network 10.0.3.0/24
VRF 0 BGP routing table entry for 10.0.3.0/24
Paths: (1 available, best #1, table Default-IP-Routing-Table)
  Not advertised to any peer
  Original VRF 0
  Local, (Received from a RR-client)
    172.31.0.65 from 172.31.0.65 (172.31.0.65)
      Origin IGP metric 0, localpref 100, valid, internal, best
      Community: 10:1

      Last update: Wed Dec 29 22:43:10 2021
```

If one member for each neighbor becomes out of SLA, the preferable route map is still applied:

```
Branch1_A_FGT (root) # diagnose sys sdwan health-check
Health Check(HUB):
Seq(1 H1_T11): state(alive), packet-loss(0.000%) latency(120.207), jitter(0.018), mos
(4.338), bandwidth-up(999999), bandwidth-dw(999997), bandwidth-bi(1999996) sla_map=0x0
Seq(4 H1_T22): state(alive), packet-loss(0.000%) latency(0.182), jitter(0.008), mos(4.404),
bandwidth-up(999999), bandwidth-dw(999998), bandwidth-bi(1999997) sla_map=0x1
Seq(6 H2_T11): state(alive), packet-loss(0.000%) latency(120.102), jitter(0.009), mos
(4.404), bandwidth-up(999999), bandwidth-dw(999998), bandwidth-bi(1999997) sla_map=0x0
Seq(9 H2_T22): state(alive), packet-loss(0.000%) latency(0.176), jitter(0.009), mos(4.404),
bandwidth-up(999999), bandwidth-dw(999997), bandwidth-bi(1999996) sla_map=0x1
```

```
# diagnose sys sdwan neighbor
Neighbor(172.31.0.1): member(1 4 )role(standalone)
        Health-check(HUB:1)  sla-pass selected alive
Neighbor(172.31.0.2): member(6 9 )role(standalone)
        Health-check(HUB:1)  sla-pass selected alive
```

```
Hub_1_FGT (root) # get router info bgp network 10.0.3.0/24
VRF 0 BGP routing table entry for 10.0.3.0/24
Paths: (1 available, best #1, table Default-IP-Routing-Table)
  Not advertised to any peer
```

```
  Original VRF 0
  Local, (Received from a RR-client)
    172.31.0.65 from 172.31.0.65 (172.31.0.65)
      Origin IGP metric 0, localpref 100, valid, internal, best
      Community: 10:1
      Last update: Thu Dec 30 10:44:47 2021
```

```
Hub_2_FGT (root) # get router info bgp network 10.0.3.0/24
VRF 0 BGP routing table entry for 10.0.3.0/24
Paths: (1 available, best #1, table Default-IP-Routing-Table)
  Not advertised to any peer
  Original VRF 0
  Local, (Received from a RR-client)
    172.31.0.65 from 172.31.0.65 (172.31.0.65)
      Origin IGP metric 0, localpref 100, valid, internal, best
      Community: 10:1
      Last update: Wed Dec 29 22:43:10 2021
```

If both members for Hub_1 become out of SLA, the default route map is applied:

```
Branch1_A_FGT (root) # diagnose sys sdwan health-check
Health Check(HUB):
Seq(1 H1_T11): state(alive), packet-loss(0.000%) latency(120.194), jitter(0.018), mos
(4.338), bandwidth-up(999999), bandwidth-dw(999997), bandwidth-bi(1999996) sla_map=0x0
Seq(4 H1_T22): state(alive), packet-loss(0.000%) latency(120.167), jitter(0.006), mos
(4.338), bandwidth-up(999999), bandwidth-dw(999998), bandwidth-bi(1999997) sla_map=0x0
Seq(6 H2_T11): state(alive), packet-loss(0.000%) latency(120.180), jitter(0.012), mos
(4.338), bandwidth-up(999999), bandwidth-dw(999998), bandwidth-bi(1999997) sla_map=0x0
Seq(9 H2_T22): state(alive), packet-loss(0.000%) latency(0.170), jitter(0.005), mos(4.404),
bandwidth-up(999999), bandwidth-dw(999997), bandwidth-bi(1999996) sla_map=0x1
```

```
# diagnose sys sdwan  neighbor
Neighbor(172.31.0.1): member(1 4 )role(standalone)
        Health-check(HUB:1)  sla-fail alive
Neighbor(172.31.0.2): member(6 9 )role(standalone)
        Health-check(HUB:1)  sla-pass selected alive
```

```
Hub_1_FGT (root) # get router info bgp network 10.0.3.0/24
VRF 0 BGP routing table entry for 10.0.3.0/24
Paths: (1 available, best #1, table Default-IP-Routing-Table)
  Not advertised to any peer
  Original VRF 0
  Local, (Received from a RR-client)
    172.31.0.65 from 172.31.0.65 (172.31.0.65)
      Origin IGP metric 0, localpref 100, valid, internal, best
      Community: 10:2
      Last update: Thu Dec 30 10:57:33 2021
```

```
Hub_2_FGT (root) # get router info bgp network 10.0.3.0/24
VRF 0 BGP routing table entry for 10.0.3.0/24
Paths: (1 available, best #1, table Default-IP-Routing-Table)
  Not advertised to any peer
  Original VRF 0
  Local, (Received from a RR-client)
    172.31.0.65 from 172.31.0.65 (172.31.0.65)
      Origin IGP metric 0, localpref 100, valid, internal, best
      Community: 10:1
      Last update: Wed Dec 29 22:43:10 2021
```

VPN overlay

The following topics provide instructions on SD-WAN VPN overlays:

ADVPN and shortcut paths

This topic provides an example of how to use SD-WAN and ADVPN together.

ADVPN (Auto Discovery VPN) is an IPsec technology that allows a traditional hub-and-spoke VPN's spokes to establish dynamic, on-demand, direct tunnels between each other to avoid routing through the topology's hub device. The primary advantage is that it provides full meshing capabilities to a standard hub-and-spoke topology. This greatly reduces the provisioning effort for full spoke-to-spoke low delay reachability, and addresses the scalability issues associated with very large fully meshed VPN networks.

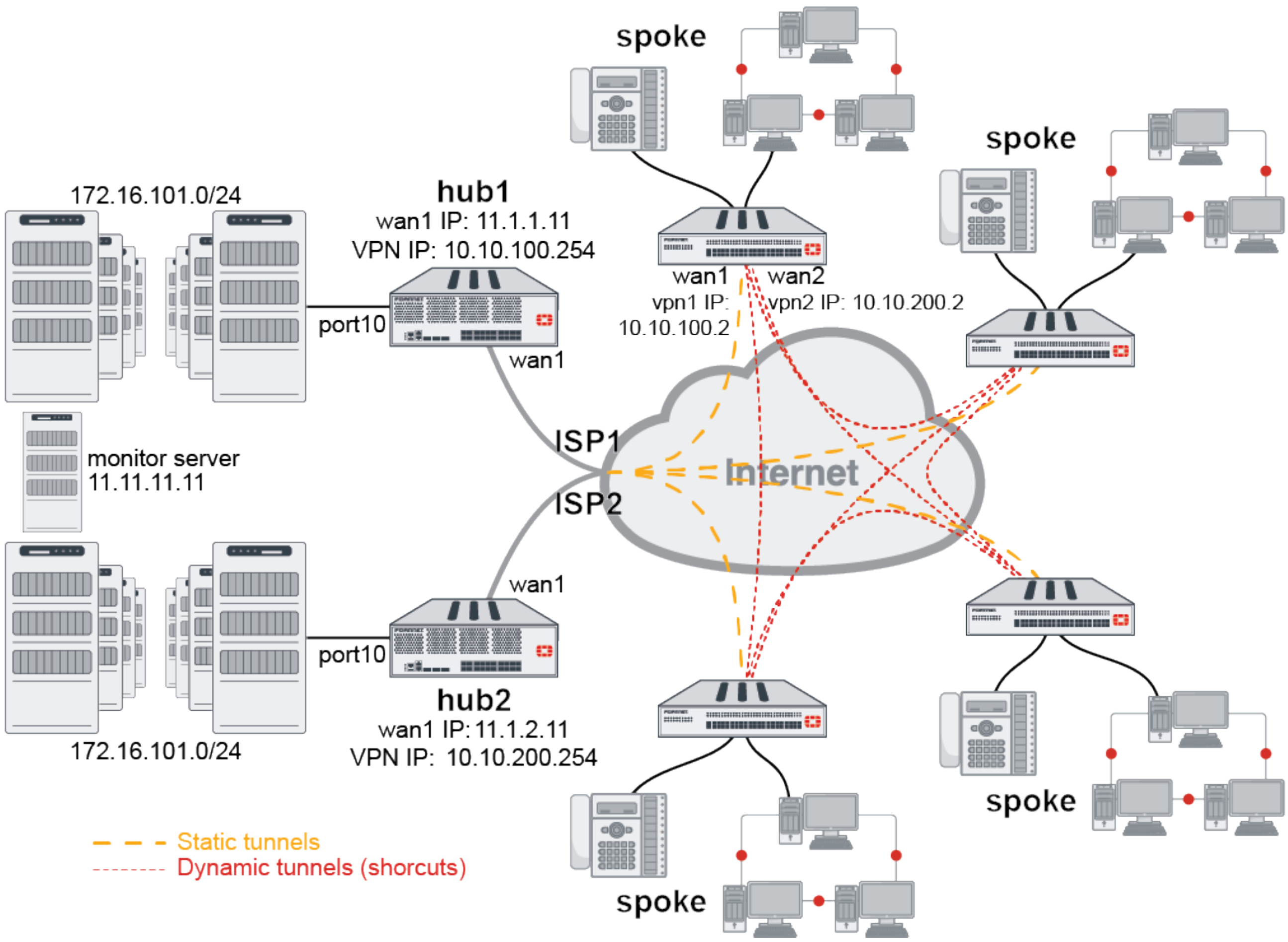
If a customer's head office and branch offices all have two or more internet connections, they can build a dual-hub ADVPN network. Combined with SD-WAN technology, the customer can load-balance traffic to other offices on multiple dynamic tunnels, control specific traffic using specific connections, or choose better performance connections dynamically.

SD-WAN load-balance mode rules (or services) do not support ADVPN members. Other modes' rules, such as SLA and priority, support ADVPN members.

This topic covers three parts:

1. Configure dual-hub ADVPN with multiple branches.
2. Configure BGP to exchange routing information among hubs and spokes.
3. Configure SD-WAN on spoke to do load-balancing and control traffic.

Configuration example



A typical ADVPN configuration with SD-WAN usually has two hubs, and each spoke connects to two ISPs and establishes VPN tunnels with both hubs.

This example shows a hub-and-spoke configuration using two hubs and one spoke:

- Hub1 and Hub2 both use wan1 to connect to the ISPs and port10 to connect to internal network.
- Spoke1 uses wan1 to connect to ISP1 and wan2 to connect to ISP2.
- wan1 sets up VPN to hub1.
- wan2 sets up VPN to hub2.

The SD-WAN is configured on the spoke. It uses the two VPN interfaces as members and two rules to control traffic to headquarters or other spokes using ADVPN VPN interfaces. You can create more rules if required.

For this example:

- Use SD-WAN member 1 (via ISP1) and its dynamic shortcuts for financial department traffic if member 1 meets SLA requirements. If it doesn't meet SLA requirements, it will use SD-WAN member 2 (via ISP2).
- Use SD-WAN member 2 (via ISP2) and its dynamic shortcuts for engineering department traffic.
- Load balance other traffic going to hubs and other spokes between these two members.
- Set up all other traffic to go with their original ISP connection. All other traffic does not go through SD-WAN.
- Set up basic network configuration to let all hubs and spokes connect to their ISPs and the Internet.

Hub internal network	172.16.101.0/24
Spoke1 internal network	10.1.100.0/24
ADVPN 1 network	10.10.100.0/24
ADVPN 2 network	10.10.200.0/24
Hub1 wan1 IP	11.1.1.11
Hub2 wan1 IP	11.1.2.11
Hub1 VPN IP	10.10.100.254
Hub2 VPN IP	10.10.200.254
Spoke1 to hub1 VPN IP	10.10.100.2
Spoke1 to hub2 VPN IP	10.10.200.2
Ping server in Headquarters	11.11.11.11
Internal subnet of spoke1	22.1.1.0/24
Internal subnet of spoke2	33.1.1.0/24
Firewall addresses	Configure hub_subnets and spoke_subnets before using in policies. These can be customized.

The GUI does not support some ADVPN related options, such as auto-discovery-sender, auto-discovery-receiver, auto-discovery-forwarder, and IBGP neighbor-group setting, so this example only provides CLI configuration commands.

Hub1 sample configuration

To configure the IPsec phase1 and phase2 interface:

```
config vpn ipsec phase1-interface
    edit "hub-phase1"
        set type dynamic
        set interface "wan1"
        set peertype any
        set net-device disable
        set proposal aes128-sha256 aes256-sha256 3des-sha256 aes128-sha1 aes256-sha1 3des-
sha1
        set add-route disable
        set dpd on-idle
        set auto-discovery-sender enable
        set psksecret sample
        set dpd-retryinterval 5
    next
end
config vpn ipsec phase2-interface
    edit "hub-phase2"
        set phase1name "hub-phase1"
        set proposal aes128-sha1 aes256-sha1 3des-sha1 aes128-sha256 aes256-sha256 3des-
sha256
    next
end
```

When `net-device` is disabled, a tunnel ID is generated for each dynamic tunnel. This ID, in the form of an IP address, is used as the gateway in the route entry to that tunnel. The `tunnel-search` option is removed in FortiOS 7.0.0 and later.

To configure the VPN interface and BGP:

```
config system interface
    edit "hub-phase1"
        set ip 10.10.100.254 255.255.255.255
        set remote-ip 10.10.100.253 255.255.255.0
    next
end
config router bgp
    set as 65505
    config neighbor-group
        edit "advpn"
            set link-down-failover enable
            set remote-as 65505
            set route-reflector-client enable
        next
    end
    config neighbor-range
        edit 1
            set prefix 10.10.100.0 255.255.255.0
            set neighbor-group "advpn"
        next
    end
    config network
        edit 1
            set prefix 172.16.101.0 255.255.255.0
        next
        edit 2
            set prefix 11.11.11.0 255.255.255.0
         next
    end
end
```

To configure the firewall policy:

```
config firewall policy
    edit 1
        set name "spoke2hub"
        set srcintf "hub-phase1"
        set dstintf "port10"
        set srcaddr "spoke_subnets"
        set dstaddr "hub_subnets"
        set action accept
        set schedule "always"
        set service "ALL"
        set comments "allow traffic from spokes to headquarter"
    next
    edit 2
        set name "spoke2spoke"
        set srcintf "hub-phase1"
```

```
        set dstintf "hub-phase1"
        set srcaddr "spoke_subnets"
        set dstaddr "spoke_subnets"
        set action accept
        set schedule "always"
        set service "ALL"
        set comments "allow traffic from spokes to spokes"
    next
    edit 3
        set name "internal2spoke"
        set srcintf "port10"
        set dstintf "hub-phase1"
        set srcaddr "hub_subnets"
        set dstaddr "spoke_subnets"
        set action accept
        set schedule "always"
        set service "ALL"
        set comments "allow traffic from headquarter to spokes"
    next
end
```

Hub2 sample configuration

Hub2 configuration is the same as hub1 except the wan1 IP address, VPN interface IP address, and BGP neighbor-range prefix.

To configure the IPsec phase1 and phase2 interface:

```
config vpn ipsec phase1-interface
    edit "hub-phase1"
        set type dynamic
        set interface "wan1"
        set peertype any
        set net-device disable
        set proposal aes128-sha256 aes256-sha256 3des-sha256 aes128-sha1 aes256-sha1 3des-
sha1
        set add-route disable
        set dpd on-idle
        set auto-discovery-sender enable
        set psksecret sample
        set dpd-retryinterval 5
    next
end
config vpn ipsec phase2-interface
    edit "hub-phase2"
        set phase1name "hub-phase1"
        set proposal aes128-sha1 aes256-sha1 3des-sha1 aes128-sha256 aes256-sha256 3des-
sha256
    next
end
```

To configure the VPN interface and BGP:

```
config system interface
    edit "hub-phase1"
```

```
        set ip 10.10.200.254 255.255.255.255
        set remote-ip 10.10.200.253 255.255.255.0
    next
end
config router bgp
    set as 65505
    config neighbor-group
        edit "advpn"
            set link-down-failover enable
            set remote-as 65505
            set route-reflector-client enable
        next
    end
    config neighbor-range
        edit 1
            set prefix 10.10.200.0 255.255.255.0
            set neighbor-group "advpn"
        next
    end
    config network
        edit 1
            set prefix 172.16.101.0 255.255.255.0
        next
        edit 2
            set prefix 11.11.11.0 255.255.255.0
         next
    end
end
```

To configure the firewall policy:

```
config firewall policy
    edit 1
        set name "spoke2hub"
        set srcintf "hub-phase1"
        set dstintf "port10"
        set srcaddr "spoke_subnets"
        set dstaddr "hub_subnets"
        set action accept
        set schedule "always"
        set service "ALL"
        set comments "allow traffic from spokes to headquarter"
    next
    edit 2
        set name "spoke2spoke"
        set srcintf "hub-phase1"
        set dstintf "hub-phase1"
        set srcaddr "spoke_subnets"
        set dstaddr "spoke_subnets"
        set action accept
        set schedule "always"
        set service "ALL"
        set comments "allow traffic from spokes to spokes"
    next
    edit 3
        set name "internal2spoke"
```

```
        set srcintf "port10"
        set dstintf "hub-phase1"
        set srcaddr "hub_subnets"
        set dstaddr "spoke_subnets"
        set action accept
        set schedule "always"
        set service "ALL"
        set comments "allow traffic from headquarter to spokes"
    next
end
```

Spoke1 sample configuration

To configure the IPsec phase1 and phase2 interface:

```
config vpn ipsec phase1-interface
    edit "spoke1-phase1"
        set interface "wan1"
        set peertype any
        set net-device enable
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set add-route disable
        set dpd on-idle
        set auto-discovery-receiver enable
        set remote-gw 11.1.1.11
        set psksecret sample
        set dpd-retryinterval 5
    next
    edit "spoke1-2-phase1"
        set interface "wan2"
        set peertype any
        set net-device enable
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set add-route disable
        set dpd on-idle
        set auto-discovery-receiver enable
        set remote-gw 11.1.2.11
        set psksecret sample
        set dpd-retryinterval 5
    next
end
config vpn ipsec phase2-interface
    edit "spoke1-phase2"
        set phase1name "spoke1-phase1"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm aes256gcm
chacha20poly1305
        set auto-negotiate enable
    next
    edit "spoke1-2-phase2"
        set phase1name "spoke1-2-phase1"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm aes256gcm
chacha20poly1305
        set auto-negotiate enable
    next
end
```

To configure the VPN interface and BGP:

```
config system interface
    edit "spoke1-phase1"
        set ip 10.10.100.2 255.255.255.255
        set remote-ip 10.10.100.254 255.255.255.0
    next
    edit "spoke1-2-phase1"
        set ip 10.10.200.2 255.255.255.255
        set remote-ip 10.10.200.254 255.255.255.0
    next
end
config router bgp
    set as 65505
    config neighbor
        edit "10.10.100.254"
            set advertisement-interval 1
            set link-down-failover enable
            set remote-as 65505
        next
        edit "10.10.200.254"
            set advertisement-interval 1
            set link-down-failover enable
            set remote-as 65505
        next
    end
    config network
        edit 1
            set prefix 10.1.100.0 255.255.255.0
        next
    end
end
```

To configure SD-WAN:

```
config system sdwan
    set status enable
    config members
        edit 1
            set interface "spoke1-phase1"
        next
        edit 2
            set interface "spoke1-2-phase1"
        next
    end
    config health-check
        edit "ping"
            set server "11.11.11.11"
            set members 1 2
            config sla
                edit 1
                    set latency-threshold 200
                    set jitter-threshold 50
                    set packetloss-threshold 5
                next
            end
```

```
        next
    end
    config service
        edit 1
            set mode sla
            set dst "financial-department"
            config sla
                edit "ping"
                    set id 1
                next
            end
            set priority-members 1 2
        next
        edit 2
            set priority-members 2
            set dst "engineering-department"
        next
    end
end
```

If no SD-WAN zone is specified, members are added to the default *virtual-wan-link* zone.

To configure the firewall policy:

```
config firewall policy
    edit 1
        set name "outbound_advpn"
        set srcintf "internal"
        set dstintf "virtual-wan-link"
        set srcaddr "spoke_subnets"
        set dstaddr "spoke_subnets" "hub_subnets"
        set action accept
        set schedule "always"
        set service "ALL"
        set comments "allow internal traffic going out to headquarter and other spokes"
    next
    edit 2
        set name "inbound_advpn"
        set srcintf "virtual-wan-link"
        set dstintf "internal"
        set srcaddr "spoke_subnets" "hub_subnets"
        set dstaddr "spoke_subnets"
        set action accept
        set schedule "always"
        set service "ALL"
        set comments "allow headquarter and other spokes traffic coming in"
    next
end
```

Troubleshooting ADVPN and shortcut paths

Before spoke vs spoke shortcut VPN is established

Use the following CLI commands to check status before spoke vs spoke shortcut VPN is established.

```
# get router info bgp summary
BGP router identifier 2.2.2.2, local AS number 65505
BGP table version is 13
3 BGP AS-PATH entries
0 BGP community entries

Neighbor        V         AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
10.10.100.254   4      65505    3286    3270       11    0    0 00:02:15        5
10.10.200.254   4      65505    3365    3319       12    0    0 00:02:14        5

Total number of neighbors 2
```

```
# get router info routing-table bgp

Routing table for VRF=0
B*      0.0.0.0/0 [200/0] via 10.10.200.254, spoke1-2-phase1, 00:00:58
                  [200/0] via 10.10.100.254, spoke1-phase1, 00:00:58
B       1.1.1.1/32 [200/0] via 11.1.1.1 (recursive via 12.1.1.1), 00:01:29
                   [200/0] via 11.1.1.1 (recursive via 12.1.1.1), 00:01:29
B       11.11.11.0/24 [200/0] via 10.10.200.254, spoke1-2-phase1, 00:01:29
                      [200/0] via 10.10.100.254, spoke1-phase1, 00:01:29
B       33.1.1.0/24 [200/0] via 10.10.200.3, spoke1-2-phase1, 00:00:58
                    [200/0] via 10.10.100.3, spoke1-phase1, 00:00:58
                    [200/0] via 10.10.200.3, spoke1-2-phase1, 00:00:58
                    [200/0] via 10.10.100.3, spoke1-phase1, 00:00:58
```

```
# diagnose vpn tunnel list
list all ipsec tunnel in vd 3
------------------------------------------------------
name=spoke1-phase1 ver=1 serial=5 12.1.1.2:0->11.1.1.11:0 tun_id=11.1.1.11 dst_mtu=15324
bound_if=48 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/536 options[0218]=npu create_dev
frag-rfc  accept_traffic=1

proxyid_num=1 child_num=0 refcnt=22 ilast=0 olast=0 ad=r/2
stat: rxp=1 txp=185 rxb=16428 txb=11111
dpd: mode=on-demand on=1 idle=20000ms retry=3 count=0 seqno=4
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=spoke1 proto=0 sa=1 ref=4 serial=1 auto-negotiate adr
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=6 options=1a227 type=00 soft=0 mtu=15262 expire=42820/0B replaywin=2048
       seqno=ba esn=0 replaywin_lastseq=00000002 itn=0 qat=0
  life: type=01 bytes=0/0 timeout=42903/43200
  dec: spi=03e01a2a esp=aes key=16 56e673f0df05186aa657f55cbb631c13
       ah=sha1 key=20 b0d50597d9bed763c42469461b03da8041f87e88
  enc: spi=2ead61bc esp=aes key=16 fe0ccd4a3ec19fe6d520c437eb6b8897
       ah=sha1 key=20 e3e669bd6df41b88eadaacba66463706f26fb53a
  dec:pkts/bytes=1/16368, enc:pkts/bytes=185/22360
  npu_flag=03 npu_rgwy=11.1.1.11 npu_lgwy=12.1.1.2 npu_selid=0 dec_npuid=1 enc_npuid=1
------------------------------------------------------
name=spoke1-2-phase1 ver=1 serial=6 112.1.1.2:0->11.1.2.11:0 tun_id=11.1.2.11 dst_mtu=15324
```

```
bound_if=90 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/536 options[0218]=npu create_dev
frag-rfc  accept_traffic=1

proxyid_num=1 child_num=0 refcnt=21 ilast=0 olast=0 ad=r/2
stat: rxp=1 txp=186 rxb=16498 txb=11163
dpd: mode=on-demand on=1 idle=20000ms retry=3 count=0 seqno=74
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=spoke1-2 proto=0 sa=1 ref=4 serial=1 auto-negotiate adr
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=6 options=1a227 type=00 soft=0 mtu=15262 expire=42818/0B replaywin=2048
       seqno=bb esn=0 replaywin_lastseq=00000002 itn=0 qat=0
  life: type=01 bytes=0/0 timeout=42901/43200
  dec: spi=03e01a2b esp=aes key=16 fe49f5042a5ad236250bf53312db1346
       ah=sha1 key=20 5dbb15c8cbc046c284bb1c6425dac2b3e15bec85
  enc: spi=2ead61bd esp=aes key=16 d6d97be52c3cccb9e88f28a9db64ac46
       ah=sha1 key=20 e20916ae6ea2295c2fbd5cbc8b8f5dd8b17f52f1
  dec:pkts/bytes=1/16438, enc:pkts/bytes=186/22480
  npu_flag=03 npu_rgwy=11.1.2.11 npu_lgwy=112.1.1.2 npu_selid=1 dec_npuid=1 enc_npuid=1
```

```
# diagnose sys sdwan service

Service(1): Address Mode(IPV4) flags=0x0
  TOS(0x0/0x0), Protocol(0: 1->65535), Mode(sla)
  Member sub interface:
  Members:
    1: Seq_num(1), alive, sla(0x1), cfg_order(0), cost(0), selected
    2: Seq_num(2), alive, sla(0x1), cfg_order(1), cost(0), selected
  Dst address: 33.1.1.1-33.1.1.100

Service(2): Address Mode(IPV4) flags=0x0
  TOS(0x0/0x0), Protocol(0: 1->65535), Mode(manual)
  Member sub interface:
  Members:
    1: Seq_num(2), alive, selected
  Dst address: 33.1.1.101-33.1.1.200
```

```
# diagnose firewall proute list
list route policy info(vf=vd2):

id=2132869121 vwl_service=1 vwl_mbr_seq=1 dscp_tag=0xff 0xff flags=0x0 tos=0x00 tos_
mask=0x00 protocol=0 sport=0:65535 iif=0 dport=1-65535 oif=70 oif=71
destination(1): 33.1.1.1-33.1.1.100
source wildcard(1): 0.0.0.0/0.0.0.0

id=2132869122 vwl_service=2 vwl_mbr_seq=2 dscp_tag=0xff 0xff flags=0x0 tos=0x00 tos_
mask=0x00 protocol=0 sport=0:65535 iif=0 dport=1-65535 oif=71
destination(1): 33.1.1.101-33.1.1.200
source wildcard(1): 0.0.0.0/0.0.0.0
```

After spoke vs spoke shortcut VPN is established

Use the following CLI commands to check status after spoke vs spoke shortcut VPN is established.

```
# get router info routing-table bgp

Routing table for VRF=0
```

```
B*      0.0.0.0/0 [200/0] via 10.10.200.254, spoke1-2-phase1, 00:01:33
                  [200/0] via 10.10.100.254, spoke1-phase1, 00:01:33
B       1.1.1.1/32 [200/0] via 11.1.1.1 (recursive via 12.1.1.1), 00:02:04
                   [200/0] via 11.1.1.1 (recursive via 12.1.1.1), 00:02:04
B       11.11.11.0/24 [200/0] via 10.10.200.254, spoke1-2-phase1, 00:02:04
                      [200/0] via 10.10.100.254, spoke1-phase1, 00:02:04
B       33.1.1.0/24 [200/0] via 10.10.200.3, spoke1-2-phase1_0, 00:01:33
                    [200/0] via 10.10.100.3, spoke1-phase1_0, 00:01:33
                    [200/0] via 10.10.200.3, spoke1-2-phase1_0, 00:01:33
                    [200/0] via 10.10.100.3, spoke1-phase1_0, 00:01:33
```

```
# diagnose sys sdwan service

Service(1): Address Mode(IPV4) flags=0x0
  TOS(0x0/0x0), Protocol(0: 1->65535), Mode(sla)
  Member sub interface:
    1: seq_num(1), interface(spoke1-phase1):
       1: spoke1-phase1_0(111)
    2: seq_num(2), interface(spoke1-2-phase1):
       1: spoke1-2-phase1_0(113)
  Members:
    1: Seq_num(1), alive, sla(0x1), cfg_order(0), cost(0), selected
    2: Seq_num(2), alive, sla(0x1), cfg_order(1), cost(0), selected
  Dst address: 33.1.1.1-33.1.1.100

Service(2): Address Mode(IPV4) flags=0x0
  TOS(0x0/0x0), Protocol(0: 1->65535), Mode(manual)
  Member sub interface:
    1: seq_num(2), interface(spoke1-2-phase1):
       1: spoke1-2-phase1_0(113)
  Members:
    1: Seq_num(2), alive, selected
  Dst address: 33.1.1.101-33.1.1.200
```

```
# diagnose vpn tunnel list
list all ipsec tunnel in vd 3
------------------------------------------------------
name=spoke1-phase1 ver=1 serial=5 12.1.1.2:0->11.1.1.11:0 tun_id=11.1.1.11 dst_mtu=15324
bound_if=48 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/536 options[0218]=npu create_dev
frag-rfc  accept_traffic=1

proxyid_num=1 child_num=1 refcnt=20 ilast=0 olast=0 ad=r/2
stat: rxp=1 txp=759 rxb=16428 txb=48627
dpd: mode=on-demand on=1 idle=20000ms retry=3 count=0 seqno=4
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=vd2-1 proto=0 sa=1 ref=5 serial=1 auto-negotiate adr
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=6 options=1a227 type=00 soft=0 mtu=15262 expire=42536/0B replaywin=2048
       seqno=2f8 esn=0 replaywin_lastseq=00000002 itn=0 qat=0
  life: type=01 bytes=0/0 timeout=42898/43200
  dec: spi=03e01a42 esp=aes key=16 1f131bda108d33909d49fc2778bd08bb
       ah=sha1 key=20 14131d3f0da9b741a2fd13d530b0553aa1f58983
  enc: spi=2ead61d8 esp=aes key=16 81ed24d5cd7bb59f4a80dceb5a560e1f
       ah=sha1 key=20 d2ccc2f3223ce16514e75f672cd88c4b4f48b681
  dec:pkts/bytes=1/16360, enc:pkts/bytes=759/94434
  npu_flag=03 npu_rgwy=11.1.1.11 npu_lgwy=12.1.1.2 npu_selid=0 dec_npuid=1 enc_npuid=1
```

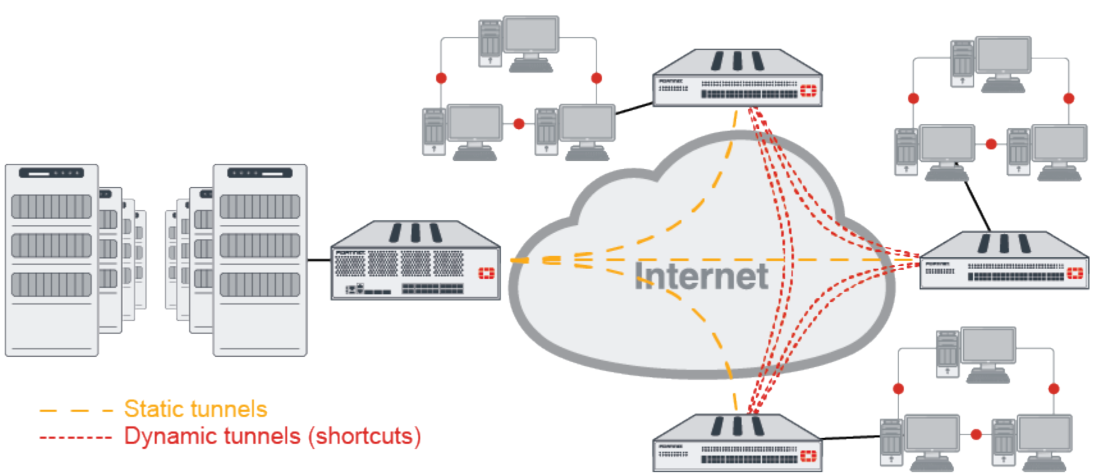
```
------------------------------------------------------
name=spoke1-2-phase1 ver=1 serial=6 112.1.1.2:0->11.1.2.11:0 tun_id=11.1.2.11 dst_mtu=15324
bound_if=90 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/536 options[0218]=npu create_dev
frag-rfc  accept_traffic=1

proxyid_num=1 child_num=1 refcnt=19 ilast=0 olast=0 ad=r/2
stat: rxp=1 txp=756 rxb=16450 txb=48460
dpd: mode=on-demand on=1 idle=20000ms retry=3 count=0 seqno=74
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=vd2-2 proto=0 sa=1 ref=5 serial=1 auto-negotiate adr
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=6 options=1a227 type=00 soft=0 mtu=15262 expire=42538/0B replaywin=2048
       seqno=2f5 esn=0 replaywin_lastseq=00000002 itn=0 qat=0
  life: type=01 bytes=0/0 timeout=42900/43200
  dec: spi=03e01a43 esp=aes key=16 7fc87561369f88b56d08bfda769eb45b
       ah=sha1 key=20 0ed554ef231c5ac16dc2e71d1907d7347dda33d6
  enc: spi=2ead61d9 esp=aes key=16 00286687aa1762e7d8216881d6720ef3
       ah=sha1 key=20 59d5eec6299ebcf038c190860774e2833074d7c3
  dec:pkts/bytes=1/16382, enc:pkts/bytes=756/94058
  npu_flag=03 npu_rgwy=11.1.2.11 npu_lgwy=112.1.1.2 npu_selid=1 dec_npuid=1 enc_npuid=1
------------------------------------------------------
name=spoke1-phase1_0 ver=1 serial=55 12.1.1.2:0->13.1.1.3:0 tun_id=13.1.1.3 dst_mtu=15324
bound_if=48 lgwy=static/1 tun=intf/0 mode=dial_inst/3 encap=none/728 options[02d8]=npu
create_dev no-sysctl rgwy-chg frag-rfc  accept_traffic=1

 parent=vd2-1 index=0
proxyid_num=1 child_num=0 refcnt=18 ilast=8 olast=8 ad=r/2
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-demand on=1 idle=20000ms retry=3 count=0 seqno=0
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=vd2-1 proto=0 sa=1 ref=2 serial=1 auto-negotiate adr
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=3 options=1a227 type=00 soft=0 mtu=15262 expire=42893/0B replaywin=2048
       seqno=1 esn=0 replaywin_lastseq=00000000 itn=0 qat=0
  life: type=01 bytes=0/0 timeout=42901/43200
  dec: spi=03e01a44 esp=aes key=16 c3b77a98e3002220e2373b73af14df6e
       ah=sha1 key=20 d18d107c248564933874f60999d6082fd7a78948
  enc: spi=864f6dba esp=aes key=16 eb6181806ccb9bac37931f9eadd4d5eb
       ah=sha1 key=20 ab788f7a372877a5603c4ede1be89a592fc21873
  dec:pkts/bytes=0/0, enc:pkts/bytes=0/0
  npu_flag=00 npu_rgwy=13.1.1.3 npu_lgwy=12.1.1.2 npu_selid=51 dec_npuid=0 enc_npuid=0
------------------------------------------------------
name=spoke1-2-phase1_0 ver=1 serial=57 112.1.1.2:0->113.1.1.3:0 tun_id=113.1.1.3 dst_
mtu=15324
bound_if=90 lgwy=static/1 tun=intf/0 mode=dial_inst/3 encap=none/728 options[02d8]=npu
create_dev no-sysctl rgwy-chg frag-rfc  accept_traffic=1

 parent=vd2-2 index=0
proxyid_num=1 child_num=0 refcnt=17 ilast=5 olast=5 ad=r/2
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-demand on=1 idle=20000ms retry=3 count=0 seqno=0
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=vd2-2 proto=0 sa=1 ref=3 serial=1 auto-negotiate adr
  src: 0:0.0.0.0/0.0.0.0:0
```

```
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=3 options=1a227 type=00 soft=0 mtu=15262 expire=42900/0B replaywin=2048
       seqno=1 esn=0 replaywin_lastseq=00000000 itn=0 qat=0
  life: type=01 bytes=0/0 timeout=42901/43200
  dec: spi=03e01a45 esp=aes key=16 0beb519ed9f800e8b4c0aa4e1df7da35
       ah=sha1 key=20 bc9f38db5296cce4208a69f1cc8a9f7ef4803c37
  enc: spi=864f6dbb esp=aes key=16 1d26e3556afcdb9f8e3e33b563b44228
       ah=sha1 key=20 564d05ef6f7437e1fd0a88d5fee7b6567f9d387e
  dec:pkts/bytes=0/0, enc:pkts/bytes=0/0
  npu_flag=00 npu_rgwy=113.1.1.3 npu_lgwy=112.1.1.2 npu_selid=53 dec_npuid=0 enc_npuid=0
```

```
# diagnose firewall proute list
list route policy info(vf=vd2):

id=2132869121 vwl_service=1 vwl_mbr_seq=1 dscp_tag=0xff 0xff flags=0x0 tos=0x00 tos_
mask=0x00 protocol=0 sport=0:65535 iif=0 dport=1-65535 oif=111 oif=70 oif=113 oif=71
destination(1): 33.1.1.1-33.1.1.100
source wildcard(1): 0.0.0.0/0.0.0.0

id=2132869122 vwl_service=2 vwl_mbr_seq=2 dscp_tag=0xff 0xff flags=0x0 tos=0x00 tos_
mask=0x00 protocol=0 sport=0:65535 iif=0 dport=1-65535 oif=113 oif=71
destination(1): 33.1.1.101-33.1.1.200
source wildcard(1): 0.0.0.0/0.0.0.0
```

SD-WAN monitor on ADVPN shortcuts

SD-WAN monitors ADVPN shortcut link quality by dynamically creating link monitors for each ADVPN link. The dynamic link monitor on the spoke will use ICMP probes and the IP address of the gateway as the monitored server. These ICMP probes will not be counted as actual user traffic that keeps the spoke-to-spoke tunnel alive.



- When no shortcut is established:

```
# diagnose sys sdwan health-check
Health Check(ping):
Seq(1 tunnel-1): state(alive), packet-loss(0.000%) latency(0.038), jitter(0.006) sla_
map=0x3
Seq(2 tunnel-2): state(alive), packet-loss(0.000%) latency(0.035), jitter(0.004) sla_
map=0x3
```

- When one shortcut is established:

```
# diagnose sys sdwan health-check
Health Check(ping):
Seq(1 tunnel-1): state(alive), packet-loss(0.000%) latency(0.039), jitter(0.003) sla_
map=0x3
Seq(1 tunnel-1_0): state(alive), packet-loss(0.000%) latency(0.060), jitter(0.023) sla_
map=0x3
Seq(2 tunnel-2): state(alive), packet-loss(0.000%) latency(0.035), jitter(0.002) sla_
map=0x3
```

- When more than one shortcut is established:

```
# diagnose sys sdwan health-check
Health Check(ping):
Seq(1 tunnel-1): state(alive), packet-loss(0.000%) latency(0.036), jitter(0.004) sla_
map=0x3
Seq(1 tunnel-1_0): state(alive), packet-loss(0.000%) latency(0.041), jitter(0.009) sla_
map=0x3
Seq(2 tunnel-2): state(alive), packet-loss(0.000%) latency(0.030), jitter(0.005) sla_
map=0x3
Seq(2 tunnel-2_0): state(alive), packet-loss(0.000%) latency(0.031), jitter(0.004) sla_
map=0x3
```

Hold down time to support SD-WAN service strategies

In a hub and spoke SD-WAN topology with shortcuts created over ADVPN, a downed or recovered shortcut can affect which member is selected by an SD-WAN service strategy. When a downed shortcut tunnel recovers and the shortcut is added back into the service strategy, the shortcut is held at a low priority until the hold down time has elapsed.

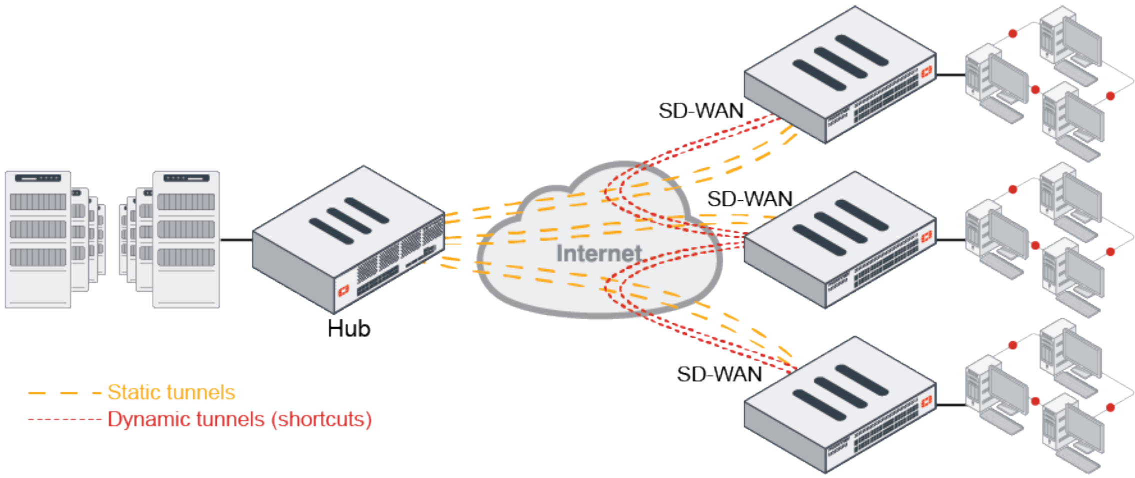
By default, the hold down time is zero seconds. It can be set to 0 - 10000000 seconds.

To configure the hold down time:

```
config system sdwan
    config service
        edit 1
            set hold-down-time <integer>
        next
    end
end
```

Example

In this example, the hold down time is set to 15 seconds, and then the SD-WAN service is looked at before and after the hold down elapses after a downed shortcut recovers.



To configure the hold down time:

```
config system sdwan
    config service
        edit 1
            set hold-down-time 15
        next
    end
end
```

To view which SD-WAN member is selected before and after the hold down time elapses:

Before the hold down time has elapsed:

```
# diagnose sys sdwan service
Service(1): Address Mode(IPV4) flags=0x200
  Gen(34), TOS(0x0/0x0), Protocol(0: 1->65535), Mode(priority), link-cost-factor(packet-
loss), link-cost-threshold(0), heath-check(ping)
Hold down time(15) seconds, Hold start at 2003 second, now 2010
  Member sub interface(4):
    1: seq_num(1), interface(vd2-1):
       1: vd2-1_0(86)
    3: seq_num(2), interface(vd2-2):
       1: vd2-2_0(88)

  Members(4):
    1: Seq_num(1 vd2-1), alive, packet loss: 27.000%, selected
    2: Seq_num(2 vd2-2_0), alive, packet loss: 0.000%, selected
    3: Seq_num(2 vd2-2), alive, packet loss: 0.000%, selected
    4: Seq_num(1 vd2-1_0), alive, packet loss: 61.000%, selected
  Dst address(1):
        33.1.1.101-33.1.1.200
```

After the hold down time has elapsed:

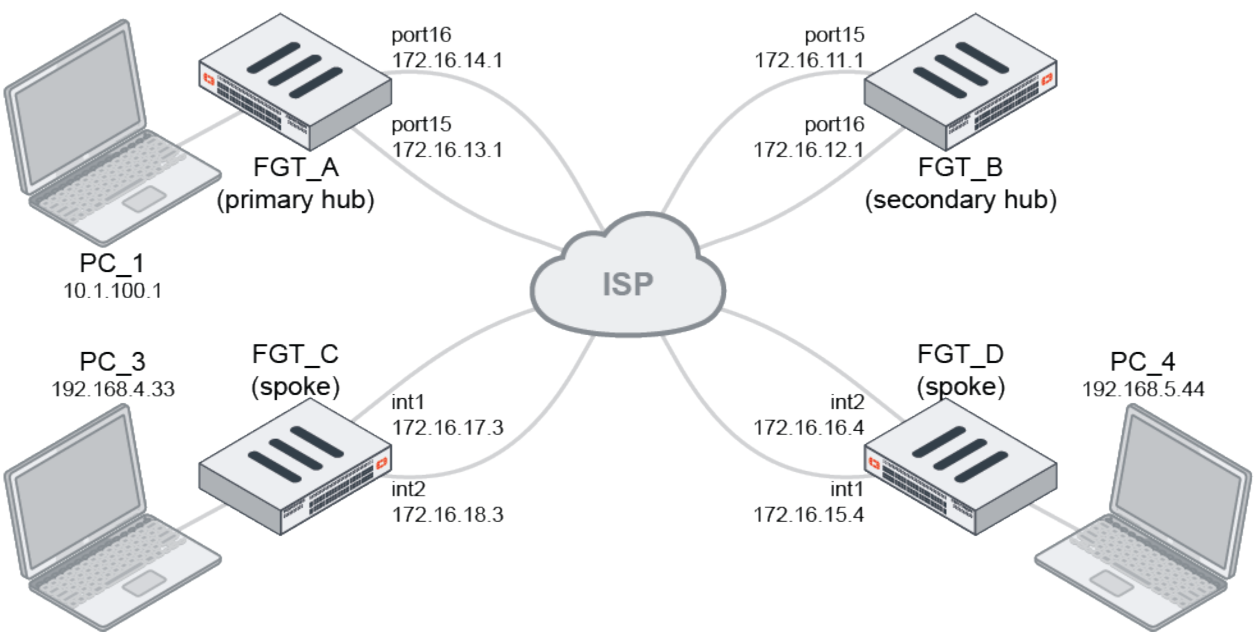
```
# diagnose sys sdwan service
Service(1): Address Mode(IPV4) flags=0x200
  Gen(35), TOS(0x0/0x0), Protocol(0: 1->65535), Mode(priority), link-cost-factor(packet-
loss), link-cost-threshold(0), heath-check(ping)
Hold down time(15) seconds, Hold start at 2018 second, now 2019
```

```
Member sub interface(4):

  2: seq_num(2), interface(vd2-2):
     1: vd2-2_0(88)
  3: seq_num(1), interface(vd2-1):
     1: vd2-1_0(86)
Members(4):
  1: Seq_num(2 vd2-2_0), alive, packet loss: 0.000%, selected
  2: Seq_num(2 vd2-2), alive, packet loss: 0.000%, selected
  3: Seq_num(1 vd2-1), alive, packet loss: 24.000%, selected
  4: Seq_num(1 vd2-1_0), alive, packet loss: 44.000%, selected
Dst address(1):
      33.1.1.101-33.1.1.200\
```

SD-WAN integration with OCVPN

OCVPN has the capability to enable SD-WAN in order to dynamically add its tunnel interfaces as SD-WAN members. Users can configure SD-WAN health checks and service rules to direct traffic over the OCVPN tunnels.

The following example uses a dual hub and spoke topology. Each hub and spoke has two WAN link connections to the ISP. The spokes generate two IPsec tunnels to each hub (four tunnels in total). BGP neighbors are established over each tunnel and routes from the hubs and other spokes learned from all neighbors, which forms an ECMP scenario. All tunnels are placed as SD-WAN members, so traffic can be distributed across tunnels based on the configured SD-WAN service rules.



To integrate SD-WAN with OCVPN in the GUI:

1. Configure the primary hub:
   a. Go to *VPN > Overlay Controller VPN* and set the *Status* to *Enable*.
   b. For *Role*, select *Primary Hub*.

c. Enter the WAN interfaces (*port15* and *port16*) and tunnel IP allocation block (*10.254.0.0/16*).

> The WAN interface is position sensitive, meaning a tunnel will be created with the first position interface on the hub to the first position interface on the spoke, and so on. In this example, FGT_A (primary hub) will create two tunnels with FGT_C (spoke):
> - FGT_A port15 <==> FGT_C internal1
> - FGT_A port16 <==> FGT_C internal2

d. Enable *Auto-discovery shortcuts*.

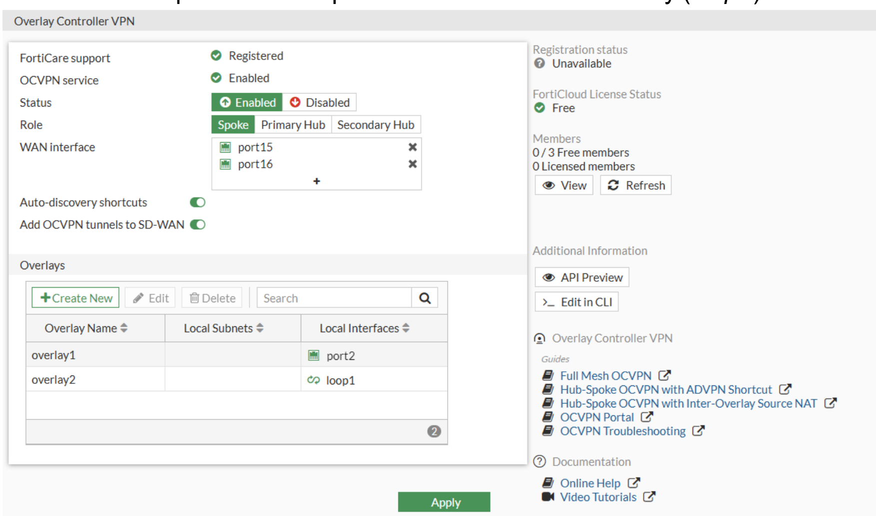
e. Enable *Add OCVPN tunnels to SD-WAN*. The IPsec tunnels will be added automatically to the SD-WAN members if SD-WAN is enabled.

2. Configure the overlays on the primary hub:

a. In the *Overlays* section, click *Create New*.

b. Enter a name and add the local interface (*port2*). Note the overlay is either based on local subnets or local interfaces, but not both.
By default, inter-overlay traffic is not enabled. Toggle *Allow traffic from other overlays* to enable it.

c. Click *OK* and repeat these steps to create the second overlay (*loop1*).



d. Click *Apply*.

3. Configure the secondary hub with the same settings as the primary hub.

4. Configure the spoke:

a. Go to *VPN > Overlay Controller VPN* and set the *Status* to *Enable*.

b. For *Role*, select *Spoke*.

c. Enter the WAN interfaces (*internal1* and *internal2*).

d. Enable *Auto-discovery shortcuts*.

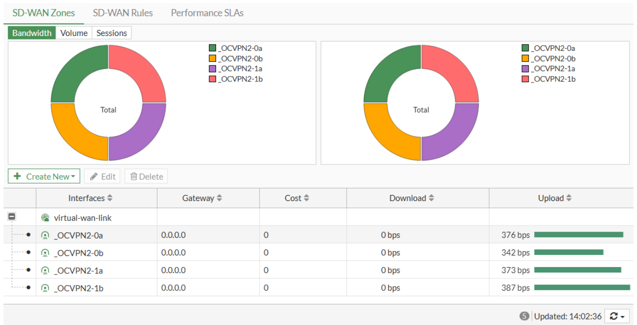
e. Enable *Add OCVPN tunnels to SD-WAN*. The IPsec tunnels will be added automatically to the SD-WAN members if SD-WAN is enabled.

f. Configure the overlays.

> The overlay names on the spokes must match the names on the hub for the traffic to be allowed through the same overlay.

g. Click *Apply*.

5. Configure the other spoke with the same settings.
6. On a spoke, go to *Network > SD-WAN* and select the *SD-WAN Zones* tab to view the configuration generated by OCVPN.



Firewall policies will be automatically generated by OCVPN between the local interfaces and the SD-WAN interface. Each policy will define the proper local and remote networks for its source and destination addresses.

To integrate SD-WAN with OCVPN in the CLI:

1. Configure the primary hub:

```
config vpn ocvpn
    set role primary-hub
    set sdwan enable
    set wan-interface "port15" "port16"
    set ip-allocation-block 10.254.0.0 255.255.0.0
    config overlays
        edit "overlay1"
            config subnets
                edit 1
                    set type interface
                    set interface "port2"
                next
            end
        next
        edit "overlay2"
            config subnets
                edit 1
                    set type interface
                    set interface "loop1"
                next
            end
        next
    end
end
```

2. Configure the secondary hub with the same settings as the primary hub.
3. Configure the spoke:

```
config vpn ocvpn
    set status enable
    set sdwan enable
    set wan-interface "internal1" "internal2"
    config overlays
        edit "overlay1"
            config subnets
```

```
                edit 1
                    set type interface
                    set interface "wan2"
                next
            end
        next
        edit "overlay2"
            config subnets
                edit 1
                    set type interface
                    set interface "loop1"
                next
            end
        next
    end
end
```

4. Configure the other spoke with the same settings.
5. Configure SD-WAN:

```
config system sdwan
    set status enable
    config members
        edit 1
            set interface "_OCVPN2-0a"
        next
        edit 2
            set interface "_OCVPN2-0b"
        next
        edit 3
            set interface "_OCVPN2-1a"
        next
        edit 4
            set interface "_OCVPN2-1b"
        next
    end
end
```

Firewall policies will be automatically generated by OCVPN between the local interfaces and the SD-WAN interface. Each policy will define the proper local and remote networks for its source and destination addresses.

If no SD-WAN zone is specified, members are added to the default *virtual-wan-link* zone.

To verify the integration is working after the ADVPN shortcut is triggered:

1. Check the routing table on the spoke:

```
FGT_C # get router info routing-table all
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
```

```
        * - candidate default

Routing table for VRF=0
S*      0.0.0.0/0 [10/0] via 172.16.17.2, internal1
                  [10/0] via 172.16.18.2, internal2
B       10.1.100.0/24 [200/0] via 10.254.7.254, _OCVPN2-0a, 00:10:24
                      [200/0] via 10.254.15.254, _OCVPN2-0b, 00:10:24
B       10.1.200.0/24 [200/0] via 10.254.7.254, _OCVPN2-0a, 00:10:24
                      [200/0] via 10.254.15.254, _OCVPN2-0b, 00:10:24
B       10.2.100.0/24 [200/0] via 10.254.71.254, _OCVPN2-1a, 00:10:15
                      [200/0] via 10.254.79.254, _OCVPN2-1b, 00:10:15
B       10.2.200.0/24 [200/0] via 10.254.71.254, _OCVPN2-1a, 00:10:15
                      [200/0] via 10.254.79.254, _OCVPN2-1b, 00:10:15
B       10.254.0.0/16 [200/0] via 10.254.7.254, _OCVPN2-0a, 00:10:15
                      [200/0] via 10.254.15.254, _OCVPN2-0b, 00:10:15
                      [200/0] via 10.254.71.254, _OCVPN2-1a, 00:10:15
                      [200/0] via 10.254.79.254, _OCVPN2-1b, 00:10:15
C       10.254.0.0/21 is directly connected, _OCVPN2-0a
C       10.254.0.1/32 is directly connected, _OCVPN2-0a
C       10.254.8.0/21 is directly connected, _OCVPN2-0b
C       10.254.8.1/32 is directly connected, _OCVPN2-0b
C       10.254.64.0/21 is directly connected, _OCVPN2-1a
C       10.254.64.1/32 is directly connected, _OCVPN2-1b_0 <==shortcut tunnel
C       10.254.64.2/32 is directly connected, _OCVPN2-1a
C       10.254.72.0/21 is directly connected, _OCVPN2-1b
C       10.254.72.2/32 is directly connected, _OCVPN2-1b
                       is directly connected, _OCVPN2-1b_0
C       172.16.17.0/24 is directly connected, internal1
C       172.16.18.0/24 is directly connected, internal2
C       172.16.200.0/24 is directly connected, wan1
C       192.168.1.0/24 is directly connected, internal
C       192.168.4.0/24 is directly connected, wan2
B       192.168.5.0/24 [200/0] via 10.254.0.2, _OCVPN2-0a, 00:00:10
                       [200/0] via 10.254.8.2, _OCVPN2-0b, 00:00:10
                       [200/0] via 10.254.0.2, _OCVPN2-0a, 00:00:10
                       [200/0] via 10.254.8.2, _OCVPN2-0b, 00:00:10
                       [200/0] via 10.254.64.1, _OCVPN2-1b_0, 00:00:10
                       [200/0] via 10.254.72.1, _OCVPN2-1b, 00:00:10
                       [200/0] via 10.254.64.1, _OCVPN2-1b_0, 00:00:10
                       [200/0] via 10.254.72.1, _OCVPN2-1b, 00:00:10
C       192.168.44.0/24 is directly connected, loop1
B       192.168.55.0/24 [200/0] via 10.254.0.2, _OCVPN2-0a, 00:00:10
                        [200/0] via 10.254.8.2, _OCVPN2-0b, 00:00:10
                        [200/0] via 10.254.0.2, _OCVPN2-0a, 00:00:10
                        [200/0] via 10.254.8.2, _OCVPN2-0b, 00:00:10
                        [200/0] via 10.254.64.1, _OCVPN2-1b_0, 00:00:10
                        [200/0] via 10.254.72.1, _OCVPN2-1b, 00:00:10
                        [200/0] via 10.254.64.1, _OCVPN2-1b_0, 00:00:10
                        [200/0] via 10.254.72.1, _OCVPN2-1b, 00:00:10
```

2. Check the VPN tunnel state:

```
FGT_C # diagnose vpn tunnel list

list all ipsec tunnel in vd 0
------------------------------------------------------
name=_OCVPN2-1b_0 ver=2 serial=1c 172.16.18.3:0->172.16.15.4:0 tun_id=172.16.15.4 dst_
```

```
mtu=1500
bound_if=9 lgwy=static/1 tun=intf/0 mode=dial_inst/3 encap=none/728 options[02d8]=npu
create_dev no-sysctl rgwy-chg frag-rfc  accept_traffic=1 overlay_id=4

 parent=_OCVPN2-1b index=0
proxyid_num=1 child_num=0 refcnt=15 ilast=0 olast=0 ad=r/2
stat: rxp=641 txp=1025 rxb=16436 txb=16446
dpd: mode=on-idle on=1 idle=20000ms retry=3 count=0 seqno=0
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=_OCVPN2-1b proto=0 sa=1 ref=3 serial=1 auto-negotiate adr
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=6 options=1a227 type=00 soft=0 mtu=1438 expire=42650/0B replaywin=1024
       seqno=407 esn=0 replaywin_lastseq=00000280 itn=0 qat=0 hash_search_len=1
  life: type=01 bytes=0/0 timeout=43186/43200
  dec: spi=90f03d9d esp=aes key=16 6cb33685bbc67d5c85488e0176ecf7b0
       ah=sha1 key=20 7d11b3babe62c840bf444b7b1f637b4324722a71
  enc: spi=7bc94bda esp=aes key=16 b4d8fc731d411eb24448b4077a5872ca
       ah=sha1 key=20 b724064d827304a6d80385ed4914461108b7312f
  dec:pkts/bytes=641/16368, enc:pkts/bytes=2053/123426
  npu_flag=03 npu_rgwy=172.16.15.4 npu_lgwy=172.16.18.3 npu_selid=1f dec_npuid=1 enc_
npuid=1
------------------------------------------------------
name=_OCVPN2-0a ver=2 serial=18 172.16.17.3:0->172.16.13.1:0 tun_id=172.16.17.3 dst_
mtu=1500
bound_if=8 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/536 options[0218]=npu create_
dev frag-rfc  accept_traffic=1 overlay_id=1

proxyid_num=1 child_num=0 refcnt=20 ilast=0 olast=0 ad=r/2
stat: rxp=1665 txp=2922 rxb=278598 txb=70241
dpd: mode=on-idle on=1 idle=20000ms retry=3 count=0 seqno=7
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=_OCVPN2-0a proto=0 sa=1 ref=4 serial=1 auto-negotiate adr
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=6 options=1a227 type=00 soft=0 mtu=1438 expire=41599/0B replaywin=1024
       seqno=890 esn=0 replaywin_lastseq=00000680 itn=0 qat=0 hash_search_len=1
  life: type=01 bytes=0/0 timeout=42899/43200
  dec: spi=90f03d95 esp=aes key=16 a6ffcc197bb1b46ec745d0b595cdd69a
       ah=sha1 key=20 8007c134e41edf282f95daf9c9033d688ef05ccc
  enc: spi=a1bf21bf esp=aes key=16 ead05be389b0dec222f969e2f9c46b1d
       ah=sha1 key=20 b04105d34d4b0e61b018f2e60591f9b1510783bb
  dec:pkts/bytes=1665/278538, enc:pkts/bytes=4237/265074
  npu_flag=03 npu_rgwy=172.16.13.1 npu_lgwy=172.16.17.3 npu_selid=1b dec_npuid=1 enc_
npuid=1
------------------------------------------------------
name=_OCVPN2-1a ver=2 serial=1a 172.16.17.3:0->172.16.11.1:0 tun_id=172.16.11.1 dst_
mtu=1500
bound_if=8 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/536 options[0218]=npu create_
dev frag-rfc  accept_traffic=1 overlay_id=3

proxyid_num=1 child_num=0 refcnt=17 ilast=0 olast=0 ad=r/2
stat: rxp=1 txp=2913 rxb=16376 txb=69642
dpd: mode=on-idle on=1 idle=20000ms retry=3 count=0 seqno=5
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=_OCVPN2-1a proto=0 sa=1 ref=28 serial=1 auto-negotiate adr
```

```
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=6 options=1a227 type=00 soft=0 mtu=1438 expire=41653/0B replaywin=1024
       seqno=887 esn=0 replaywin_lastseq=00000002 itn=0 qat=0 hash_search_len=1
  life: type=01 bytes=0/0 timeout=42900/43200
  dec: spi=90f03d9b esp=aes key=16 ee03f5b0f617a26c6177e91d60abf90b
       ah=sha1 key=20 f60cbbc4ebbd6d0327d23137da707b7ab2dc49e6
  enc: spi=a543a7d3 esp=aes key=16 1d37efab13a5c0347b582b2198b15cb8
       ah=sha1 key=20 427ee4c82bac6f26f0bcabfe04328c7f57ce682e
  dec:pkts/bytes=1/16316, enc:pkts/bytes=4229/264036
  npu_flag=03 npu_rgwy=172.16.11.1 npu_lgwy=172.16.17.3 npu_selid=1d dec_npuid=1 enc_
npuid=1
------------------------------------------------------
name=_OCVPN2-0b ver=2 serial=19 172.16.18.3:0->172.16.14.1:0 tun_id=172.16.14.1 dst_
mtu=1500
bound_if=9 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/536 options[0218]=npu create_
dev frag-rfc  accept_traffic=1 overlay_id=2

proxyid_num=1 child_num=0 refcnt=20 ilast=0 olast=0 ad=r/2
stat: rxp=1665 txp=2917 rxb=278576 txb=69755
dpd: mode=on-idle on=1 idle=20000ms retry=3 count=0 seqno=7
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=_OCVPN2-0b proto=0 sa=1 ref=4 serial=1 auto-negotiate adr
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=6 options=1a227 type=00 soft=0 mtu=1438 expire=41599/0B replaywin=1024
       seqno=88b esn=0 replaywin_lastseq=00000680 itn=0 qat=0 hash_search_len=1
  life: type=01 bytes=0/0 timeout=42899/43200
  dec: spi=90f03d96 esp=aes key=16 9d7eb233c1d095b30796c3711d53f2fd
       ah=sha1 key=20 d8feacd42b5e0ba8b5e38647b2f2734c94644bd1
  enc: spi=a1bf21c0 esp=aes key=16 d2c0984bf86dc504c5475230b24034f0
       ah=sha1 key=20 3946e4033e1f42b0d9a843b94448f56fd5b57bee
  dec:pkts/bytes=1665/278516, enc:pkts/bytes=4233/264411
  npu_flag=03 npu_rgwy=172.16.14.1 npu_lgwy=172.16.18.3 npu_selid=1c dec_npuid=1 enc_
npuid=1
------------------------------------------------------
name=_OCVPN2-1b ver=2 serial=1b 172.16.18.3:0->172.16.12.1:0 tun_id=172.16.12.1 dst_
mtu=1500
bound_if=9 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/536 options[0218]=npu create_
dev frag-rfc  accept_traffic=1 overlay_id=4

proxyid_num=1 child_num=1 refcnt=19 ilast=1 olast=0 ad=r/2
stat: rxp=1 txp=2922 rxb=16430 txb=70173
dpd: mode=on-idle on=1 idle=20000ms retry=3 count=0 seqno=4
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=_OCVPN2-1b proto=0 sa=1 ref=28 serial=1 auto-negotiate adr
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=6 options=1a227 type=00 soft=0 mtu=1438 expire=41656/0B replaywin=1024
       seqno=890 esn=0 replaywin_lastseq=00000002 itn=0 qat=0 hash_search_len=1
  life: type=01 bytes=0/0 timeout=42903/43200
  dec: spi=90f03d9c esp=aes key=16 a655767c1ed6cff4575857eb3981ad81
       ah=sha1 key=20 bfc2bccd7103a201be2641d4c6147d437d2c3f70
  enc: spi=a543a7d4 esp=aes key=16 7221b814e483165b01edfdc8260d261a
       ah=sha1 key=20 d54819643c2f1b20da2aea4282d50a1f1bc1d72a
  dec:pkts/bytes=1/16370, enc:pkts/bytes=4238/265164
```

```
  npu_flag=03 npu_rgwy=172.16.12.1 npu_lgwy=172.16.18.3 npu_selid=1e dec_npuid=1 enc_
npuid=1
```

3. Check the SD-WAN state:

```
FGT_C # diagnose sys sdwan health-check
Health Check(Default_DNS):
Health Check(Default_Office_365):
Health Check(Default_Gmail):
Health Check(Default_AWS):
Health Check(Default_Google Search):
Health Check(Default_FortiGuard):
Health Check(ocvpn):
Seq(1 _OCVPN2-0a): state(alive), packet-loss(0.000%) latency(0.364), jitter(0.028) sla_
map=0x0
Seq(2 _OCVPN2-0b): state(alive), packet-loss(0.000%) latency(0.287), jitter(0.026) sla_
map=0x0
Seq(3 _OCVPN2-1a): state(dead), packet-loss(100.000%) sla_map=0x0
Seq(4 _OCVPN2-1b): state(dead), packet-loss(100.000%) sla_map=0x0
Seq(4 _OCVPN2-1b_0): state(alive), packet-loss(0.000%) latency(0.289), jitter(0.029)
sla_map=0x0
```

Adaptive Forward Error Correction

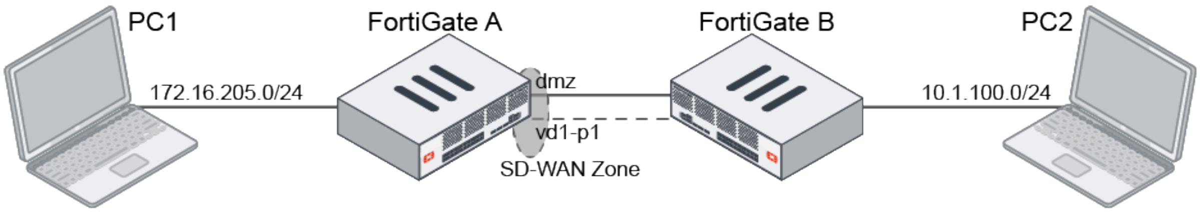
Forward Error Correction (FEC) is used to control and correct errors in data transmission by sending redundant data across the VPN in anticipation of dropped packets occurring during transit. The mechanism sends out *x* number of redundant packets for every *y* number of base packets.

Adaptive FEC considers link conditions and dynamically adjusts the FEC packet ratio:

- The FEC base and redundant packet relationship is dynamically adjusted based on changes to the network SLA metrics defined in the SD-WAN SLA health checks. For example, when there is no or low packet loss in the network, FEC can work on a low redundant level sending only one redundant packet for every 10 base packets. As packet loss increases, the number of redundant packets sent can rise accordingly.
- FEC can be applied only to streams that are sensitive to packet loss. For Example, policies that allow the UDP based VoIP protocol can enable FEC, while TCP based traffic policies do not. This reduces unnecessary bandwidth consumption by FEC.
- Because FEC does not support NPU offloading, the ability to specify streams and policies that do not require FEC allows those traffic to be offloaded. This means that all traffic suffers a performance impact.

In this example, an IPsec tunnel is configured between two FortiGates that both have FEC enabled. The tunnel is an SD-WAN zone, and an SLA health-check is used to monitor the quality of the VPN overlay. The intention is to apply FEC to UDP traffic that is passing through the VPN overlay, while allowing all other traffic to pass through without FEC. An FEC profile is configured to adaptively increase redundant levels if the link quality exceeds a 10% packet loss threshold, or the bandwidth exceeds 950 Mbps.

The DMZ interface and IPsec tunnel vd1-p1 are SD-WAN members. FEC is enabled on vd1-p1, and health-check works on vd1-p1.



To configure the FortiGates:

1. On both FortiGates, enable FEC and NPU offloading on the IPsec tunnel vd1-p1:

```
config vpn ipsec phase1-interface
    edit "vd1-p1"
        set npu-offload enable
        set fec-egress enable
        set fec-ingress enable
    next
end
```

2. On FortiGate A, configure SD-WAN:

   The VPN overlay member (vd1-p1) must be included in the health-check and configured as the higher priority member in the SD-WAN rule.

```
config system sdwan
    set status enable
    config zone
        edit "virtual-wan-link"
        next
    end
    config members
        edit 1
            set interface "dmz"
            set gateway 172.16.208.2
        next
        edit 2
            set interface "vd1-p1"
        next
    end
    config health-check
        edit "1"
            set server "2.2.2.2"
            set members 2
            config sla
                edit 1
                next
            end
        next
    end
    config service
        edit 1
            set name "1"
            set dst "all"
            set src "172.16.205.0"
            set priority-members 2 1
```

```
        next
    end
end
```

3. On FortiGate A, create a policy to specify performing FEC on UDP traffic, and a policy for other traffic:

```
config firewall policy
    edit 1
        set srcintf "port5"
        set dstintf "virtual-wan-link"
        set action accept
        set srcaddr "172.16.205.0"
        set dstaddr "all"
        set schedule "always"
        set service "ALL_UDP"
        set fec enable
    next
    edit 2
        set srcintf "any"
        set dstintf "any"
        set action accept
        set srcaddr "all"
        set dstaddr "all"
        set schedule "always"
        set service "ALL"
    next
end
```

4. On FortiGate A, configure FEC mapping to bind network SLA metrics and FEC base and redundant packets:

```
config vpn ipsec fec
    edit "m1"
        config mappings
            edit 1
                set base 8
                set redundant 2
                set packet-loss-threshold 10
            next
            edit 2
                set base 9
                set redundant 3
                set bandwidth-up-threshold 950000
            next
        end
    next
end
```

The mappings are matched from top to bottom: packet loss greater than 10% with eight base and two redundant packets, and then uploading bandwidth greater than 950 Mbps with nine base and three redundant packets.

5. On FortiGate A, apply the FEC mappings on vd1-p1:

```
config vpn ipsec phase1-interface
    edit "vd1-p1"
        set fec-health-check "1"
        set fec-mapping-profile "m1"
        set fec-base 10
        set fec-redundant 1
```

```
    next
end
```

The FEC base and redundant values are used when the link quality has not exceeded the limits specified in the FEC profile mapping. If `fec-codec` is set to `xor` the base and redundant packet values will not be updated.

To verify the results:

1. Send TCP and UDP traffic from PC1 to PC2, then check the sessions on FortiGate A:

```
# diagnose sys session list

session info: proto=6 proto_state=01 duration=12 expire=3587 timeout=3600 flags=00000000
socktype=0 sockport=0 av_idx=0 use=3
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/255
state=may_dirty npu
statistic(bytes/packets/allow_err): org=112/2/1 reply=112/2/1 tuples=2
tx speed(Bps/kbps): 0/0 rx speed(Bps/kbps): 0/0
orgin->sink: org pre->post, reply pre->post dev=15->102/102->15
gwy=172.16.209.2/172.16.205.11
hook=pre dir=org act=noop 172.16.205.11:39176->10.1.100.22:5001(0.0.0.0:0)
hook=post dir=reply act=noop 10.1.100.22:5001->172.16.205.11:39176(0.0.0.0:0)
pos/(before,after) 0/(0,0), 0/(0,0)
misc=0 policy_id=2 pol_uuid_idx=719 auth_info=0 chk_client_info=0 vd=0
serial=00020f7a tos=ff/ff app_list=0 app=0 url_cat=0
sdwan_mbr_seq=2 sdwan_service_id=1
rpdb_link_id=ff000001 rpdb_svc_id=0 ngfwid=n/a
npu_state=0x5000c00
npu info: flag=0x82/0x81, offload=8/8, ips_offload=0/0, epid=249/74, ipid=74/86,
vlan=0x0000/0x0000
vlifid=74/249, vtag_in=0x0000/0x0001 in_npu=1/1, out_npu=1/1, fwd_en=0/0, qid=5/5

session info: proto=17 proto_state=00 duration=0 expire=180 timeout=0 flags=00000000
socktype=0 sockport=0 av_idx=0 use=4
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/255
state=may_dirty fec
statistic(bytes/packets/allow_err): org=100366/67/1 reply=0/0/0 tuples=2
tx speed(Bps/kbps): 0/0 rx speed(Bps/kbps): 0/0
orgin->sink: org pre->post, reply pre->post dev=15->102/102->15 gwy=172.16.209.2/0.0.0.0
hook=pre dir=org act=noop 172.16.205.11:49052->10.1.100.22:5001(0.0.0.0:0)
hook=post dir=reply act=noop 10.1.100.22:5001->172.16.205.11:49052(0.0.0.0:0)
misc=0 policy_id=1 pol_uuid_idx=593 auth_info=0 chk_client_info=0 vd=0
serial=000210fa tos=ff/ff app_list=0 app=0 url_cat=0
sdwan_mbr_seq=2 sdwan_service_id=1
rpdb_link_id=ff000001 rpdb_svc_id=0 ngfwid=n/a
npu_state=0x5040000
no_ofld_reason:  non-npu-intf
```

   Non-FEC protected TCP traffic is offloaded, while FEC protected UDP traffic is not offloaded

2. On FortiGate A, check the health-check result and the corresponding FEC base and redundant packets:

```
# diagnose sys sdwan health-check
Health Check(1):
Seq(2 vd1-p1): state(alive), packet-loss(0.000%) latency(0.168), jitter(0.021),
bandwidth-up(999999), bandwidth-dw(999998), bandwidth-bi(1999997) sla_map=0x1
```

Because bandwidth-up is more than 950000kbps, base and redundant are set to 9 and 3:

```
# diagnose vpn tunnel fec vd1-p1
egress:
    enabled=1 base=9 redundant=3 codec=0 timeout=10(ms)
    encode=6621 encode_timeout=6621 encode_fail=0
    tx_data=6880 tx_parity=18601
ingress:
    enabled=1 timeout=50(ms)
    fasm_cnt=0 fasm_full=0
    ipsec_fec_chk_fail=0 complete=0
    rx_data=0 rx_parity=0
    recover=0 recover_timeout=0 recover_fail=0
    rx=0 rx_fail=0
```

3. Make packet loss more than 10%, then check the health-check result and the corresponding FEC base and redundant packets again:

```
# diagnose sys sdwan  health-check
Health Check(1):
Seq(2 vd1-p1): state(alive), packet-loss(15.000%) latency(0.168), jitter(0.017),
bandwidth-up(999999), bandwidth-dw(999998), bandwidth-bi(1999997) sla_map=0x0
```

Because packet loss is more than 10%, entry one in FEC mapping is first matched, and base and redundant are set to 8 and 2:

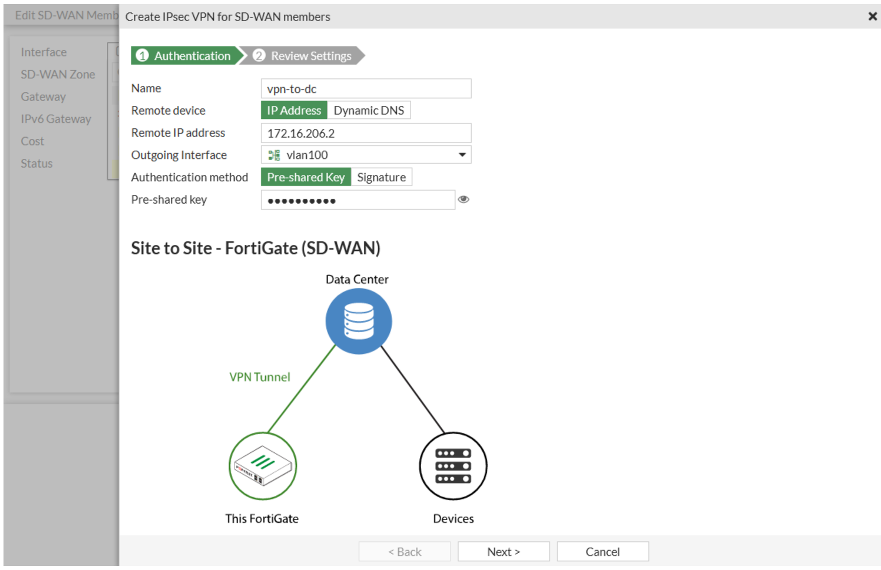
```
# diagnose vpn tunnel fec  vd1-p1
egress:
    enabled=1 base=8 redundant=2 codec=0 timeout=10(ms)
    encode=6670 encode_timeout=6670 encode_fail=0
    tx_data=6976 tx_parity=18748
ingress:
    enabled=1 timeout=50(ms)
    fasm_cnt=0 fasm_full=0
    ipsec_fec_chk_fail=0 complete=0
    rx_data=0 rx_parity=0
    recover=0 recover_timeout=0 recover_fail=0
    rx=0 rx_fail=0
```

Dual VPN tunnel wizard

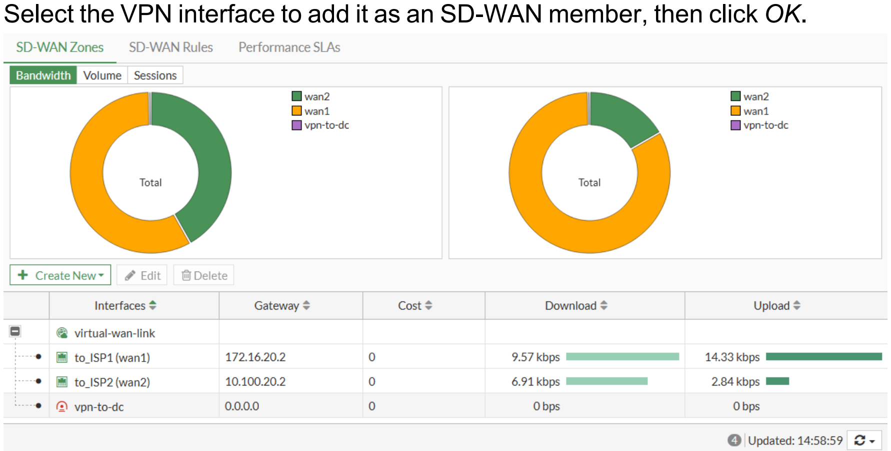
This wizard is used to automatically set up multiple VPN tunnels to the same destination over multiple outgoing interfaces. This includes automatically configuring IPsec, routing, and firewall settings, avoiding cumbersome and error-prone configuration steps.

To create a new SD-WAN VPN interface using the tunnel wizard:

1. Go to *Network > SD-WAN*, select the *SD-WAN Zones* tab, and click *Create New > SD-WAN Member*.
2. In the *Interface* drop-down, click *+VPN*. The *Create IPsec VPN for SD-WAN members* pane opens.



3. Enter the required information, then click *Next*.
4. Review the settings then click *Create*.
5. Click *Close* to return to the SD-WAN page.
   The newly created VPN interface will be highlighted in the *Interface* drop-down list.
6. Select the VPN interface to add it as an SD-WAN member, then click *OK*.



Duplicate packets on other zone members

When duplication rules are used, packets are duplicated on other good links within the SD-WAN zone and de-duplicated on the destination FortiGate. Use `force` mode to force duplication on other links within the SD-WAN zone, or use `on-demand` mode to trigger duplication only when SLA fails on the selected member.

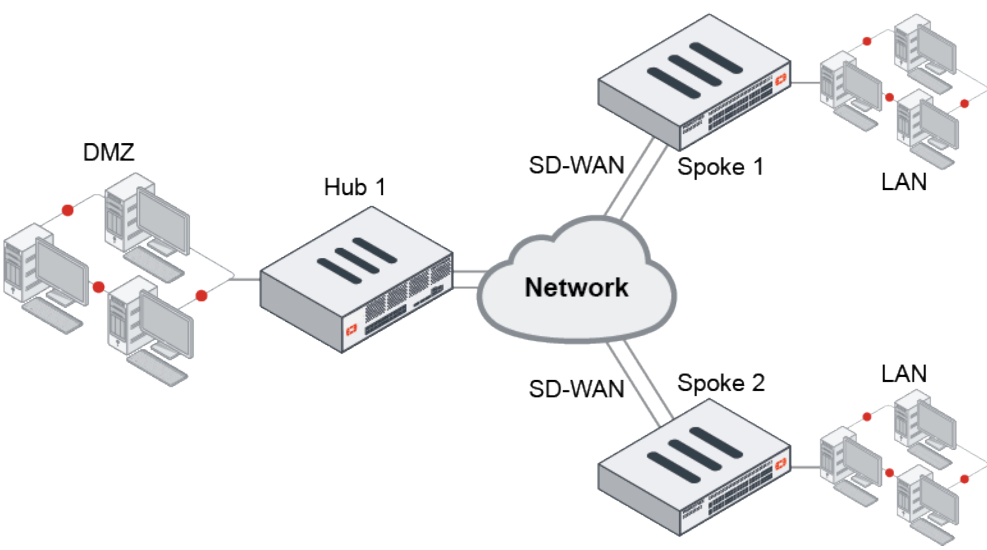
The duplication rule is configured in the CLI by using the `config duplication` command. The following options can be configured:

Parameter	Description
srcaddr	Source address or address group names.
dstaddr	Destination address or address group names.
srcaddr6	Source IPv6 address or IPv6 address group names.
dstaddr6	Destination IPv6 address or IPv6 address group names.
srcintf	Incoming (ingress) interfaces or zones.
dstintf	Outgoing (egress) interfaces or zones.
service	Service and service group names.
packet-duplication	Configure packet duplication method. • disable: Disable packet duplication (default). • force: Duplicate packets across all interface members of the SD-WAN zone. • on-demand: Duplicate packets across all interface members of the SD-WAN zone based on the link quality.
packet-de-duplication	Enable/disable discarding of packets that have been duplicated (default = disable).

The `duplication-max-num <integer>` option under `config system sdwan` is the maximum number of interface members that a packet is duplicated on in the SD-WAN zone (2 - 4, default = 2). If this value is set to 3, the original packet plus two more copies are created. If there are three member interfaces in the SD-WAN zone and the `duplication-max-num` is set to 2, the packet duplication follows the configuration order, so the packets are duplicated on the second member.

Example

The packet duplication feature works best in a spoke-spoke or hub-and-spoke topology. In this example, a hub-and-spoke ADVPN topology is used. Before shortcuts are established, Hub 1 forwards the duplicate packets from Spoke 1 to Spoke 2. Once shortcuts are established, Hub 1 is transparent, and duplicate packets are exchanged directly between the spokes.



To configure packet duplication between Spoke 1 and Spoke 2:

1. Configure Spoke 1:

```
config system sdwan
    set status enable
    config zone
        edit "virtual-wan-link"
        next
        edit "sdwanzone_v4"
        next
    end
    config members
        edit 1
            set interface "t1"
            set zone "sdwanzone_v4"
        next
        edit 4
            set interface "t21"
            set zone "sdwanzone_v4"
        next
        edit 2
            set interface "t2"
            set zone "sdwanzone_v4"
        next
    end
    config health-check
        edit "h1"
            set server "10.34.1.1"
            set interval 1000
            set failtime 10
            set members 1 2
            config sla
                edit 1
                    set packetloss-threshold 40
                next
            end
        next
    end
    config duplication
        edit 1
            set srcaddr "all"
            set dstaddr "all"
            set srcintf "port1"
            set dstintf "sdwanzone_v4"
            set service "ALL"
            set packet-duplication force
            set packet-de-duplication enable
        next
    end
end
```

2. Configure Spoke 2 with similar settings.

Duplicate packets based on SD-WAN rules

SD-WAN duplication rules can specify SD-WAN service rules to trigger packet duplication. This allows the duplication to occur based on an SD-WAN rule instead of the source, destination, and service parameters in the duplication rule.

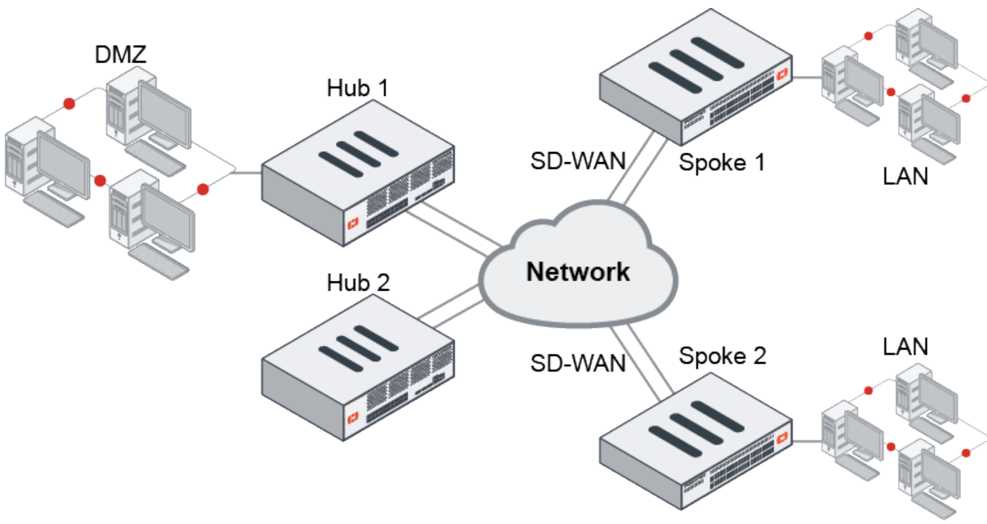
1. Packets can be forced to duplicate to all members of the same SD-WAN zone. See Duplicate packets on other zone members on page 660 for details.

   For example, in Spoke 1 set `packet-duplication` to `force` so that when a client sends a packet to the server, it is duplicated to all members of the same zone as long as its health check is alive. If a members health check is dead, then the member is removed from the SD-WAN duplication zone.
2. Packets can be duplicated to other members of the SD-WAN zone on-demand only when the condition of the link is not good enough.

   Set `packet-duplication` to `on-demand`. If `sla-match-service` is disabled, when all the SLAs of the member exceed threshold (sla_map=0), the packet is duplicated. But when the SLAs are within threshold (sla_map!=0), the packet is not duplicated.

   If `sla-match-service` is enabled, then only the SLA health checks and targets used in the service rule need to exceed threshold in order to trigger packet duplication.
3. Packets can be duplicated to all members of the same SD-WAN zone when the traffic matches one or more regular SD-WAN service rules.

The following example shows the third type of packet duplication.



In this example, SD-WAN is configured with three members: vpn1, vpn2, and vpn3. Service rule 1 controls all traffic from 10.100.20.0/24 to 172.16.100.0/24 using member 1.

To send a duplicate of the traffic that matches service rule 1 using member 2, members 1 and 2 are added to the same SD-WAN zone, and a duplication rule is configured with service-id set to 1.

To send a duplicate of the traffic that matches service rule 1 using member 2:

```
config system sdwan
    set status enable
    config zone
        edit "virtual-wan-link"
        next
        edit "zone2"
```

```
        next
    end
    config members
        edit 1
            set interface "vpn1"
        next
        edit 2
            set interface "vpn2"
        next
        edit 3
            set interface "vpn3"
            set zone "zone2"
        next
    end
    config service
        edit 1
            set dst "172.16.100.0"
            set src "10.100.20.0"
            set priority-members 1
        next
    end
    config duplication
        edit 1
            set service-id 1
            set packet-duplication force
        next
    end
end
```

Speed tests run from the hub to the spokes in dial-up IPsec tunnels

In a hub and spoke SD-WAN topology that uses dial-up VPN overlays, QoS can be applied on individual tunnels based on the measured bandwidth between the hub and spokes. The FortiGate can use the built in speed test to dynamically populate the egress bandwidth to individual dial-up tunnels from the hub.

SD-WAN members on a spoke can switch routes when the speed test is running from the hub to the spoke. The speed test results can be cached for reuse when a tunnel comes back after going down.

CLI commands

Allow upload speed tests to be run from the hub to spokes on demand for dial-up IPsec tunnel:

```
config system speed-test-schedule
    edit <interface>
        set dynamic-server {enable | disable}
    next
end
```

`<interface>`	The dial-up IPsec tunnel interface on the hub.
`dynamic-server {enable | disable}`	Enable/disable the dynamic speed test server (default = disable).

> To limit the maximum and minimum bandwidth used in the speed test, enable `set update-inbandwidth` and `set update-outbandwidth`. See Scheduled interface speedtest on page 527 for more information.

```
config system global
   set speedtest-server {enable | disable}
end
```

`speedtest-server {enable | disable}`	Enable/disable the speed test server on the spoke (default = disable). This setting must be enabled on spoke FortiGates. This enables iPerf in server mode, which listens on the default iPerf TCP port 5201.

Allow an SD-WAN member on the spoke to switch routes when it is on speed test from the hub to spokes:

```
config system sdwan
    set speedtest-bypass-routing {enable | disable}
    config neighbor
        edit <bgp neighbor>
            set mode speedtest
        next
    end
end
```

`speedtest-bypass-routing {enable | disable}`	Enable/disable bypass routing when doing a speed test on an SD-WAN member (default = disable).
`set mode speedtest`	Use the speed test to select the neighbor.

Manually run uploading speed test on the physical interfaces of each tunnel of an dial-up IPsec interface:

```
execute speed-test-dynamic <interface> <tunnel_name> <'y'/'n'> <max-out> <min-out>
```

`<interface>`	IPsec phase1 interface name.
`<tunnel_name>`	The tunnel name, or `all` for all tunnels.
`<'y'/'n'>`	Apply the result to the tunnels' shaper or not.
`<max-out>`	The maximum speed used in a speed test, in kbps.
`<min-out>`	The minimum speed used in a speed test, in kbps.

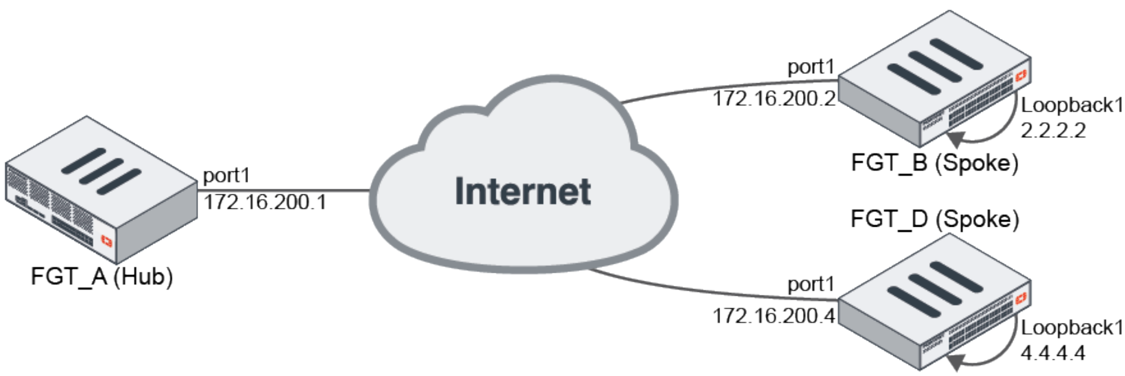
Manually run a non-blocking uploading speed test:

```
diagnose netlink interface speed-test-tunnel <interface> <tunnel_name>
```

Debug and test commands:

`diagnose debug application speedtest <int>`	Enable debug of the speed test module in the forticron daemon.
`diagnose debug application speedtestd <int>`	Enable debug of the speed test server daemon.
`diagnose test application forticron 9`	List the scheduled speed tests.
`diagnose test application forticron 10`	Show the cached speed test results.
`diagnose test application forticron 11`	Write the cached speed test results to disk.
`diagnose test application forticron 12`	Load the speed test results from disk.
`diagnose test application forticron 99`	Cancel all pending speed tests.

Example



In this example, the hub is configured as a VPN dial-up server and both of the spokes are connected to the hub. It is assumed that the VPN configuration is already done, with a dynamic gateway type and kernel device creation (`net-device`) disabled. Only one SD-WAN interface is used, so there is only one VPN overlay member in the SD-WAN zone. Multiple WAN interfaces and VPN overlays could be used.

The VPN interfaces and IP addresses are:

FortiGate	Interface	IP Address
FGT_A (Hub)	hub-phase1	10.10.100.254
FGT_B (Spoke)	spoke11-p1	10.10.100.2
FGT_D (Spoke)	spoke21-p1	10.10.100.3

A recurring speed test is configured that runs on the hub over the dial-up interfaces. The speed tests are performed over the underlay interface from the hub to the spoke. Each spoke is configured to operate as a speed test server and to allow the speed test to run on its underlay interface. The spokes establish BGP peering with the hub over the VPN interface, and advertises its loopback network to the hub. The specific configuration is only shown for FGT_B.

When the speed test is running, routing through the VPN overlay can be bypassed, and route maps are used to filter the routes that are advertised to peers. The spoke's route map does not advertise any routes to the peer, forcing the hub to use others paths to reach the spoke's network.

When no speed tests are running, the spoke's route map allows its network to be advertised on the hub.

When the speed test is complete, the measured egress bandwidth is dynamically applied to the VPN tunnel on the hub, and the result is cached for future use, in case the tunnel is disconnected and reconnected again.

To configure the hub FortiGate (FGT_A):

1. Configure a shaping profile:

```
config firewall shaping-profile
    edit "profile_1"
        config shaping-entries
            edit 1
                set class-id 2
                set priority low
                set guaranteed-bandwidth-percentage 10
                set maximum-bandwidth-percentage 10
            next
        end
        set default-class-id 2
    next
end
```

   Three classes are used in the profile for low, medium, and high priority traffic. Each class is assigned a guaranteed and maximum bandwidth as a percentage of the measured bandwidth from the speed test.

2. Use the shaping profile in the interface:

```
config system interface
    edit "hub-phase1"
        set egress-shaping-profile "profile_1"
    next
end
```

3. Configure a schedule to use for the speed tests:

```
config firewall schedule recurring
    edit "speedtest_recurring"
        set start 01:00
        set end 23:00
        set day monday tuesday wednesday thursday friday saturday
    next
end
```

4. Configure the speed test schedule:

```
config system speed-test-schedule
    edit "hub-phase1"
        set schedules "speedtest_recurring"
        set dynamic-server enable
    next
end
```

To configure the spoke FortiGates (FGT_B and FGT_D):

1. Enable the speed test daemon:

```
config system global
    set speedtest-server enable
end
```

2. Allow speed tests on the interface:

```
config system interface
    edit "port1"
        append allowaccess speed-test
    next
end
```

3. Configure SD-WAN with bypass routing enabled for speed tests on member *spoke11-p1*:

```
config system sdwan
    set speedtest-bypass-routing enable
    config members
        edit 1
            set interface "spoke11-p1"
        next
    end
    config neighbor
        edit "10.10.100.254"
            set member 1
            set mode speedtest
        next
    end
end
```

4. Configure BGP routing:

```
config router route-map
    edit "No_Speed-Test"
        config rule
            edit 1
                set action permit
            next
        end
    next
    edit "Start_Speed-Test"
        config rule
            edit 1
                set action deny
            next
        end
    next
end
```

```
config router bgp
    set as 65412
    config neighbor
        edit "10.10.100.254"
            set remote-as 65412
            set route-map-out "Start_Speed-Test"
            set route-map-out-preferable "No_Speed-Test"
```

```
            next
        end
        config network
            edit 1
                set prefix 2.2.2.2 255.255.255.255
            next
            edit 2
                set prefix 10.1.100.0 255.255.255.0
            next
        end
    end
```

To manually run the speed test:

```
# execute speed-test-dynamic hub-phase1 all y 1000 100
Start testing the speed of each tunnel of hub-phase1
[6400d9] hub-phase1_0: physical_intf=port1, local_ip=172.16.200.1, server_ip=172.16.200.2
Wait for test 6400d9 to finish...
Speed-test result for test ID 6400d9:
    Completed
    measured upload bandwidth is 1002 kbps
    measured time Sun Jun 20 15:56:34 2021

The tested out-bandwidth is more than the set maximum accepted value 1000. Will update the
tunnel's shaper by the set update-outbandwidth-maximum.
Apply shaping profile 'profile_1' with bandwidth 1000 to tunnel hub-phase1_0 of interface
hub-phase1
[6400e0] hub-phase1_1: physical_intf=port1, local_ip=172.16.200.1, server_ip=172.16.200.4
Wait for test 6400e0 to finish...
Speed-test result for test ID 6400e0:
    Completed
    measured upload bandwidth is 1002 kbps
    measured time Sun Jun 20 15:56:39 2021

The tested out-bandwidth is more than the set maximum accepted value 1000. Will update the
tunnel's shaper by the set update-outbandwidth-maximum.
Apply shaping profile 'profile_1' with bandwidth 1000 to tunnel hub-phase1_1 of interface
hub-phase1
```

```
# diagnose netlink interface speed-test-tunnel hub-phase1 all
send speed test request for tunnel 'hub-phase1_0' of 'hub-phase1': 172.16.200.1 ->
172.16.200.2
send speed test request for tunnel 'hub-phase1_1' of 'hub-phase1': 172.16.200.1 ->
172.16.200.4
```

Results

1. Before the speed test starts, FGT_A can receive the route from FGT_B by BGP:

```
# get router info routing-table bgp
Routing table for VRF=0
B       2.2.2.2/32 [200/0] via 10.10.100.2 (recursive via 172.16.200.2, hub-phase1),
00:00:10
B       10.1.100.0/24 [200/0] via 10.10.100.2 (recursive via 172.16.200.2, hub-phase1),
00:00:10
```

2. At the scheduled time, the speed test starts for the hub-phase1 interface from hub to spoke:

```
# diagnose test application forticron 9
Speed test schedules:
    Interface     Server     Update    Up/Down-limit (kbps)                Days
H:M     TOS     Schedule
---------------------------------------------------------------------------------------
-----------------------------------
    hub-phase1    dynamic                                                  1111111
14:41    0x00    speedtest_recurring
Active schedules:
        64002f: hub-phase1(port1) 172.16.200.2     hub-phase1_1
        64002e: hub-phase1(port1) 172.16.200.4     hub-phase1_0
```

The `diagnose debug application speedtest -1` command can be used on both the hub and spokes to check the speed test execution.

3. While the speed test is running, FGT_A does not receive the route from FGT_B by BGP:

```
#  get router info routing-table bgp
Routing table for VRF=0
```

4. Speed tests results can be dynamically applied to the dial-up tunnel for egress traffic shaping:

```
# diagnose vpn tunnel list
------------------------------------------------------
name=hub-phase1_0 ver=2 serial=c 172.16.200.1:0->172.16.200.4:0 tun_id=172.16.200.4 dst_
mtu=1500 dpd-link=on remote_location=0.0.0.0 weight=1
...
egress traffic control:
        bandwidth=737210(kbps) lock_hit=0 default_class=2 n_active_class=3
        class-id=2      allocated-bandwidth=73720(kbps)         guaranteed-
bandwidth=73720(kbps)
                        max-bandwidth=73720(kbps)       current-bandwidth=0(kbps)
                        priority=low    forwarded_bytes=52
                        dropped_packets=0       dropped_bytes=0
        class-id=3      allocated-bandwidth=221163(kbps)        guaranteed-
bandwidth=221162(kbps)
                        max-bandwidth=294883(kbps)      current-bandwidth=0(kbps)
                        priority=medium         forwarded_bytes=0
                        dropped_packets=0       dropped_bytes=0
        class-id=4      allocated-bandwidth=442325(kbps)        guaranteed-
bandwidth=147441(kbps)
                        max-bandwidth=442325(kbps)      current-bandwidth=0(kbps)
                        priority=high   forwarded_bytes=0
                        dropped_packets=0       dropped_bytes=0
------------------------------------------------------
name=hub-phase1_1 ver=2 serial=d 172.16.200.1:0->172.16.200.2:0 tun_id=172.16.200.2 dst_
mtu=1500 dpd-link=on remote_location=0.0.0.0 weight=1
...
egress traffic control:
        bandwidth=726813(kbps) lock_hit=0 default_class=2 n_active_class=3
        class-id=2      allocated-bandwidth=72681(kbps)         guaranteed-
bandwidth=72681(kbps)
                        max-bandwidth=72681(kbps)       current-bandwidth=0(kbps)
                        priority=low    forwarded_bytes=123
                        dropped_packets=0       dropped_bytes=0
        class-id=3      allocated-bandwidth=218044(kbps)        guaranteed-
bandwidth=218043(kbps)
                        max-bandwidth=290725(kbps)      current-bandwidth=0(kbps)
```

```
                        priority=medium         forwarded_bytes=0
                        dropped_packets=0       dropped_bytes=0
        class-id=4      allocated-bandwidth=436087(kbps)        guaranteed-
bandwidth=145362(kbps)
                        max-bandwidth=436087(kbps)      current-bandwidth=0(kbps)
                        priority=high   forwarded_bytes=0
                        dropped_packets=0       dropped_bytes=0
```

5. Speed test results can be cached, indexed, and written to disk:

```
# diagnose test application forticron 10
Speed test results:
1: vdom=root, phase1intf=hub-phase1, peer-id='spoke11-p1', bandwidth=737210, last_
log=1624226603
2: vdom=root, phase1intf=hub-phase1, peer-id='spoke21-p1', bandwidth=726813, last_
log=1624226614
```

```
# diagnose test application forticron 11
Write 2 logs to disk.
```

```
# diagnose test application forticron 12
load 2 results.
```

Disable then reenable the IPsec VPN tunnel and the cached speed test results can be applied to the tunnel again:

```
# diagnose vpn tunnel list
------------------------------------------------------
name=hub-phase1_0 ver=2 serial=c 172.16.200.1:0->172.16.200.4:0 tun_id=172.16.200.4 dst_
mtu=1500 dpd-link=on remote_location=0.0.0.0 weight=1
...
egress traffic control:
        bandwidth=737210(kbps) lock_hit=0 default_class=2 n_active_class=3
------------------------------------------------------
name=hub-phase1_1 ver=2 serial=d 172.16.200.1:0->172.16.200.2:0 tun_id=172.16.200.2 dst_
mtu=1500 dpd-link=on remote_location=0.0.0.0 weight=1
...
egress traffic control:
        bandwidth=726813(kbps) lock_hit=0 default_class=2 n_active_class=3
```

Interface based QoS on individual child tunnels based on speed test results

In a hub and spoke SD-WAN topology that uses dial-up VPN overlays, QoS can be applied on individual tunnels based on the measured bandwidth between the hub and spokes. The FortiGate can use the built in speed test to dynamically populate the egress bandwidth to individual dial-up tunnels from the hub.

A bandwidth limit, derived from the speed test, and a traffic shaping profile can be applied on the dial-up IPsec tunnel interface on the hub. A class ID and percentage based QoS settings can be applied to individual child tunnels using a traffic shaping policy and profile.

CLI commands

If the interface is an IPsec dial-up server, then egress shaping profile type can only be set to `policing`; it cannot be set to `queuing`:

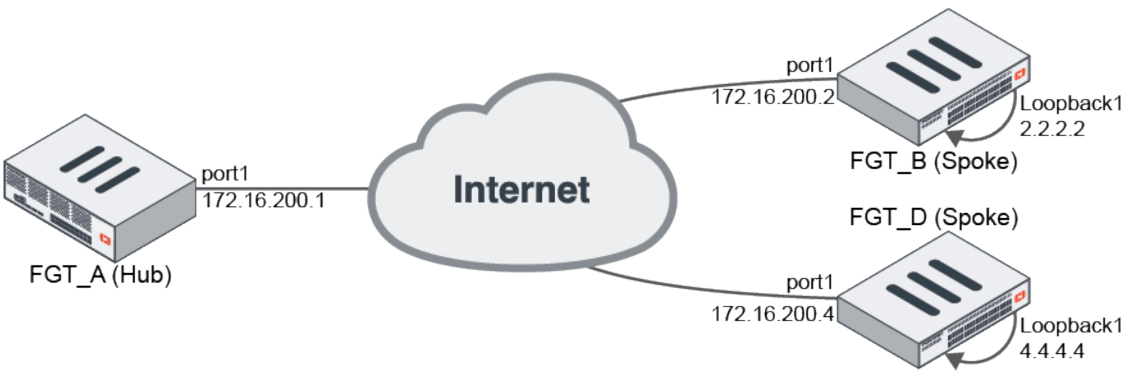
```
config firewall shaping-profile
    edit <profile-name>
```

```
        set type policing
    next
end
```

The outbandwidth value is dynamically obtained from the speed test results for each individual child tunnel, and should not be set manually:

```
config system interface
    edit <dialup-server-phase1-name>
        set egress-shaping-profile <profile-name>
        set outbandwidth <bandwidth>
    next
end
```

Example



In this example, the hub is configured as a VPN dial-up server and both of the spokes are connected to the hub. It is assumed that the VPN configuration is already done, with a dynamic gateway type and kernel device creation (`net-device`) disabled. Only one SD-WAN interface is used, so there is only one VPN overlay member in the SD-WAN zone. Multiple WAN interfaces and VPN overlays could be used.

The VPN interfaces and IP addresses are:

FortiGate	Interface	IP Address
FGT_A (Hub)	hub-phase1	10.10.100.254
FGT_B (Spoke)	spoke11-p1	10.10.100.2
FGT_D (Spoke)	spoke21-p1	10.10.100.3

The hub VPN has two child tunnels, one to each spoke.

The speed test configuration is shown in Speed tests run from the hub to the spokes in dial-up IPsec tunnels on page 664. This example shows applying a shaping profile to the hub's tunnel interface in order to apply interface based traffic shaping to the child tunnels.

A traffic shaping policy is used to match and assign traffic to the classes in the shaping profile.

To configure the hub FortiGate (FGT_A) and check the results:

1. Configure the hub FortiGate (FGT_A) as in Speed tests run from the hub to the spokes in dial-up IPsec tunnels on page 664.

2. Configure the shaping profile:

```
config firewall shaping-profile
    edit "profile_1"
        config shaping-entries
            edit 1
                set class-id 2
                set priority low
                set guaranteed-bandwidth-percentage 10
                set maximum-bandwidth-percentage 10
            next
            edit 2
                set class-id 3
                set priority medium
                set guaranteed-bandwidth-percentage 30
                set maximum-bandwidth-percentage 40
            next
            edit 3
                set class-id 4
                set priority high
                set guaranteed-bandwidth-percentage 20
                set maximum-bandwidth-percentage 60
            next
        end
        set default-class-id 2
    next
end
```

3. Configure a traffic shaping policy:

```
config firewall shaping-policy
    edit 2
        set service "ALL"
        set schedule "always"
        set dstintf "hub-phase1"
        set class-id 3
        set srcaddr "all"
        set dstaddr "all"
    next
end
```

In this example, all traffic through the hub-phase1 interface is put into class ID 3. Class IDs an be assigned based on your traffic requirements.

4. At the schedules time, the speed test will start for the hub-phase1 interface from the hub to the spokes. The speed test results can then be dynamically applied on individual child tunnels as egress traffic shaping, and the class ID percentage based QoS settings is applicable on them as templates.

```
# diagnose vpn tunnel list
------------------------------------------------------
name=hub-phase1_0 ver=2 serial=c 172.16.200.1:0->172.16.200.4:0 tun_id=172.16.200.4 dst_
mtu=1500 dpd-link=on remote_location=0.0.0.0 weight=1
...
egress traffic control:
        bandwidth=737210(kbps) lock_hit=0 default_class=2 n_active_class=3
        class-id=2      allocated-bandwidth=73720(kbps)         guaranteed-
bandwidth=73720(kbps)
                        max-bandwidth=73720(kbps)       current-bandwidth=0(kbps)
```

```
                        priority=low    forwarded_bytes=52
                        dropped_packets=0       dropped_bytes=0
        class-id=3      allocated-bandwidth=221163(kbps)        guaranteed-
bandwidth=221162(kbps)
                        max-bandwidth=294883(kbps)      current-bandwidth=0(kbps)
                        priority=medium         forwarded_bytes=0
                        dropped_packets=0       dropped_bytes=0
        class-id=4      allocated-bandwidth=442325(kbps)        guaranteed-
bandwidth=147441(kbps)
                        max-bandwidth=442325(kbps)      current-bandwidth=0(kbps)
                        priority=high   forwarded_bytes=0
                        dropped_packets=0       dropped_bytes=0
------------------------------------------------------
name=hub-phase1_1 ver=2 serial=d 172.16.200.1:0->172.16.200.2:0 tun_id=172.16.200.2 dst_
mtu=1500 dpd-link=on remote_location=0.0.0.0 weight=1
...
egress traffic control:
        bandwidth=726813(kbps) lock_hit=0 default_class=2 n_active_class=3
        class-id=2      allocated-bandwidth=72681(kbps)         guaranteed-
bandwidth=72681(kbps)
                        max-bandwidth=72681(kbps)       current-bandwidth=0(kbps)
                        priority=low    forwarded_bytes=123
                        dropped_packets=0       dropped_bytes=0
        class-id=3      allocated-bandwidth=218044(kbps)        guaranteed-
bandwidth=218043(kbps)
                        max-bandwidth=290725(kbps)      current-bandwidth=0(kbps)
                        priority=medium         forwarded_bytes=0
                        dropped_packets=0       dropped_bytes=0
        class-id=4      allocated-bandwidth=436087(kbps)        guaranteed-
bandwidth=145362(kbps)
                        max-bandwidth=436087(kbps)      current-bandwidth=0(kbps)
                        priority=high   forwarded_bytes=0
                        dropped_packets=0       dropped_bytes=0
```

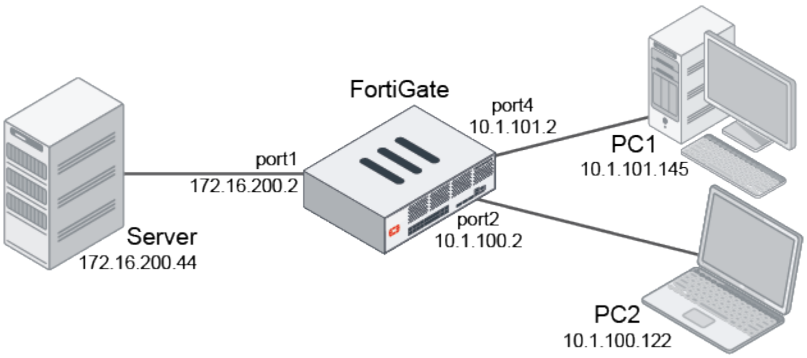
The guaranteed and maximum bandwidths equal 10% of the speed test result, as expected.

Use SSL VPN interfaces in zones

SSL VPN interfaces can be used in zones, simplifying firewall policy configuration in some scenarios.

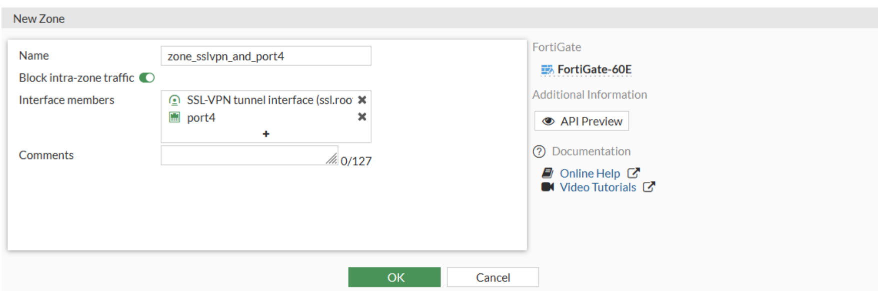
Example

In this example, a zone is created that includes a physical interface (port4) and an SSL VPN interface. The zone is used as the source interface in a firewall policy. PC1 is used for regular access with a firewall policy, and PC2 uses the SSL VPN for access.



To create a zone that includes the port4 and ssl.root interfaces in the GUI:

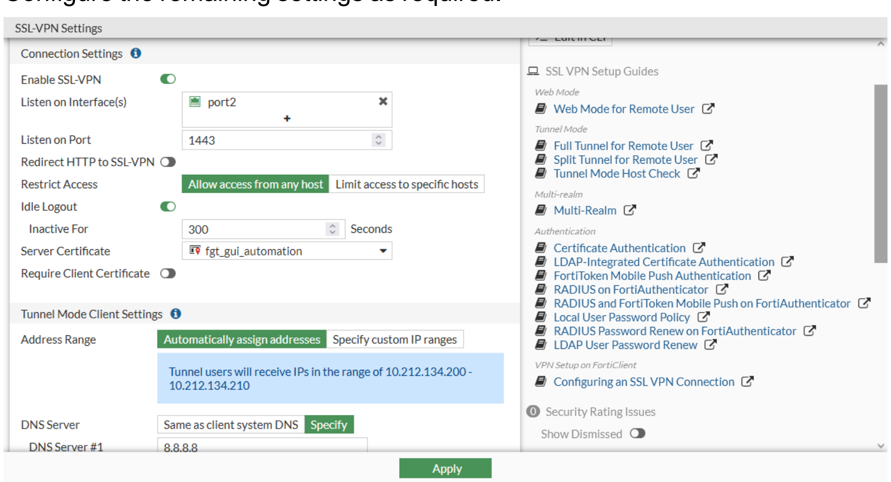
1. Go to *Network > Interfaces* and click *Create New > Zone*.
2. Set the name of the zone, such as *zone_sslvpn_and_port4*.
3. Add *port4* and *ssl.root* to the *Interface members*.



4. Click *OK*.

To configure SSL VPN settings in the GUI:

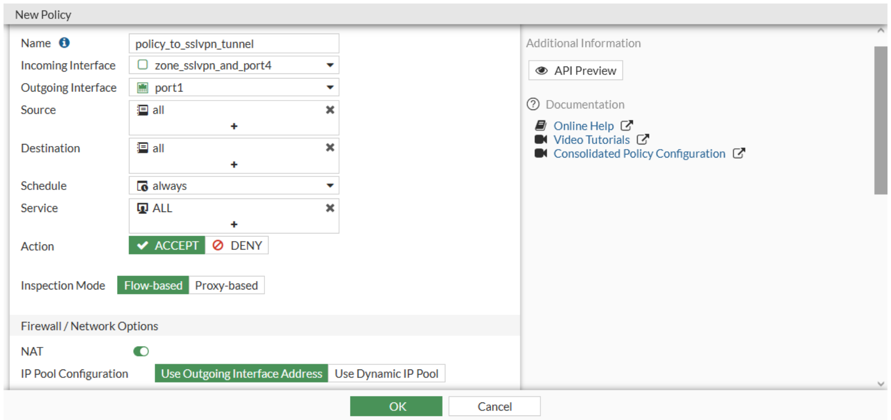
1. Go to *VPN > SSL-VPN Settings*.
2. Set *Listen on Interface(s)* to *port2*.
3. Set *Listen on Port* to *1443*.
4. Configure the remaining settings as required.



5. Click *Apply*.

To configure a firewall policy with the zone as the source interface in the GUI:

1. Go to *Policy & Objects > Firewall Policy* and click *Create New*.
2. Set the policy name, such as *policy_to_sslvpn_tunnel*.
3. Set *Incoming Interface* to *zone_sslvpn_and_port4*.
4. Set *Outgoing Interface* to *port1*.
5. Configure the remaining settings as required.



6. Click *OK*.

To configure the zone, SSL VPN, and policy in the CLI:

1. Create a zone that includes the port4 and ssl.root interfaces:

```
config system zone
    edit "zone_sslvpn_and_port4"
        set interface "port4" "ssl.root"
    next
end
```

2. Configure SSL VPN settings with port2 as the source interface:

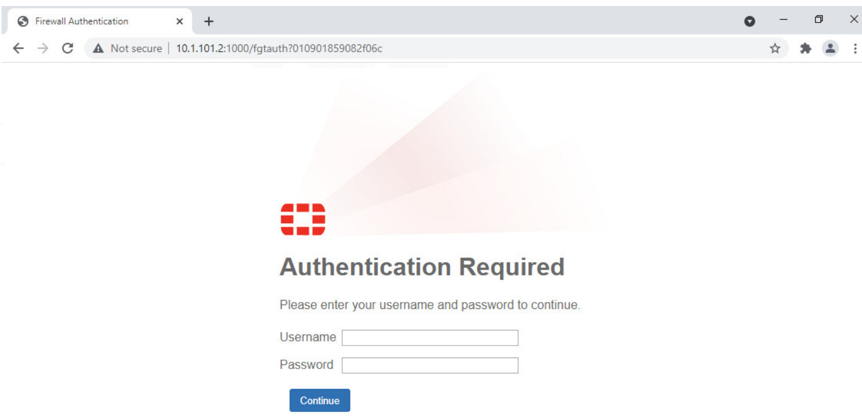
```
config vpn ssl settings
    set servercert "fgt_gui_automation"
    set tunnel-ip-pools "SSLVPN_TUNNEL_ADDR1"
    set tunnel-ipv6-pools "SSLVPN_TUNNEL_IPv6_ADDR1"
    set dns-server1 8.8.8.8
    set dns-server2 8.8.4.4
    set port 1443
    set source-interface "port2"
    set source-address "all"
    set source-address6 "all"
    set default-portal "web-access"
end
```

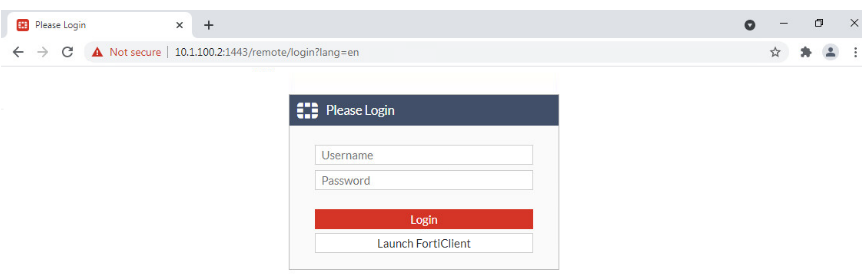
3. Configure a firewall policy with the zone as the source interface:

```
config firewall policy
    edit 2
        set name "policy_to_sslvpn_tunnel"
        set srcintf "zone_sslvpn_and_port4"
        set dstintf "port1"
```

```
            set action accept
            set srcaddr "all"
            set dstaddr "all"
            set schedule "always"
            set service "ALL"
            set logtraffic all
            set nat enable
            set users "u1"
        next
    end
```

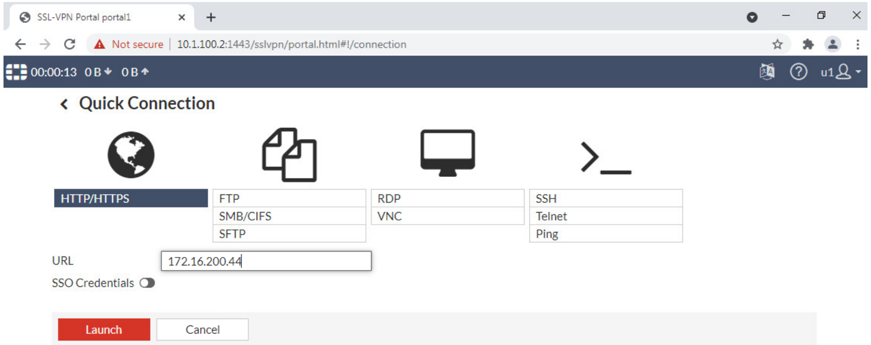
To test the configuration:

1. On PC1, open a browser and try to access the server at 172.16.200.44.
   You are redirected to the authentication page.
   

2. Enter the *Username* and *Password*, then click *Continue*.
   You are redirected back to the server.
3. On PC2, access the SSL VPN web portal.
   

4. Enter the *Username* and *Password*, then click *Login*.

5. Access the server using the bookmark.

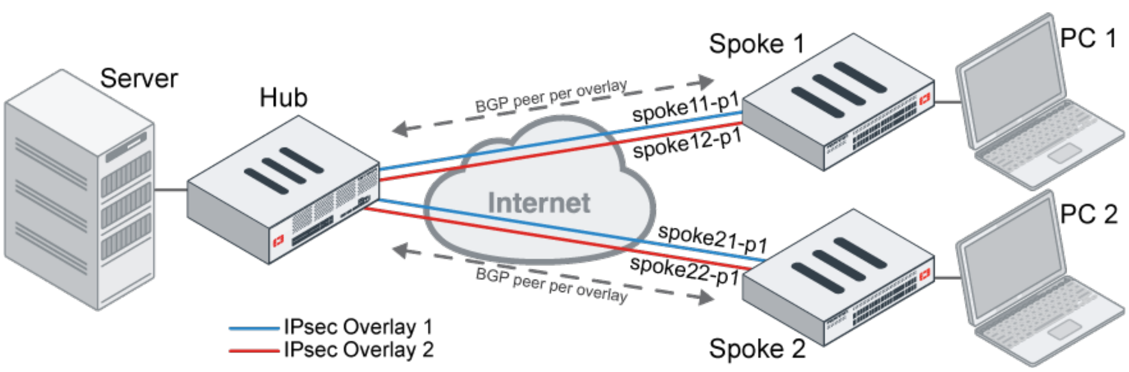


SD-WAN in large scale deployments

Phase 2 selectors can be used to inject IKE routes on the ADVPN shortcut tunnel. When configuration method (`mode-cfg`) is enabled in IPsec phase 1 configuration, enabling `mode-cfg-allow-client-selector` allows custom phase 2 selectors to be configured. By also enabling the addition of a route to the peer destination selector (`add-route`) in the phase 1 configuration, IKE routes based on the phase 2 selectors can be injected. This means that routes do not need to be reflected on the hub to propagate them between spokes, avoiding possible BGP daemon process load issues and improving network scalability in a large-scale ADVPN network.

Route map rules can apply priorities to BGP routes. On the hub, priorities can be set in a route map's rules, and the route map can be applied on BGP routes. This allows the hub to mark the preferred path learned from the spokes with a higher priority, instead of using multiple SD-WAN policy routes on the hub. When a preferred outbound route map (`route-map-out-preferable`) is also configured in an SD-WAN neighbor on the spoke, deploying SD-WAN rules on the hub to steer traffic from the hub to a spoke is unnecessary.

SD-WAN members' local cost can be exchanged on the ADVPN shortcut tunnel so that spokes can use the remote cost as tiebreak to select a preferred shortcut. If multiple shortcuts originate from the same member to different members on the same remote spoke, then the remote cost on the shortcuts is used as the tiebreak to decide which shortcut is preferred.



In this example, SD-WAN is configured on an ADVPN network with a BGP neighbor per overlay.

Instead of reflecting BGP routes with the route-reflector on the hub, when the shortcuts are triggered, IKE routes on the shortcuts are directly injected based on the configured phase 2 selectors to allow routes to be exchanged between spokes.

Routes between the hub and the spokes are exchanged by BGP, and the spokes use the default route to send spoke-to-spoke traffic to the hub and trigger the shortcuts.

Instead of configuring SD-WAN rules on the hub, different priorities are configured on the BGP routes by matching different BGP communities to steer traffic from the hub to the spokes.

To configure Spoke 1:

1. Configure phase 1:

```
config vpn ipsec phase1-interface
    edit "spoke11-p1"
        ...
        set ike-version 2
        set net-device enable
        set add-route enable
        set mode-cfg enable
        set auto-discovery-receiver enable
        set mode-cfg-allow-client-selector enable
        ...
    next
    edit "spoke12-p1"
       ...
        set ike-version 2
        set net-device enable
        set add-route enable
        set mode-cfg enable
        set auto-discovery-receiver enable
        set mode-cfg-allow-client-selector enable
    next
end
```

2. Configure phase 2:

```
config vpn ipsec phase2-interface
    edit "spoke11-p2"
        ...
        set src-name "LAN_Net"
        set dst-name "all"
    next
    edit "spoke12-p2"
        ...
        set src-name "LAN_Net"
        set dst-name "all"
    next
end
```

3. Configure an address group:
   Spoke 1 uses LAN subnet 10.1-3.100.0/24.

```
config firewall addrgrp
    edit "LAN_Net"
        set member "10.1.100.0" "10.2.100.0" "10.3.100.0"
    next
end
```

4. Configure route maps:

- If overlay 1 to the hub is in SLA, attach "65000:1" to the BGP routes advertised to the hub over overlay 1.
- If overlay 2 to the hub is in SLA, attach "65000:2" to the BGP routes advertised to the hub over overlay 2.
- If any overlay to the hub is out of SLA, attach "65000:9999" to the BGP routes advertised to the hub over any overlay.

```
config router route-map
    edit "HUB_CARRIER1"
        config rule
            edit 1
                set set-community "65000:1"
                ...
            next
        end
        ...
    next
    edit "HUB_CARRIER2"
        config rule
            edit 1
                set set-community "65000:2"
                ...
            next
        end
        ...
    next
    edit "HUB_BAD"
        config rule
            edit 1
                set set-community "65000:9999"
                ...
            next
        end
        ...
    next
end
```

5. Configure BGP and SD-WAN members and neighbors:

```
config router bgp
    set as 65412
    config neighbor
        edit "10.10.15.253"
            set remote-as 65412
            set route-map-out "HUB_BAD"
            set route-map-out-preferable "HUB_CARRIER1"
            ...
        next
        edit "10.10.16.253"
            set remote-as 65412
            set route-map-out "HUB_BAD"
            set route-map-out-preferable "HUB_CARRIER2"
            ...
        next
    end
end
```

```
config system sdwan
    config members
```

```
        edit 1
            set interface "spoke11-p1"
        next
        edit 2
            set interface "spoke12-p1"
        next
    end
    config neighbor
        edit "10.10.15.253"
            set member 1
            set health-check "1"
            set sla-id 1
        next
        edit "10.10.16.253"
            set member 2
            set health-check "11"
            set sla-id 1
        next
    end
end
```

To configure Spoke 2:

1. Configure phase 1:

```
config vpn ipsec phase1-interface
    edit "spoke21-p1"
        ...
        set ike-version 2
        set net-device enable
        set add-route enable
        set mode-cfg enable
        set auto-discovery-receiver enable
        set mode-cfg-allow-client-selector enable
        ...
    next
    edit "spoke22-p1"
       ...
        set ike-version 2
        set net-device enable
        set add-route enable
        set mode-cfg enable
        set auto-discovery-receiver enable
        set mode-cfg-allow-client-selector enable
    next
end
```

2. Configure phase 2:

```
config vpn ipsec phase2-interface
    edit "spoke21-p2"
        ...
        set src-name "LAN_Net"
        set dst-name "all"
    next
    edit "spoke22-p2"
        ...
```

```
        set src-name "LAN_Net"
        set dst-name "all"
    next
end
```

3. Configure an address group:

   Spoke 2 uses LAN subnet 192.168.5-7.0/24.

```
config firewall addrgrp
    edit "LAN_Net"
        set member "192.168.5.0" "192.168.6.0" "192.168.7.0"
    next
end
```

4. Configure route maps:
   - If overlay 1 to the hub is in SLA, attach "65000:1" to the BGP routes advertised to the hub over overlay 1.
   - If overlay 2 to the hub is in SLA, attach "65000:2" to the BGP routes advertised to the hub over overlay 2.
   - If any overlay to the hub is out of SLA, attach "65000:9999" to the BGP routes advertised to the hub over any overlay.

```
config router route-map
    edit "HUB_CARRIER1"
        config rule
            edit 1
                set set-community "65000:1"
                ...
            next
        end
        ...
    next
    edit "HUB_CARRIER2"
        config rule
            edit 1
                set set-community "65000:2"
                ...
            next
        end
        ...
    next
    edit "HUB_BAD"
        config rule
            edit 1
                set set-community "65000:9999"
                ...
            next
        end
        ...
    next
end
```

5. Configure BGP and SD-WAN members and neighbors:

```
config router bgp
    set as 65412
    config neighbor
        edit "10.10.15.253"
            set remote-as 65412
```

```
            set route-map-out "HUB_BAD"
            set route-map-out-preferable "HUB_CARRIER1"
            ...
        next
        edit "10.10.16.253"
            set remote-as 65412
            set route-map-out "HUB_BAD"
            set route-map-out-preferable "HUB_CARRIER2"
            ...
        next
    end
end
```

```
config system sdwan
    config members
        edit 1
            set interface "spoke21-p1"
            set cost 100
        next
        edit 2
            set interface "spoke22-p1"
            set cost 200
        next
    end
    config neighbor
        edit "10.10.15.253"
            set member 1
            set health-check "1"
            set sla-id 1
        next
        edit "10.10.16.253"
            set member 2
            set health-check "11"
            set sla-id 1
        next
    end
end
```

To configure the hub:

1. Configure the route maps:
    - Set the priority to 100 for routes with community 65000:1, indicating that they are in SLA for overlay 1.
    - Set the priority to 200 for routes with community 65000:2, indicating that they are in SLA for overlay 2.
    - Set the priority to 9999 for routes with community 65000:9999, indicating that they are out of SLA for any overlay.

```
config router route-map
     edit "Set_Pri"
        config rule
            edit 1
                set match-community "comm_65000:1"
                set set-priority 100
            next
            edit 2
                set match-community "comm_65000:2"
                set set-priority 200
```

```
            next
            edit 3
                set match-community "comm_65000:9999"
                set set-priority 9999
            next
        end
    next
end
```

2. Configure BGP:

```
config router bgp
    set as 65412
    config neighbor-group
        edit "advpn"
            set remote-as 65412
            set route-map-in "Set_Pri"
            ...
        next
        edit "advpn2"
            set remote-as 65412
            set route-map-in "Set_Pri"
            ...
        next
    end
    config neighbor-range
        edit 1
            set prefix 10.10.15.0 255.255.255.0
            set neighbor-group "advpn"
        next
        edit 2
            set prefix 10.10.16.0 255.255.255.0
            set neighbor-group "advpn2"
        next
    end
end
```

To test the configuration:

1. Check the routing tables on the spokes:

   Spoke 1:

```
spoke-1 (root) # get router info routing-table all
B*      0.0.0.0/0 [200/0] via 10.10.15.253 (recursive is directly connected, spoke11-
p1), 00:01:17, [1/0]         // default route to hub
                         [200/0] via 10.10.16.253 (recursive is directly connected,
spoke12-p1), 00:01:17, [1/0]
B       9.0.0.0/24 [200/0] via 10.10.15.253 (recursive is directly connected, spoke11-
p1), 00:01:17, [1/0]        // route to the server behind hub
                          [200/0] via 10.10.16.253 (recursive is directly connected,
spoke12-p1), 00:01:17, [1/0]
C       10.1.100.0/24 is directly connected, port2            // route to PC 1
C       10.10.15.0/24 is directly connected, spoke11-p1       // overlay 1
C       10.10.15.1/32 is directly connected, spoke11-p1
C       10.10.16.0/24 is directly connected, spoke12-p1       // overlay 2
C       10.10.16.1/32 is directly connected, spoke12-p1
```

Spoke 2:

```
spoke-2 (root) # get router info routing-table all
B*      0.0.0.0/0 [200/0] via 10.10.15.253 (recursive is directly connected, spoke21-
p1), 00:46:14, [1/0]         // default route to hub
                         [200/0] via 10.10.16.253 (recursive is directly connected,
spoke22-p1), 00:46:14, [1/0]
B       9.0.0.0/24 [200/0] via 10.10.15.253 (recursive is directly connected, spoke21-
p1), 00:46:18, [1/0]        // route to the server behind hub
                          [200/0] via 10.10.16.253 (recursive is directly connected,
spoke22-p1), 00:46:18, [1/0]
C       10.10.15.0/24 is directly connected, spoke21-p1        // overlay 1
C       10.10.15.2/32 is directly connected, spoke21-p1
C       10.10.16.0/24 is directly connected, spoke22-p1        // overlay 2
C       10.10.16.2/32 is directly connected, spoke22-p1
C       192.168.5.0/24 is directly connected, port2            // route to PC 2
```

2. Send traffic from PC 1 to PC 2 and trigger the shortcut:

   The IKE routes on the shortcut are directly injected based on the phase 2 selectors, and spoke-to-spoke traffic then goes directly through the shortcut instead of going through the hub.

   Spoke 1:

```
spoke-1 (root) # get router info routing-table static
S       192.168.5.0/24 [15/0] via spoke11-p1_0 tunnel 172.16.200.4 vrf 0, [1/0]
S       192.168.6.0/24 [15/0] via spoke11-p1_0 tunnel 172.16.200.4 vrf 0, [1/0]
S       192.168.7.0/24 [15/0] via spoke11-p1_0 tunnel 172.16.200.4 vrf 0, [1/0]
```

```
spoke-1 (root) # diagnose sniffer packet any 'host 192.168.5.44' 4
interfaces=[any]
filters=[host 192.168.5.44]
1.446306 port2 in 10.1.100.22 -> 192.168.5.44: icmp: echo request
1.446327 spoke11-p1_0 out 10.1.100.22 -> 192.168.5.44: icmp: echo request
1.446521 spoke11-p1_0 in 192.168.5.44 -> 10.1.100.22: icmp: echo reply
1.446536 port2 out 192.168.5.44 -> 10.1.100.22: icmp: echo reply
```

   Spoke 2:

```
spoke-2 (root) # get router info routing-table static
S       10.1.100.0/24 [15/0] via spoke21-p1_0 tunnel 10.10.15.1 vrf 0, [1/0]
S       10.2.100.0/24 [15/0] via spoke21-p1_0 tunnel 10.10.15.1 vrf 0, [1/0]
S       10.3.100.0/24 [15/0] via spoke21-p1_0 tunnel 10.10.15.1 vrf 0, [1/0]
```

3. Confirm that the overlays are in SLA on the spokes:

   Spoke 1:

```
spoke-1 (root) # diagnose sys sdwan  neighbor
Neighbor(10.10.15.253): member(1)role(standalone)
        Health-check(1:1)  sla-pass selected alive
Neighbor(10.10.16.253): member(2)role(standalone)
        Health-check(11:1)  sla-pass selected alive
```

   Spoke 2:

```
spoke-2 (root) # diagnose sys sdwan neighbor
Neighbor(10.10.15.253): member(1)role(standalone)
        Health-check(1:1)  sla-pass selected alive
Neighbor(10.10.16.253): member(2)role(standalone)
        Health-check(11:1)  sla-pass selected alive
```

4. On the hub, check that the routes received from the spokes have the expected priorities:

```
hub (root) # diagnose ip route list | grep proto=11
tab=254 vf=0 scope=0 type=1 proto=11 prio=100 0.0.0.0/0.0.0.0/0->10.1.100.0/24
pref=0.0.0.0 gwy=10.10.15.1 dev=101(hub-phase1)
tab=254 vf=0 scope=0 type=1 proto=11 prio=200 0.0.0.0/0.0.0.0/0->10.1.100.0/24
pref=0.0.0.0 gwy=10.10.16.1 dev=102(hub2-phase1)
tab=254 vf=0 scope=0 type=1 proto=11 prio=100 0.0.0.0/0.0.0.0/0->192.168.5.0/24
pref=0.0.0.0 gwy=10.10.15.2 dev=101(hub-phase1)
tab=254 vf=0 scope=0 type=1 proto=11 prio=200 0.0.0.0/0.0.0.0/0->192.168.5.0/24
pref=0.0.0.0 gwy=10.10.16.2 dev=102(hub2-phase1)
```

The priority set by the hub's route map is based on the community string received from the spoke. The route with a lower priority value is selected, so traffic to Spoke 1 goes out on the hub-phase1 tunnel:

```
hub (root) # diagnose sniffer packet any 'host 9.0.0.2' 4
interfaces=[any]
filters=[host 9.0.0.2]
2.735456 R190 in 9.0.0.2 -> 10.1.100.22: icmp: echo request
2.735508 hub-phase1 out 9.0.0.2 -> 10.1.100.22: icmp: echo request
2.735813 hub-phase1 in 10.1.100.22 -> 9.0.0.2: icmp: echo reply
2.735854 R190 out 10.1.100.22 -> 9.0.0.2: icmp: echo reply
```

5. If overlay 1 goes out of SLA, the priorities of the routes on the hub are updated and traffic from the hub to Spoke 1 goes through overlay 2:

Spoke 1:

```
spoke-1 (root) # diagnose sys sdwan  neighbor
Neighbor(10.10.15.253): member(1)role(standalone)
        Health-check(1:1)  sla-fail alive
Neighbor(10.10.16.253): member(2)role(standalone)
        Health-check(11:1)  sla-pass selected alive
```

Spoke 2:

```
spoke-2 (root) # diagnose sys sdwan neighbor
Neighbor(10.10.15.253): member(1)role(standalone)
        Health-check(1:1)  sla-fail alive
Neighbor(10.10.16.253): member(2)role(standalone)
        Health-check(11:1)  sla-pass selected alive
```

Hub:

```
hub (root) # diagnose ip route list | grep proto=11
tab=254 vf=0 scope=0 type=1 proto=11 prio=200 0.0.0.0/0.0.0.0/0->10.1.100.0/24
pref=0.0.0.0 gwy=10.10.16.1 dev=102(hub2-phase1)
tab=254 vf=0 scope=0 type=1 proto=11 prio=9999 0.0.0.0/0.0.0.0/0->10.1.100.0/24
pref=0.0.0.0 gwy=10.10.15.1 dev=101(hub-phase1)
tab=254 vf=0 scope=0 type=1 proto=11 prio=200 0.0.0.0/0.0.0.0/0->192.168.5.0/24
pref=0.0.0.0 gwy=10.10.16.2 dev=102(hub2-phase1)
tab=254 vf=0 scope=0 type=1 proto=11 prio=9999 0.0.0.0/0.0.0.0/0->192.168.5.0/24
pref=0.0.0.0 gwy=10.10.15.2 dev=101(hub-phase1)
```

```
hub (root) # diagnose sniffer packet any 'host 9.0.0.2' 4
interfaces=[any]
filters=[host 9.0.0.2]
3.550181 R190 in 9.0.0.2 -> 10.1.100.22: icmp: echo request
3.550234 hub2-phase1 out 9.0.0.2 -> 10.1.100.22: icmp: echo request
```

```
3.550713 hub2-phase1 in 10.1.100.22 -> 9.0.0.2: icmp: echo reply
3.550735 R190 out 10.1.100.22 -> 9.0.0.2: icmp: echo reply
```

6. Trigger shortcuts between Spoke 1 and Spoke 2:
   - Shortcuts spoke11-p1_1 and spoke11-p1_0 originate from spoke11-p1.
   - spoke11-p1_1 corresponds to spoke21-p1_0 on Spoke 2.
   - spoke11-p1_0 corresponds to spoke22-p1_0 on Spoke 2.

   Spoke 1:

```
spoke-1 (root) # diagnose sys sdwan service

Service(1): Address Mode(IPV4) flags=0x200 use-shortcut-sla
 Tie break: cfg
  Gen(12), TOS(0x0/0x0), Protocol(0: 1->65535), Mode(sla), sla-compare-number
  Service role: standalone
  Member sub interface(4):
    3: seq_num(1), interface(spoke11-p1):
       1: spoke11-p1_0(75)
       2: spoke11-p1_1(76)
  Members(4):
    1: Seq_num(1 spoke11-p1_1), alive, sla(0x1), gid(0), remote cost(100), cfg_order(0),
local cost(0), selected
    2: Seq_num(1 spoke11-p1_0), alive, sla(0x1), gid(0), remote cost(200), cfg_order(0),
local cost(0), selected
    3: Seq_num(1 spoke11-p1), alive, sla(0x1), gid(0), cfg_order(0), local cost(0),
selected
    4: Seq_num(2 spoke12-p1), alive, sla(0x2), gid(0), cfg_order(1), local cost(0),
selected
  Src address(1):
        10.1.100.0-10.1.100.255

  Dst address(1):
        0.0.0.0-255.255.255.255
```

   Spoke 2:

```
spoke-2 (root) # diagnose sys sdwan service

Service(1): Address Mode(IPV4) flags=0x200 use-shortcut-sla
 Tie break: cfg
  Gen(9), TOS(0x0/0x0), Protocol(0: 1->65535), Mode(sla), sla-compare-number
  Service role: standalone
  Member sub interface(4):
    2: seq_num(1), interface(spoke21-p1):
       1: spoke21-p1_0(68)
    4: seq_num(2), interface(spoke22-p1):
       1: spoke22-p1_0(67)
  Members(4):
    1: Seq_num(1 spoke21-p1_0), alive, sla(0x1), gid(0), cfg_order(0), local cost(100),
selected
    2: Seq_num(1 spoke21-p1), alive, sla(0x1), gid(0), cfg_order(0), local cost(100),
selected
    3: Seq_num(2 spoke22-p1_0), alive, sla(0x2), gid(0), cfg_order(1), local cost(200),
selected
    4: Seq_num(2 spoke22-p1), alive, sla(0x2), gid(0), cfg_order(1), local cost(200),
selected
```

```
  Src address(1):
        192.168.5.0-192.168.5.255

  Dst address(1):
        0.0.0.0-255.255.255.255
```

7. On Spoke 2, increase the cost of spoke21-p1_0 to 300.

```
spoke-2 (root) # config system sdwan
    config members
        edit 1
            set interface "spoke21-p1"
            set cost 300
        next
    end
end
```

The new cost is learned by the spoke11-p1_1 shortcut on Spoke 1, and that shortcut is no longer preferred due to its higher remote cost:

Spoke 1:

```
spoke-1 (root) # diagnose sys sdwan service

Service(1): Address Mode(IPV4) flags=0x200 use-shortcut-sla
 Tie break: cfg
  Gen(13), TOS(0x0/0x0), Protocol(0: 1->65535), Mode(sla), sla-compare-number
  Service role: standalone
  Member sub interface(4):
    3: seq_num(1), interface(spoke11-p1):
       1: spoke11-p1_0(78)
       2: spoke11-p1_1(79)
  Members(4):
    1: Seq_num(1 spoke11-p1_0), alive, sla(0x1), gid(0), remote cost(200), cfg_order(0),
local cost(0), selected
    2: Seq_num(1 spoke11-p1_1), alive, sla(0x1), gid(0), remote cost(300), cfg_order(0),
local cost(0), selected
    3: Seq_num(1 spoke11-p1), alive, sla(0x1), gid(0), cfg_order(0), local cost(0),
selected
    4: Seq_num(2 spoke12-p1), alive, sla(0x2), gid(0), cfg_order(1), local cost(0),
selected
  Src address(1):
        10.1.100.0-10.1.100.255

  Dst address(1):
        0.0.0.0-255.255.255.255
```

Spoke 2:

```
spoke-2 (root) # diagnose sys sdwan service

Service(1): Address Mode(IPV4) flags=0x200 use-shortcut-sla
 Tie break: cfg
  Gen(1), TOS(0x0/0x0), Protocol(0: 1->65535), Mode(sla), sla-compare-number
  Service role: standalone
  Member sub interface(4):
    2: seq_num(2), interface(spoke22-p1):
       1: spoke22-p1_0(70)
```

```
        4: seq_num(1), interface(spoke21-p1):
           1: spoke21-p1_0(71)
      Members(4):
        1: Seq_num(2 spoke22-p1_0), alive, sla(0x2), gid(0), cfg_order(1), local cost(200),
    selected
        2: Seq_num(2 spoke22-p1), alive, sla(0x2), gid(0), cfg_order(1), local cost(200),
    selected
        3: Seq_num(1 spoke21-p1_0), alive, sla(0x1), gid(0), cfg_order(0), local cost(300),
    selected
        4: Seq_num(1 spoke21-p1), alive, sla(0x1), gid(0), cfg_order(0), local cost(300),
    selected
      Src address(1):
            192.168.5.0-192.168.5.255

      Dst address(1):
            0.0.0.0-255.255.255.255
```

Advanced configuration

The following topics provide instructions on SD-WAN advanced configuration:

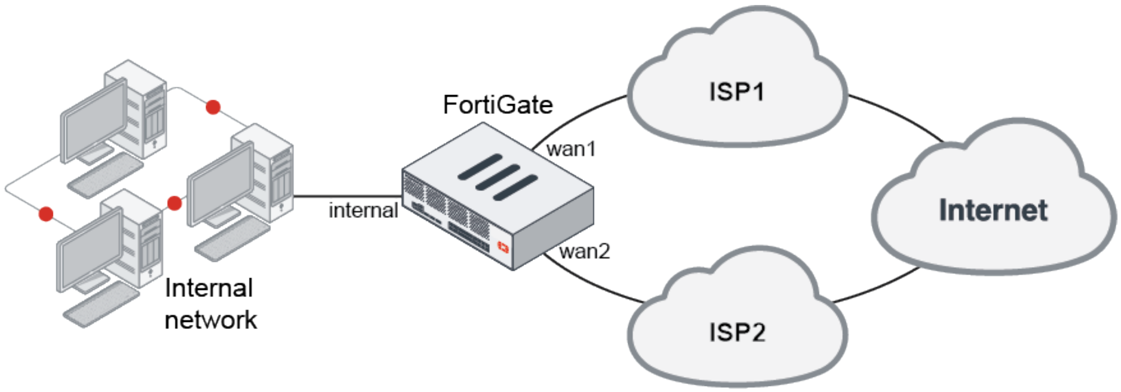
See also Packet distribution for aggregate static IPsec tunnels in SD-WAN on page 1487.

SD-WAN with FGCP HA

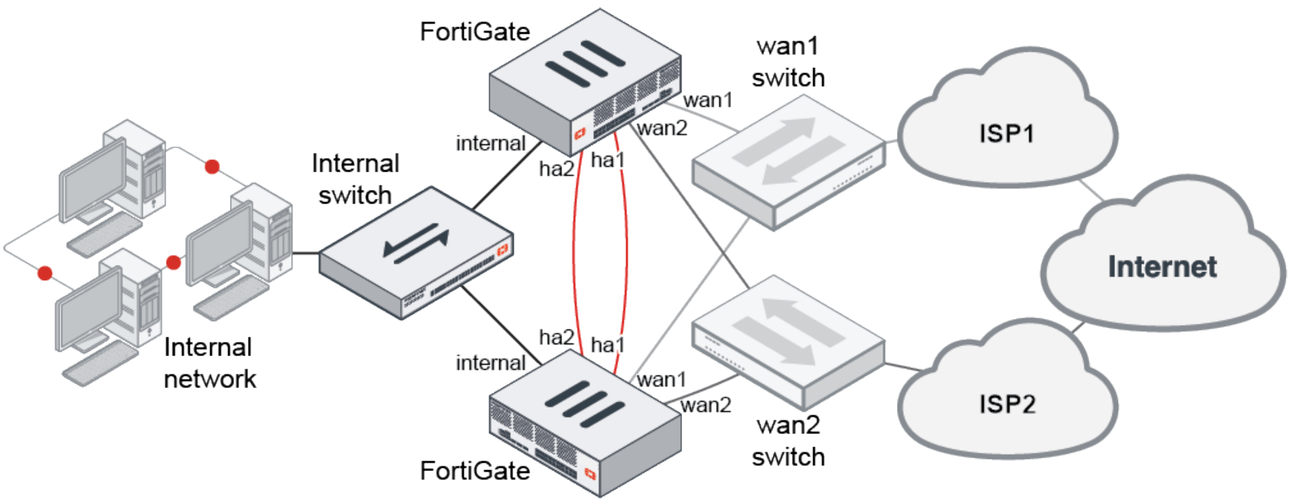
This example shows how to convert a standalone FortiGate SD-WAN solution to a FGCP HA cluster with full-mesh WAN set up. This configuration allows you to load balance your internet traffic between multiple ISP links. It also provides redundancy for your internet connection if your primary ISP in unavailable, or if one of the FortiGates in the HA cluster fails.

This example assumes that a standalone FortiGate has already been configured for SD-WAN by following the SD-WAN quick start on page 487.

Standalone FortiGate:



FGCP HA cluster:



The following devices are required to convert the topology to HA:

- A second FortiGate that is the same model running the same firmware version.
- Two switches for connecting each FortiGate's WAN interface to the corresponding ISP modem.

Before you begin:

- Ensure that the licenses and subscriptions on both HA members match.
- Ensure that there are one or more ports reserved for HA heartbeat.
- Ensure you have physical access to both HA members.

Enabling HA and re-cabling the WAN interfaces will cause network interruptions. This procedure should be performed during a maintenance window.

Configuring the standalone FortiGate for HA

After running the following commands, the FortiGate negotiates to establish an HA cluster. You might temporarily lose connectivity with the FortiGate as FGCP negotiations take place and the MAC addresses of the FortiGate interfaces are changed to HA virtual MAC addresses.

This configurations sets the HA mode to active-passive.

The ha1 and ha2 interfaces are configured as the heartbeat interfaces, with priorities set to 200 and 100 respectively. Setting different priorities for the heartbeat interfaces is a best practice, but is not required.

If you have more than one cluster on the same network, each cluster should have a different group ID. Changing the group ID changes the cluster interface's virtual MAC addresses. If the group IP causes a MAC address conflict on your network, select a different group ID.

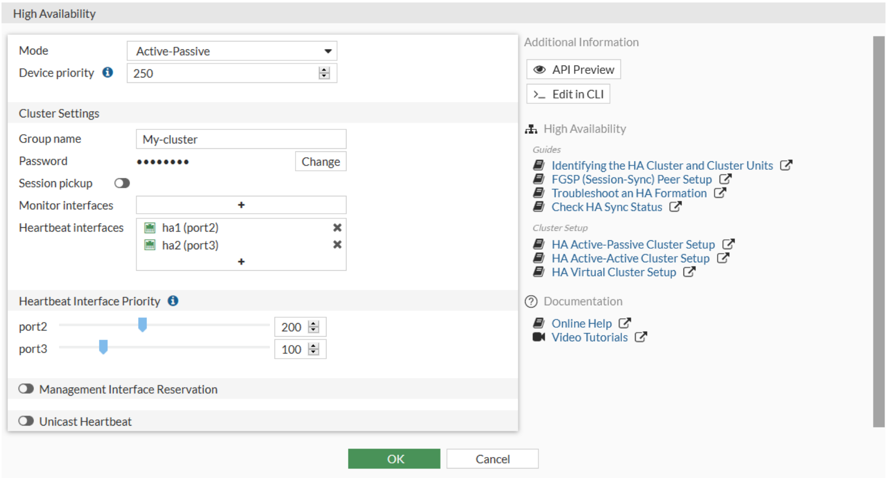
Enabling override and increasing the device priority means that this FortiGate always becomes the primary unit.

To configure the standalone FortiGate for HA in the GUI:

1. Go to *System* > *Settings* and change the *Host name* so that the FortiGate can be easily identified as the primary unit.
2. Go to *System* > *HA* and configure the following options:

Mode	Active-Passive
Device priority	250
Group name	My-cluster
Password	<password>
Heartbeat interfaces	ha1 and ha2
Heartbeat Interface Priority	**port2** (ha1): 200 **port3** (ha2): 100

Override and the group ID can only be configured from the CLI.



3. Click *OK*.
   Connectivity with the FortiGate will temporarily be lost.

To configure the standalone FortiGate for HA in the CLI:

1. Change the host name so that the FortiGate can be easily identified:

```
config system global
    set hostname primary_FG
end
```

2. Configure HA:

```
config system ha
    set mode a-p
    set group-id 100
    set group-name My-cluster
    set password <password>
    set priority 250
    set override enable
    set hbdev ha1 200 ha2 100
end
```

If HA mode does not start after running the above steps, ensure that none of the FortiGate's interfaces use DHCP or PPPoE addressing.

Configuring the secondary FortiGate for HA

The secondary FortiGate must be the same model and running the same firmware version as the primary FortiGate. The HA settings are the same as the for the primary unit, except the secondary device has a lower priority and override is not enabled.

It is best practice to reset the FortiGate to factory default settings prior to configuring HA. This reduces the chance of synchronization problems.

```
# execute factoryreset
This operation will reset the system to factory default!
Do you want to continue? (y/n) y
```

This is unnecessary if the device is new from the factory.

To configure the secondary FortiGate for HA in the GUI:

1. Go to *System* > *Settings* and change the *Host name* so that the FortiGate can be easily identified as the backup unit.
2. Go to *System* > *HA* and configure the options the same as for the primary FortiGate, except with a lower priority:

Mode	Active-Passive
Device priority	128
Group name	My-cluster
Password	<password>
Heartbeat interfaces	ha1 and ha2
Heartbeat Interface Priority	**port2** (ha1): 200 **port3** (ha2): 100

3. Click *OK*.

To configure the secondary FortiGate for HA in the CLI:

1. Change the host name so that the secondary FortiGate can be easily identified:

```
config system global
    set hostname secondary_FG
end
```

2. Configure HA:

```
config system ha
    set mode a-p
    set group-id 100
    set group-name My-cluster
    set password <password>
    set priority 128
    set hbdev ha1 200 ha2 100
end
```

Connecting the heartbeat interfaces between the FortiGates

To connect and check the heartbeat interfaces:

1. Connect the heartbeat interfaces ha1 and ha2 between the primary and secondary FortiGate.
   a. An HA primary device is selected. Because the primary FortiGate has a higher priority and override enabled, it assumes the role of HA primary.

   b. The secondary FortiGate synchronizes its configuration from the primary device.

2. Verify that the checksums match between the primary and secondary FortiGates:

```
# diagnose sys ha checksum cluster

================== FG5H0XXXXXXXXXX0 ==================

is_manage_primary()=1, is_root_primary()=1
debugzone
global: 2b e9 81 38 c2 9d 4f db b7 0e 1f 49 42 c6 1e fb
root: af a6 48 c5 c2 9a 8b 81 a5 53 fb 27 e9 ae 01 6a
all: 89 1f 63 77 48 8a 30 ee 57 06 ca eb 71 e6 8e ad

checksum
global: 2b e9 81 38 c2 9d 4f db b7 0e 1f 49 42 c6 1e fb
root: af a6 48 c5 c2 9a 8b 81 a5 53 fb 27 e9 ae 01 6a
all: 89 1f 63 77 48 8a 30 ee 57 06 ca eb 71 e6 8e ad

================== FG5H0XXXXXXXXXX1 ==================

is_manage_primary()=0, is_root_primary()=0
debugzone
global: 2b e9 81 38 c2 9d 4f db b7 0e 1f 49 42 c6 1e fb
root: af a6 48 c5 c2 9a 8b 81 a5 53 fb 27 e9 ae 01 6a
all: 89 1f 63 77 48 8a 30 ee 57 06 ca eb 71 e6 8e ad

checksum
global: 2b e9 81 38 c2 9d 4f db b7 0e 1f 49 42 c6 1e fb
root: af a6 48 c5 c2 9a 8b 81 a5 53 fb 27 e9 ae 01 6a
all: 89 1f 63 77 48 8a 30 ee 57 06 ca eb 71 e6 8e ad
```

   If all of the cluster members have identical checksums, then their configurations are synchronized. If the checksums are not the same, wait for a few minutes, then repeat the command. Some parts of the configuration might take a significant amount of time to synchronize (tens of minutes).

Connecting other traffic interfaces

After the device configurations are synchronized, you can connect the rest of the traffic interfaces. Making these connections will disrupt traffic as cables are disconnected and reconnected.

Switches must be used between the cluster and the ISPs, and between the cluster and the internal network, as shown in the topology diagram.

Checking cluster operations

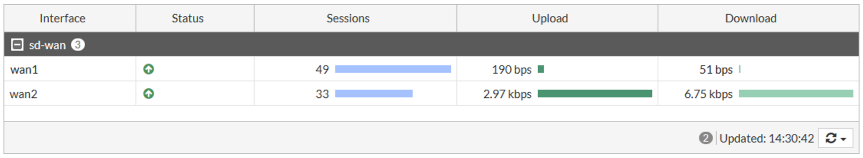
The *HA Status* dashboard widget shows the synchronization status. Hover over the host names of each FortiGate in the widget to verify that they are synchronized and have the same checksum.

To view more information about the cluster status, including the number of sessions passing through the cluster members, go to *System* > *HA*.

See Check HA synchronization status on page 2044 for more information.

Results

1. Browse the internet on a computer in the internal network.
2. Go to *Network > SD-WAN* and select the *SD-WAN Zones* tab to see the bandwidth, volume, and sessions for traffic on the SD-WAN interfaces. See Results on page 492 for details.
3. Go to *Dashboard > Network*, and expand the *SD-WAN* widget to see information about each interface, such as the number of sessions and the bit rate.



Testing HA failover

All traffic should currently be flowing through the primary FortiGate. If it becomes unavailable, traffic fails over to the secondary FortiGate. When the primary FortiGate rejoins the cluster, the secondary FortiGate continues to operate as the primary FortiGate.

To test this, ping a reliable IP address from a computer in the internal network, and then power off the primary FortiGate.

There will be a momentary pause in the ping results until traffic diverts to the backup FortiGate, allowing the ping traffic to continue:

```
64 bytes from 184.25.76.114: icmp_seq=69 ttl=52 time=8.719 ms\
64 bytes from 184.25.76.114: icmp_seq=70 ttl=52 time=8.822 ms\
64 bytes from 184.25.76.114: icmp_seq=74 ttl=52 time=8.901 ms\
Request timeout for icmp_seq 75\
64 bytes from 184.25.76.114: icmp_seq=76 ttl=52 time=8.860 ms\
64 bytes from 184.25.76.114: icmp_seq=77 ttl=52 time=9.174 ms\
64 bytes from 184.25.76.114: icmp_seq=83 ttl=52 time=8.639 ms}
```

If you are using port monitoring, you can also unplug the primary FortiGate's internet facing interface to test failover.

After the secondary FortiGate becomes the primary, you can log into the cluster using the same IP address as before the fail over. If the primary FortiGate is powered off, you will be logged into the backup FortiGate. Check the host name to verify what device you have logged into. The FortiGate continues to operate in HA mode, and if you restart the primary FortiGate, it will rejoin the cluster and act as the backup FortiGate. Traffic is not disrupted when the restarted FortiGate rejoins the cluster.

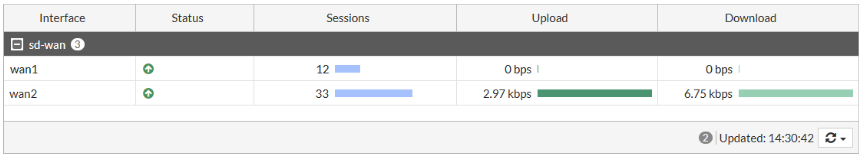
You can also use the CLI to force an HA failover. See Force HA failover for testing and demonstrations on page 2069 for information.

Testing ISP failover

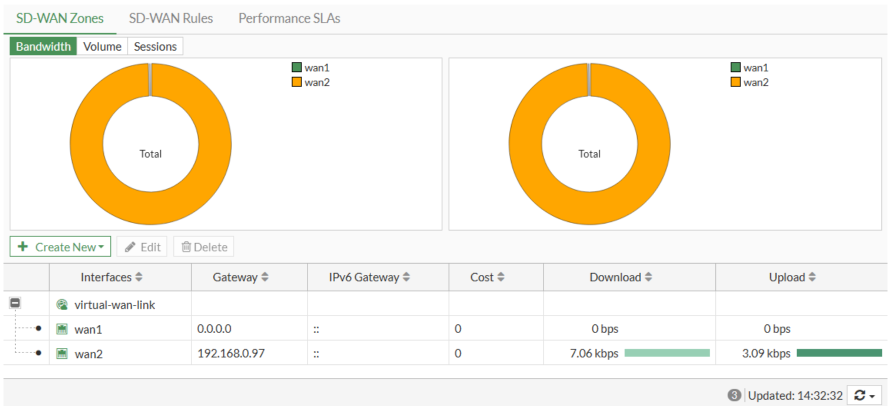
To test a failover of the redundant internet configuration, you need to simulate a failed internet connection to one of the ports. You can do this by disconnecting power from the wan1 switch, or by disconnecting the wan1 interfaces of both FortiGates from ISP1.

After disconnecting, verify that users still have internet access

- Go to *Dashboard > Network*, and expand the *SD-WAN* widget. The *Upload* and *Download* columns for wan1 show that traffic is not going through that interface.



- Go to *Network > SD-WAN* and select the *SD-WAN Zones* tab. The *Bandwidth*, *Volume*, and *Sessions* tabs show that traffic is entirely diverted to wan2.



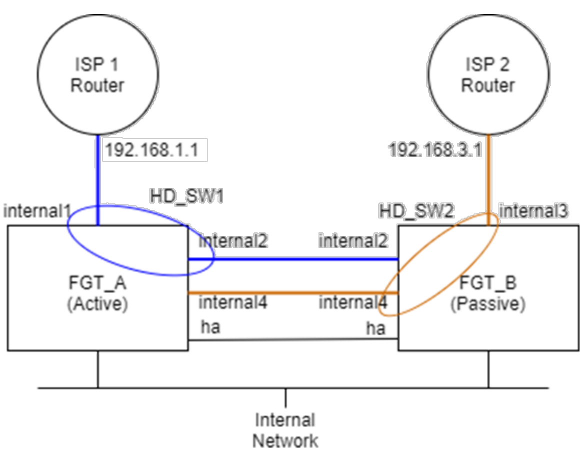
Users on the network should not notice the wan1 failure. If you are using the wan1 gateway IP address to connect to the administrator dashboard, it will appear as though you are still connecting through wan1.

After verifying a successful failover, reestablish the connection to ISP1.

Configuring SD-WAN in an HA cluster using internal hardware switches

In this SD-WAN configuration, two FortiGates in an active-passive (A-P) HA pair are used to provide hardware redundancy. Instead of using external switches to provide a mesh network connection to the ISP routers, the FortiGates use their built-in hardware switches to connect to the ISP routers.

Only FortiGate models that have hardware switches can be used for this solution. Ports in a software switch are not in a forwarding state when a FortiGate is acting as a secondary device in a A-P cluster.



In this topology:

- Two hardware switches are created, HD_SW1 and HD_SW2.
- HD_SW1 is used to connect to ISP 1 Router and includes the internal1 and internal2 ports.
- HD_SW2 is used to connect to ISP 2 Router and includes the internal3 and internal4 ports.
- Another interface on each device is used as the HA heartbeat interface, connecting the two FortiGates in HA.

The FortiGates create two hardware switches to connect to ISP 1 and ISP2. When FGT_A is the primary device, it reaches ISP 1 on internal1 in HD_SW1 and ISP 2 on internal4 in HD_SW2. When FGT_B is the primary device, it reaches ISP 1 on internal2 in HD_SW1 and ISP 2 on internal3 on HD_SW2.

HA failover

This is not a standard HA configuration with external switches. In the case of a device failure, one of the ISPs will no longer be available because the switch that is connected to it will be down.

For example, If FGT_A loses power, HA failover will occur and FGT_B will become the primary unit. Its connection to internal2 on HD_SW1 will also be down, so it will be unable to connect to ISP 1. Its SD-WAN SLAs will be broken, and traffic will only be routed through ISP 2.

A link on a hardware switch cannot be monitored in HA monitor, so it is impossible to perform link failure when a port in either of the hardware switches fails. Performing a link failure is unnecessary in this configuration though, because any link failure on the hardware switch will be experienced by both cluster members. SD-WAN SLA health checks should be used to monitor the health of each ISP.

Failure on a hardware switch or ISP router

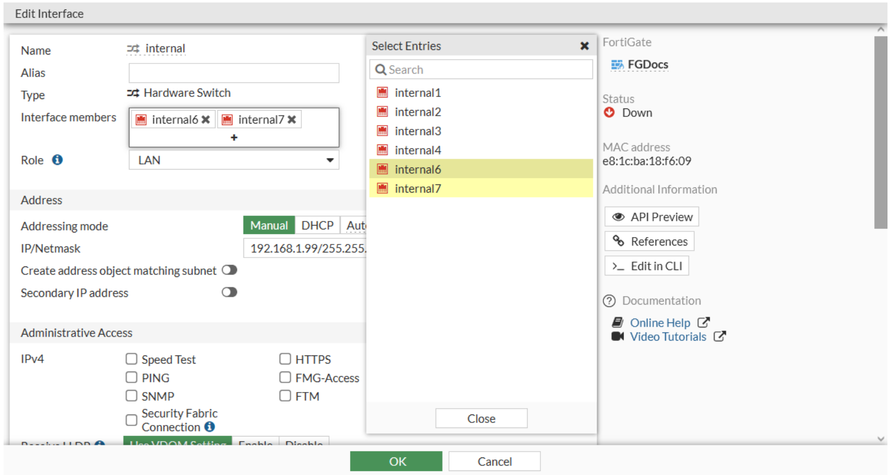
If a hardware switch or switch interface is down, or the ISP router is down, the SD-WAN can detect the broken SLA and continue routing to the other ISP.

For example, if FGT_A is the primary unit, and ISP 2 Router becomes unreachable, the SLA health checks on SD-WAN will detect the broken SLA and cause traffic to stop routing to ISP 2.

Configuration

To configure the HA A-P cluster with internal hardware switches:

1. Configure two FortiGates with internal switches in an A-P HA cluster (follow the steps in HA active-passive cluster setup on page 2035), starting by connecting the heartbeat interface.
2. When the HA cluster is up, connect to the primary FortiGate's GUI.
3. Remove the existing interface members from the default hardware switch:
   a. Go to *Network > Interfaces*.
   b. In the *LAN* section, double-click the *internal* interface to edit it.
   c. In *Interface Members*, remove all of the interfaces



d. Click *OK*.

4. Configure the hardware switch interfaces for the two ISPs:

a. Go to *Network > Interfaces* and click *Create New > Interface*.

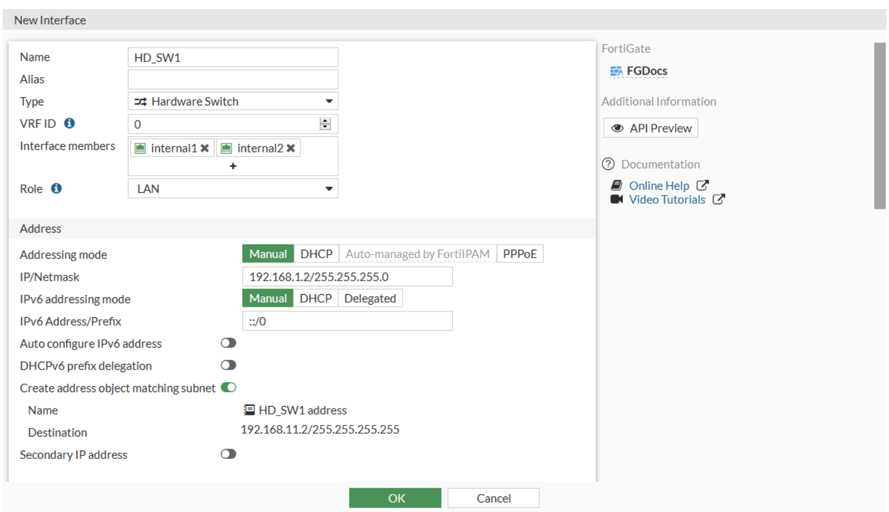
b. Enter a name (*HD_SW1*).

c. Set *Type* to *Hardware Switch*.

d. In *Interface Members*, add two interfaces (*internal1* and *internal2*).

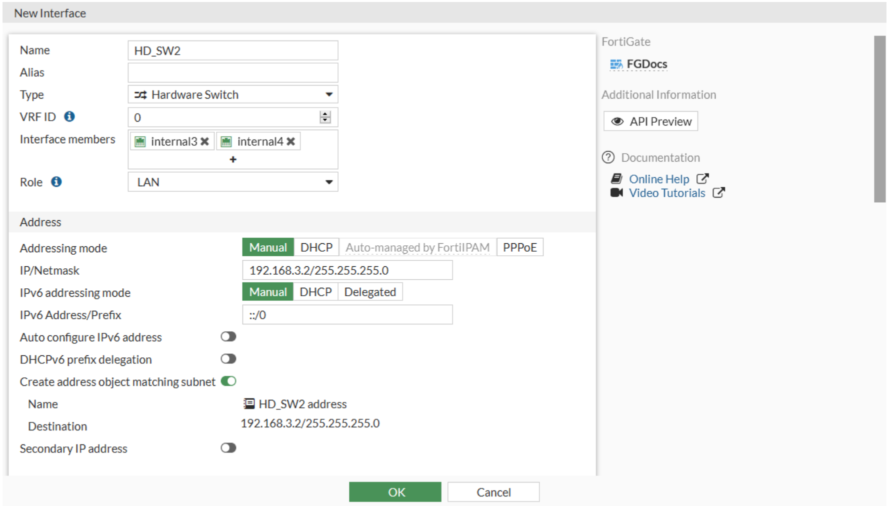
e. Set *IP/Netmask* to 192.168.1.2/24.

f. Configure the remaining settings as needed.



g. Click *OK*.

h. Repeat these steps to create a second hardware switch interface (*HD_SW2*) with two interface members (*internal3* and *internal4*) and *IP/Netmask* set to *192.168.3.2/24*.



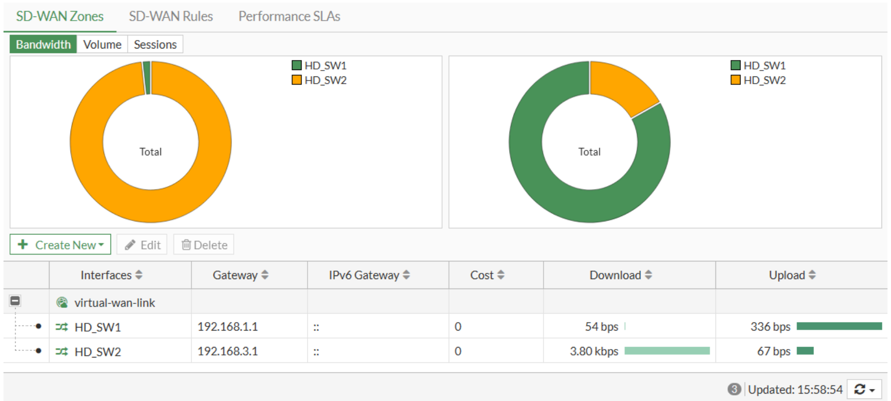
To connect the devices as shown in the topology:

1. Connect the incoming interface to the internal switch on both FortiGates.
2. On FGT_A, connect internal1 of HD_SW1 to ISP 1 Router.
3. On FGT_B, connect internal3 of HD_SW2 to ISP 2 Router.
4. For HD_SW1, connect FGT_A internal2 directly to FGT_B internal2.
5. For HD_SW2, connect FGT_A internal4 directly to FGT_B internal4.

To configure SD-WAN:

The primary FortiGate makes all the SD-WAN decisions.

1. On the primary FortiGate, go to *Network > SD-WAN*, select the *SD-WAN Zones* tab, and click *Create New > SD-WAN Member*.
2. In the *Interface* dropdown, select *HD_SW1*.
3. Leave *SD-WAN Zone* set to *virtual-wan-link*.
4. Enter the *Gateway* address *192.168.1.1*.
5. Click *OK*.
6. Repeat these steps to add the second interface (*HD_SW2*) with the gateway *192.168.3.1*.
7. Click *Apply*.



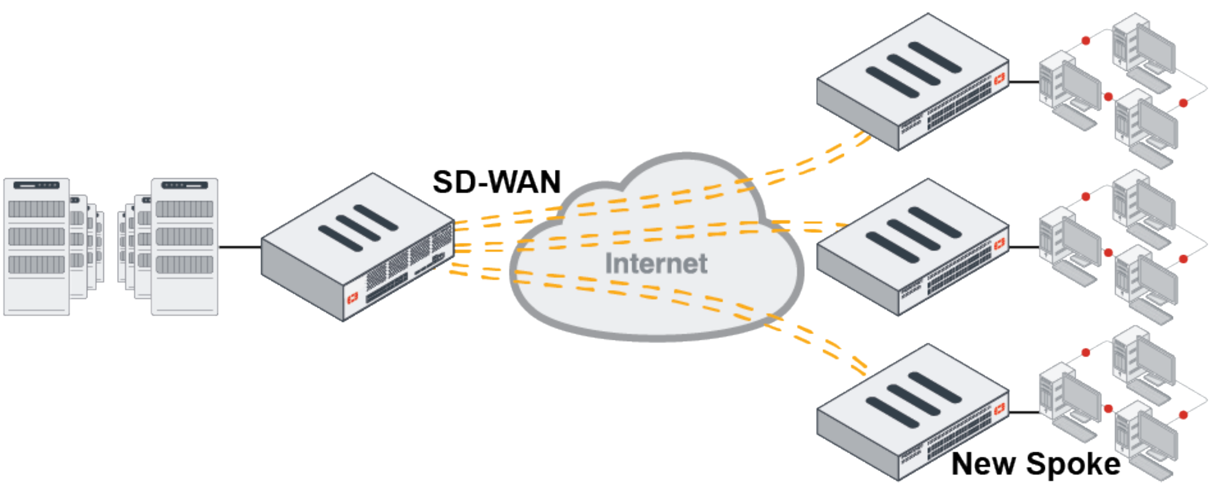
8. Create a health check:
   a. Go to *Network > SD-WAN*, select the *Performance SLA* tab, and click *Create New*.
   b. Set *Name* to *GW_HC*.
   c. Set *Protocol* to *Ping* and *Servers* to *8.8.8.8*.
   d. Set *Participants* to *All SD-WAN Members*.
   e. Enable *SLA Target* and leave the default values.
   f. Click *OK*.
9. Create SD-WAN rules as needed. The SLA health check can be used to determine when the ISP connections are in or out of SLA, and to failover accordingly.

SD-WAN configuration portability

When configuring SD-WAN, adding interfaces to members is optional.

This allows the SD-WAN to be configured without associating any interfaces to SD-WAN members. It also allows a configuration to be copied directly from one device to another, without requiring the devices to have interfaces with the same names.

After the configuration is created, add interfaces to the members make it functional.

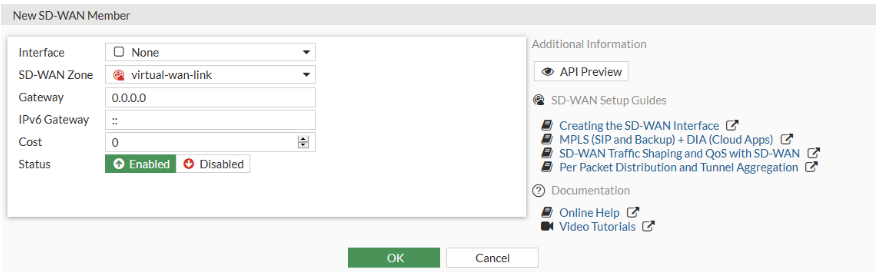


Example 1

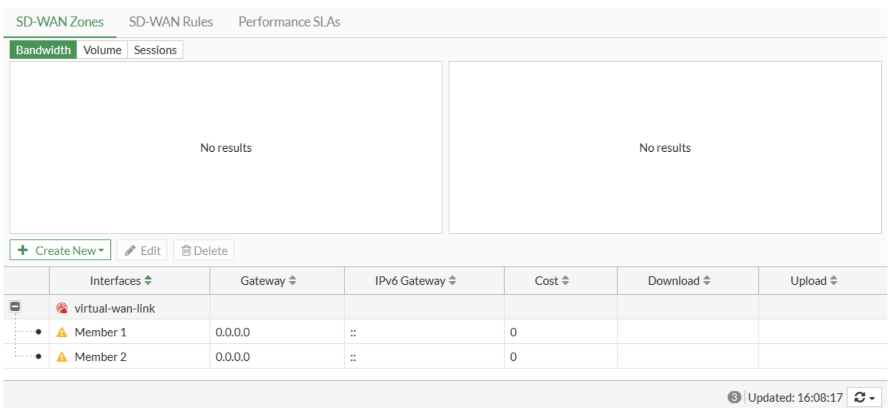
In this example, we create a template with two SD-WAN members configured without assigned interfaces that are used in a performance SLA and SD-WAN rule. The template can be used to configure new devices, as in Example 2 on page 704. Interfaces are then assigned to the members, and the configuration becomes active.

To create the SD-WAN members in the GUI:

1. Go to *Network > SD-WAN*, select the *SD-WAN Zones* tab, and click *Create New > SD-WAN Member*.
2. Leave all the settings set to their default values and click *OK*.



3. Repeat the above steps to create a second member.
   The empty members are listed on the *SD-WAN Zones* tab.

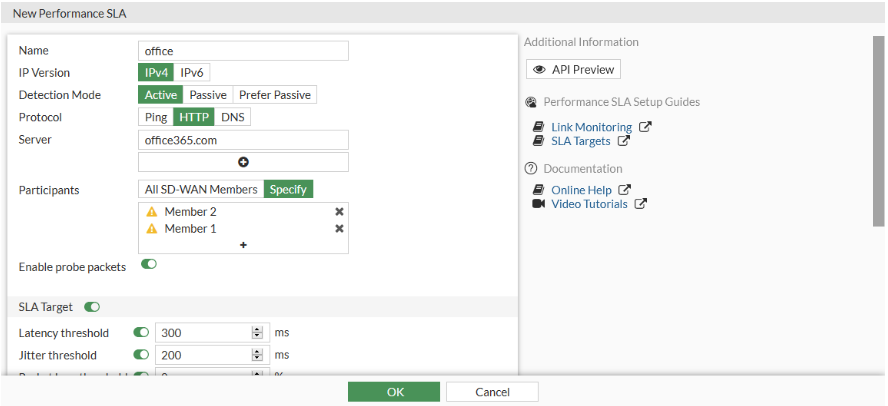


   The members are disabled until interfaces are configured, but can still be used in performance SLAs and SD-WAN rules.

To create a performance SLA in the GUI:

1. Go to *Network > SD-WAN* and select the *Performance SLAs* tab.
2. Click *Create New*.

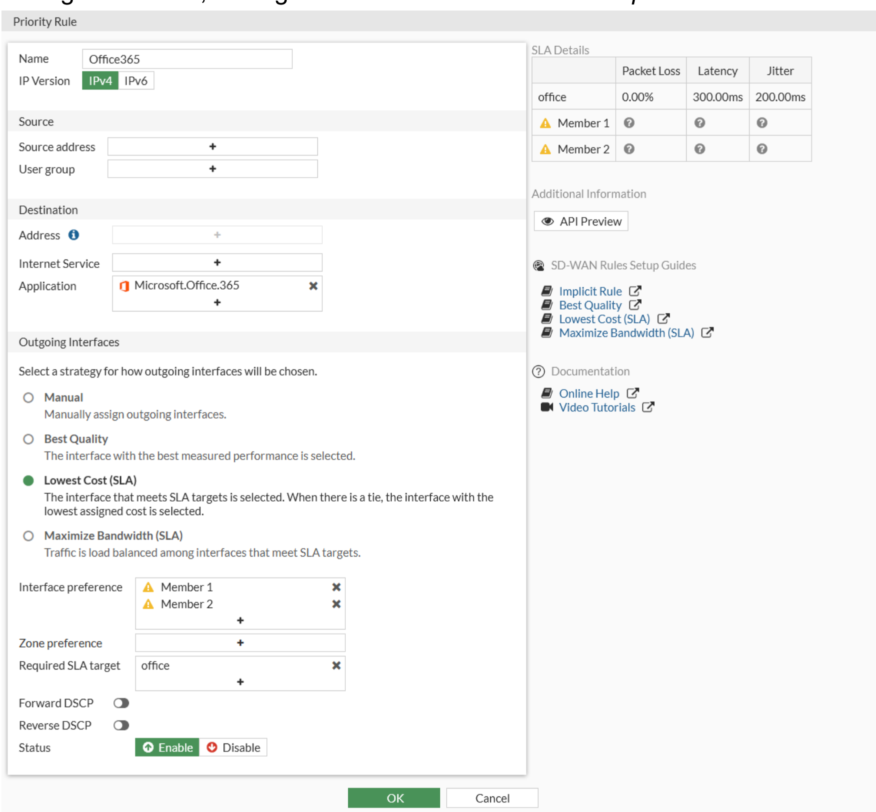
3. Configure the performance SLA, specifying the empty members as participants.



4. Click *OK*.

To create an SD-WAN rule in the GUI:

1. Go to *Network > SD-WAN*, select the *SD-WAN Rules* tab, and click *Create New*.
2. Configure the rule, adding both members to the *Interface preference* field:

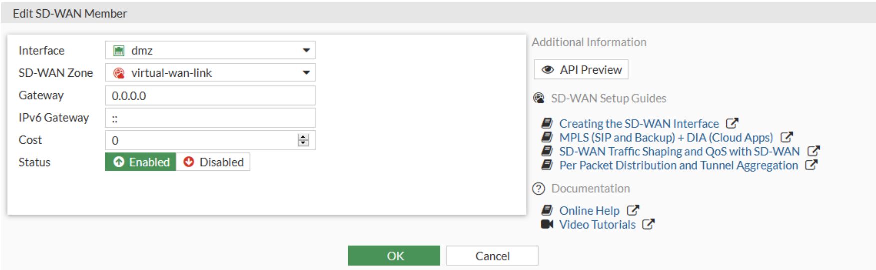


3. Click *OK*.

To assign interfaces to the SD-WAN members in the GUI:

1. Go to *Network > SD-WAN* and select the *SD-WAN Zones* tab.
2. Edit the first member

3. Set *Interface* to an actual interface.



4. Click *OK*.
5. Repeat the above steps to assign an interface to the second member.

To configure the SD-WAN in the CLI:

1. Create SD-WAN members:

```
config system sdwan
    set status enable
    config members
        edit 1
        next
        edit 2
        next
    end
end
```

2. Create a health check (performance SLA):

```
config system sdwan
    config health-check
        edit "office"
            set server "office365.com"
            set protocol http
            set sla-fail-log-period 300
            set sla-pass-log-period 300
            set members 2 1
            config sla
                edit 1
                    set latency-threshold 300
                    set jitter-threshold 200
                next
                edit 2
                    set link-cost-factor latency
                    set latency-threshold 20
                next
            end
        next
    end
end
```

3. Create a service (rule):

```
config system sdwan
    config service
        edit 3
            set name "Office365"
```

```
                set mode sla
                set internet-service enable
                set internet-service-app-ctrl 33182
                config sla
                    edit "office"
                        set id 2
                    next
                end
                set priority-members 1 2
            next
        end
    end
```

The SD-WAN configuration can now be used in as a template for new spokes, as in Example 2 on page 704.

To assign interfaces to the SD-WAN members in the CLI:

```
config system sdwan
    config members
        edit 1
            set interface "_OCVPN4-0.0"
        next
        edit 2
            set interface "_OCVPN4-0.1"
        next
    end
end
```

If no SD-WAN zone is specified, members are added to the default *virtual-wan-link* zone.

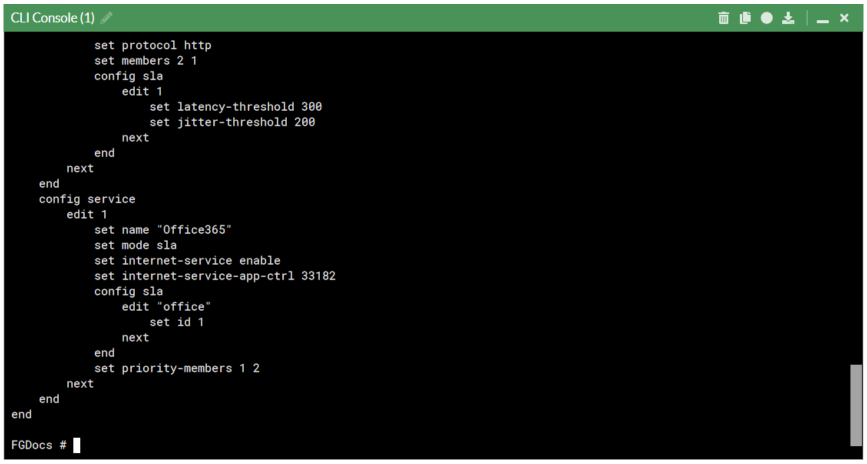
Example 2

In this example, the configuration from Example 1 is copied onto a new FortiGate.

Using the CLI console and the GUI

To copy the SD-WAN configuration from the original FortiGate:

1. Optionally, change the console screen paging setting. See Screen paging on page 37 for details.
2. Open the CLI console.
3. If necessary, click *Clear console* to empty the console.
4. Enter the following command:
   ```
   show system sdwan
   ```
5. Either click *Download* and open the file in a text editor, or click *Copy to clipboard* and paste the content into a text editor.



6. Edit the CLI configuration as necessary. For example, the first line that shows the `show` command should be deleted, and the default health checks can be removed.
7. If required, save the CLI configuration as a text file.

To paste the SD-WAN configuration onto a new FortiGate:

1. Copy the SD-WAN configuration from the text editor.
2. On the new FortiGate, open the CLI console.
3. Press *Ctrl* + *v* to paste the CLI commands.
4. In necessary, press *Enter* to apply the last `end` command.
   The SD-WAN configuration is copied to the new FortiGate.
   If the interfaces do not exist, the SD-WAN members are created without interfaces, and are disabled until interfaces are configured.

To assign interfaces to the SD-WAN members:

1. Go to *Network* > *SD-WAN* and select the *SD-WAN Zones* tab.
2. Edit the first member
3. Set *Interface* to an actual interface.
4. Click *OK*.
5. Repeat the above steps to assign an interface to the second member.

Using a terminal emulator

The following instructions use PuTTy. The steps may vary in other terminal emulators.

To copy the SD-WAN configuration from the original FortiGate:

1. Connect to the FortiGate. See Connecting to the CLI on page 29 for details.
2. Enter the following command:
   ```
   show system sdwan
   ```
3. Select the output, press *Ctrl* + *c* to copy it, and then paste it into a text editor.
4. Edit the CLI configuration as necessary. For example, the default health checks can be removed.
5. If required, save the CLI configuration as a text file.

To paste the SD-WAN configuration onto a new FortiGate:

1. Connect to the new FortiGate. See Connecting to the CLI on page 29 for details.
2. Copy the SD-WAN configuration from the text editor.
3. Right-click to paste the SD-WAN configuration.
4. In necessary, press *Enter* to apply the last `end` command.
   The SD-WAN configuration is copied to the new FortiGate.
   If the interfaces do not exist, the SD-WAN members are created without interfaces, and are disabled until interfaces are configured.

To assign interfaces to the SD-WAN members in the CLI:

```
config system sdwan
    config members
        edit 1
            set interface "_OCVPN4-0.0"
        next
        edit 2
            set interface "_OCVPN4-0.1"
        next
    end
end
```

If no SD-WAN zone is specified, members are added to the default *virtual-wan-link* zone.

SD-WAN segmentation over a single overlay

SD-WAN, VPN, and BGP configurations support L3 VPN segmentation over a single overlay. In these configurations, a hub and spoke SD-WAN deployment requires that branch sites, or spokes, are able to accommodate multiple companies or departments, and each company's subnet is separated by a different VRF. A subnet on one VRF cannot communicate with a subnet on another VRF between different branches, but can communicate with the same VRF.

SD-WAN options

VRF-aware SD-WAN health checks

SD-WAN on the originating spoke can tag the health check probes with the correct VRF when transmitting to a multi-VRF tunnel. The hub can then forward the probes to the correct health check server in the same VRF as the hub.

```
config system sdwan
   config health-check
       edit <name>
           set vrf <vrf id>
           set source <address>
       next
   end
end
```

`vrf <vrf id>`	Virtual Routing Forwarding ID.
`source <address>`	Source IP address used in the health-check packet to the server.

Overlay stickiness

When a hub has multiple overlays, traffic received on one overlay should egress on the same overlay when possible. The `service-sla-tie-break` option ensures overlay stickiness. In SD-WAN service rules, options are available to ensure that traffic received in a zone stays in that zone.

```
config system sdwan
    config zone
        edit <name>
            set service-sla-tie-break input-device
        next
    end
    config service
        edit <id>
            set input-zone <zone>
            set tie-break input-device
        next
    end
end
```

`service-sla-tie-break input-device`	Members that meet the SLA are selected by matching the input device.
`input-zone <zone>`	Source input-zone name.
`tie-break input-device`	Members that meet the SLA are selected by matching the input device.

IPsec options

Configurable rate limit for shortcut offers sent by the hub

By default, the hub sends a shortcut offer to a spoke every five seconds. If the hub continues to send offers that keep failing, and there are a large number of spokes, this can cause a high load on the hub. This setting makes the interval between shortcut offers configurable.

```
config vpn ipsec phase1-interface
    edit <name>
        set auto-discovery-offer-interval <interval>
    next
end
```

`auto-discovery-offer-interval <interval>`	Interval between shortcut offer messages, in seconds (1 - 300, default = 5).

Segmentation over a single overlay

Segmentation requires that VRF info is encapsulated within the IPsec VPN tunnel. This setting enables multi-VRF IPSEC tunnels.

```
config vpn ipsec phase1-interface
    edit <name>
        set encapsulation vpn-id-ipip
    next
end
```

`encapsulation vpn-id-ipip`	VPN ID with IPIP encapsulation.

VPN configuration for BGP

The role of a VRF can be specified, along with other VRF details. Up to 64 VRFs can be configured per VDOM for any device.

```
config router bgp
    config vrf
        edit <vrf>
            set role {standalone | ce | pe}
            set rd <string>
            set export-rt <route_target>
            set import-rt <route_target>
            set import-route-map <route_map>
            config leak-target
                edit <vrf>
                    set route-map <route-map>
                    set interface <interface>
                next
            end
        next
    end
end
```

`role {standalone | ce | pe}`	VRF role: standalone, customer edge (CE), or provider edge (PE).
`rd <string>`	Route Distinguisher: AA|AA:NN. This option is only available when the role is CE.
`export-rt <route_target>`	List of export route target. This option is only available when the role is CE.
`import-rt <route_target>`	List of import route target. This option is only available when the role is CE.
`import-route-map <route_map>`	Import route map. This option is only available when the role is CE.
`route-map <route-map>`	Route map of VRF leaking.
`interface <interface>`	Interface that is used to leak routes to the target VRF.

In FortiOS 7.0, `config vrf` was `config vrf-leak`, and `config leak-target` was `config target`.

Display BGP routes by VRF and neighbor

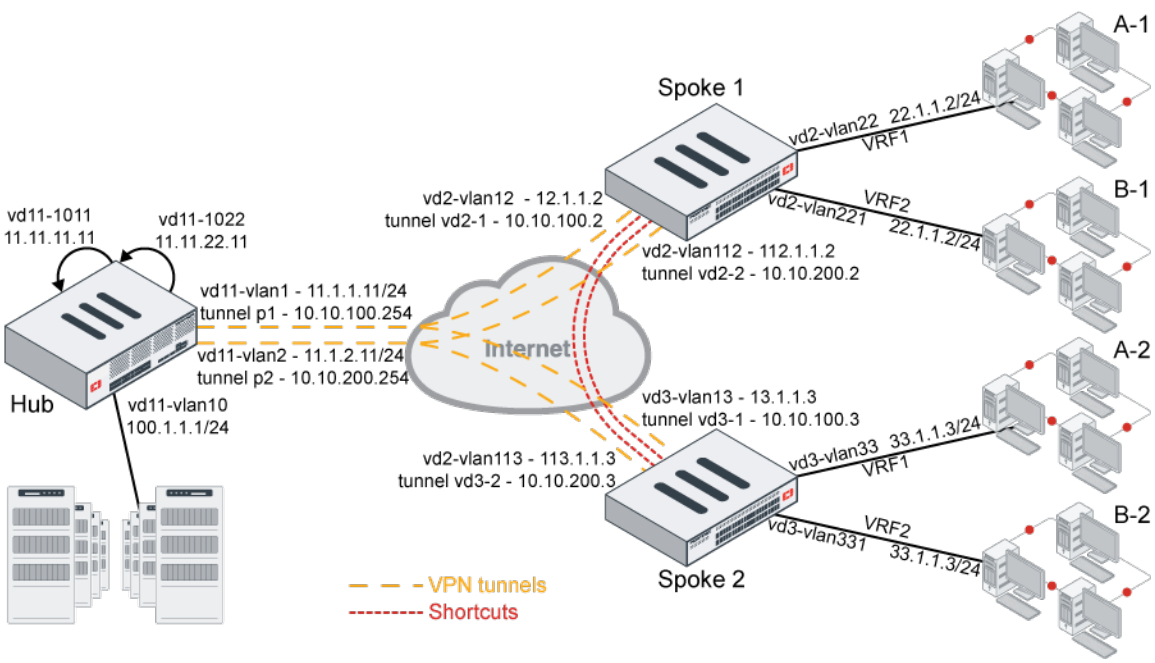
```
# diagnose ip router bgp set-filter vrf <vrf>
# diagnose ip router bgp set-filter neighbor <neighbor address>
# diagnose ip router bgp set-filter reset
# execute router clear bgp vpnv4 unicast soft {in | out}
# get router info filter show
# get router info filter vrf {vrf | all}
```

Examples

In example 1, multiple companies (or departments of a company) share the ADVPN. Company A and company B each have two branches in two different locations. Company A's branches (A-1 and A-2) can talk to each other using the VPN shortcut, but not to company B's branches (B-1 and B-2). Likewise, company B's branches can talk to each other using the VPN shortcut, but not to company A's branches. Traffic can share the tunnels and shortcuts, but cannot be mixed up.

Example 2 shows that performance SLA health checks can be sent from a spoke's VRF to the loopback on the hub that is in the same VRF.

Example 3 shows that when traffic is ingress on the hub on one overlay, it will preferably egress on the same overlay.



Example 1

In this example, two spokes each have two tunnels to the hub.

- Each spoke has two VRFs behind it that can use the same IP address or subnets.
- The computers in VRF1 behind spoke 1 can talk to the computers in VRF1 behind spoke 2, but not to any of the computers in the VRF2s behind either spoke.
- The computers in VRF2 behind spoke 1 can talk to the computers in VRF2 behind spoke 2, but not to any of the computers in the VRF1s behind either spoke.

To configure the hub:

```
config router bgp
    set as 65505
    set router-id 11.11.11.11
    set ibgp-multipath enable
    set additional-path enable
    set additional-path-vpnv4 enable
    set cluster-id 11.12.13.14
    set additional-path-select 3
    config neighbor-group
        edit "gr1"
            set capability-graceful-restart enable
            set capability-default-originate enable
            set next-hop-self-rr enable
            set soft-reconfiguration-vpnv4 enable
            set remote-as 65505
            set additional-path both
            set additional-path-vpnv4 both
            set adv-additional-path 3
            set route-reflector-client enable
            set route-reflector-client-vpnv4 enable
        next
        edit "gr2"
            set capability-graceful-restart enable
            set capability-default-originate enable
            set next-hop-self-rr enable
            set soft-reconfiguration-vpnv4 enable
            set remote-as 65505
            set additional-path both
            set additional-path-vpnv4 both
            set adv-additional-path 3
            set route-reflector-client enable
            set route-reflector-client-vpnv4 enable
        next
    end
    config neighbor-range
        edit 1
            set prefix 10.10.100.0 255.255.255.0
            set neighbor-group "gr1"
        next
        edit 2
            set prefix 10.10.200.0 255.255.255.0
            set neighbor-group "gr2"
        next
    end
    config network
        edit 12
            set prefix 11.11.11.11 255.255.255.255
        next
        edit 22
            set prefix 11.11.22.11 255.255.255.255
        next
        edit 10
            set prefix 100.1.1.0 255.255.255.0
        next
```

```
        edit 33
            set prefix 11.1.1.0 255.255.255.0
        next
    end
    config vrf
        edit "0"
            set role pe
        next
        edit "1"
            set role ce
            set rd "1:1"
            set export-rt "1:1"
            set import-rt "1:1"
        next
        edit "2"
            set role ce
            set rd "2:1"
            set export-rt "2:1"
            set import-rt "2:1"
        next
    end
end
```

```
config vpn ipsec phase1-interface
    edit "p1"
        set type dynamic
        set interface "vd11-vlan1"
        set peertype any
        set net-device disable
        set proposal aes128-sha1
        set add-route disable
        set dpd on-idle
        set dhgrp 5
        set auto-discovery-sender enable
        set auto-discovery-offer-interval 10
        set encapsulation vpn-id-ipip
        set psksecret **********
        set dpd-retryinterval 60
    next
    edit "p2"
        set type dynamic
        set interface "vd11-vlan2"
        set peertype any
        set net-device disable
        set proposal aes128-sha1
        set add-route disable
        set dpd on-idle
        set dhgrp 5
        set auto-discovery-sender enable
        set auto-discovery-offer-interval 10
        set encapsulation vpn-id-ipip
        set psksecret **********
        set dpd-retryinterval 60
    next
end
```

```
config vpn ipsec phase2-interface
    edit "p1"
        set phase1name "p1"
        set proposal aes128-sha1
        set dhgrp 5
    next
    edit "p2"
        set phase1name "p2"
        set proposal aes128-sha1
        set dhgrp 5
    next
end
```

To configure a spoke:

```
config router bgp
    set as 65505
    set router-id 2.2.2.2
    set ebgp-multipath enable
    set ibgp-multipath enable
    set network-import-check disable
    set additional-path enable
    set additional-path6 enable
    set additional-path-vpnv4 enable
    set recursive-next-hop enable
    set graceful-restart enable
    set additional-path-select 4
    config neighbor
        edit "10.10.100.254"
            set capability-dynamic enable
            set capability-graceful-restart-vpnv4 enable
            set soft-reconfiguration enable
            set soft-reconfiguration-vpnv4 enable
            set remote-as 65505
            set additional-path both
            set additional-path-vpnv4 both
            set adv-additional-path 3
        next
        edit "10.10.200.254"
            set capability-dynamic enable
            set capability-graceful-restart-vpnv4 enable
            set soft-reconfiguration enable
            set soft-reconfiguration-vpnv4 enable
            set remote-as 65505
            set additional-path both
            set additional-path-vpnv4 both
            set adv-additional-path 3
        next
    end
    config network
        edit 3
            set prefix 22.1.1.0 255.255.255.0
        next
        edit 4
            set prefix 12.12.12.0 255.255.255.0
        next
```

```
    end
    config vrf
        edit "0"
            set role pe
        next
        edit "1"
            set role ce
            set rd "1:1"
            set export-rt "1:1"
            set import-rt "1:1"
        next
        edit "2"
            set role ce
            set rd "2:1"
            set export-rt "2:1"
            set import-rt "2:1"
        next
    end
end

config vpn ipsec phase1-interface
    edit "vd2-1"
        set interface "vd2-vlan12"
        set peertype any
        set net-device enable
        set proposal aes128-sha1
        set add-route disable
        set dhgrp 5
        set idle-timeout enable
        set idle-timeoutinterval 5
        set auto-discovery-receiver enable
        set encapsulation vpn-id-ipip
        set remote-gw 11.1.1.11
        set psksecret **********
    next
    edit "vd2-2"
        set interface "vd2-vlan112"
        set peertype any
        set net-device enable
        set proposal aes128-sha1
        set add-route disable
        set dhgrp 5
        set auto-discovery-receiver enable
        set encapsulation vpn-id-ipip
        set remote-gw 11.1.2.11
        set psksecret **********
    next
end

config vpn ipsec phase2-interface
    edit "vd2-1"
        set phase1name "vd2-1"
        set proposal aes128-sha1
        set dhgrp 5
        set auto-negotiate enable
    next
    edit "vd2-2"
```

```
        set phase1name "vd2-2"
        set proposal aes128-sha1
        set dhgrp 5
        set auto-negotiate enable
    next
end
config system sdwan
    set status enable
    config zone
        edit "virtual-wan-link"
        next
        edit "SASE"
        next
        edit "zon2"
        next
    end
    config members
        edit 1
            set interface "vd2-1"
            set cost 10
        next
        edit 2
            set interface "vd2-2"
            set cost 20
        next
    end
    config health-check
        edit "ping"
            set server "11.11.11.11"
            set members 1 2
            config sla
                edit 1
                    set latency-threshold 200
                    set jitter-threshold 50
                next
            end
        next
        edit "1"
            set server "22.1.1.2"
            set vrf 1
            set members 1 2
        next
    end
    config service
        edit 2
            set mode sla
            set dst "100-200"
            config sla
                edit "ping"
                    set id 1
                next
            end
            set priority-members 2
            set use-shortcut-sla disable
        next
        edit 1
```

```
            set name "test-tag"
            set mode sla
            set dst "001-100"
            config sla
                edit "ping"
                    set id 1
                next
            end
            set priority-members 1 2
        next
    end
end
```

To check the spoke 1 routes:

```
# get router info routing-table bgp
Routing table for VRF=0
B*      0.0.0.0/0 [200/0] via 10.10.100.254 (recursive via vd2-1 tunnel 11.1.1.11 vrf 0),
04:42:57, [1/0]
                  [200/0] via 10.10.200.254 (recursive via vd2-2 tunnel 11.1.2.11 vrf 0),
04:42:57, [1/0]
B       1.1.1.1/32 [200/0] via 11.1.1.1 [2] (recursive via 12.1.1.1, vd2-vlan12), 04:42:57,
[1/0]
B       1.222.222.222/32 [200/0] via 11.1.1.1 [2] (recursive via 12.1.1.1, vd2-vlan12),
04:42:57, [1/0]
B       11.11.11.11/32 [200/0] via 10.10.100.254 (recursive via vd2-1 tunnel 11.1.1.11 vrf
0), 04:42:57, [1/0]
                       [200/0] via 10.10.200.254 (recursive via vd2-2 tunnel 11.1.2.11 vrf
0), 04:42:57, [1/0]
B       33.1.1.0/24 [200/0] via 10.10.100.254 [2] (recursive via vd2-1 tunnel 11.1.1.11 vrf
0), 04:42:57, [1/0]
                    [200/0] via 10.10.200.254 [2] (recursive via vd2-2 tunnel 11.1.2.11 vrf
0), 04:42:57, [1/0]
B       100.1.1.0/24 [200/0] via 10.10.100.254 (recursive via vd2-1 tunnel 11.1.1.11 vrf 0),
04:42:57, [1/0]
                     [200/0] via 10.10.200.254 (recursive via vd2-2 tunnel 11.1.2.11 vrf 0),
04:42:57, [1/0]

Routing table for VRF=1
B V     33.1.1.0/24 [200/0] via 10.10.100.3 [2] (recursive via vd2-1 tunnel 11.1.1.11 vrf
0), 04:42:57, [1/0]
                    [200/0] via 10.10.200.3 [2] (recursive is directly connected, vd2-2_0),
04:42:57, [1/0]

Routing table for VRF=2
B V     33.1.1.0/24 [200/0] via 10.10.100.3 [2] (recursive via vd2-1 tunnel 11.1.1.11 vrf
0), 04:42:56, [1/0]
                    [200/0] via 10.10.200.3 [2] (recursive is directly connected, vd2-2_0),
04:42:56, [1/0]
```

VRF1 routes:

```
# get router info filter vrf 1
# get router info routing-table bgp
Routing table for VRF=1
B V     33.1.1.0/24 [200/0] via 10.10.100.3 [2] (recursive via vd2-1 tunnel 11.1.1.11 vrf
0), 04:44:11, [1/0]
```

```
                    [200/0] via 10.10.200.3 [2] (recursive is directly connected, vd2-2_0),
04:44:11, [1/0]
```

To test the configuration on shortcut 1:

1. From VRF1 of spoke 1 ping VRF1 of spoke 2 and from VRF2 of spoke 1 ping VRF2 spoke 2. Both VRF1 and VRF2 source and destination IP addresses are the same, so you can see how the traffic is isolated
2. Check sessions on spoke 1:

   The output `vd=<vdom ID>:<VRF ID>` indicates that sessions are created in and stay in the corresponding VRFs.

   - User at 22.1.1.22 in VRF1 on spoke 1 pings 33.1.1.33 in VRF1 on spoke2.

     ```
     # diagnose sys session list
     session info: proto=1 proto_state=00 duration=21 expire=42 timeout=0 flags=00000000
     socktype=0 sockport=0 av_idx=0 use=3
     origin-shaper=
     reply-shaper=
     per_ip_shaper=
     class_id=0 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/255
     state=may_dirty
     statistic(bytes/packets/allow_err): org=420/5/1 reply=420/5/1 tuples=2
     tx speed(Bps/kbps): 0/0 rx speed(Bps/kbps): 0/0
     orgin->sink: org pre->post, reply pre->post dev=89->131/131->89
     gwy=10.10.200.3/22.1.1.22
     hook=pre dir=org act=noop 22.1.1.22:48417->33.1.1.33:8(0.0.0.0:0)
     hook=post dir=reply act=noop 33.1.1.33:48417->22.1.1.22:0(0.0.0.0:0)
     src_mac=02:4c:a5:fc:6a:7f
     misc=0 policy_id=1 pol_uuid_idx=566 auth_info=0 chk_client_info=0 vd=1:1
     serial=00092eee tos=ff/ff app_list=0 app=0 url_cat=0
     sdwan_mbr_seq=0 sdwan_service_id=1
     rpdb_link_id=ff000001 ngfwid=n/a
     npu_state=0x5040001 no_offload
     no_ofld_reason:  disabled-by-policy non-npu-intf
     ```

   - User at 22.1.1.22 in VRF2 on spoke 1 pings 33.1.1.33 in VRF2 on spoke2:

     ```
     # diagnose sys session list
     session info: proto=1 proto_state=00 duration=4 expire=56 timeout=0 flags=00000000
     socktype=0 sockport=0 av_idx=0 use=3
     origin-shaper=
     reply-shaper=
     per_ip_shaper=
     class_id=0 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/255
     state=may_dirty
     statistic(bytes/packets/allow_err): org=168/2/1 reply=168/2/1 tuples=2
     tx speed(Bps/kbps): 39/0 rx speed(Bps/kbps): 39/0
     orgin->sink: org pre->post, reply pre->post dev=113->131/131->113
     gwy=10.10.200.3/22.1.1.22
     hook=pre dir=org act=noop 22.1.1.22:55841->33.1.1.33:8(0.0.0.0:0)
     hook=post dir=reply act=noop 33.1.1.33:55841->22.1.1.22:0(0.0.0.0:0)
     src_mac=02:4c:a5:fc:6a:7f
     misc=0 policy_id=1 pol_uuid_idx=566 auth_info=0 chk_client_info=0 vd=1:2
     serial=00092f77 tos=ff/ff app_list=0 app=0 url_cat=0
     sdwan_mbr_seq=0 sdwan_service_id=1
     rpdb_link_id=ff000001 ngfwid=n/a
     npu_state=0x5040001 no_offload
     no_ofld_reason:  disabled-by-policy non-npu-intf
     ```

3. Check sessions on spoke 2:

   The output `vd=<vdom ID>:<VRF ID>` indicates that sessions are created in and stay in the corresponding VRFs.

   - User at 22.1.1.22 in VRF1 on spoke 1 pings 33.1.1.33 in VRF1 on spoke 2:

```
# diagnose sys session list
session info: proto=1 proto_state=00 duration=11 expire=49 timeout=0 flags=00000000
socktype=0 sockport=0 av_idx=0 use=3
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/255
state=may_dirty npu
statistic(bytes/packets/allow_err): org=168/2/1 reply=168/2/1 tuples=2
tx speed(Bps/kbps): 14/0 rx speed(Bps/kbps): 14/0
orgin->sink: org pre->post, reply pre->post dev=132->92/92->132
gwy=33.1.1.33/10.10.200.2
hook=pre dir=org act=noop 22.1.1.22:27733->33.1.1.33:8(0.0.0.0:0)
hook=post dir=reply act=noop 33.1.1.33:27733->22.1.1.22:0(0.0.0.0:0)
misc=0 policy_id=1 pol_uuid_idx=630 auth_info=0 chk_client_info=0 vd=6:1
serial=000a29fd tos=ff/ff app_list=0 app=0 url_cat=0
rpdb_link_id=00000000 ngfwid=n/a
npu_state=0x4000001 no_offload
npu info: flag=0x00/0x82, offload=0/0, ips_offload=0/0, epid=0/0, ipid=0/0,
vlan=0x0000/0x0000
vlifid=0/0, vtag_in=0x0000/0x0000 in_npu=0/0, out_npu=0/0, fwd_en=0/0, qid=0/0
no_ofld_reason:  disabled-by-policy
```

   - User at 22.1.1.22 in VRF2 on spoke 1 pings 33.1.1.33 in VRF2 on spoke 2:

```
# diagnose sys session list
session info: proto=1 proto_state=00 duration=17 expire=43 timeout=0 flags=00000000
socktype=0 sockport=0 av_idx=0 use=3
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/255
state=may_dirty npu
statistic(bytes/packets/allow_err): org=168/2/1 reply=168/2/1 tuples=2
tx speed(Bps/kbps): 9/0 rx speed(Bps/kbps): 9/0
orgin->sink: org pre->post, reply pre->post dev=132->115/115->132
gwy=33.1.1.33/10.10.200.2
hook=pre dir=org act=noop 22.1.1.22:24917->33.1.1.33:8(0.0.0.0:0)
hook=post dir=reply act=noop 33.1.1.33:24917->22.1.1.22:0(0.0.0.0:0)
dst_mac=02:4c:a5:fc:6a:7f
misc=0 policy_id=1 pol_uuid_idx=630 auth_info=0 chk_client_info=0 vd=6:2
serial=000a29ca tos=ff/ff app_list=0 app=0 url_cat=0
rpdb_link_id=00000000 ngfwid=n/a
npu_state=0x4000001 no_offload
npu info: flag=0x00/0x82, offload=0/0, ips_offload=0/0, epid=0/0, ipid=0/0,
vlan=0x0000/0x0000
vlifid=0/0, vtag_in=0x0000/0x0000 in_npu=0/0, out_npu=0/0, fwd_en=0/0, qid=0/0
no_ofld_reason:  disabled-by-policy
```

To test the configuration on shortcut 2:

1. From VRF1 of spoke 1 ping VRF1 of spoke 2 and from VRF2 of spoke 1 ping VRF2 spoke 2. Both VRF1 and VRF2 source and destination IP addresses are the same, so you can see how the traffic is isolated
2. Check sessions on spoke 1:

   The output `vd=<vdom ID>:<VRF ID>` indicates that sessions are created in and stay in the corresponding VRFs.
   - User at 22.1.1.22 in VRF1 on spoke 1 pings 33.1.1.133 in VRF1 on spoke 2:

     ```
     # diagnose sys session listsession info: proto=1 proto_state=00 duration=17 expire=45
     timeout=0 flags=00000000 socktype=0 sockport=0 av_idx=0 use=3
     origin-shaper=
     reply-shaper=
     per_ip_shaper=
     class_id=0 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/255
     state=may_dirty
     statistic(bytes/packets/allow_err): org=336/4/1 reply=336/4/1 tuples=2
     tx speed(Bps/kbps): 19/0 rx speed(Bps/kbps): 19/0
     orgin->sink: org pre->post, reply pre->post dev=89->137/137->89
     gwy=10.10.200.3/22.1.1.22
     hook=pre dir=org act=noop 22.1.1.22:25968->33.1.1.133:8(0.0.0.0:0)
     hook=post dir=reply act=noop 33.1.1.133:25968->22.1.1.22:0(0.0.0.0:0)
     src_mac=02:4c:a5:fc:6a:7f
     misc=0 policy_id=1 pol_uuid_idx=566 auth_info=0 chk_client_info=0 vd=1:1
     serial=000aa475 tos=ff/ff app_list=0 app=0 url_cat=0
     sdwan_mbr_seq=0 sdwan_service_id=2
     rpdb_link_id=ff000002 ngfwid=n/a
     npu_state=0x5040001 no_offload
     no_ofld_reason:  disabled-by-policy non-npu-intf
     ```

   - User at 22.1.1.22 in VRF2 on spoke 1 pings 33.1.1.133 in VRF2 on spoke 2:

     ```
     # diagnose sys session listsession info: proto=1 proto_state=00 duration=8 expire=53
     timeout=0 flags=00000000 socktype=0 sockport=0 av_idx=0 use=3
     origin-shaper=
     reply-shaper=
     per_ip_shaper=
     class_id=0 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/255
     state=may_dirty
     statistic(bytes/packets/allow_err): org=252/3/1 reply=252/3/1 tuples=2
     tx speed(Bps/kbps): 30/0 rx speed(Bps/kbps): 30/0
     orgin->sink: org pre->post, reply pre->post dev=113->137/137->113
     gwy=10.10.200.3/22.1.1.22
     hook=pre dir=org act=noop 22.1.1.22:28528->33.1.1.133:8(0.0.0.0:0)
     hook=post dir=reply act=noop 33.1.1.133:28528->22.1.1.22:0(0.0.0.0:0)
     src_mac=02:4c:a5:fc:6a:7f
     misc=0 policy_id=1 pol_uuid_idx=566 auth_info=0 chk_client_info=0 vd=1:2
     serial=000aa49f tos=ff/ff app_list=0 app=0 url_cat=0
     sdwan_mbr_seq=0 sdwan_service_id=2
     rpdb_link_id=ff000002 ngfwid=n/a
     npu_state=0x5040001 no_offload
     no_ofld_reason:  disabled-by-policy non-npu-intf
     ```

3. Check sessions on spoke 2:

   The output `vd=<vdom ID>:<VRF ID>` indicates that sessions are created in and stay in the corresponding VRFs.

- User at 22.1.1.22 in VRF1 on spoke 1 pings 33.1.1.133 in VRF1 on spoke 2:

```
# diagnose sys session list
session info: proto=1 proto_state=00 duration=24 expire=38 timeout=0 flags=00000000
socktype=0 sockport=0 av_idx=0 use=3
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/255
state=may_dirty npu
statistic(bytes/packets/allow_err): org=336/4/1 reply=336/4/1 tuples=2
tx speed(Bps/kbps): 13/0 rx speed(Bps/kbps): 13/0
orgin->sink: org pre->post, reply pre->post dev=138->92/92->138
gwy=33.1.1.133/10.10.200.2
hook=pre dir=org act=noop 22.1.1.22:25968->33.1.1.133:8(0.0.0.0:0)
hook=post dir=reply act=noop 33.1.1.133:25968->22.1.1.22:0(0.0.0.0:0)
misc=0 policy_id=1 pol_uuid_idx=630 auth_info=0 chk_client_info=0 vd=6:1
serial=000aa476 tos=ff/ff app_list=0 app=0 url_cat=0
rpdb_link_id=00000000 ngfwid=n/a
npu_state=0x4000001 no_offload
npu info: flag=0x00/0x82, offload=0/0, ips_offload=0/0, epid=0/0, ipid=0/0,
vlan=0x0000/0x0000
vlifid=0/0, vtag_in=0x0000/0x0000 in_npu=0/0, out_npu=0/0, fwd_en=0/0, qid=0/0
no_ofld_reason:  disabled-by-policy
```

- User at 22.1.1.22 in VRF2 on spoke 1 pings 33.1.1.133 in VRF2 on spoke2:

```
# diagnose sys session list
session info: proto=1 proto_state=00 duration=15 expire=46 timeout=0 flags=00000000
socktype=0 sockport=0 av_idx=0 use=3
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/255
state=may_dirty npu
statistic(bytes/packets/allow_err): org=252/3/1 reply=252/3/1 tuples=2
tx speed(Bps/kbps): 16/0 rx speed(Bps/kbps): 16/0
orgin->sink: org pre->post, reply pre->post dev=138->115/115->138
gwy=33.1.1.133/10.10.200.2
hook=pre dir=org act=noop 22.1.1.22:28528->33.1.1.133:8(0.0.0.0:0)
hook=post dir=reply act=noop 33.1.1.133:28528->22.1.1.22:0(0.0.0.0:0)
misc=0 policy_id=1 pol_uuid_idx=630 auth_info=0 chk_client_info=0 vd=6:2
serial=000aa4a0 tos=ff/ff app_list=0 app=0 url_cat=0
rpdb_link_id=00000000 ngfwid=n/a
npu_state=0x4000001 no_offload
npu info: flag=0x00/0x82, offload=0/0, ips_offload=0/0, epid=0/0, ipid=0/0,
vlan=0x0000/0x0000
vlifid=0/0, vtag_in=0x0000/0x0000 in_npu=0/0, out_npu=0/0, fwd_en=0/0, qid=0/0
no_ofld_reason:  disabled-by-policy
```

Example 2

In this example, SLA health checks are sent from a spoke's VRF to the loopback on the hub that is in the same VRF.

To configure the health check:

```
config system sdwan
    config health-check
        edit "1"
            set server "11.11.22.11"
            set vrf 1
            set source 22.1.1.2
            set members 1 2
            config sla
                edit 1
                    set latency-threshold 200
                    set jitter-threshold 50
                next
            end
        next
    end
end
```

To check the health check status:

```
# diagnose sys sdwan health-check status  1
Health Check(1):
Seq(1 vd2-1): state(alive), packet-loss(0.000%) latency(0.023), jitter(0.002), mos(4.404),
bandwidth-up(0), bandwidth-dw(0), bandwidth-bi(0) sla_map=0x1
Seq(2 vd2-2): state(alive), packet-loss(0.000%) latency(0.022), jitter(0.002), mos(4.404),
bandwidth-up(0), bandwidth-dw(0), bandwidth-bi(0) sla_map=0x1
```

Example 3

In this example, when traffic from spoke 1 arrives at the hub on tunnel 1, it will egress the hub on tunnel 1 to go to other spokes. If traffic arrives on tunnel 2, it will egress on tunnel 2, and not tunnel 1.

To configure SD-WAN on the hub:

```
config system sdwan
    set status enable
    config zone
        edit "virtual-wan-link"
            set service-sla-tie-break input-device
        next
    end
    config members
        edit 1
            set interface "p1"
        next
        edit 2
            set interface "p2"
        next
    end
    config health-check
        edit "1"
            set server "22.1.1.2"
            set members 1 2
            config sla
```

```
                edit 1
                next
            end
        next
    end
    config service
        edit 1
            set mode sla
            set dst "all"
            config sla
                edit "1"
                    set id 1
                next
            end
            set priority-members 1 2
            set tie-break input-device
        next
    end
end
```

To verify that traffic stays in the same overlay on ingress and egress on the hub:

1. Confirm that the SD-WAN service rule has `Tie break` set to `input-device` so that, when SLAs are met on all of the members, traffic prefers to egress on the same member as the input device:

```
# diagnose sys sdwan service

Service(1): Address Mode(IPV4) flags=0x200 use-shortcut-sla
 Tie break: input-device
  Gen(1), TOS(0x0/0x0), Protocol(0: 1->65535), Mode(sla), sla-compare-order
  Members(2):
    1: Seq_num(1 p1), alive, sla(0x1), gid(0), cfg_order(0), local cost(0), selected
    2: Seq_num(2 p2), alive, sla(0x1), gid(0), cfg_order(1), local cost(0), selected
  Dst address(1):
        0.0.0.0-255.255.255.255
```

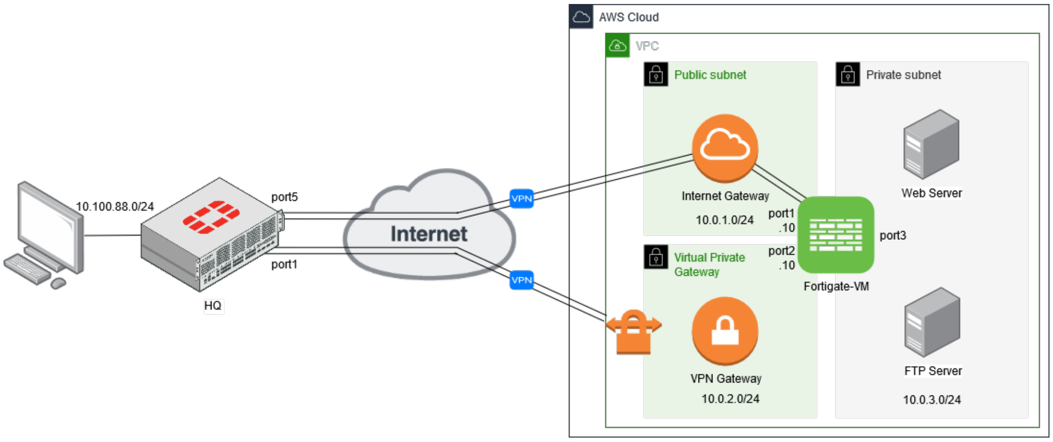
2. Use `diagnose sniffer packet` commands to verify that traffic ingress and egress are on the same overlay.

SD-WAN cloud on-ramp

In this example, you configure a connection to a new cloud deployment that has some remote servers. SD-WAN is used to steer traffic through the required overlay tunnel.

The on-premise FortiGate has two internet connections, each with a single VPN connection. The two VPN gateways are configured on the cloud for redundancy, one terminating at the FortiGate-VM, and the other at the native AWS VPN Gateway.

This example uses AWS as the Infrastructure as a Service (IaaS) provider, but the same configuration can also apply to other services. A full mesh VPN setup is not shown, but can be added later if required.



To connect to the servers that are behind the cloud FortiGate-VM, virtual IP addresses (VIPs) are configured on port2 to map to the servers:

- VPN traffic terminating on port1 is routed to the VIP on port2 to access the web servers.
- VPN traffic terminating on the VPN gateway accesses the VIPs on port2 directly.

There are four major steps to configure this setup:

1. Configuring the VPN overlay between the HQ FortiGate and cloud FortiGate-VM on page 722
2. Configuring the VPN overlay between the HQ FortiGate and AWS native VPN gateway on page 727
3. Configuring the VIP to access the remote servers on page 730
4. Configuring the SD-WAN to steer traffic between the overlays on page 733

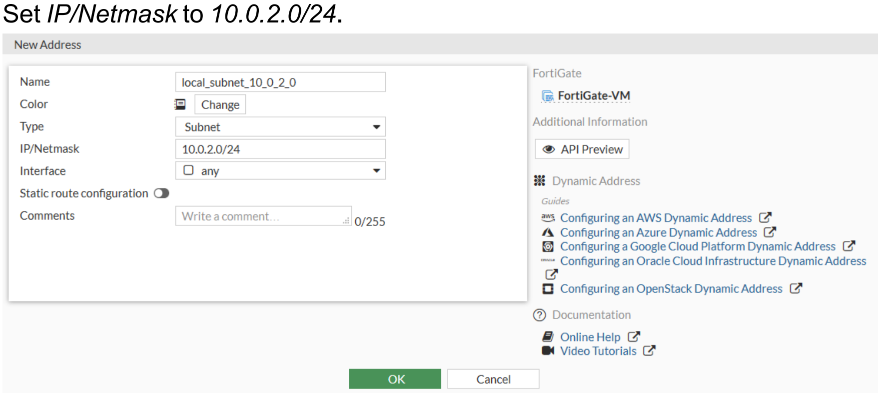
After the configuration is complete, verify the traffic to ensure that the configuration is working as expected, see Verifying the traffic on page 737.

Configuring the VPN overlay between the HQ FortiGate and cloud FortiGate-VM

Configure the cloud FortiGate-VM

To create an address for the VPN gateway:

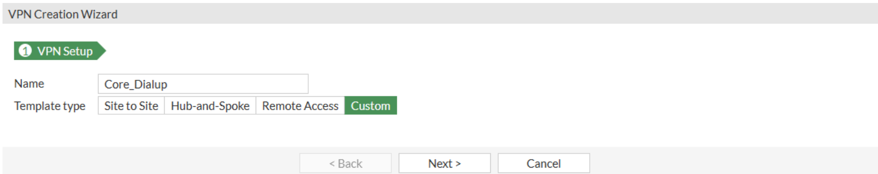
1. Go to *Policy & Objects > Addresses* and click *Create New > Address*.
2. Set *Name* to *local_subnet_10_0_2_0*.
3. Set *IP/Netmask* to *10.0.2.0/24*.



4. Click *OK*.

To configure a custom IPsec VPN:

1. Go to *VPN > IPsec Wizard*.
2. Set *Name* to *Core_Dialup*.
3. Set *Template type* to *Custom*.

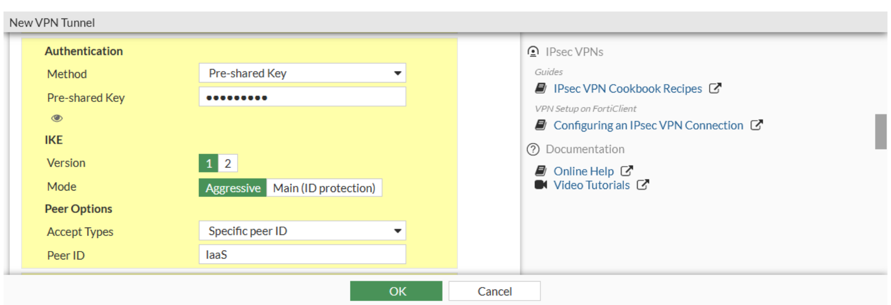


4. Click *Next*.
5. Configure *Network* settings:

Remote Gateway	Dialup User
Interface	port1
NAT Traversal	Enable

6. Configure *Authentication* settings:

Method	Pre-shared Key
Pre-shared Key	Enter the pre-shared key.
Version	1
Mode	Aggressive This setting allows the peer ID to be specified.
Accept Types	Specific peer ID
Peer ID	IaaS The other end of the tunnel needs to have its local ID set to IaaS.



7. Leave the default *Phase 1 Proposal* settings and disable *XAUTH*.

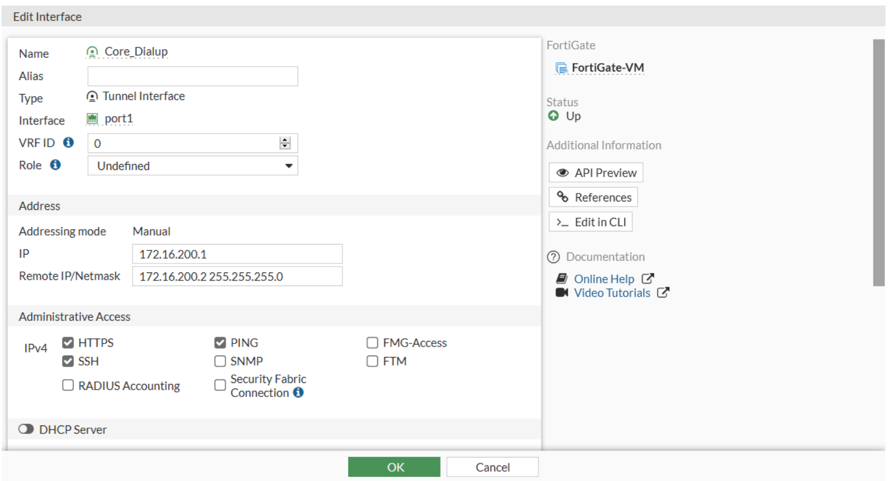
8. Configure the *Phase 2 Selector* settings:

Name	Ent_Core
Local Address	Named Address - *local_subnet_10_0_2_0*
Remote Address	Named Address - *all* This setting allows traffic originating from both the remote subnet 10.100.88.0 and the health checks from the VPN interface on the remote FortiGate. For increased security, each subnet can be specified individually.

9. Click *OK*.

To configure remote and local tunnel IP addresses:

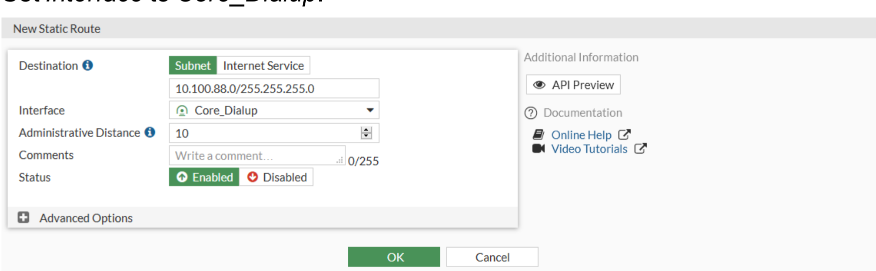
1. Go to *Network > Interfaces* and edit the *Core_Dialup* interface under *port1*.
2. Set *IP* to *172.16.200.1*.
3. Set *Remote IP/Netmask* to *172.16.200.2 255.255.255.0*. This is where remote health check traffic will come from.
4. Enable *Administrative access* for *HTTPS*, *PING*, and *SSH*.



5. Click *OK*.

To configure a route to the remote subnet through the tunnel:

1. Go to *Network > Static Routes* and click *Create New*.
2. Set *Destination* to *Subnet* and enter the IP address and netmask: *10.100.88.0/255.255.255.0*.
3. Set *Interface* to *Core_Dialup*.

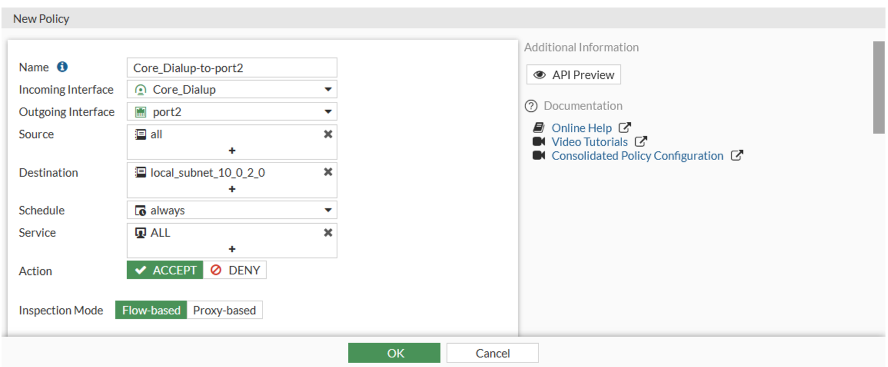


4. Click *OK*.

To configure a firewall policy to allow traffic from the tunnel to port2:

1. Go to *Policy & Objects > Firewall Policy* and click *Create New*.
2. Configure the following:

Name	Core_Dialup-to-port2
Incoming Interface	Core_Dialup
Outgoing Interface	port2
Source	all
Destination	local_subnet_10_0_2_0
Schedule	always
Service	ALL
Action	ACCEPT



3. Configure the remaining settings as required.
4. Click *OK*.

Configure the HQ FortiGate

To create an address for the VPN gateway:

1. Go to *Policy & Objects > Addresses* and click *Create New > Address*.
2. Set *Name* to *remote_subnet_10_0_2_0*.
3. Set *IP/Netmask* to *10.0.2.0/24*.
4. Click *OK*.

To configure a custom IPsec VPN:

1. Go to *VPN > IPsec Wizard*.
2. Set *Name* to *FGT_AWS_Tun*.
3. Set *Template type* to *Custom*.
4. Click *Next*.
5. Configure *Network* settings:

Remote Gateway	Static IP Address
IP Address	100.21.29.17
Interface	port5
NAT Traversal	Enable

6. Configure *Authentication* settings:

Method	Pre-shared Key
Pre-shared Key	Enter the pre-shared key.
Version	1
Mode	Aggressive This setting allows the peer ID to be specified.
Accept Types	Any peer ID

7. Leave the default *Phase 1 Proposal* settings, except set *Local ID* to *IaaS*.
8. Disable *XAUTH*.
9. Configure the *Phase 2 Selector* settings:

Name	FGT_AWS_Tun
Local Address	Named Address - *all* This setting allows traffic originating from both the local subnet 10.100.88.0 and the health checks from the VPN interface. For increased security, each subnet can be specified individually.
Remote Address	Named Address - *remote_subnet_10_0_2_0*

10. Click *OK*.

To configure local and remote tunnel IP addresses:

1. Go to *Network > Interfaces* and edit the *FGT_AWS_Tun* interface under *port5*.
2. Set *IP* to *172.16.200.2*.
3. Set *Remote IP/Netmask* to *172.16.200.1 255.255.255.0*.
4. Enable *Administrative access* for *HTTPS*, *PING*, and *SSH*.
5. Click *OK*.

Routing is defined when creating the SD-WAN interface. The firewall policy is created after the SD-WAN interface is defined.

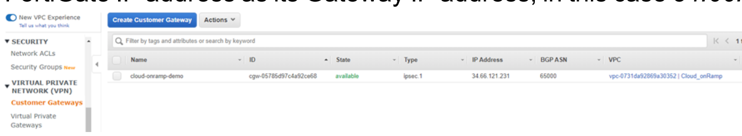
Configuring the VPN overlay between the HQ FortiGate and AWS native VPN gateway

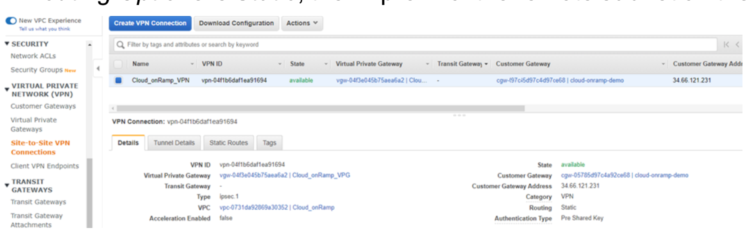
This example uses static routing. It is assumed that the AWS VPN Gateway is already configured, and that proper routing is applied on the corresponding subnet.

Verify the AWS configuration

See Creating routing tables and associate subnets in the AWS Administration Guide for configuration details.

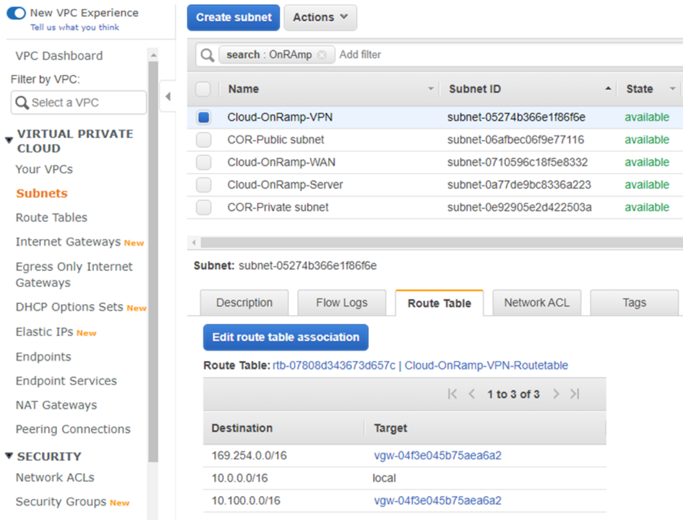
To check the AWS configuration:

1. Go to *Virtual Private Network (VPN) > Customer Gateways* to confirm that the customer gateway defines the FortiGate IP address as its Gateway IP address, in this case *34.66.121.231*.

2. Go to *Virtual Private Network (VPN) > Virtual Private Gateways* to confirm that a virtual private gateway (VPG) has been created. In this case it is attached to the *Cloud_onRamp* VPC that contains the FortiGate and servers.

3. Go to *Virtual Private Network (VPN) > Site-to-Site VPN Connections* to confirm that site-to-site VPN connections have been created and attached to the customer gateway and virtual private gateway.
If *Routing Options* is *Static*, the IP prefix of the remote subnet on the HQ FortiGate (10.100.88.0) is entered here.

AWS site-to-site VPN always creates two VPN tunnels for redundancy. In this example, only Tunnel 1 is used.

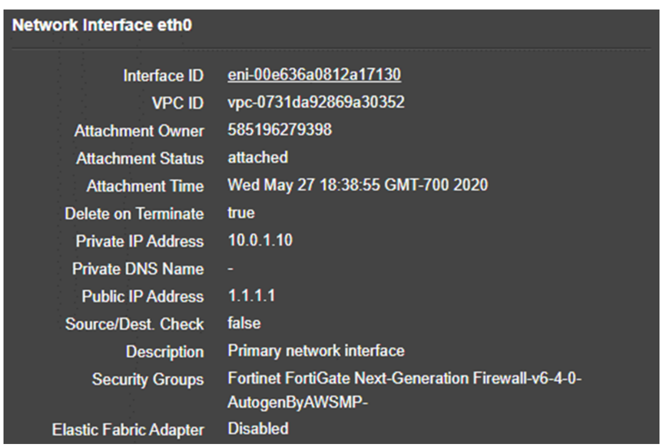

4. Click *Download Configuration* to download the FortiGate's tunnel configurations. The configuration can be referred to when configuring the FortiGate VPN.

5. The new VPG is attached to your VPC, but to successfully route traffic to the VPG, proper routing must be defined. Go to *Virtual Private Cloud > Subnets*, select the *Cloud-OnRamp-VPN*, and select the *Route Table* tab to verify that there are at least two routes to send traffic over the VPG.



- *169.254.0.0/24* defines the tunnel IP address. Health check traffic originating from the FortiGate will come from this IP range.
- *10.100.0.0/16* defines the remote subnet from the HQ FortiGate.
- Both routes point to the just created VPG *vgw-04xxxx*.

6. On the cloud FortiGate-VM EC2 instances, ensure that port1 and port2 both have *Source/Dest. Check* set to *false*. This allows the FortiGate to accept and route traffic to and from a different network.
   If you launched the instance from the AWS marketplace, this setting defaults to *true*.



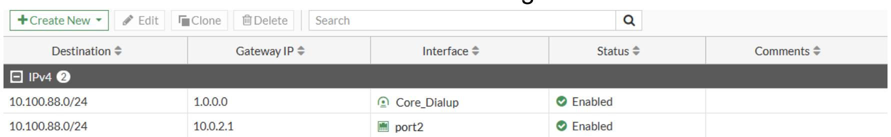
Configure routing to the VPG on the cloud FortiGate-VM

To configure routing to the VPG on the cloud FortiGate-VM:

1. Go to *Network > Static Routes* and click *Create New*.
2. Set *Destination* to *Subnet* and enter the IP address and netmask: *10.100.88.0/255.255.255.0*.
3. Set *Gateway Address* to *Specify* and enter *10.0.2.1*.

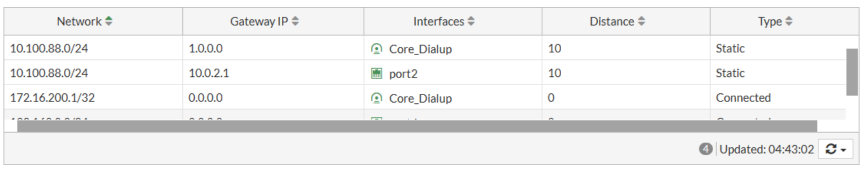
4. Set *Interface* to *port2*.
   The new route must have the same *Administrative Distance* as the route that was created for traffic through the *Core_Dialup* tunnel to ensure that both routes are added to the routing table (see To configure a route to the remote subnet through the tunnel).

   The *Gateway Address* is arbitrarily set to 10.0.2.1. The VPG does not have an IP address, but the address defined here allows the FortiGate to route traffic out of port2, while AWS routes the traffic based on its routing table.
5. Click *OK*.
6. Go to *Network > Static Routes* to view the configured static routes:



Destination	Gateway IP	Interface	Status	Comments
IPv4 2				
10.100.88.0/24	1.0.0.0	Core_Dialup	Enabled	
10.100.88.0/24	10.0.2.1	port2	Enabled	

7. If *Optimal* dashboards is selected, go to *Dashboard > Network* and expand the Routing widget to view the routing table.
   If *Comprehensive* dashboards is selected, go to *Dashboard > Routing Monitor* and select *Static & Dynamic* in the widget toolbar to view the routing table:



Network	Gateway IP	Interfaces	Distance	Type
10.100.88.0/24	1.0.0.0	Core_Dialup	10	Static
10.100.88.0/24	10.0.2.1	port2	10	Static
172.16.200.1/32	0.0.0.0	Core_Dialup	0	Connected

Configure IPsec VPN on the HQ FortiGate

To configure a custom IPsec VPN:

1. Go to *VPN > IPsec Wizard*.
2. Set *Name* to *AWS_VPG*.
3. Set *Template type* to *Custom*.
4. Click *Next*.
5. Configure *Network* settings:

Remote Gateway	Static IP Address
IP Address	34.210.19.225 This address is taken from the downloaded AWS configuration file.
Interface	port1
NAT Traversal	Enable

6. Configure *Authentication* settings:

Method	Pre-shared Key
Pre-shared Key	Enter the pre-shared key.
Version	1
Mode	Main

7. Configure the *Phase 1 Proposal* settings using information from the downloaded AWS configuration file.

8. Disable *XAUTH*.
9. Configure the *Phase 2 Selector* settings:

Name	AWS_VPG
Local Address	Named Address - *all* This setting allows traffic originating from both the local subnet 10.100.88.0 and the health checks from the VPN interface. For increased security, each subnet can be specified individually.
Remote Address	Named Address - *remote_subnet_10_0_2_0*

10. Click *OK*.

To configure local and remote tunnel IP addresses:

1. Go to *Network > Interfaces* and edit the *AWS_VPG* interface under *port1*.
2. Set *IP* to *169.254.55.154*.
3. Set *Remote IP/Netmask* to *169.254.55.153 255.255.255.0*.
4. Enable *Administrative access* for *HTTPS* and *PING*.
5. Click *OK*.

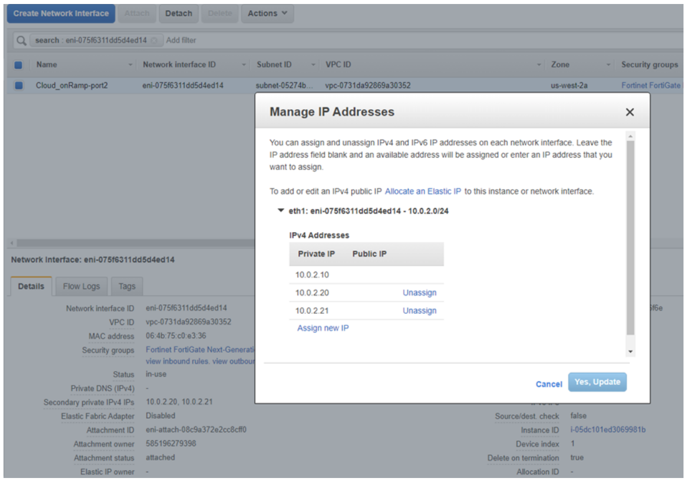
Routing is defined when creating the SD-WAN interface. The firewall policy is created after the SD-WAN interface is defined.

Configuring the VIP to access the remote servers

VIPs, interface IP addresses, and policies are created on the cloud FortiGate-VM to allow access to the remote servers.

To configure additional private IPs on AWS for the FortiGate VIP:

1. On the FortiGate EC2 instance, edit the *Elastic Network Interface* that corresponds to *port2*. In this example, Network Interface *eth1*.
2. Go to *Actions > Manage IP Addresses*.
3. Add two private IP address in the 10.0.2.0/24 subnet.
   These address will be used in the VIPs on the FortiGate. This ensures that traffic to these IP addresses is routed to the FortiGate by AWS.

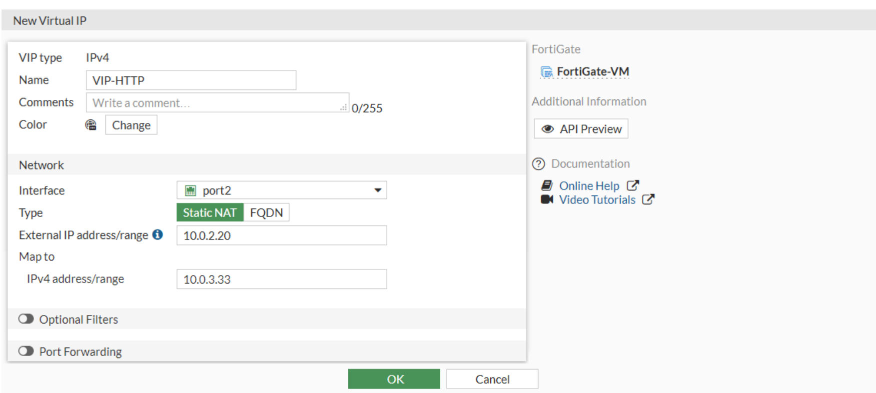


4. Click *Yes, Update*.

To configure VIPs on the cloud FortiGate-VM:

1. Go to *Policy & Objects* > *Virtual IPs* and click *Create New* > *Virtual IP*.
2. Configure the following:

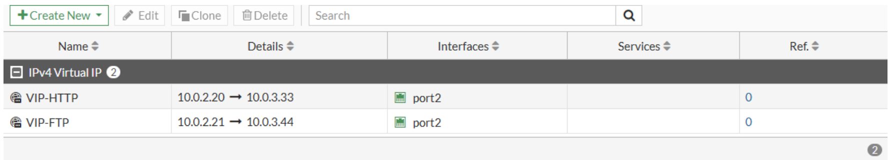
Name	VIP-HTTP
Interface	port2
External IP address/range	10.0.2.20
Map to IPv4 address/range	10.0.3.33



3. Click *OK*.

4. Create a second VIP for the FTP server with the following settings:

Name	VIP-FTP
Interface	port2
External IP address/range	10.0.2.21
Map to IPv4 address/range	10.0.3.44



To configure firewall policies to allow traffic from port2 to port3:

1. Go to *Policy & Objects > Firewall Policy* and click *Create New*.
2. Configure the following:

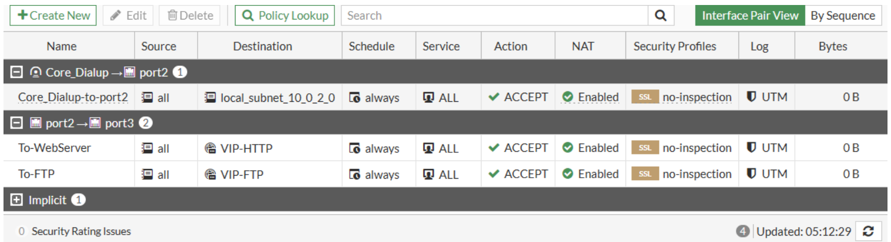
Name	To-WebServer
Incoming Interface	port2
Outgoing Interface	port3
Source	all
Destination	VIP-HTTP
Schedule	always
Service	ALL
Action	ACCEPT
NAT	Enabled

3. Configure the remaining settings as required.
4. Click *OK*.
5. Create a second policy for the FTP VIP with the following settings:

Name	To-FTP
Incoming Interface	port2
Outgoing Interface	port3
Source	all
Destination	VIP-FTP
Schedule	always
Service	ALL
Action	ACCEPT

NAT	Enabled

6. Click *OK*.



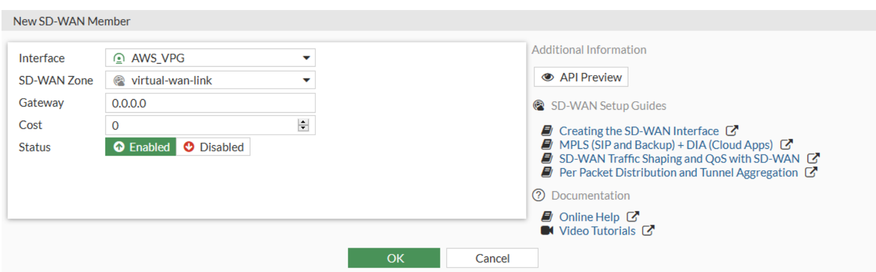
Configuring the SD-WAN to steer traffic between the overlays

Configure the HQ FortiGate to use two overlay tunnels for SD-WAN, steering HTTPS and HTTP traffic through the FGT_ AWS_Tun tunnel, and SSH and FTP throguh the AWS_VPG tunnel.

1. Add SD-WAN member interfaces
2. Configure a route to the remote network
3. Configure firewall policies
4. Configure a health check
5. Configure SD-WAN rules

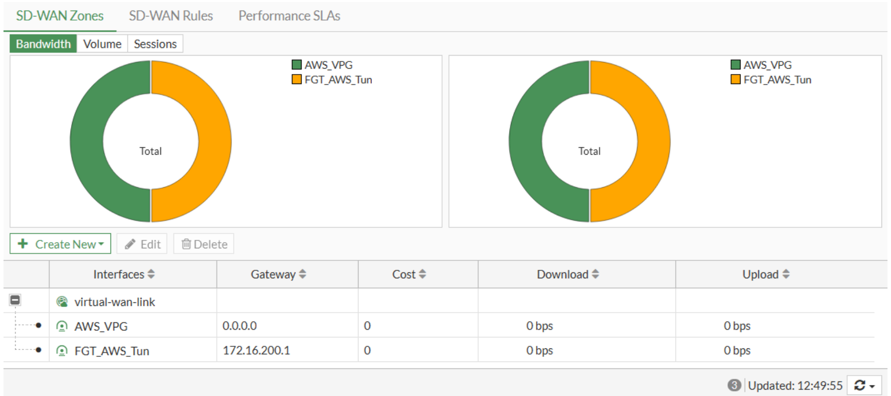
To add SD-WAN member interfaces:

1. Go to *Network > SD-WAN*, select the *SD-WAN Zones* tab, and click *Create New > SD-WAN Member*.
2. Set *Interface* to *AWS_VPG* then click *OK*.



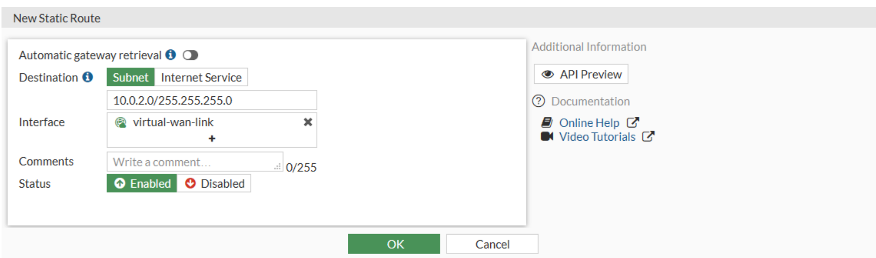
3. Click *Create New > SD-WAN Member* again.
4. Set *Interface* to *FGT_AWS_Tun*.
5. Set *Gateway* to *172.16.200.1*.

6. Click *OK*.



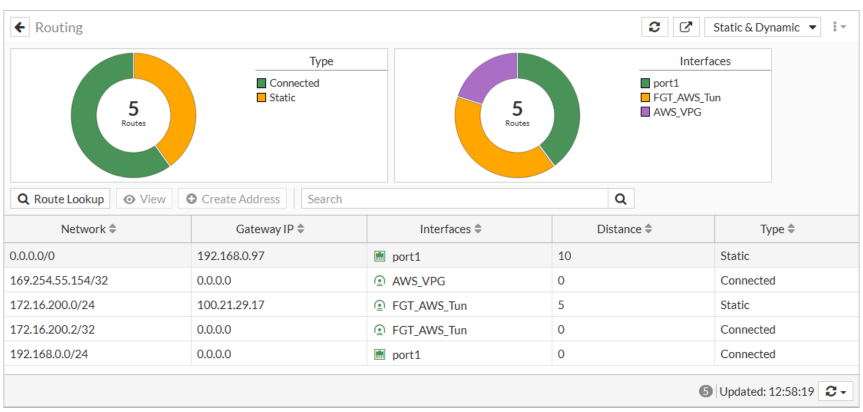
To configure a route to the remote network 10.0.2.0/24:

1. Go to *Network > Static Routes* and click *Create New*.
2. Set *Destination* to *Subnet* and enter the IP address and netmask: *10.0.2.0/255.255.255.0*.
3. Set *Interface* to *virtual-wan-link*.



4. Click *OK*.

   Individual routes to each tunnel are automatically added to the routing table with the same distance:



To configure firewall policies to allow traffic from the internal subnet to SD-WAN:

1. Go to *Policy & Objects > Firewall Policy* and click *Create New*.
2. Configure the following:

Name	ISFW-to-IaaS

Incoming Interface	port3
Outgoing Interface	virtual-wan-link
Source	all
Destination	all
Schedule	always
Service	ALL
Action	ACCEPT
NAT	Enabled

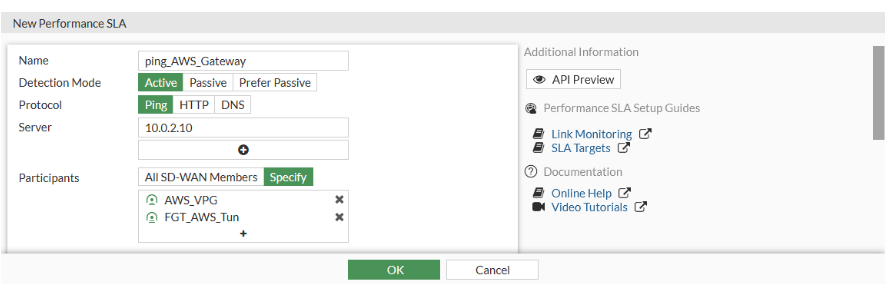
3. Configure the remaining settings as required.
4. Click *OK*.
   Once the firewall policies are configured, the VPN tunnels should come up when there is traffic.

To configure a health check to monitor the status of the tunnels:

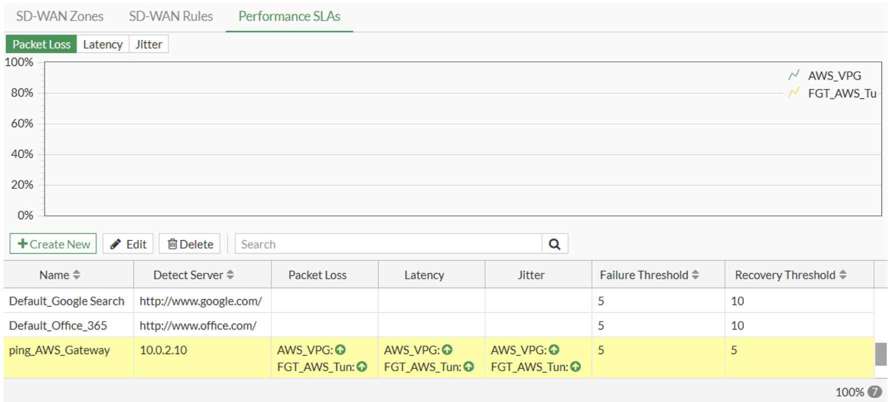
As you are accessing the servers on the 10.0.2.0/24 subnet, it is preferable to use the FortiGate port2 interface as the ping server for detection. This ensures that, if the gateway is not reachable in either tunnel, its routes are brought down and traffic continues on the other tunnel.

1. Go to *Network > SD-WAN*, select the *Performance SLAs* tab, and click *Create New*.
2. Configure the following:

Name	ping_AWS_Gateway
Protocol	Ping
Server	10.0.2.10
Participants	Specify Add AWS_VPG and FGT_AWS_Tun as participants.



3. Click *OK*.



Health check probes originate from the VPN interface's IP address. This is why the phase2 selectors are configured with *Local Address* set to *all*.

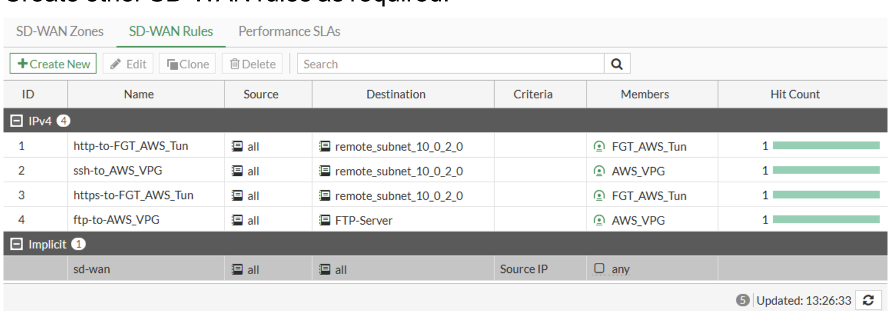
To configure SD-WAN rules to steer traffic:

HTTPS and HTTP traffic is steered to the FGT_AWS_Tun tunnel, and SSH and FTP traffic is steered to the AWS_VPG tunnel. The Manual algorithm is used in this example.

1. Go to *Network > SD-WAN*, select the *SD-WAN Rules* tab, and click *Create New*.
2. Configure the following:

Name	http-to-FGT_AWS_Tun
Source Address	all
Address	remote_subnet_10_0_2_0
Protocol	TCP
Port range	80 - 80
Outgoing Interfaces	Manual
Interface preference	FGT_AWS_Tun

3. Click *OK*.
4. Create other SD-WAN rules as required:



Verifying the traffic

To verify that pings are sent across the IPsec VPN tunnels

- On the HQ FortiGate, run the following CLI command:

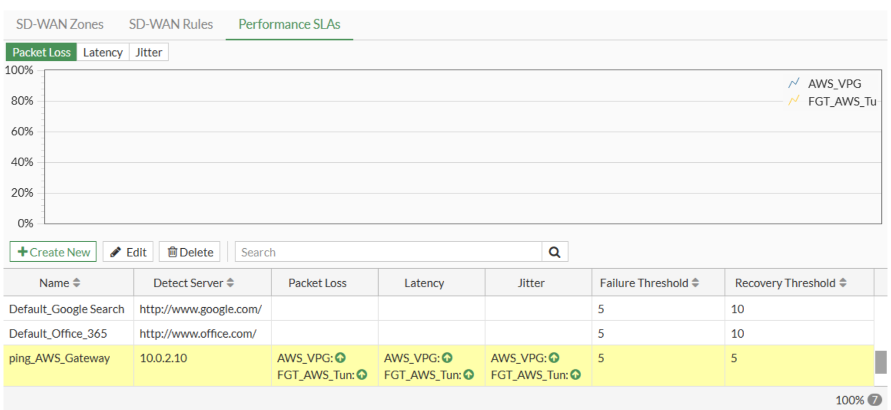
```
# diagnose sniffer packet any 'host 10.0.2.10' 4 0 1 interfaces=[any]
Using Original Sniffing Mode
interfaces=[any]
filters=[host 10.0.2.10]
pcap_snapshot: snaplen raised from 0 to 262144
2021-06-05 11:35:14.822600 AWS_VPG out 169.254.55.154 -> 10.0.2.10: icmp: echo request
2021-06-05 11:35:14.822789 FGT_AWS_Tun out 172.16.200.2 -> 10.0.2.10: icmp: echo request
2021-06-05 11:35:14.877862 FGT_AWS_Tun in 10.0.2.10 -> 172.16.200.2: icmp: echo reply
2021-06-05 11:35:14.878887 AWS_VPG in 10.0.2.10 -> 169.254.55.154: icmp: echo reply
```

- On the cloud FortiGate-VM, run the following CLI command:

```
# diagnose sniffer packet any 'host 10.0.2.10' 4 0 1 interfaces=[any]
Using Original Sniffing Mode
interfaces=[any]
filters=[host 10.0.2.10]
pcap_snapshot: snaplen raised from 0 to 262144
2021-06-05 11:37:57.176329 port2 in 169.254.55.154 -> 10.0.2.10: icmp: echo request
2021-06-05 11:37:57.176363 port2 out 10.0.2.10 -> 169.254.55.154: icmp: echo reply
2021-06-05 11:37:57.176505 Core_Dialup in 172.16.200.2 -> 10.0.2.10: icmp: echo request
2021-06-05 11:37:57.176514 Core_Dialup out 10.0.2.10 -> 172.16.200.2: icmp: echo reply
```

To verify the SLA health checks on the HQ FortiGate:

1. Go to *Network > SD-WAN*, select the *Performance SLAs* tab, select *Packet Loss*, and click the *ping_AWS_Gateway* SLA:

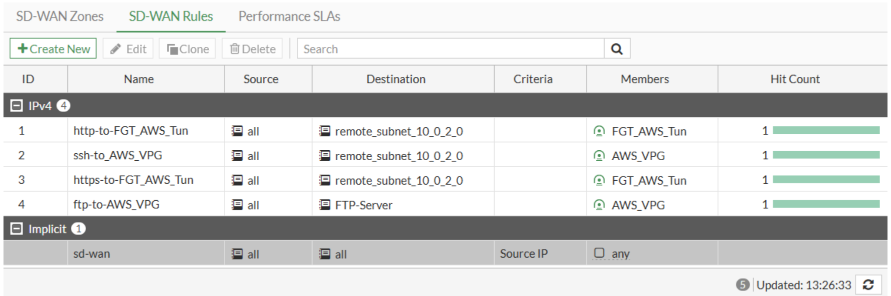


2. Run the following CLI command:

```
# diagnose sys sdwan health-check
…
Seq(1 AWS_VPG): state(alive), packet-loss(0.000%) latency(56.221), jitter(0.290) sla_
map=0x0
Seq(2 FGT_AWS_Tun): state(alive), packet-loss(0.000%) latency(55.039), jitter(0.223)
sla_map=0x0
```

To verify service rules:

1. Go to *Network > SD-WAN* and select the *SD-WAN Rules* tab:



2. Run the following CLI command:

```
# diagnose sys sdwan service

Service(1): Address Mode(IPV4) flags=0x0
  Gen(1), TOS(0x0/0x0), Protocol(6: 80->80), Mode(manual)
  Members:
    1: Seq_num(2 FGT_AWS_Tun), alive, selected
  Src address:
        0.0.0.0-255.255.255.255
  Dst address:
        10.0.2.0-10.0.2.255

Service(2): Address Mode(IPV4) flags=0x0
  Gen(1), TOS(0x0/0x0), Protocol(6: 22->22), Mode(manual)
  Members:
    1: Seq_num(1 AWS_VPG), alive, selected
  Src address:
        0.0.0.0-255.255.255.255
  Dst address:
        10.0.2.0-10.0.2.255

Service(3): Address Mode(IPV4) flags=0x0
  Gen(1), TOS(0x0/0x0), Protocol(6: 443->443), Mode(manual)
  Members:
    1: Seq_num(2 FGT_AWS_Tun), alive, selected
  Src address:
        0.0.0.0-255.255.255.255
  Dst address:
        10.0.2.0-10.0.2.255

Service(4): Address Mode(IPV4) flags=0x0
  Gen(1), TOS(0x0/0x0), Protocol(0: 1->65535), Mode(manual)
  Members:
    1: Seq_num(1 AWS_VPG), alive, selected
  Src address:
        0.0.0.0-255.255.255.255
  Dst address:
        10.0.2.21-10.0.2.21
```

To verify that sessions are going to the correct tunnel:

1. Run the following CLI command to verify that HTTPS and HTTP traffic destined for the Web server at 10.0.2.20 uses FGT_AWS_Tun:

```
# diagnose sys session filter dst 10.0.2.20
# diagnose sys session list

session info: proto=6 proto_state=11 duration=2 expire=3597 timeout=3600 flags=00000000
socktype=0 sockport=0 av_idx=0 use=4
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=FGT_AWS_Tun/ vlan_cos=0/255
state=log may_dirty npu f00 csf_syncd_log app_valid
statistic(bytes/packets/allow_err): org=593/4/1 reply=3689/5/1 tuples=3
tx speed(Bps/kbps): 264/2 rx speed(Bps/kbps): 1646/13
orgin->sink: org pre->post, reply pre->post dev=0->18/18->0 gwy=172.16.200.1/0.0.0.0
hook=post dir=org act=snat 10.100.88.101:55589->10.0.2.20:80(172.16.200.2:55589)
hook=pre dir=reply act=dnat 10.0.2.20:80->172.16.200.2:55589(10.100.88.101:55589)
hook=post dir=reply act=noop 10.0.2.20:80->10.100.88.101:55589(0.0.0.0:0)
pos/(before,after) 0/(0,0), 0/(0,0)
src_mac=00:09:0f:00:03:01
misc=0 policy_id=32 auth_info=0 chk_client_info=0 vd=0
serial=00b7442c tos=ff/ff app_list=2000 app=34050 url_cat=0
sdwan_mbr_seq=0 sdwan_service_id=0
rpdb_link_id= ff000001 rpdb_svc_id=2154552596 ngfwid=n/a
npu_state=0x3041008

session info: proto=6 proto_state=66 duration=1 expire=3 timeout=3600 flags=00000000
socktype=0 sockport=0 av_idx=0 use=4
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=FGT_AWS_Tun/ vlan_cos=0/255
state=log may_dirty ndr f00 csf_syncd_log
statistic(bytes/packets/allow_err): org=48/1/0 reply=40/1/1 tuples=3
tx speed(Bps/kbps): 26/0 rx speed(Bps/kbps): 22/0
orgin->sink: org pre->post, reply pre->post dev=5->18/18->5
gwy=172.16.200.1/10.100.88.101
hook=post dir=org act=snat 10.100.88.101:55621->10.0.2.20:443(172.16.200.2:55621)
hook=pre dir=reply act=dnat 10.0.2.20:443->172.16.200.2:55621(10.100.88.101:55621)
hook=post dir=reply act=noop 10.0.2.20:443->10.100.88.101:55621(0.0.0.0:0)
pos/(before,after) 0/(0,0), 0/(0,0)
src_mac=00:09:0f:00:03:01
misc=0 policy_id=32 auth_info=0 chk_client_info=0 vd=0
serial=00b74b50 tos=ff/ff app_list=2000 app=0 url_cat=0
sdwan_mbr_seq=0 sdwan_service_id=0
rpdb_link_id= ff000003 rpdb_svc_id=2154552596 ngfwid=n/a
npu_state=0x3041008
```

2. Run the following CLI command to verify that SSH and FTP traffic destined for the FTP server at 10.0.2.21 uses AWS_VPG:

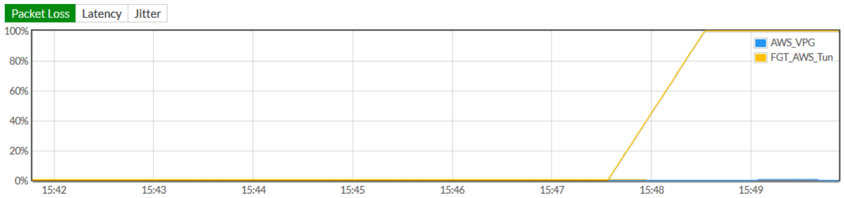
```
# diagnose sys session filter dst 10.0.2.20
# diagnose sys session list
```

```
session info: proto=6 proto_state=11 duration=197 expire=3403 timeout=3600
flags=00000000 socktype=0 sockport=0 av_idx=0 use=4
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=AWS_VPG/ helper=ftp vlan_cos=0/255
state=log may_dirty ndr npu f00 csf_syncd_log app_valid
statistic(bytes/packets/allow_err): org=580/12/1 reply=863/13/1 tuples=3
tx speed(Bps/kbps): 2/0 rx speed(Bps/kbps): 4/0
orgin->sink: org pre->post, reply pre->post dev=5->17/17->5
gwy=169.254.55.153/10.100.88.101
hook=post dir=org act=snat 10.100.88.101:55528->10.0.2.21:21(169.254.55.154:55528)
hook=pre dir=reply act=dnat 10.0.2.21:21->169.254.55.154:55528(10.100.88.101:55528)
hook=post dir=reply act=noop 10.0.2.21:21->10.100.88.101:55528(0.0.0.0:0)
pos/(before,after) 0/(0,0), 0/(0,0)
src_mac=00:09:0f:00:03:01
misc=0 policy_id=32 auth_info=0 chk_client_info=0 vd=0
serial=00b72a5f tos=ff/ff app_list=2000 app=15896 url_cat=0
sdwan_mbr_seq=0 sdwan_service_id=0
rpdb_link_id= ff000004 rpdb_svc_id=2149689849 ngfwid=n/a
npu_state=0x3041008

session info: proto=6 proto_state=11 duration=3 expire=3596 timeout=3600 flags=00000000
socktype=0 sockport=0 av_idx=0 use=4
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=AWS_VPG/ vlan_cos=0/255
state=log may_dirty ndr npu f00 csf_syncd_log app_valid
statistic(bytes/packets/allow_err): org=1496/6/1 reply=1541/5/1 tuples=3
tx speed(Bps/kbps): 416/3 rx speed(Bps/kbps): 429/3
orgin->sink: org pre->post, reply pre->post dev=5->17/17->5
gwy=169.254.55.153/10.100.88.101
hook=post dir=org act=snat 10.100.88.101:55644->10.0.2.21:22(169.254.55.154:55644)
hook=pre dir=reply act=dnat 10.0.2.21:22->169.254.55.154:55644(10.100.88.101:55644)
hook=post dir=reply act=noop 10.0.2.21:22->10.100.88.101:55644(0.0.0.0:0)
pos/(before,after) 0/(0,0), 0/(0,0)
src_mac=00:09:0f:00:03:01
misc=0 policy_id=32 auth_info=0 chk_client_info=0 vd=0
serial=00b75287 tos=ff/ff app_list=2000 app=16060 url_cat=0
sdwan_mbr_seq=0 sdwan_service_id=0
rpdb_link_id= ff000002 rpdb_svc_id=2149689849 ngfwid=n/a
npu_state=0x3041008
```

To simulate an issue on an overlay VPN tunnel:

On the cloud FortiGate-VM, disable the firewall policy allowing Core_Dialup to port2.

1. Health-checks through the FGT_AWS_Tun tunnel fail:
   a. Go to *Network > SD-WAN*, select the *Performance SLAs* tab, select *Packet Loss*, and click the *ping_AWS_Gateway* SLA:



+ Create New | Edit | Delete | Search

Name	Detect Server	Packet Loss	Latency	Jitter	Failure Threshold	Recovery Threshold
Default_AWS	http://aws.amazon.com/				5	10
Default_DNS	208.91.112.53 208.91.112.52 (System DNS)				5	10
Default_FortiGuard	http://fortiguard.com/				5	10
Default_Gmail	gmail.com				5	10
Default_Google Search	http://www.google.com/				5	10
Default_Office_365	http://www.office.com/				5	10
ping_AWS_Gateway	10.0.2.10	AWS_VPG: 0.00% FGT_AWS_Tun:	AWS_VPG: 54.98ms FGT_AWS_Tun:	AWS_VPG: 0.41ms FGT_AWS_Tun:	5	5

b. Run the following CLI command:

```
# diagnose sys sdwan health-check
...
Seq(1 AWS_VPG): state(alive), packet-loss(0.000%) latency(52.746), jitter(0.713) sla_
map=0x0
Seq(2 FGT_AWS_Tun): state(dead), packet-loss(19.000%) sla_map=0x0
```

2. Service rules show that the member is down:

a. Go to *Network > SD-WAN* and select the *SD-WAN Rules* tab:

+ Create New | Edit | Delete | Search

ID	Name	Source	Destination	Criteria	Members	Hit Count
IPv4 4						
1	http-to-FGT_AWS_Tun	all	remote_subnet_10_0_2_0		FGT_AWS_Tun	1
2	ssh-to_AWS_VPG	all	remote_subnet_10_0_2_0		AWS_VPG	2
3	https-to-FGT_AWS_Tun	all	remote_subnet_10_0_2_0		FGT_AWS_Tun	1
4	ftp_to_AWS_VPG	all	FTP-Server		AWS_VPG	2
Implicit 1						
	sd-wan	all	all	Source IP	any	

b. Run the following CLI command:

```
# diagnose sys sdwan service

Service(1): Address Mode(IPV4) flags=0x0
  Gen(2), TOS(0x0/0x0), Protocol(6: 80->80), Mode(manual)
  Members:
    1: Seq_num(2 FGT_AWS_Tun), dead
  Src address:
        0.0.0.0-255.255.255.255
  Dst address:
        10.0.2.0-10.0.2.255

Service(2): Address Mode(IPV4) flags=0x0
  Gen(1), TOS(0x0/0x0), Protocol(6: 22->22), Mode(manual)
  Members:
    1: Seq_num(1 AWS_VPG), alive, selected
  Src address:
        0.0.0.0-255.255.255.255
  Dst address:
        10.0.2.0-10.0.2.255
```

```
Service(3): Address Mode(IPV4) flags=0x0
  Gen(2), TOS(0x0/0x0), Protocol(6: 443->443), Mode(manual)
  Members:
    1: Seq_num(2 FGT_AWS_Tun), dead
  Src address:
        0.0.0.0-255.255.255.255
  Dst address:
        10.0.2.0-10.0.2.255

Service(4): Address Mode(IPV4) flags=0x0
  Gen(1), TOS(0x0/0x0), Protocol(0: 1->65535), Mode(manual)
  Members:
    1: Seq_num(1 AWS_VPG), alive, selected
  Src address:
        0.0.0.0-255.255.255.255
  Dst address:
        10.0.2.21-10.0.2.21
```

3. Sessions are redirected to the working tunnel:
   a. Run the following CLI command:

```
# diagnose sys session list

session info: proto=6 proto_state=11 duration=3 expire=3596 timeout=3600
flags=00000000 socktype=0 sockport=0 av_idx=0 use=4
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=AWS_VPG/ vlan_cos=0/255
state=log may_dirty ndr npu f00 csf_syncd_log app_valid
statistic(bytes/packets/allow_err): org=504/4/1 reply=620/3/1 tuples=3
tx speed(Bps/kbps): 150/1 rx speed(Bps/kbps): 184/1
orgin->sink: org pre->post, reply pre->post dev=0->17/17->0
gwy=169.254.55.153/0.0.0.0
hook=post dir=org act=snat 10.100.88.101:56373->10.0.2.20:80(169.254.55.154:56373)
hook=pre dir=reply act=dnat 10.0.2.20:80->169.254.55.154:56373(10.100.88.101:56373)
hook=post dir=reply act=noop 10.0.2.20:80->10.100.88.101:56373(0.0.0.0:0)
pos/(before,after) 0/(0,0), 0/(0,0)
src_mac=00:09:0f:00:03:01
misc=0 policy_id=32 auth_info=0 chk_client_info=0 vd=0
serial=00b87199 tos=ff/ff app_list=2000 app=34050 url_cat=0
rpdb_link_id= 80000000 rpdb_svc_id=0 ngfwid=n/a
npu_state=0x3041008

session info: proto=6 proto_state=66 duration=3 expire=1 timeout=3600 flags=00000000
socktype=0 sockport=0 av_idx=0 use=4
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=AWS_VPG/ vlan_cos=0/255
state=log may_dirty ndr f00 csf_syncd_log
statistic(bytes/packets/allow_err): org=48/1/0 reply=40/1/1 tuples=3
tx speed(Bps/kbps): 15/0 rx speed(Bps/kbps): 12/0
orgin->sink: org pre->post, reply pre->post dev=5->17/17->5
gwy=169.254.55.153/10.100.88.101
hook=post dir=org act=snat 10.100.88.101:56383->10.0.2.20:443(169.254.55.154:56383)
```

```
hook=pre dir=reply act=dnat 10.0.2.20:443->169.254.55.154:56383(10.100.88.101:56383)
hook=post dir=reply act=noop 10.0.2.20:443->10.100.88.101:56383(0.0.0.0:0)
pos/(before,after) 0/(0,0), 0/(0,0)
src_mac=00:09:0f:00:03:01
misc=0 policy_id=32 auth_info=0 chk_client_info=0 vd=0
serial=00b876bb tos=ff/ff app_list=2000 app=0 url_cat=0
rpdb_link_id= 80000000 rpdb_svc_id=0 ngfwid=n/a
npu_state=0x3041008
total session 2
```

4. Routes to the *FGT_AWS_Tun* tunnel are removed:
   a. If *Optimal* dashboards is selected, go to *Dashboard* > *Network* and expand the Routing widget to view the routing table.
   If *Comprehensive* dashboards is selected, go to *Dashboard* > *Routing Monitor* and select *Static & Dynamic* in the widget toolbar to view the routing table:

Network	Gateway IP	Interfaces	Distance	IP Version	Type
IPv4 40					
0.0.0.0/0	10.100.64.254	Internet_A (port1)	1	IPv4	Static
0.0.0.0/0	10.100.65.254	Internet_B (port5)	1	IPv4	Static
10.0.2.0/24	169.254.55.153	AWS_VPG	1	IPv4	Static
10.0.10.0/24	0.0.0.0	VPN_A_Tunnel (Branch-HQ-A)	0	IPv4	Connected
10.0.10.1/32	0.0.0.0	VPN_A_Tunnel (Branch-HQ-A)	0	IPv4	Connected

   b. Run the following CLI command:

```
# get router info routing-table all
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default

Routing table for VRF=0
S*      0.0.0.0/0 [1/0] via 10.100.64.254, port1
                  [1/0] via 10.100.65.254, port5
S       10.0.2.0/24 [1/0] via 169.254.55.153, AWS_VPG
C       10.0.10.0/24 is directly connected, Branch-HQ-A
C       10.0.10.1/32 is directly connected, Branch-HQ-A
…
```

Troubleshooting SD-WAN

The following topics provide instructions on SD-WAN troubleshooting:

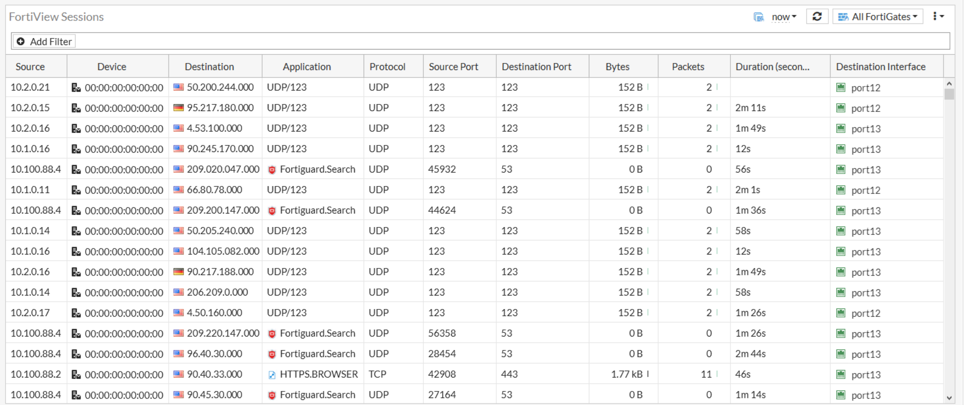
- Tracking SD-WAN sessions on page 744
- Understanding SD-WAN related logs on page 744
- SD-WAN related diagnose commands on page 747
- SD-WAN bandwidth monitoring service on page 751
- Using SNMP to monitor health check on page 754

Tracking SD-WAN sessions

You can check the destination interface in *Dashboard > FortiView Sessions* in order to see which port the traffic is being forwarded to.

The example below demonstrates a source-based load-balance between two SD-WAN members:

- If the source IP address is an *even* number, it will go to *port13*.
- If the source IP address is an *odd* number, it will go to *port12*.


FortiView Sessions

now

All FortiGates

Add Filter

Source	Device	Destination	Application	Protocol	Source Port	Destination Port	Bytes	Packets	Duration (secon...	Destination Interface
10.2.0.21	00:00:00:00:00:00	50.200.244.000	UDP/123	UDP	123	123	152 B	2		port12
10.2.0.15	00:00:00:00:00:00	95.217.180.000	UDP/123	UDP	123	123	152 B	2	2m 11s	port12
10.2.0.16	00:00:00:00:00:00	4.53.100.000	UDP/123	UDP	123	123	152 B	2	1m 49s	port13
10.1.0.16	00:00:00:00:00:00	90.245.170.000	UDP/123	UDP	123	123	152 B	2	12s	port13
10.100.88.4	00:00:00:00:00:00	209.020.047.000	Fortiguard.Search	UDP	45932	53	0 B	0	56s	port13
10.1.0.11	00:00:00:00:00:00	66.80.78.000	UDP/123	UDP	123	123	152 B	2	2m 1s	port12
10.100.88.4	00:00:00:00:00:00	209.200.147.000	Fortiguard.Search	UDP	44624	53	0 B	0	1m 36s	port13
10.1.0.14	00:00:00:00:00:00	50.205.240.000	UDP/123	UDP	123	123	152 B	2	58s	port13
10.1.0.16	00:00:00:00:00:00	104.105.082.000	UDP/123	UDP	123	123	152 B	2	12s	port13
10.2.0.16	00:00:00:00:00:00	90.217.188.000	UDP/123	UDP	123	123	152 B	2	1m 49s	port13
10.1.0.14	00:00:00:00:00:00	206.209.0.000	UDP/123	UDP	123	123	152 B	2	58s	port13
10.2.0.17	00:00:00:00:00:00	4.50.160.000	UDP/123	UDP	123	123	152 B	2	1m 26s	port12
10.100.88.4	00:00:00:00:00:00	209.220.147.000	Fortiguard.Search	UDP	56358	53	0 B	0	1m 26s	port13
10.100.88.4	00:00:00:00:00:00	96.40.30.000	Fortiguard.Search	UDP	28454	53	0 B	0	2m 44s	port13
10.100.88.2	00:00:00:00:00:00	90.40.33.000	HTTPS.BROWSER	TCP	42908	443	1.77 kB	11	46s	port13
10.100.88.4	00:00:00:00:00:00	90.45.30.000	Fortiguard.Search	UDP	27164	53	0 B	0	1m 14s	port13

Understanding SD-WAN related logs

This topic lists the SD-WAN related logs and explains when the logs will be triggered.

Health-check detects a failure:

- When health-check detects a failure, it will record a log:

```
1: date=2021-04-20 time=17:06:31 eventtime=1618963591590008160 tz="-0700"
logid="0100022921" type="event" subtype="system" level="critical" vd="root"
logdesc="Routing information changed" name="test" interface="R150" status="down"
msg="Static route on interface R150 may be removed by health-check test. Route:
(10.100.1.2->10.100.2.22 ping-down)"
```

- When health-check detects a recovery, it will record a log:

```
2: date=2021-04-20 time=17:11:46 eventtime=1618963906950174240 tz="-0700"
logid="0100022921" type="event" subtype="system" level="critical" vd="root"
logdesc="Routing information changed" name="test" interface="R150" status="up"
msg="Static route on interface R150 may be added by health-check test. Route:
(10.100.1.2->10.100.2.22 ping-up)"
```

Health-check has an SLA target and detects SLA qualification changes:

- When health-check has an SLA target and detects SLA changes, and changes to fail:

```
1: date=2021-04-20 time=21:32:33 eventtime=1618979553388763760 tz="-0700"
logid="0113022923" type="event" subtype="sdwan" level="notice" vd="root" logdesc="SDWAN
status" eventtype="Health Check" healthcheck="test" slatargetid=1 oldvalue="2"
newvalue="1" msg="Number of pass member changed."
```

```
2: date=2021-04-20 time=21:32:33 eventtime=1618979553388751880 tz="-0700"
logid="0113022923" type="event" subtype="sdwan" level="notice" vd="root" logdesc="SDWAN
status" eventtype="Health Check" healthcheck="test" slatargetid=1 member="1" msg="Member
status changed. Member out-of-sla."
```

- When health-check has an SLA target and detects SLA changes, and changes to pass:

```
1: date=2021-04-20 time=21:38:49 eventtime=1618979929908765200 tz="-0700"
logid="0113022923" type="event" subtype="sdwan" level="notice" vd="root" logdesc="SDWAN
status" eventtype="Health Check" healthcheck="test" slatargetid=1 oldvalue="1"
newvalue="2" msg="Number of pass member changed."
```

```
2: date=2021-04-20 time=21:38:49 eventtime=1618979929908754060 tz="-0700"
logid="0113022923" type="event" subtype="sdwan" level="information" vd="root"
logdesc="SDWAN status" eventtype="Health Check" healthcheck="test" slatargetid=1
member="1" msg="Member status changed. Member in sla."
```

SD-WAN calculates a link's session/bandwidth over/under its ratio and stops/resumes traffic:

- When SD-WAN calculates a link's session/bandwidth over its configured ratio and stops forwarding traffic:

```
1: date=2021-04-20 time=21:55:14 eventtime=1618980914728863220 tz="-0700"
logid="0113022924" type="event" subtype="sdwan" level="notice" vd="root" logdesc="SDWAN
volume status" eventtype="Volume" interface="R160" member="2" msg="Member enters into
conservative status with limited ablity to receive new sessions for too much traffic."
```

- When SD-WAN calculates a link's session/bandwidth according to its ratio and resumes forwarding traffic:

```
2: date=2021-04-20 time=22:12:52 eventtime=1618981972698753360 tz="-0700"
logid="0113022924" type="event" subtype="sdwan" level="notice" vd="root" logdesc="SDWAN
volume status" eventtype="Volume" interface="R160" member="2" msg="Member resume normal
status to receive new sessions for internal adjustment"
```

The SLA mode service rule's SLA qualified member changes:

- When the SLA mode service rule's SLA qualified member changes. In this example R150 fails the SLA check, but is still alive:

```
1: date=2021-04-20 time=22:40:46 eventtime=1618983646428803040 tz="-0700"
logid="0113022923" type="event" subtype="sdwan" level="notice" vd="root" logdesc="SDWAN
status" eventtype="Service" serviceid=1 service="test" seq="2,1" msg="Service
prioritized by SLA will be redirected in sequence order."
```

- When the SLA mode service rule's SLA qualified member changes. In this example R150 changes from fail to pass:

```
2: date=2021-04-20 time=22:41:51 eventtime=1618983711678827920 tz="-0700"
logid="0113022923" type="event" subtype="sdwan" level="notice" vd="root" logdesc="SDWAN
status" eventtype="Service" serviceid=1 service="test" seq="1,2" msg="Service
prioritized by SLA will be redirected in sequence order."
```

The priority mode service rule member's link status changes:

- When priority mode service rule member's link status changes. In this example R150 changes to better than R160, and both are still alive:

```
1: date=2021-04-20 time=22:56:55 eventtime=1618984615708804760 tz="-0700"
logid="0113022923" type="event" subtype="sdwan" level="notice" vd="root" logdesc="SDWAN
status" eventtype="Service" serviceid=1 service="test" metric="packet-loss" seq="2,1"
msg="Service prioritized by performance metric will be redirected in sequence order."
```

- When priority mode service rule member's link status changes. In this example R160 changes to better than R150, and both are still alive:

```
2: date=2021-04-20 time=22:56:58 eventtime=1618984618278852140 tz="-0700"
logid="0113022923" type="event" subtype="sdwan" level="notice" vd="root" logdesc="SDWAN
status" eventtype="Service" serviceid=1 service="test" metric="packet-loss" seq="1,2"
msg="Service prioritized by performance metric will be redirected in sequence order."
```

SD-WAN member is used in service and it fails the health-check:

- When SD-WAN member fails the health-check, it will stop forwarding traffic:

```
1: date=2021-04-20 time=23:04:32 eventtime=1618985072898756700 tz="-0700"
logid="0113022923" type="event" subtype="sdwan" level="notice" vd="root" logdesc="SDWAN
status" eventtype="Service" interface="R150" member="1" serviceid=1 service="test"
gateway=10.100.1.1 msg="Member link is unreachable or miss threshold. Stop forwarding
traffic. "
```

- When SD-WAN member passes the health-check again, it will resume forwarding logs:

```
2: date=2021-04-20 time=23:06:08 eventtime=1618985168018789600 tz="-0700"
logid="0113022923" type="event" subtype="sdwan" level="notice" vd="root" logdesc="SDWAN
status" eventtype="Service" interface="R150" member="1" serviceid=1 service="test"
gateway=10.100.1.1 msg="Member link is available. Start forwarding traffic. "
```

Load-balance mode service rule's SLA qualified member changes:

- When load-balance mode service rule's SLA qualified member changes. In this example R150 changes to not meet SLA:

```
1: date=2021-04-20 time=23:10:24 eventtime=1618985425048820800 tz="-0700"
logid="0113022923" type="event" subtype="sdwan" level="notice" vd="root" logdesc="SDWAN
status" eventtype="Service" serviceid=1 service="test" member="2(R160)" msg="Service
will be load balanced among members with available routing."
```

- When load-balance mode service rule's SLA qualified member changes. In this example R150 changes to meet SLA:

```
2: date=2021-04-20 time=23:11:34 eventtime=1618985494478807100 tz="-0700"
logid="0113022923" type="event" subtype="sdwan" level="notice" vd="root" logdesc="SDWAN
status" eventtype="Service" serviceid=1 service="test" member="2(R160),1(R150)"
msg="Service will be load balanced among members with available routing."
```

SLA link status logs, generated with interval sla-fail-log-period or sla-pass-log-period:

- When SLA fails, SLA link status logs will be generated with interval sla-fail-log-period:

```
1: date=2021-04-20 time=23:18:10 eventtime=1618985890469018260 tz="-0700"
logid="0113022925" type="event" subtype="sdwan" level="notice" vd="root" logdesc="SDWAN
SLA information" eventtype="SLA" healthcheck="test" slatargetid=1 interface="R150"
status="up" latency="0.061" jitter="0.004" packetloss="2.000%"
inbandwidthavailable="0kbps" outbandwidthavailable="200.00Mbps"
bibandwidthavailable="200.00Mbps" inbandwidthused="1kbps" outbandwidthused="1kbps"
bibandwidthused="2kbps" slamap="0x0" metric="packetloss" msg="Health Check SLA status.
SLA failed due to being over the performance metric threshold."
```

- When SLA passes, SLA link status logs will be generated with interval sla-pass-log-period:

```
2: date=2021-04-20 time=23:18:12 eventtime=1618985892509027220 tz="-0700"
logid="0113022925" type="event" subtype="sdwan" level="information" vd="root"
logdesc="SDWAN SLA information" eventtype="SLA" healthcheck="test" slatargetid=1
interface="R150" status="up" latency="0.060" jitter="0.003" packetloss="0.000%"
inbandwidthavailable="0kbps" outbandwidthavailable="200.00Mbps"
bibandwidthavailable="200.00Mbps" inbandwidthused="1kbps" outbandwidthused="1kbps"
bibandwidthused="2kbps" slamap="0x1" msg="Health Check SLA status."
```

SD-WAN related diagnose commands

This topic lists the SD-WAN related diagnose commands and related output.

To check SD-WAN health-check status:

```
FGT # diagnose sys sdwan health-check
Health Check(server):
Seq(1 R150): state(alive), packet-loss(0.000%) latency(0.110), jitter(0.024) sla_map=0x0
Seq(2 R160): state(alive), packet-loss(0.000%) latency(0.068), jitter(0.009) sla_map=0x0
```

```
FGT # diagnose sys sdwan health-check
Health Check(ping):
Seq(1 R150): state(alive), packet-loss(0.000%) latency(0.100), jitter(0.017) sla_map=0x0
Seq(2 R160): state(dead), packet-loss(100.000%) sla_map=0x0
```

```
FGT # diagnose sys sdwan health-check google
Health Check(google):
Seq(1 R150): state(alive), packet-loss(0.000%) latency(0.081), jitter(0.019) sla_map=0x0
Seq(2 R160): state(alive), packet-loss(0.000%) latency(0.060), jitter(0.004) sla_map=0x0
```

To check SD-WAN member status:

- When SD-WAN load-balance mode is *source-ip-based*/*source-dest-ip-based*.

```
FGT # diagnose sys sdwan member
Member(1): interface: R150, gateway: 10.100.1.1 2000:10:100:1::1, priority: 0 1024,
weight: 0
Member(2): interface: R160, gateway: 10.100.1.5 2000:10:100:1::5, priority: 0 1024,
weight: 0
```

- When SD-WAN load-balance mode is *weight-based*.

```
FGT # diagnose sys sdwan member
Member(1): interface: R150, gateway: 10.100.1.1 2000:10:100:1::1, priority: 0 1024,
weight: 33
  Session count: 15
Member(2): interface: R160, gateway: 10.100.1.5 2000:10:100:1::5, priority: 0 1024,
weight: 66
  Session count: 1
```

- When SD-WAN load-balance mode is *measured-volume-based*.
  - Both members are under volume and still have room:

```
FGT # diagnose sys sdwan member
Member(1): interface: R150, gateway: 10.100.1.1 2000:10:100:1::1, priority: 0 1024,
```

```
weight: 33
  Config volume ratio: 33, last reading: 218067B, volume room 33MB
Member(2): interface: R160, gateway: 10.100.1.5 2000:10:100:1::5, priority: 0 1024,
weight: 66
  Config volume ratio: 66, last reading: 202317B, volume room 66MB
```

- Some members are overloaded and some still have room:

```
FGT # diagnose sys sdwan member
Member(1): interface: R150, gateway: 10.100.1.1 2000:10:100:1::1, priority: 0 1024,
weight: 0
  Config volume ratio: 33, last reading: 1287767633B, overload volume 517MB
Member(2): interface: R160, gateway: 10.100.1.5 2000:10:100:1::5, priority: 0 1024,
weight: 63
  Config volume ratio: 66, last reading: 1686997898B, volume room 63MB
```

- When SD-WAN load balance mode is *usage-based*/*spillover*.
  - When no spillover occurs:

```
FGT # diagnose sys sdwan member
Member(1): interface: R150, gateway: 10.100.1.1 2000:10:100:1::1, priority: 0 1024,
weight: 255
  Egress-spillover-threshold: 400kbit/s, ingress-spillover-threshold: 300kbit/s
  Egress-overbps=0, ingress-overbps=0
Member(2): interface: R160, gateway: 10.100.1.5 2000:10:100:1::5, priority: 0 1024,
weight: 254
  Egress-spillover-threshold: 0kbit/s, ingress-spillover-threshold: 0kbit/s
  Egress-overbps=0, ingress-overbps=0
```

  - When member has reached limit and spillover occurs:

```
FGT # diagnose sys sdwan member
Member(1): interface: R150, gateway: 10.100.1.1 2000:10:100:1::1, priority: 0 1024,
weight: 255
  Egress-spillover-threshold: 400kbit/s, ingress-spillover-threshold: 300kbit/s
  Egress-overbps=1, ingress-overbps=0
Member(2): interface: R160, gateway: 10.100.1.5 2000:10:100:1::5, priority: 0 1024,
weight: 254
  Egress-spillover-threshold: 0kbit/s, ingress-spillover-threshold: 0kbit/s
  Egress-overbps=0, ingress-overbps=0
```

  - You can also use the `diagnose netlink dstmac list` command to check if you are over the limit.

```
FGT # diagnose netlink dstmac list R150
dev=R150 mac=00:00:00:00:00:00 vwl rx_tcp_mss=0 tx_tcp_mss=0 egress_overspill_
threshold=50000 egress_bytes=100982 egress_over_bps=1 ingress_overspill_
threshold=37500 ingress_bytes=40 ingress_over_bps=0 sampler_rate=0 vwl_zone_id=1
intf_qua=0
```

To check SD-WAN service rules status:

- *Manual mode* service rules.

```
FGT # diagnose sys sdwan service
Service(1): Address Mode(IPV4) flags=0x200
  Gen(1), TOS(0x0/0x0), Protocol(0: 1->65535), Mode(manual)
  Members(2):
    1: Seq_num(1 R150), alive, selected
    2: Seq_num(2 R160), alive, selected
```

```
  Dst address(1):
        10.100.21.0-10.100.21.255
```

- *Auto mode* service rules.

```
FGT # diagnose sys sdwan service
Service(1): Address Mode(IPV4) flags=0x200
  Gen(1), TOS(0x0/0x0), Protocol(0: 1->65535), Mode(auto), link-cost-factor(latency),
link-cost-threshold(10), heath-check(ping)
  Members(2):
    1: Seq_num(2 R160), alive, latency: 0.066, selected
    2: Seq_num(1 R150), alive, latency: 0.093
  Dst address(1):
        10.100.21.0-10.100.21.255
```

- *Priority mode* service rules.

```
FGT # diagnose sys sdwan service
Service(1): Address Mode(IPV4) flags=0x200
  Gen(1), TOS(0x0/0x0), Protocol(0: 1->65535), Mode(priority), link-cost-factor
(latency), link-cost-threshold(10), heath-check(ping)
  Members(2):
    1: Seq_num(2 R160), alive, latency: 0.059, selected
    2: Seq_num(1 R150), alive, latency: 0.077, selected
  Dst address(1):
        10.100.21.0-10.100.21.255
```

- *Load-balance* mode service rules.

```
FGT # diagnose sys sdwan service
Service(1): Address Mode(IPV4) flags=0x200
  Gen(1), TOS(0x0/0x0), Protocol(0: 1->65535), Mode(load-balance hash-mode=round-robin)
  Members(2):
    1: Seq_num(1 R150), alive, sla(0x1), gid(2), num of pass(1), selected
    2: Seq_num(2 R160), alive, sla(0x1), gid(2), num of pass(1), selected
  Dst address(1):
        10.100.21.0-10.100.21.255
```

- *SLA mode* service rules.

```
FGT # diagnose sys sdwan service
Service(1): Address Mode(IPV4) flags=0x200
  Gen(1), TOS(0x0/0x0), Protocol(0: 1->65535), Mode(sla), sla-compare-order
  Members(2):
    1: Seq_num(1 R150), alive, sla(0x1), gid(0), cfg_order(0), cost(0), selected
    2: Seq_num(2 R160), alive, sla(0x1), gid(0), cfg_order(1), cost(0), selected
  Dst address(1):
        10.100.21.0-10.100.21.255
```

To check interface logs from the past 15 minutes:

```
FGT (root) # diagnose sys sdwan intf-sla-log R150
Timestamp: Wed Apr 21 16:58:27 2021, used inbandwidth: 655bps, used outbandwidth:
81655306bps, used bibandwidth: 81655961bps, tx bys: 3413479982bytes, rx bytes: 207769bytes.
Timestamp: Wed Apr 21 16:58:37 2021, used inbandwidth: 649bps, used outbandwidth:
81655540bps, used bibandwidth: 81656189bps, tx bys: 3515590414bytes, rx bytes: 208529bytes.
Timestamp: Wed Apr 21 16:58:47 2021, used inbandwidth: 655bps, used outbandwidth:
81655546bps, used bibandwidth: 81656201bps, tx bys: 3617700886bytes, rx bytes: 209329bytes.
Timestamp: Wed Apr 21 16:58:57 2021, used inbandwidth: 620bps, used outbandwidth:
```

```
81671580bps, used bibandwidth: 81672200bps, tx bys: 3719811318bytes, rx bytes: 210089bytes.
Timestamp: Wed Apr 21 16:59:07 2021, used inbandwidth: 620bps, used outbandwidth:
81671580bps, used bibandwidth: 81672200bps, tx bys: 3821921790bytes, rx bytes: 210889bytes.
Timestamp: Wed Apr 21 16:59:17 2021, used inbandwidth: 665bps, used outbandwidth:
81688152bps, used bibandwidth: 81688817bps, tx bys: 3924030936bytes, rx bytes: 211926bytes.
Timestamp: Wed Apr 21 16:59:27 2021, used inbandwidth: 671bps, used outbandwidth:
81688159bps, used bibandwidth: 81688830bps, tx bys: 4026141408bytes, rx bytes: 212726bytes.
```

To check SLA logs in the past 10 minutes:

```
FGT (root) # diagnose sys sdwan sla-log ping 1
Timestamp: Wed Apr 21 17:10:11 2021, vdom root, health-check ping, interface: R150, status:
up, latency: 0.079, jitter: 0.023, packet loss: 0.000%.
Timestamp: Wed Apr 21 17:10:12 2021, vdom root, health-check ping, interface: R150, status:
up, latency: 0.079, jitter: 0.023, packet loss: 0.000%.
Timestamp: Wed Apr 21 17:10:12 2021, vdom root, health-check ping, interface: R150, status:
up, latency: 0.081, jitter: 0.024, packet loss: 0.000%.
Timestamp: Wed Apr 21 17:10:13 2021, vdom root, health-check ping, interface: R150, status:
up, latency: 0.081, jitter: 0.025, packet loss: 0.000%.
Timestamp: Wed Apr 21 17:10:13 2021, vdom root, health-check ping, interface: R150, status:
up, latency: 0.082, jitter: 0.026, packet loss: 0.000%.
Timestamp: Wed Apr 21 17:10:14 2021, vdom root, health-check ping, interface: R150, status:
up, latency: 0.083, jitter: 0.026, packet loss: 0.000%.
Timestamp: Wed Apr 21 17:10:14 2021, vdom root, health-check ping, interface: R150, status:
up, latency: 0.084, jitter: 0.026, packet loss: 0.000%.
```

To check Application Control used in SD-WAN and the matching IP addresses:

```
FGT # diagnose sys sdwan internet-service-app-ctrl-list
Gmail(15817 4294836957): 64.233.191.19 6 443 Thu Apr 22 10:10:34 2021
Gmail(15817 4294836957): 142.250.128.83 6 443 Thu Apr 22 10:06:47 2021
Facebook(15832 4294836806): 69.171.250.35 6 443 Thu Apr 22 10:12:00 2021
Amazon(16492 4294836342): 3.226.60.231 6 443 Thu Apr 22 10:10:57 2021
Amazon(16492 4294836342): 52.46.135.211 6 443 Thu Apr 22 10:10:58 2021
Amazon(16492 4294836342): 52.46.141.85 6 443 Thu Apr 22 10:10:58 2021
Amazon(16492 4294836342): 52.46.155.13 6 443 Thu Apr 22 10:10:58 2021
Amazon(16492 4294836342): 54.82.242.32 6 443 Thu Apr 22 10:10:59 2021
YouTube(31077 4294838537): 74.125.202.138 6 443 Thu Apr 22 10:06:51 2021
YouTube(31077 4294838537): 108.177.121.119 6 443 Thu Apr 22 10:08:24 2021
YouTube(31077 4294838537): 142.250.136.119 6 443 Thu Apr 22 10:02:02 2021
YouTube(31077 4294838537): 142.250.136.132 6 443 Thu Apr 22 10:08:16 2021
YouTube(31077 4294838537): 142.250.148.100 6 443 Thu Apr 22 10:07:28 2021
YouTube(31077 4294838537): 142.250.148.132 6 443 Thu Apr 22 10:10:32 2021
YouTube(31077 4294838537): 172.253.119.91 6 443 Thu Apr 22 10:02:01 2021
YouTube(31077 4294838537): 184.150.64.211 6 443 Thu Apr 22 10:04:36 2021
YouTube(31077 4294838537): 184.150.168.175 6 443 Thu Apr 22 10:02:26 2021
YouTube(31077 4294838537): 184.150.168.211 6 443 Thu Apr 22 10:02:26 2021
YouTube(31077 4294838537): 184.150.186.141 6 443 Thu Apr 22 10:02:26 2021
YouTube(31077 4294838537): 209.85.145.190 6 443 Thu Apr 22 10:10:36 2021
YouTube(31077 4294838537): 209.85.200.132 6 443 Thu Apr 22 10:02:03 2021
```

To check the dynamic tunnel status:

```
# diagnose sys link-monitor interface <name> <name>_0
```

For example:

```
# diagnose sys link-monitor interface vd2-2
Interface(vd2-2): state(up, since Tue Jun 15 12:31:28 2021), bandwidth(up:1299bps,
down:0bps), session count(IPv4:2, IPv6:0), tx(2409919 bytes), rx(5292290 bytes), latency
(0.03), jitter(0.00), packet-loss(0.00).

# diagnose sys link-monitor interface vd2-2 vd2-2_0
Interface(vd2-2_0): state(up, since Tue Jun 15 15:21:52 2021), bandwidth(up:640bps,
down:0bps), session count(IPv4:0, IPv6:0), tx(102242 bytes), rx(16388 bytes), latency(0.03),
jitter(0.00), packet-loss(0.00).
```

To check BGP learned routes and determine if they are used in SD-WAN service:

```
FGT # get router info bgp network 10.100.11.0/24
VRF 0 BGP routing table entry for 10.100.11.0/24
Paths: (2 available, best #2, table Default-IP-Routing-Table)
  Advertised to non peer-group peers:
   10.100.1.1
  Original VRF 0
  20 10
    10.100.1.1 from 10.100.1.1 (5.5.5.5)
      Origin incomplete metric 0, route tag 15, localpref 100, valid, external, best
      Community: 30:5
      Advertised Path ID: 2
       Last update: Thu Apr 22 10:27:27 2021

  Original VRF 0
  20 10
    10.100.1.5 from 10.100.1.5 (6.6.6.6)
      Origin incomplete metric 0, route tag 15, localpref 100, valid, external, best
      Community: 30:5
      Advertised Path ID: 1
       Last update: Thu Apr 22 10:25:50 2021

FGT # diagnose sys sdwan route-tag-list
Route-tag: 15, address: v4(1), v6(0)Last write/now: 6543391 6566007
        service(1), last read route-tag 15 at 6543420
Prefix(24): Address list(1):
        10.100.11.0-10.100.11.255 oif: 50 48

FGT # diagnose firewall proute list
list route policy info(vf=root):
id=2133196801(0x7f260001) vwl_service=1(DataCenter) vwl_mbr_seq=1 2 dscp_tag=0xff 0xff
flags=0x40 order-addr tos=0x00 tos_mask=0x00 protocol=0 sport=0-65535 iif=0 dport=1-65535
oif=48(R150) oif=50(R160)
destination(1): 10.100.11.0-10.100.11.255
source wildcard(1): 0.0.0.0/0.0.0.0
hit_count=0 last_used=2021-04-22 10:25:10
```

SD-WAN bandwidth monitoring service

The bandwidth measuring tool is used to detect true upload and download speeds. Bandwidth tests can be run on demand or automated using a script, and can be useful when configuring SD-WAN SLA and rules to balance SD-WAN traffic.

The speed test tool requires a valid SD-WAN Bandwidth Monitoring Service license.

The speed test tool is compatible with iperf3.6 with SSL support. It can test the upload bandwidth to the FortiGate Cloud speed test service. It can initiate the server connection and send download requests to the server. The tool can be run up to 10 times a day .

FortiGate downloads the speed test server list. The list expires after 24 hours. One of the speed test servers is selected, based on user input. The speed test runs, testing upload and download speeds. The test results are shown in the command terminal.

To download the speed test server list:

```
# execute speed-test-server download
Download completed.
```

To check the speed test server list:

```
# execute speed-test-server list
AWS_West valid
        Host: 34.210.67.183 5204 fortinet
        Host: 34.210.67.183 5205 fortinet
        Host: 34.210.67.183 5206 fortinet
        Host: 34.210.67.183 5207 fortinet
Google_West valid
        Host: 35.197.55.210 5204 fortinet
        Host: 35.197.55.210 5205 fortinet
        Host: 35.197.55.210 5206 fortinet
        Host: 35.197.55.210 5207 fortinet
        Host: 35.230.2.124 5204 fortinet
        Host: 35.230.2.124 5205 fortinet
        Host: 35.230.2.124 5206 fortinet
        Host: 35.230.2.124 5207 fortinet
        Host: 35.197.18.234 5204 fortinet
        Host: 35.197.18.234 5205 fortinet
        Host: 35.197.18.234 5206 fortinet
        Host: 35.197.18.234 5207 fortinet
```

To run the speed test:

You can run the speed test without specifying a server. The system will automatically choose one server from the list and run the speed test.

```
# execute speed-test auto
The license is valid to run speed test.
Speed test quota for 2/1 is 9
current vdom=root
Run in uploading mode.
Connecting to host 35.230.2.124, port 5206
[ 16] local 172.16.78.185 port 2475 connected to 35.230.2.124 port 5206
[ ID] Interval Transfer Bitrate Retr Cwnd
[ 16] 0.00-1.01 sec 11.0 MBytes 91.4 Mbits/sec 0 486 KBytes
[ 16] 1.01-2.00 sec 11.6 MBytes 98.4 Mbits/sec 0 790 KBytes
[ 16] 2.00-3.01 sec 11.0 MBytes 91.6 Mbits/sec 15 543 KBytes
[ 16] 3.01-4.01 sec 11.2 MBytes 94.2 Mbits/sec 1 421 KBytes
[ 16] 4.01-5.01 sec 11.2 MBytes 93.5 Mbits/sec 0 461 KBytes
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval Transfer Bitrate Retr
```

```
[ 16] 0.00-5.01 sec 56.1 MBytes 93.8 Mbits/sec 16 sender
[ 16] 0.00-5.06 sec 55.8 MBytes 92.6 Mbits/sec receiver

speed test Done.
Run in reverse downloading mode!
Connecting to host 35.230.2.124, port 5206
Reverse mode, remote host 35.230.2.124 is sending
[ 16] local 172.16.78.185 port 2477 connected to 35.230.2.124 port 5206
[ ID] Interval Transfer Bitrate
[ 16] 0.00-1.00 sec 10.9 MBytes 91.4 Mbits/sec
[ 16] 1.00-2.00 sec 11.2 MBytes 93.9 Mbits/sec
[ 16] 2.00-3.00 sec 11.2 MBytes 94.0 Mbits/sec
[ 16] 3.00-4.00 sec 11.2 MBytes 93.9 Mbits/sec
[ 16] 4.00-5.00 sec 10.9 MBytes 91.1 Mbits/sec
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval Transfer Bitrate Retr
[ 16] 0.00-5.03 sec 57.5 MBytes 95.9 Mbits/sec 40 sender
[ 16] 0.00-5.00 sec 55.4 MBytes 92.9 Mbits/sec receiver

speed test Done
```

To run the speed test on a server farm or data center:

```
# execute speed-test auto AWS_West
The license is valid to run speed test.
Speed test quota for 2/1 is 8
current vdom=root
Run in uploading mode.
Connecting to host 34.210.67.183, port 5205
```

To run the speed test on a local interface when there are multiple valid routes:

```
# execute speed-test port1 Google_West
The license is valid to run speed test.
Speed test quota for 2/1 is 6
bind to local ip 172.16.78.202
current vdom=root
Specified interface port1 does not comply with default outgoing interface port2 in routing
table!
Force to use the specified interface!
Run in uploading mode.
Connecting to host 35.197.18.234, port 5205
[ 11] local 172.16.78.202 port 20852 connected to 35.197.18.234 port 5205
[ ID] Interval Transfer Bitrate Retr Cwnd
[ 11] 0.00-1.01 sec 10.7 MBytes 89.0 Mbits/sec 0 392 KBytes
[ 11] 1.01-2.01 sec 10.5 MBytes 88.5 Mbits/sec 1 379 KBytes
[ 11] 2.01-3.01 sec 11.3 MBytes 94.5 Mbits/sec 0 437 KBytes
[ 11] 3.01-4.01 sec 11.2 MBytes 94.3 Mbits/sec 0 478 KBytes
[ 11] 4.01-5.00 sec 11.3 MBytes 95.2 Mbits/sec 0 503 KBytes
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval Transfer Bitrate Retr
[ 11] 0.00-5.00 sec 55.1 MBytes 92.3 Mbits/sec 1 sender
[ 11] 0.00-5.04 sec 54.5 MBytes 90.7 Mbits/sec receiver

speed test Done.
Run in reverse downloading mode!
```

```
Connecting to host 35.197.18.234, port 5205
Reverse mode, remote host 35.197.18.234 is sending
[ 11] local 172.16.78.202 port 20853 connected to 35.197.18.234 port 5205
[ ID] Interval Transfer Bitrate
[ 11] 0.00-1.00 sec 10.9 MBytes 91.1 Mbits/sec
[ 11] 1.00-2.00 sec 11.2 MBytes 94.0 Mbits/sec
[ 11] 2.00-3.00 sec 11.2 MBytes 94.0 Mbits/sec
[ 11] 3.00-4.00 sec 11.2 MBytes 94.0 Mbits/sec
[ 11] 4.00-5.00 sec 11.2 MBytes 94.0 Mbits/sec
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval Transfer Bitrate Retr
[ 11] 0.00-5.03 sec 57.4 MBytes 95.8 Mbits/sec 33 sender
[ 11] 0.00-5.00 sec 55.7 MBytes 93.4 Mbits/sec receiver

speed test Done.
```

To add a script to run a speed test automatically once every 24 hours:

```
config system auto-script
    edit "speedtest"
        set interval 86400
        set repeat 0
        set start auto
        set script "
execute speed-test-server download
execute speed-test"
    next
end
```

To view the results of the speed test script:

```
execute auto-script result speedtest
```

Using SNMP to monitor health check

You can monitor SD-WAN health check related statistics using SNMP. The MIB file can be downloaded by going to *System > SNMP* and clicking *Download FortiGate MIB File*.

The following OIDs can be monitored:

Name	OID	Description
fgVWLHealthCheckLinkNumber	.1.3.6.1.4.1.12356.101.4.9.1	The number of health check links in fgVWLHealthCheckLinkTable
fgVWLHealthCheckLinkTable	.1.3.6.1.4.1.12356.101.4.9.2	SD-WAN health check statistics table. This table has a dependent expansion relationship with fgVdTable.Only health checks with a configured member link are present in this table.

Name	OID	Description
fgVWLHealthCheckLinkTableEntry	.1.3.6.1.4.1.12356.101.4.9.2.1	SD-WAN health check statistics on a virtual domain.
fgVWLHealthCheckLinkID	.1.3.6.1.4.1.12356.101.4.9.2.1.1	SD-WAN health check link ID. Only health checks with configured member link are present in this table. Virtual-wan-link health check link IDs are only unique within a virtual domain.
fgVWLHealthCheckLinkName	.1.3.6.1.4.1.12356.101.4.9.2.1.2	Health check name.
fgVWLHealthCheckLinkSeq	.1.3.6.1.4.1.12356.101.4.9.2.1.3	SD-WAN member link sequence.
fgVWLHealthCheckLinkState	.1.3.6.1.4.1.12356.101.4.9.2.1.4	Health check state on a specific member link.
fgVWLHealthCheckLinkLatency	.1.3.6.1.4.1.12356.101.4.9.2.1.5	The average latency of a health check on a specific member link within last 30 probes, in float number.
fgVWLHealthCheckLinkJitter	.1.3.6.1.4.1.12356.101.4.9.2.1.6	The average jitter of a health check on a specific member link within last 30 probes, in float number.
fgVWLHealthCheckLinkPacketSend	.1.3.6.1.4.1.12356.101.4.9.2.1.7	The total number of packets sent by a health check on a specific member link.
fgVWLHealthCheckLinkPacketRecv	.1.3.6.1.4.1.12356.101.4.9.2.1.8	The total number of packets received by a health check on a specific member link.
fgVWLHealthCheckLinkPacketLoss	.1.3.6.1.4.1.12356.101.4.9.2.1.9	The packet loss percentage of a health check on a specific member link within last 30 probes, in float number.
fgVWLHealthCheckLinkVdom	.1.3.6.1.4.1.12356.101.4.9.2.1.10	The VDOM that the link monitor entry exists in. This name corresponds to the fgVdEntName used in fgVdTable.
fgVWLHealthCheckLinkBandwidthIn	.1.3.6.1.4.1.12356.101.4.9.2.1.11	The available bandwidth of incoming traffic detected by a health check on a specific member link, in Mbps,

Name	OID	Description
fgVWLHealthCheckLinkBandwidthOut	.1.3.6.1.4.1.12356.101.4.9.2.1.12	The available bandwidth of outgoing traffic detected by a health check on a specific member link, in Mbps.
fgVWLHealthCheckLinkBandwidthBi	.1.3.6.1.4.1.12356.101.4.9.2.1.13	The available bandwidth of bi-direction traffic detected by a health check on a specific member link, in Mbps.
fgVWLHealthCheckLinkIfName	.1.3.6.1.4.1.12356.101.4.9.2.1.14	SD-WAN member interface name.

Example

This example shows a SD-WAN health check configuration and its collected statistics.

To configure the SD-WAN health check:

```
config system sdwan
    set status enable
    config zone
        edit "virtual-wan-link"
        next
    end
    config members
        edit 1
            set interface "port1"
            set gateway 192.168.2.1
        next
        edit 2
            set interface "MPLS"
            set zone "SD-Zone2"
            set cost 20
        next
        edit 3
            set interface "port2"
        next
    end
    config health-check
        edit "pingserver"
            set server "8.8.8.8"
            set sla-fail-log-period 10
            set sla-pass-log-period 20
            set members 2 1 3
            config sla
                edit 1
                    set link-cost-factor jitter packet-loss
                    set packetloss-threshold 2
                next
            end
        next
```

```
    end
end
```

The collected statistics:

fgVWLHealthCheckLinkID	.1.3.6.1.4.1.12356.101.4.9.2.1.1	1	2	3
fgVWLHealthCheckLinkName	.1.3.6.1.4.1.12356.101.4.9.2.1.2	pingserver	pingserver	pingserver
fgVWLHealthCheckLinkSeq	.1.3.6.1.4.1.12356.101.4.9.2.1.3	2	1	3
fgVWLHealthCheckLinkState	.1.3.6.1.4.1.12356.101.4.9.2.1.4	0	0	0
fgVWLHealthCheckLinkLatency	.1.3.6.1.4.1.12356.101.4.9.2.1.5	39.302	43.124	44.348
fgVWLHealthCheckLinkJitter	.1.3.6.1.4.1.12356.101.4.9.2.1.6	4.346	3.951	5.05
fgVWLHealthCheckLinkPacketSend	.1.3.6.1.4.1.12356.101.4.9.2.1.7	3657689	3657689	3657689
fgVWLHealthCheckLinkPacketRecv	.1.3.6.1.4.1.12356.101.4.9.2.1.8	3196258	3220258	3219466
fgVWLHealthCheckLinkPacketLoss	.1.3.6.1.4.1.12356.101.4.9.2.1.9	0	0	0
fgVWLHealthCheckLinkVdom	.1.3.6.1.4.1.12356.101.4.9.2.1.10	root	root	root
fgVWLHealthCheckLinkBandwidthIn	.1.3.6.1.4.1.12356.101.4.9.2.1.11	9999963	9999937	9999999
fgVWLHealthCheckLinkBandwidthOut	.1.3.6.1.4.1.12356.101.4.9.2.1.12	9999981	9999953	9999998
fgVWLHealthCheckLinkBandwidthBi	.1.3.6.1.4.1.12356.101.4.9.2.1.13	19999944	19999890	19999997
fgVWLHealthCheckLinkIfName	.1.3.6.1.4.1.12356.101.4.9.2.1.14	MPLS	port1	port2

Policy and Objects

This section contains topics on configuring policies and traffic shaping:

Policies

The firewall policy is the axis around which most features of the FortiGate revolve. Many firewall settings end up relating to or being associated with the firewall policies and the traffic they govern. Any traffic going through a FortiGate has to be associated with a policy. These policies are essentially discrete compartmentalized sets of instructions that control the traffic flow going through the firewall. These instructions control where the traffic goes, how it is processed, if it is processed, and whether or not it is allowed to pass through the FortiGate.

When the firewall receives a connection packet, it analyzes the source address, destination address, and service (by port number). It also registers the incoming interface, the outgoing interface it needs to use, and the time of day. Using this information, the FortiGate firewall attempts to locate a security policy that matches the packet. If a policy matches the parameters, then the FortiGate takes the required action for that policy. If it is *Accept*, the traffic is allowed to proceed to the next step. If the action is *Deny* or a match cannot be found, the traffic is not allowed to proceed.

The two basic actions at the initial connection are either *Accept* or *Deny*:

- If the action is *Accept*, the policy permits communication sessions. There may be other packet processing instructions, such as requiring authentication to use the policy or restrictions on the source and destination of the traffic.
- If the action is *Deny*, the policy blocks communication sessions, and you can optionally log the denied traffic. If no security policy matches the traffic, the packets are dropped. A *Deny* security policy is needed when it is required to log the denied traffic, also called *violation traffic*.

One other action can be associated with the policy:

- *IPsec*: this is an *Accept* action that is specifically for IPsec VPNs.

Each field in a firewall policy that accepts multiple inputs, such as srcaddr and dstaddr, can accept as many inputs as there are unique objects created. The maximum number of objects depends on the model. See the Maximum Values Table for more details.

The following topics provide information on the available types of policies and configuration instructions:

The following topics provide instructions on configuring policies:

Firewall policy

The firewall policy is the axis around which most of the other features of the FortiGate firewall revolve. A large portion of the settings in the firewall at some point will end up relating to or being associated with the firewall policies and the traffic that they govern. Any traffic going through a FortiGate unit has to be associated with a policy. These policies are essentially discrete compartmentalized sets of instructions that control the traffic flow going through the firewall. These instructions control where the traffic goes, how it's processed, if it's processed, and even whether or not it's allowed to pass through the FortiGate.

The following topics provide information on the firewall policy and configuration:

Firewall policy parameters

For traffic to flow through the FortiGate firewall, there must be a policy that matches its parameters:

- Incoming interface(s)
- Outgoing interface(s)
- Source address(es)
- User(s) identity
- Destination address(es)
- Internet service(s)
- Schedule
- Service

Traffic parameters are checked against the configured policies for a match. If the parameters do not match any configured policies, the traffic is denied.

Traffic flow initiated from each direction requires a policy, that is, if sessions can be initiated from both directions, each direction requires a policy.

Just because packets can go from point A to point B on port X does not mean that the traffic can flow from point B to point A on port X. A policy must be configured for each direction.

When designing a policy, there is often reference to the traffic flow, but most communication is two-way so trying to determine the direction of the flow might be confusing. If traffic is HTTP web traffic, the user sends a request to the website, but most of the traffic flow will be coming from the website to the user or in both directions? For the purposes of determining the direction for a policy, the important factor is the direction of the initiating communication. The user is sending a request to the website, so this is the initial communication; the website is responding so the traffic is from the user's network to the Internet.

FortiOS does not perform a reverse-path check on reply traffic that matches an allowed session based on the IP tuple. The request traffic can be sent on one interface and the reply traffic could return on another interface.

Configurations in the GUI

Firewall policies can be created in the GUI by configuring the necessary parameters.

Incoming interface(s)	This is the interface or interfaces by which the traffic is first connected to the FortiGate unit. The exception being traffic that the FortiGate generates itself. This is not limited to the physical Ethernet ports found on the device. The incoming interface can also be a logical or virtual interface such as a VPN tunnel, a Virtual WAN link, or a wireless interface.
Outgoing interface(s)	This is the interface or interfaces used by traffic leaving a port once it has been processed by the firewall. Similar to incoming interfaces, it is not limited to only physical interfaces.
Source address(es)	The addresses that a policy can receive traffic from can be wide open or tightly controlled. For a public web server that the world at large should be able to access, the best choice will be *all*. If the destination is a private web server that only the branch offices of a company should be able to access, or a list of internal computers that are the only ones allowed to access an external resource, then a group of preconfigured addresses is the better strategy
User(s) identity	This parameter is based on a user identity that can be from a number of authentication authorities. It will be an account or group that has been set up in advance that can be selected from the drop down menu. The exception to this is the feature that allows the importing of LDAP Users. When the feature is used, a small wizard window will appear to guide the user through the setup. The caveat is that the LDAP server object in the *User & Authentication > LDAP Servers* section has to be already configured to allow the use of this import feature.
Destination address(es)	In the same way that the source address may need to be limited, the destination address can be used as a traffic filter. When the traffic is destined for internal resources, the specific address of the resource can be defined to better protect the other resources on the network. One of the specialized destination address options is to use a Virtual IP address. If the destination address doesn’t need to be internal, you can define policies that are only for connecting to specific addresses on the Internet.
Internet service(s)	In this context, an Internet service is a combination of one or more addresses and one or more services associated with a service found on the Internet such as an update service for software.

Schedule	The time frame that is applied to the policy. This can be something as simple as a time range that the sessions are allowed to start, such as between 8:00 am and 5:00 pm. Something more complex like business hours that include a break for lunch and time of the session's initiation may need a schedule group because it will require multiple time ranges to make up the schedule.
Service	The services chosen represent the TCP/IP suite port numbers that will most commonly be used to transport the named protocols or groups of protocols. This is different than *Application Control* which looks more closely at the packets to determine the actual protocol used to create them. A case where either side can initiate the communication, like between two internal interfaces on the FortiGate unit, would be a more likely situation to require a policy for each direction.

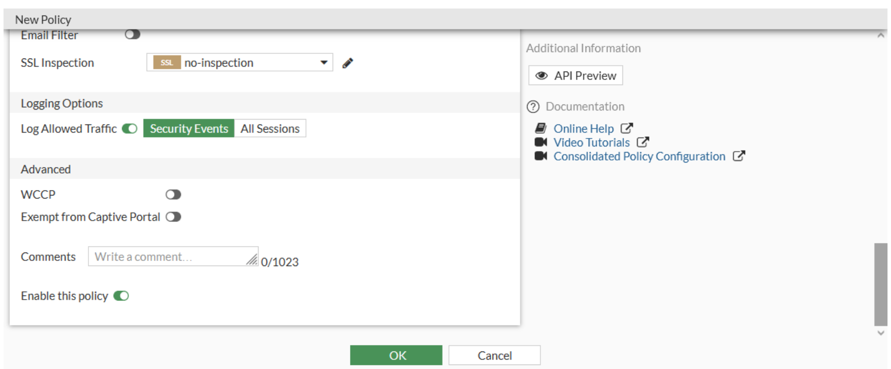
Enabling advanced policy options in the GUI

Advanced policy options can be enabled so that you can configure the options in the GUI.

To enable advanced policy options:

```
config system settings
    set gui-advanced-policy enable
end
```

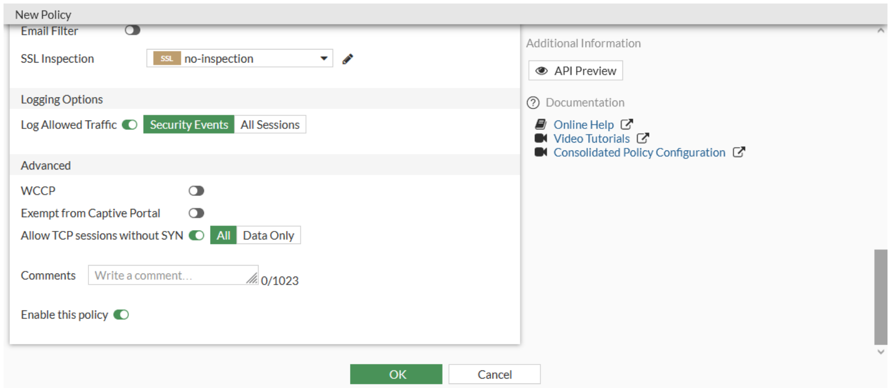
Advanced policy options are now available when creating or editing a policy in the GUI:



To enable configuring TCP sessions without SYN:

```
config system settings
    set tcp-session-without-syn enable
end
```

TCP sessions without SYN can now be configured when creating or editing a policy in the GUI:



Add Policy change summary and Policy expiration to Workflow Management

Two options, *Policy change summary* and *Policy expiration*, are included in *Workflow Management*. *Policy change summary* enforces an audit trail for changes to firewall policies. *Policy expiration* allows administrators to set a date for the firewall policy to be disabled.

There are three states for the *Policy change summary*:

- *Disable*: users will not be prompted to add a summary when editing a policy.
- *Required*: the *Policy change summary* will be enabled and will require users to add a summary when editing or creating a firewall policy.
- *Optional*: the *Policy change summary* will be enabled but users can leave the summary empty, if preferred, when editing or creating a firewall policy.

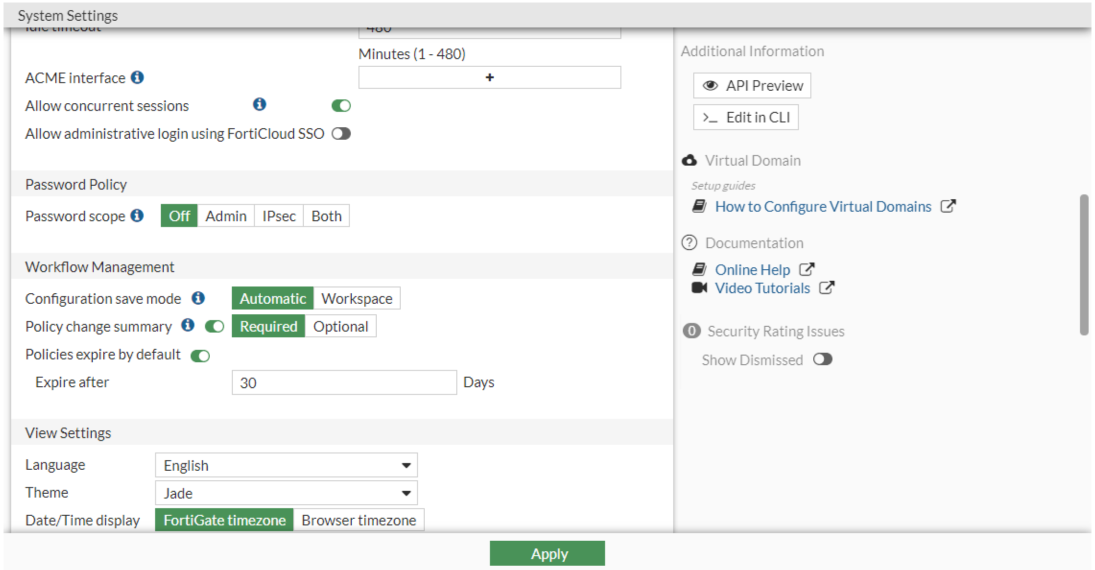
There are three states for *Policy expiration*:

- *Disable*: the firewall policy will not expire. This is the default setting for *Policy expiration*.
- *Default*: the firewall policy will expire after the default number of days.
- *Specify*: the firewall policy will expire at a set date and time.

The default value for *Policy expiration* is 30 days. This number can be changed in the CLI or in *System > Settings* in the GUI to any value between zero and 365 days. If the default value is set to zero, the *Default* state will disable the *Policy expiration*.

To configure the firewall policy change summary and default expiration in the GUI:

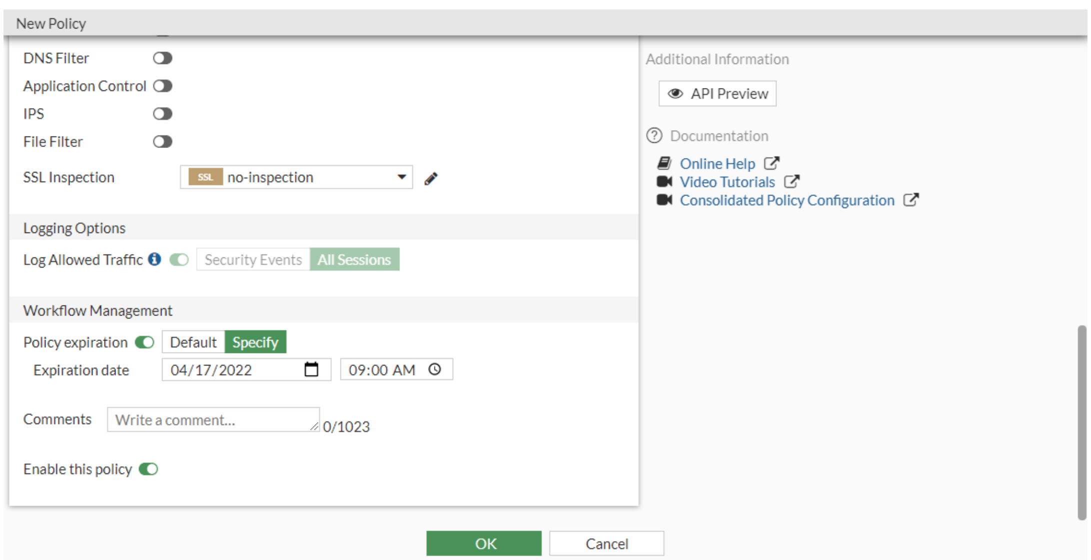
1. Go to *System > Feature Visibility*.
2. Enable *Workflow Management*.
3. Click *Apply*.
4. Go to *System > Settings*.
5. In the *Workflow Management* section, set *Policy change summary* to *Required*. *Policies expire by default* is enabled by default with an *Expire after* value of *30*.



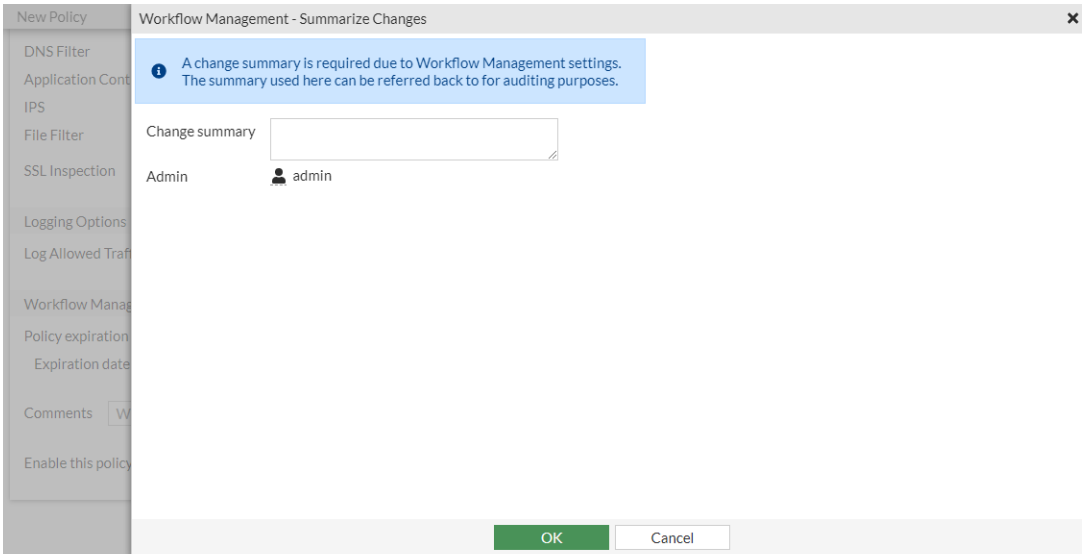
6. Click *Apply*.

To configure firewall policy expiration in the GUI:

1. Go to *Policy & Objects > Firewall Policy* and click *Create New*.
2. Name the policy and configure the necessary parameters.
3. Set *Policy expiration* to *Specify*. The *Expiration date* fields appears with the current date and time.



4. Select the date and time for the policy to expire from the *Expiration date* fields.
5. Click *OK*. The *Workflow Management - Summarize Changes* pane opens.



6. In the *Change summary* field, enter details about the changes made to the policy. These details can be referred to later for auditing purposes.
7. Click *OK*.

To configure the firewall policy change summary in the CLI:

```
config system settings
    set gui-enforce-change-summary {disable | require | optional}
end
```

To configure the policy expiration default value in the CLI:

```
config system settings
    set default-policy-expiry-days <integer>
end
```

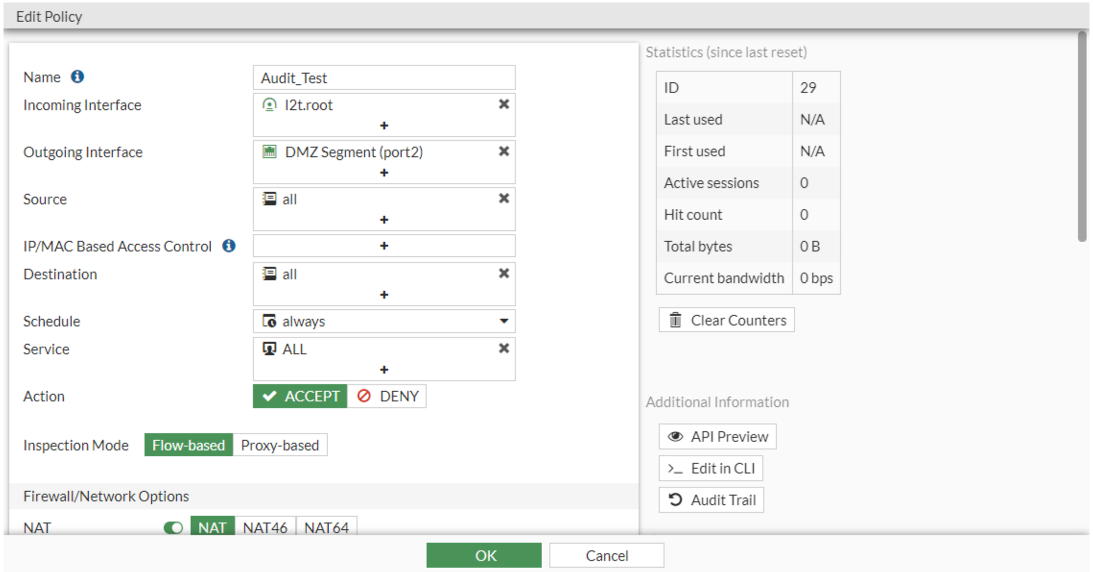
To configure firewall policy expiration in the CLI:

```
config firewall policy
    edit <id>
        set policy-expiry {enable | disable}
        set policy-expiry-date <YYYY-MM-DD HH:MM:SS>
    next
end
```

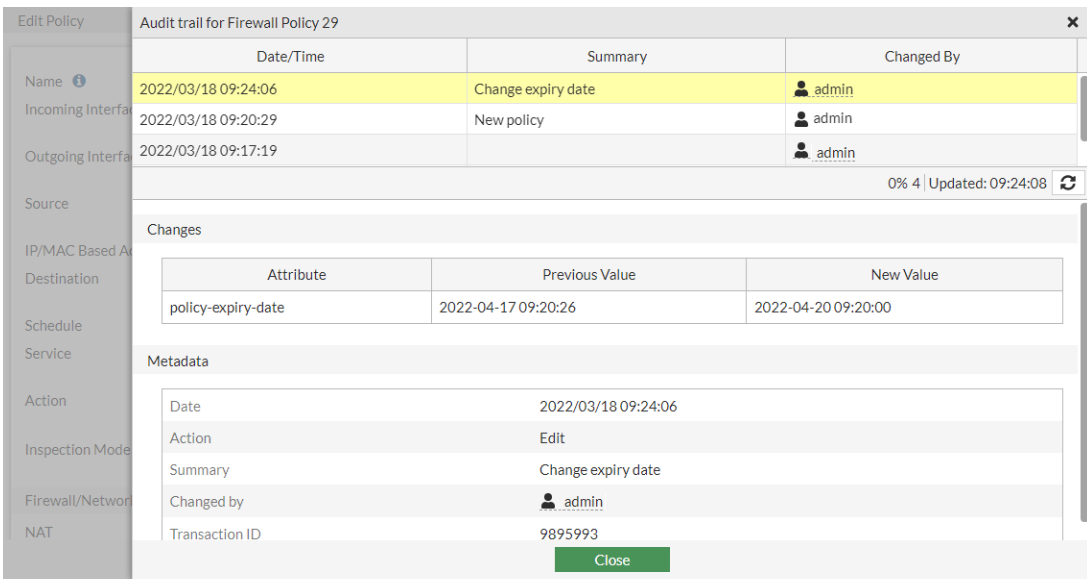
Policy change summaries are used to track changes made to a firewall policy. The *Audit trail* allow users to review the policy change summaries, including the date and time of the change and which user made the change.

To review the audit trail in the GUI:

1. Go to *Policy & Objects > Firewall Policy*.
2. Select the policy you want to review and click *Edit*.



3. In the right-side banner, click *Audit Trail*. The *Audit trail for Firewall Policy* pane opens and displays the policy change summaries for the selected policy.



4. Select an entry to review the details of the change made.
5. When you are done reviewing the *Audit Trail*, click *Close*.
6. Click *Cancel* to exit the *Edit Policy* page.

Configurations in the CLI

Firewall policies can be created in the CLI by configuring the necessary parameters. See Configurations in the GUI on page 760 for more information on the various parameters.

Parameter	Definition
srcintf	Incoming (ingress) interface.
dstintf	Outgoing (egress) interface.
srcaddr	Source IPv4 address and address group names.
dstaddr	Destination IPv4 address and address group names.
internet-service	Enable/disable use of Internet Services for this policy. If enabled, destination address and service are not used.
schedule	Schedule name.
service	Service and service group names.
anti-replay	Enable/disable checking of TCP flags per policy.
match-vip	Enable/disable matching of VIPs when used in a policy with a deny action.
auto-asic-offload	Enable/disable hardware acceleration. Available on select FortiGate models with Secure Processing Unit (SPU) hardware only.
tcp-mss-sender	Sender TCP maximum segment size (MSS).
tcp-mss-receiver	Receiver TCP maximum segment size (MSS).
session-ttl	Time-to-live (TTL) in seconds for session accepted by this policy.

Firewall anti-replay option per policy

When the global anti-replay option is disabled, the FortiGate does not check TCP flags in packets. The per policy anti-replay option overrides the global setting. This allows you to control whether or not TCP flags are checked per policy.

To enable the anti-replay option so TCP flags are checked using the CLI:

```
config firewall policy
    edit 1
        set name "policyid-1"
        set srcintf "wan2"
        set dstintf "wan1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set anti-replay enable
        set logtraffic all
        set nat enable
    next
end
```

Deny matching with a policy with a virtual IP applied

Preventing hosts with specific source addresses from accessing a server behind the FortiGate may be required in some cases. For this scenario, you should have previously configured a firewall policy with a virtual IP (VIP) object applied to it to allow such access. See Destination NAT on page 815 for details.

When denying traffic destined for a typical firewall policy without a VIP applied, you would simply configure a new firewall policy with an action of deny and with specific source addresses above the firewall policy that you want to prevent these hosts from accessing. However, the FortiGate matches firewall policies with VIPs applied differently than typical firewall policies. Policies with VIPs applied have priority over typical firewall policies.

Therefore, to block specific source traffic destined for a firewall policy specified with an action of accept and with a VIP applied, you should configure set match-vip enable on the firewall policy with a deny action that has been configured to match traffic before the firewall policy with the VIP applied.

If the policy action is set to accept, match-vip cannot be enabled.

To block VIP traffic in a deny policy:

```
config firewall policy
    edit 1
        set name "deny-policy-1"
        set srcintf "wan1"
        set dstintf "lan1"
        set srcaddr "src-hosts-to-deny-access"
        set dstaddr "all"
        set action "deny"
        set schedule "always"
        set service "all"
        set match-vip enable
    next
    edit 2
        set name "vip-policy-1"
        set srcintf "wan1"
        set dstintf "lan1"
        set srcaddr "all"
        set dstaddr "vip-object-1"
        set action "accept"
        set schedule "always"
        set service "ALL"
    next
end
```

Alternatively, to block access to a firewall policy with a VIP applied, you can configure a new VIP object configured with set src-filter <range>. Configure a new firewall policy with a deny action and with this new VIP applied, and then configure this policy to match traffic before the firewall policy with the same VIP applied with an action of accept. In this case, the firewall policy can simply have set match-vip disable, which is the default setting in a new firewall policy.

To specify a VIP with source addresses specified with a deny policy:

```
config firewall vip
    edit “vip-with-srcaddr-to-deny”
        set extip "10.1.100.199"
        set extintf "wan1"
        set mappedip "172.16.200.55"
```

```
        set src-filter "1.1.1.1/24"
    next
end
config firewall policy
    edit 3
        set name "deny-policy-3"
        set srcintf "wan1"
        set dstintf "lan1"
        set srcaddr "all"
        set dstaddr "vip-with-srcaddr-to-deny"
        set action "deny"
        set match-vip disable
        set schedule "always"
        set service "ALL"
    next
    edit 2
        set name "vip-policy-1"
        set srcintf "wan1"
        set dstintf "lan1"
        set srcaddr "all"
        set dstaddr "vip-object-1"
        set action "accept"
        set match-vip disable
        set schedule "always"
        set service "ALL"
    next
end
```

Hardware acceleration

Hardware acceleration is supported on select FortiGate devices and is enabled by default on all firewall policies to ensure optimal performance when processing network traffic traversing the FortiGate. See the Hardware Acceleration Reference Manual for details.

Typically, hardware acceleration on a specific firewall policy is disabled for one of two purposes:

- To allow CLI commands such as the packet sniffer and debug flow to display all traffic matching the policy since traffic offloaded by SPU hardware on a FortiGate device is not visible by those CLI tools.
- To troubleshoot any possible issues arising by using hardware acceleration.

To disable hardware acceleration in an IPv4 firewall policy:

```
config firewall policy
    edit 1
        set auto-asic-offload disable
    next
end
```

To disable hardware acceleration in an IPv6 firewall policy:

```
config firewall policy6
    edit 1
        set auto-asic-offload disable
    next
end
```

To disable hardware acceleration in a multicast firewall policy:

```
config firewall multicast-policy
    edit 1
        set auto-asic-offload disable
    next
end
```

TCP Maximum Segment Size (MSS)

The TCP maximum segment size (MSS) is the maximum amount of data that can be sent in a TCP segment. The MSS is the MTU size of the interface minus the 20 byte IP header and 20 byte TCP header. By reducing the TCP MSS, you can effectively reduce the MTU size of the packet.

The TCP MSS can be configured in a firewall policy, or directly on an interface. See Interface MTU packet size on page 150 for details on configuring TCP MSS directly on an interface.

To configure TCP MSS in a firewall policy:

```
config firewall policy
    edit <policy ID>
        set srcintf "internal"
        set dstintf "wan1"
        set srcaddr "10.10.10.6"
        set dstaddr "all"
        set schedule "always"
        set service "ALL"
        set tcp-mss-sender 1448
        set tcp-mss-receiver 1448
    next
end
```

Adjusting session time-to-live (TTL)

A session is a communication channel between two devices or applications across the network. Sessions allow FortiOS to inspect and act on a sequential group of packets in a session all at once instead of inspecting each packet individually. Each session has an entry in the session table that includes important information about the session.

The session time-to-live (TTL) parameter determines how long a session of a particular protocol such as TCP, UDP, or ICMP remains in the session table. To ensure proper operation of some devices or applications, the session TTL parameter may need to be increased or decreased to allow sessions to remain active in the session table for a longer or shorter duration, respectively.

To configure a modified session TTL in a firewall policy:

```
config firewall policy
    edit <policy ID>
        set srcintf "internal"
        set dstintf "wan1"
        set srcaddr "10.10.10.6"
        set dstaddr "all"
        set schedule "always"
        set service "ALL"
        set session-ttl 1800
```

```
    next
end
```

The session TTL can be set to zero or `never` to ensure a session never times out. See No session timeout on page 859 for details.

Session TTL should only be set to zero or `never` after careful consideration of:

- The connected device's or application's requirements for sessions to always stay alive
- The expectation that a connected device or application will use the same session determined by traffic using a fixed source port, fixed destination port, fixed source IP address, and fixed destination IP address.

When session TTL is set to zero or `never`, then sessions will not be cleared from the session table or expire after a specified time unless the CLI commands `diagnose system session filter <filter>` and `diagnose system session clear` are used.

If this setting is used in the case when traffic through a firewall policy can generate numerous unique sessions, then this may have unintended consequences to the FortiGate's memory usage and performance due to the session table constantly growing and not clearing out idle sessions.

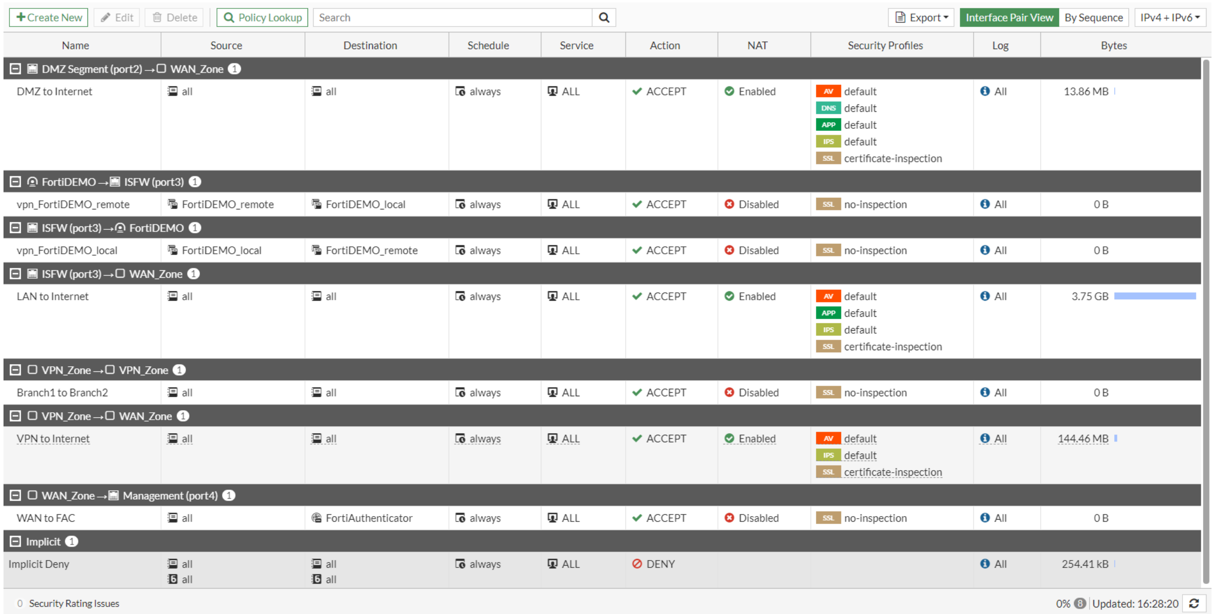
To disable session TTL in a firewall policy:

```
config firewall policy
    edit <policy ID>
        set srcintf "internal"
        set dstintf "wan1"
        set srcaddr "10.10.10.6"
        set dstaddr "all"
        set schedule "always"
        set service "ALL"
        set session-ttl never
    next
end
```

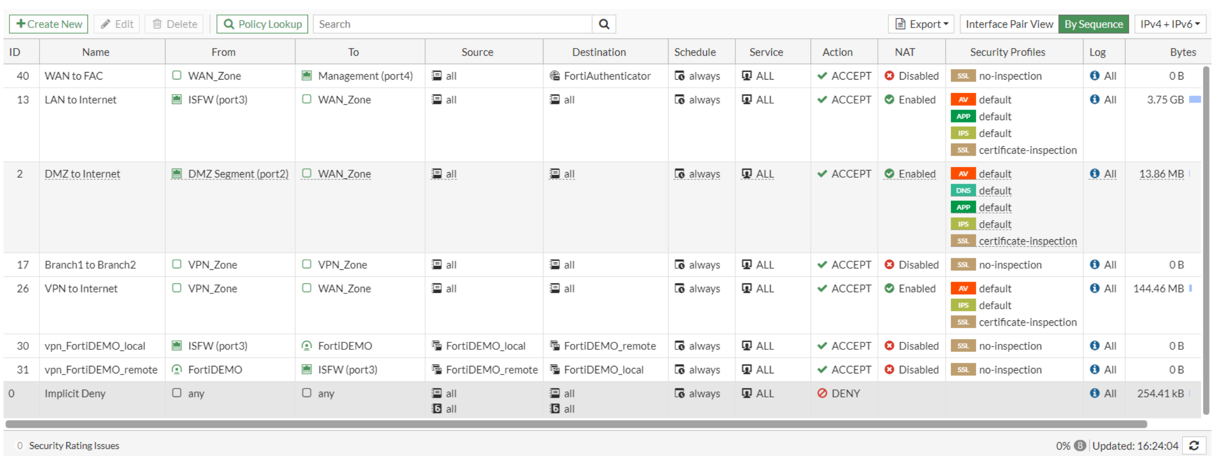
Policy views

In *Policy & Objects* policy list pages, there are two policy views: *Interface Pair View* and *By Sequence* view.

Interface Pair View displays the policies in the order that they are checked for matching traffic, grouped by the pairs of incoming and outgoing interfaces in collapsible sections.

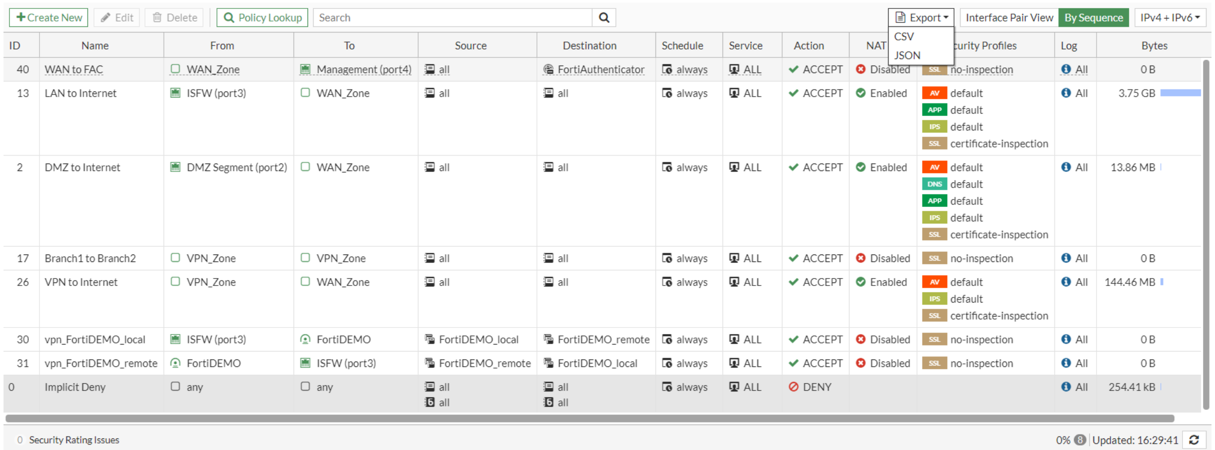


By Sequence displays policies in the order that they are checked for matching traffic without any grouping.



The default display is *Interface Pair View*. You can switch between the two views except if *any* or multiple interfaces are applied in the policy. The FortiGate automatically changes the view on the policy list page to *By Sequence* whenever there is a policy containing *any* or multiple interfaces as the *Source* or *Destination* interface. If the *Interface Pair View* is grayed out, it is likely that one or more policies have used the *any* or multiple interfaces.

You can export the current view to CSV and JSON formats by clicking *Export* and selecting *CSV* or *JSON*. The file is automatically downloaded.



Policy lookup

Firewall policy lookup is based on the `Source_interfaces/Protocol/Source_Address/Destination_Address` that matches the `source-port` and `dst-port` of the protocol. Use this tool to find out which policy matches specific traffic from a number of policies. After completing the lookup, the matching firewall policy is highlighted on the policy list page.

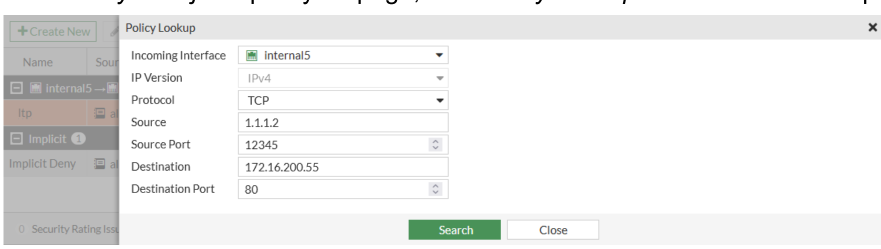
The *Policy Lookup* tool has the following requirements:

- Transparent mode does not support policy lookup function.
- When executing the policy lookup, you need to confirm whether the relevant route required for the policy work already exists.

Sample configuration

This example uses the TCP protocol to show how policy lookup works:

1. On a *Policy & Objects* policy list page, click *Policy Lookup* and enter the traffic parameters.



2. Click *Search* to display the policy lookup results.

NGFW policy

Profile-based next-generation firewall (NGFW) mode is the traditional mode where you create a profile (antivirus, web filter, and so on) and then apply the profile to a policy.

In policy-based NGFW mode, you allow applications and URL categories to be used directly in security policies, without requiring web filter or application control profiles. However, it is possible to select and apply web filter URL categories and groups.

In policy-based mode:

- Central NAT is always enabled. If no Central SNAT policy exists, you must create one. See Central SNAT on page 806 for more information.
- Pre-match rules are defined separately from security policies, and define broader rules, such as SSL inspection and user authentication.
- The IPsec wizard is not supported.

If your FortiGate operates in NAT mode, rather than enabling source NAT in individual NGFW policies, go to *Policy & Objects > Central SNAT* and add source NAT policies that apply to all matching traffic. In many cases, you may only need one SNAT policy for each interface pair.

The NGFW mode is set per VDOM, and it is only available when the VDOM inspection mode is flow-based. You can operate your entire FortiGate or individual VDOMs in NGFW policy mode. The application default port can be set as a service port in the NGFW mode using the `default-app-port-as-service` option.

In NGFW mode, administrators can configure a security policy in learn mode to monitor traffic. See Learn mode in security policies in NGFW mode on page 784 for more information.

Switching from profile-based to policy-based mode converts your policies to policy-based. To avoid issues, you could create a new VDOM for the policy-based mode. We recommend backing up your configuration before switching modes. See Configuration backups on page 65 for information.

Enabling policy-based NGFW mode

To enable policy-based NGFW mode without VDOMs in the GUI:

1. Go to *System > Settings*.
2. In *NGFW Mode*, select *Policy-based*.
3. Click *Apply*.

To enable policy-based NGFW mode with VDOMs in the GUI:

1. Go to *System > VDOM* .
2. Double-click a VDOM to edit the settings.
3. In *NGFW Mode*, select *Policy-based*.
4. Click *OK*.

To enable policy-based NGFW mode without VDOMs in the CLI:

```
config system settings
    set ngfw-mode policy-based
end
```

To enable policy-based NGFW mode with VDOMs in the CLI:

```
config vdom
    edit <vdom>
        config system settings
            set ngfw-mode policy-based
```

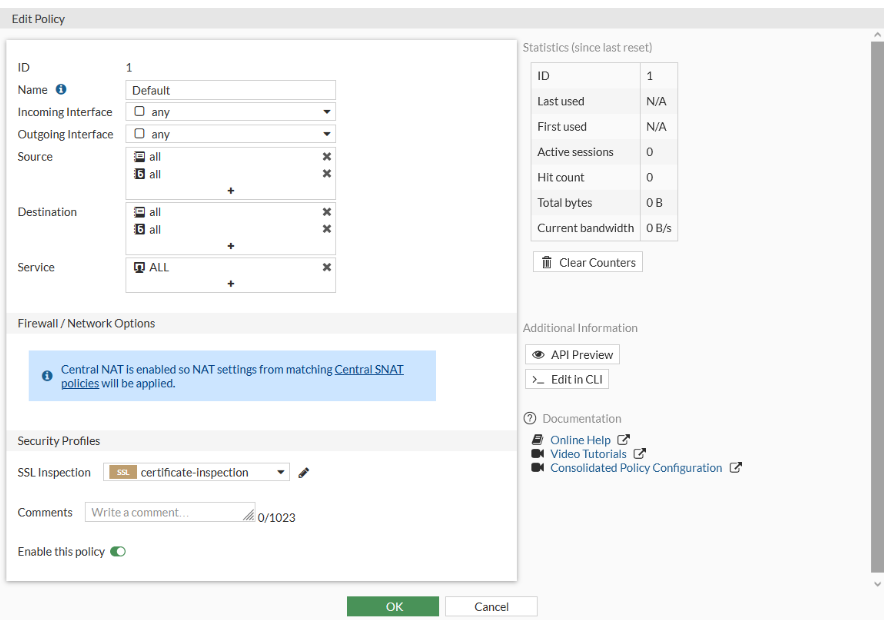
```
        end
    next
end
```

Security and SSL Inspection & Authentication policies

Security policies work with SSL Inspection & Authentication policies to inspect traffic. To allow traffic from a specific user or user group, both Security and SSL Inspection & Authentication policies must be configured. A default SSL Inspection & Authentication policy with the certificate-inspection SSL Inspection profile is preconfigured. Traffic will match the SSL Inspection & Authentication policy first. If the traffic is allowed, packets are sent to the IPS engine for application, URL category, user, and user group match, and then, if enabled, UTM inspection (antivirus, IPS, DLP, and email filter) is performed.

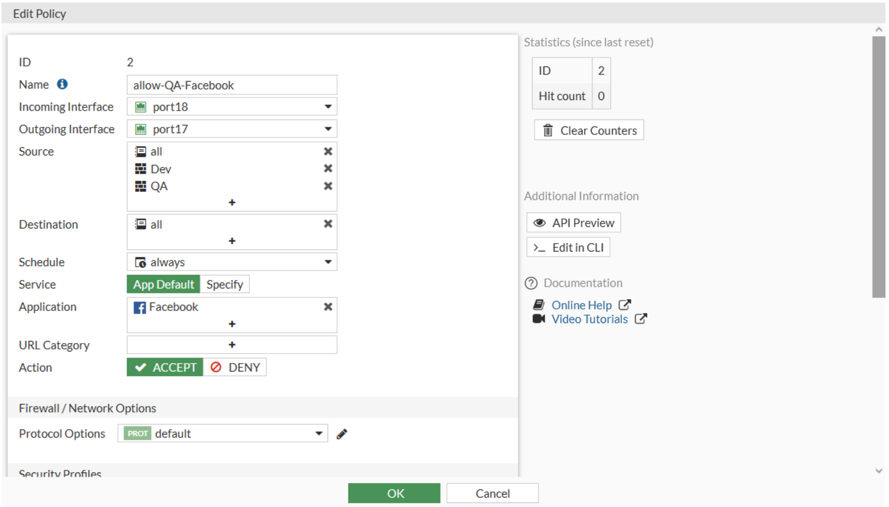
SSL Inspection & Authentication policies are used to pre-match traffic before sending the packets to the IPS engine:

- There are no schedule or action options; traffic matching the policy is always redirected to the IPS engine.
- SSL inspection, formerly configured in the VDOM settings, is configured in an SSL Inspection & Authentication policy.
- Users and user groups that require authentication must be configured in an SSL Inspection & Authentication policy.



Security policies work with SSL Inspection & Authentication policies to inspect traffic:

- Applications and URL categories can be configured directly in the policy.
- Users and user groups that require authentication must also be configured in a security policy.
- The available actions are *Accept* or *Deny*.
- The *Service* option can be used to enforce the standard port for the selected applications.
- UTM inspection is configured in a security policy.



To configure policies for Facebook and Gmail access in the CLI:

1. Configure an SSL Inspection & Authentication policy:

```
config firewall policy
    edit 1
        set name "Policy-1"
        set srcintf "port18"
        set dstintf "port17"
        set srcaddr "all"
        set dstaddr "all"
        set service "ALL"
        set ssl-ssh-profile "new-deep-inspection"
        set groups "Dev" "HR" "QA" "SYS"
    next
end
```

2. Configure security policies:

```
config firewall security-policy
    edit 2
        set name "allow-QA-Facebook"
        set srcintf "port18"
        set dstintf "port17"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set application 15832
        set groups "Dev" "QA"
    next
    edit 4
        set name "allow-QA-Email"
        set srcintf "port18"
        set dstintf "port17"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
```

```
        set url-category 23
        set groups "QA"
    next
end
```

Logs

In the application control and web filter logs, `securityid` maps to the security policy ID.

Application control log:

```
date=2019-06-17 time=16:35:47 logid="1059028704" type="utm" subtype="app-ctrl"
eventtype="signature" level="information" vd="vd1" eventtime=1560814547702405829 tz="-0700"
appid=15832 user="Jack" group="QA" srcip=10.1.100.102 dstip=157.240.3.29 srcport=56572
dstport=443 srcintf="port18" srcintfrole="undefined" dstintf="port17"
dstintfrole="undefined" proto=6 service="P2P" direction="incoming" policyid=1
sessionid=42445 appcat="Social.Media" app="Facebook" action="pass" hostname="external-sea1-
1.xx.fbcdn.net" incidentserialno=1419629662 url="/" securityid=2 msg="Social.Media:
Facebook," apprisk="medium" scertcname="*.facebook.com" scertissuer="DigiCert SHA2 High
Assurance Server CA"
```

Web filter log:

```
date=2019-06-17 time=16:42:41 logid="0317013312" type="utm" subtype="webfilter"
eventtype="ftgd_allow" level="notice" vd="vd1" eventtime=1560814961418114836 tz="-0700"
policyid=4 sessionid=43201 user="Jack" group="QA" srcip=10.1.100.102 srcport=56668
srcintf="port18" srcintfrole="undefined" dstip=172.217.3.165 dstport=443 dstintf="port17"
dstintfrole="undefined" proto=6 service="HTTPS" hostname="mail.google.com"
action="passthrough" reqtype="direct" url="/" sentbyte=709 rcvdbyte=0 direction="outgoing"
msg="URL belongs to an allowed category in policy" method="domain" cat=23 catdesc="Web-based
Email" securityid=4
```

Traffic logs:

```
date=2019-06-17 time=16:35:53 logid="0000000013" type="traffic" subtype="forward"
level="notice" vd="vd1" eventtime=1560814553778525154 tz="-0700" srcip=10.1.100.102
srcport=56572 srcintf="port18" srcintfrole="undefined" dstip=157.240.3.29 dstport=443
dstintf="port17" dstintfrole="undefined" poluuid="b740d418-8ed3-51e9-5a7b-114e99ab6370"
sessionid=42445 proto=6 action="server-rst" user="Jack" group="QA" policyid=1
policytype="consolidated" centralnatid=1 service="HTTPS" dstcountry="United States"
srccountry="Reserved" trandisp="snat" transip=172.16.200.2 transport=56572 duration=6
sentbyte=276 rcvdbyte=745 sentpkt=5 rcvdpkt=11 appid=15832 app="Facebook"
appcat="Social.Media" apprisk="medium" utmaction="allow" countapp=1 utmref=65531-294

2: date=2019-06-17 time=16:47:45 logid="0000000013" type="traffic" subtype="forward"
level="notice" vd="vd1" eventtime=1560815265058557636 tz="-0700" srcip=10.1.100.102
srcport=56668 srcintf="port18" srcintfrole="undefined" dstip=172.217.3.165 dstport=443
dstintf="port17" dstintfrole="undefined" poluuid="b740d418-8ed3-51e9-5a7b-114e99ab6370"
sessionid=43201 proto=6 action="timeout" user="Jack" group="QA" policyid=1
policytype="consolidated" centralnatid=1 service="HTTPS" dstcountry="United States"
srccountry="Reserved" trandisp="snat" transip=172.16.200.2 transport=56668 duration=303
sentbyte=406 rcvdbyte=384 sentpkt=4 rcvdpkt=4 appcat="unscanned" utmaction="allow"
countweb=1 utmref=65531-3486
```

Other NGFW policy-based mode options

You can combine *Application Control* and *Web Filter* in the same NGFW mode policy.

The following security profiles can be used in NGFW policy-based mode:

- AntiVirus
- Web Filter
- Intrusion Prevention
- File Filter
- Email Filter

Logging can also be enabled in security policies.

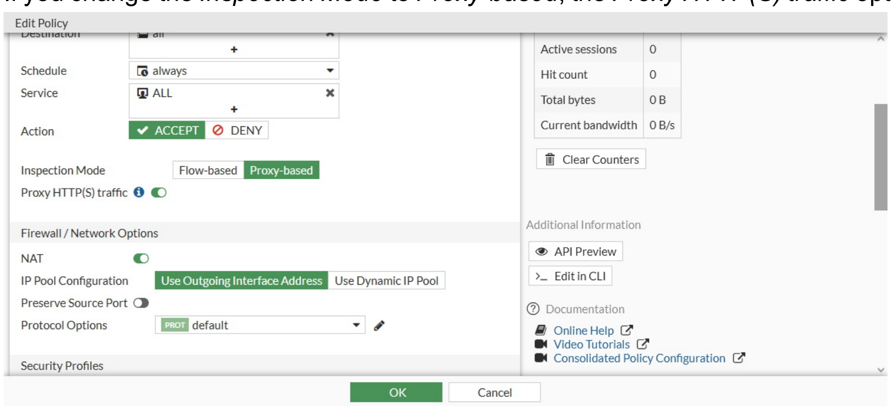
Inspection mode per policy

Inspection mode is configured on a per-policy basis in NGFW mode. This gives you more flexibility when setting up different policies.

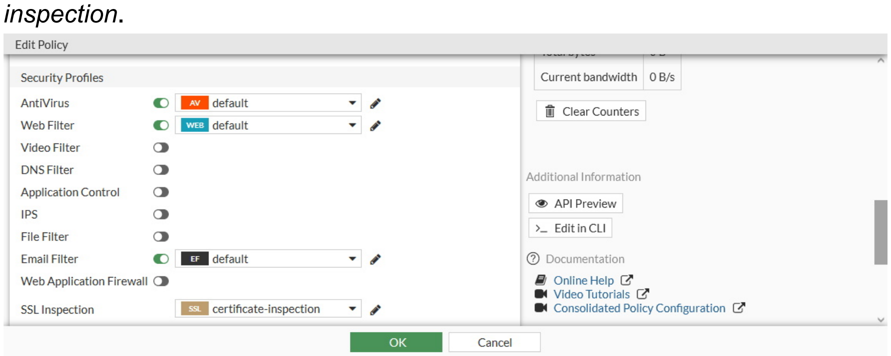
When configuring a firewall policy, you can select a *Flow-based* or *Proxy-basedInspection Mode*. The default setting is *Flow-based*.

To configure inspection mode in a policy:

1. Go to *Policy & Objects* > *Firewall Policy*.
2. Create a new policy, or edit an existing policy.
3. Configure the policy as needed.
   a. If you change the *Inspection Mode* to *Proxy-based*, the *Proxy HTTP(S) traffic* option displays.



   b. In the *Security Profiles* section, if no security profiles are enabled, the default *SSL Inspection* is *no-inspection*.
   c. In the *Security Profiles* section, if you enable any security profile, the *SSL Inspection* changes to *certificate-inspection*.



To see the inspection mode changes using the CLI:

```
config firewall policy
    edit 1
        set srcintf "wan2"
        set dstintf "wan1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set inspection-mode proxy
        set nat enable
    next
end
```

To see the HTTP and SSH policy redirect settings when inspection mode is set to proxy using the CLI:

```
config firewall policy
    edit 1
        set srcintf "wan2"
        set dstintf "wan1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set inspection-mode proxy
        set http-policy-redirect enable
        set ssh-policy-redirect enable
        set nat enable
    next
end
```

To see the default SSL-SSH policy set to no inspection using the CLI:

```
config firewall policy
    edit 1
       show fu | grep ssl-ssh-profile
        set ssl-ssh-profile "no-inspection"
    next
end
```

NGFW policy mode application default service

In NGFW policy-based mode, the application default service enforces applications running only on their default service port. The applications specified in the policy are monitored, and if traffic is detected from a nonstandard port, it is blocked, and a log entry is recorded with a *port-violation* event type.

If you are not using the default ports, and need to pick specific services, select *Specify* to select the required services.

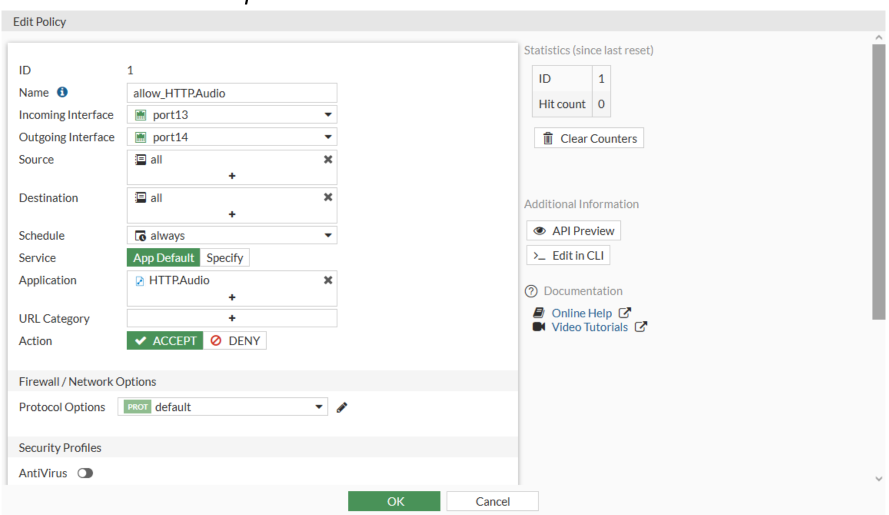
Example

In this example, the standard port is enforced for HTTPS traffic using the HTTP.Audio application.

First, an SSL Inspection & Authentication policy is created do to traffic pre-match, and then a security policy is created to allow the HTTP.Audio application when using the default port. Fetching an MP3 file from an HTTP server using port 443 is allowed, but is blocked when using a nonstandard port, such as 8443.

To enforce the HTTP.Audio application using the default port in the GUI:

1. Create a new SSL Inspection & Authentication policy, or use the default policy.
2. Go to *Policy & Objects > Security Policy*, and click *Create New*.
3. Enter a name for the policy, such as *allow_HTTP.Audio*.
4. Configure the ports as needed.
5. Set *Service* to *App Default*.
6. In the *Application* field, select *HTTP.Audio*.
7. Set the *Action* to *Accept*.



8. Click *OK*.

To enforce the HTTP.Audio application using the default port in the CLI:

1. Create a firewall policy:

```
config firewall policy
    edit 1
        set name "consolidated_all"
        set srcintf "port13"
        set dstintf "port14"
        set srcaddr "all"
        set dstaddr "all"
        set service "ALL"
        set ssl-ssh-profile "new-deep-inspection"
    next
end
```

2. Create a security policy:

```
config firewall security-policy
    edit 1
```

```
            set name "allow_HTTP.Audio"
            set srcintf "port13"
            set dstintf "port14"
            set srcaddr "all"
            set enforce-default-app-port enable
            set action accept
            set schedule "always"
            set logtraffic all
            set application 15879
        next
    end
```

Logs

The application logs show logs with an event type of `port-violation` for traffic on port 8443 that is blocked, and an event type of `signature` for traffic on port 443 that is allowed.

Blocked:

```
2: date=2019-06-18 time=16:15:40 logid="1060028736" type="utm" subtype="app-ctrl"
eventtype="port-violation" level="warning" vd="vd1" eventtime=1560899740218875746 tz="-0700"
appid=15879 srcip=10.1.100.22 dstip=172.16.200.216 srcport=52680 dstport=8443
srcintf="port13" srcintfrole="undefined" dstintf="port14" dstintfrole="undefined" proto=6
service="HTTPS" direction="incoming" policyid=1 sessionid=5041 appcat="Video/Audio"
app="HTTP.Audio" action="block" hostname="172.16.200.216" incidentserialno=1906780850
url="/app_data/story.mp3" securityid=2 msg="Video/Audio: HTTP.Audio," apprisk="elevated"
```

Allowed:

```
1: date=2019-06-18 time=16:15:49 logid="1059028704" type="utm" subtype="app-ctrl"
eventtype="signature" level="information" vd="vd1" eventtime=1560899749258579372 tz="-0700"
appid=15879 srcip=10.1.100.22 dstip=172.16.200.216 srcport=54527 dstport=443
srcintf="port13" srcintfrole="undefined" dstintf="port14" dstintfrole="undefined" proto=6
service="HTTPS" direction="incoming" policyid=1 sessionid=5064 appcat="Video/Audio"
app="HTTP.Audio" action="pass" hostname="172.16.200.216" incidentserialno=1139663486
url="/app_data/story.mp3" securityid=2 msg="Video/Audio: HTTP.Audio," apprisk="elevated"
```

Add option to set application default port as a service port

The `default-app-port-as-service` option can be used in NGFW mode to set the application default port as a service port. This allows applications to match the policy and be blocked immediately the first time that traffic hits the firewall. When this option is enabled, the NGFW policy aggregates the ports used by the applications in the policy and performs a pre-match on the traffic.

```
config system settings
    set default-app-port-as-service {enable | disable}
end
```

This option can be configured on a per-VDOM level.

This setting is enabled by default on new installations. When upgrading, the setting is disabled to retain the previous behavior.

To configure the application default port as service port:

1. Configure the VDOM settings:

```
config system settings
    set vdom-type traffic
    set opmode nat
    set ngfw-mode policy-based
    set block-land-attack disable
    set default-app-port-as-service enable
    set application-bandwidth-tracking disable
end
```

2. Configure the NGFW policy:

```
config firewall security-policy
    edit 1
        set name "test"
        set srcintf "port2"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set internet-service-src disable
        set enforce-default-app-port enable
        set action accept
    next
end
```

Sample logs

The following logging behavior occurs in NGFW mode with `default-app-port-as-service`:

- When `default-app-port-as-service` and `enforce-default-app-port` are enabled, traffic that does not match the default port is blocked immediately. Only a traffic log is generated.

Log with SSH and FTP traffic:

```
1: date=2022-02-24 time=11:16:36 eventtime=1645730197145603994 tz="-0800"
logid="0000000013" type="traffic" subtype="forward" level="notice" vd="vd1"
srcip=10.1.100.12 srcport=40402 srcintf="port2" srcintfrole="undefined"
dstip=172.16.200.55 dstport=21 dstintf="port1" dstintfrole="undefined"
srccountry="Reserved" dstcountry="Reserved" sessionid=6811 proto=6 action="deny"
policyid=0 policytype="security-policy" poluuid="7ed35582-95a2-51ec-0d21-4093cb91e67b"
policyname="Default" centralnatid=1 service="FTP" trandisp="snat" transip=172.16.200.4
transport=40402 duration=10 sentbyte=0 rcvdbyte=0 sentpkt=0 rcvdpkt=0 appcat="unscanned"
```

Log with SSH and FTP traffic with port 2121:

```
1: date=2022-02-24 time=11:19:20 eventtime=1645730360685614031 tz="-0800"
logid="0000000013" type="traffic" subtype="forward" level="notice" vd="vd1"
srcip=10.1.100.12 srcport=41362 srcintf="port2" srcintfrole="undefined"
dstip=172.16.200.55 dstport=2121 dstintf="port1" dstintfrole="undefined"
srccountry="Reserved" dstcountry="Reserved" sessionid=7213 proto=6 action="deny"
policyid=0 policytype="security-policy" poluuid="7ed35582-95a2-51ec-0d21-4093cb91e67b"
policyname="Default" centralnatid=1 service="tcp/2121" trandisp="snat"
```

```
transip=172.16.200.4 transport=41362 duration=9 sentbyte=60 rcvdbyte=0 sentpkt=1
rcvdpkt=0 appcat="unscanned"
```

- When `default-app-port-as-service` is disabled and `enforce-default-app-port` is enabled, traffic that does not match the default port is not blocked immediately. Application and traffic logs are generated.

Traffic log with SSH and FTP traffic:

```
1: date=2022-02-24 time=11:21:51 eventtime=1645730511325606916 tz="-0800"
logid="0000000013" type="traffic" subtype="forward" level="notice" vd="vd1"
srcip=10.1.100.12 srcport=40408 srcintf="port2" srcintfrole="undefined"
dstip=172.16.200.55 dstport=21 dstintf="port1" dstintfrole="undefined"
srccountry="Reserved" dstcountry="Reserved" sessionid=7522 proto=6 action="deny"
policyid=0 policytype="security-policy" poluuid="7ed35582-95a2-51ec-0d21-4093cb91e67b"
policyname="Default" centralnatid=1 service="FTP" trandisp="snat" transip=172.16.200.4
transport=40408 duration=14 sentbyte=164 rcvdbyte=171 sentpkt=3 rcvdpkt=2 appid=15896
app="FTP" appcat="Network.Service" apprisk="elevated" utmaction="block" countapp=1
utmref=65501-0
```

Application log with SSH and FTP traffic:

```
2: date=2022-02-24 time=11:21:39 eventtime=1645730499338228209 tz="-0800"
logid="1059028705" type="utm" subtype="app-ctrl" eventtype="signature" level="warning"
vd="vd1" appid=15896 srcip=10.1.100.12 srccountry="Reserved" dstip=172.16.200.55
dstcountry="Reserved" srcport=40408 dstport=21 srcintf="port2" srcintfrole="undefined"
dstintf="port1" dstintfrole="undefined" proto=6 service="FTP" direction="outgoing"
policyid=0 sessionid=7522 action="block" appcat="Network.Service" app="FTP"
incidentserialno=188744239 msg="Network.Service: FTP" apprisk="elevated"
```

Traffic log with SSH and FTP traffic with port 2121:

```
1: date=2022-02-24 time=11:24:25 eventtime=1645730665235613912 tz="-0800"
logid="0000000013" type="traffic" subtype="forward" level="notice" vd="vd1"
srcip=10.1.100.12 srcport=41366 srcintf="port2" srcintfrole="undefined"
dstip=172.16.200.55 dstport=2121 dstintf="port1" dstintfrole="undefined"
srccountry="Reserved" dstcountry="Reserved" sessionid=7876 proto=6 action="deny"
policyid=0 policytype="security-policy" poluuid="7ed35582-95a2-51ec-0d21-4093cb91e67b"
policyname="Default" centralnatid=1 service="tcp/2121" trandisp="snat"
transip=172.16.200.4 transport=41366 duration=11 sentbyte=112 rcvdbyte=171 sentpkt=2
rcvdpkt=2 appid=15896 app="FTP" appcat="Network.Service" apprisk="elevated"
utmaction="block" countapp=1 utmref=65500-0
```

Application log with SSH and FTP traffic with port 2121:

```
2: date=2022-02-24 time=11:24:16 eventtime=1645730656426052412 tz="-0800"
logid="1060028736" type="utm" subtype="app-ctrl" eventtype="port-violation"
level="warning" vd="vd1" appid=15896 srcip=10.1.100.12 srccountry="Reserved"
dstip=172.16.200.55 dstcountry="Reserved" srcport=41366 dstport=2121 srcintf="port2"
srcintfrole="undefined" dstintf="port1" dstintfrole="undefined" proto=6 service="FTP"
direction="outgoing" policyid=0 sessionid=7876 action="block" appcat="Network.Service"
app="FTP" incidentserialno=188744241 msg="Network.Service: FTP, non-default port used:
2121" apprisk="elevated"
```

Application logging in NGFW policy mode

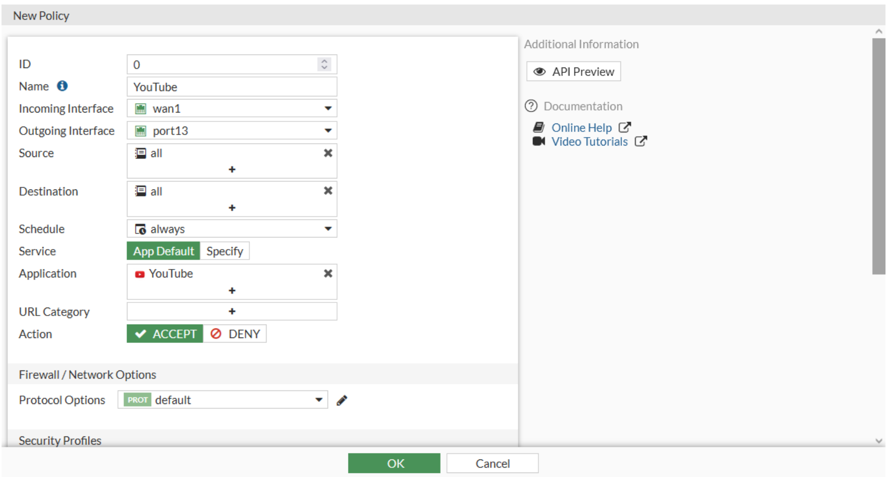
In NGFW policy mode, if an application, application category, or application group is selected on a security policy, and traffic logging is set to *UTM* or *All*, then application control logs will be generated. In addition, when a signature is set to the *ACCEPT* action under a security policy, all corresponding child signatures will be assessed and logged as well.

Under NGFW, with `default-app-port-as-service` enabled, enable APP Default. The traffic which doesn't match the default port will be blocked immediately, and there is only traffic log generated.

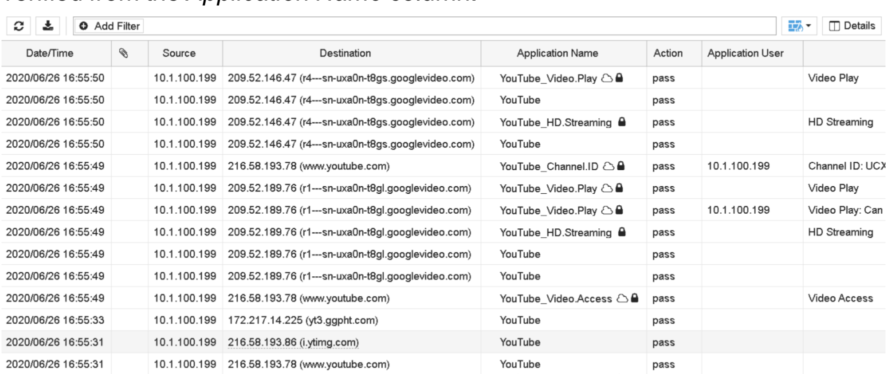
Under NGFW, with `default-app-port-as-service` disabled, enable APP Default. The traffic which doesn't match the default port will not be blocked immediately, and there is app and traffic logs generated.

To verify application logging:

1. Go to *Policy & Objects > Security Policy* and configure a new policy for YouTube.
2. Set *Action* to *ACCEPT* and *Log Allowed Traffic* to *Security Events*.



3. Configure the remaining settings as required, then click *OK*.
4. On a client system, play some YouTube videos.
5. On FortiOS, go to *Log & Report > Security Events* and view the *Application Control* logs.
   There are logs not only for *YouTube*, but also for *YouTube_Video.Play*, *YouTube_Video.Access*, and so on, as verified from the *Application Name* column.



Date/Time		Source	Destination	Application Name	Action	Application User	
2020/06/26 16:55:50		10.1.100.199	209.52.146.47 (r4---sn-uxa0n-t8gs.googlevideo.com)	YouTube_Video.Play	pass		Video Play
2020/06/26 16:55:50		10.1.100.199	209.52.146.47 (r4---sn-uxa0n-t8gs.googlevideo.com)	YouTube	pass		
2020/06/26 16:55:50		10.1.100.199	209.52.146.47 (r4---sn-uxa0n-t8gs.googlevideo.com)	YouTube_HD.Streaming	pass		HD Streaming
2020/06/26 16:55:50		10.1.100.199	209.52.146.47 (r4---sn-uxa0n-t8gs.googlevideo.com)	YouTube	pass		
2020/06/26 16:55:49		10.1.100.199	216.58.193.78 (www.youtube.com)	YouTube_Channel.ID	pass	10.1.100.199	Channel ID: UCX
2020/06/26 16:55:49		10.1.100.199	209.52.189.76 (r1---sn-uxa0n-t8gl.googlevideo.com)	YouTube_Video.Play	pass		Video Play
2020/06/26 16:55:49		10.1.100.199	209.52.189.76 (r1---sn-uxa0n-t8gl.googlevideo.com)	YouTube_Video.Play	pass	10.1.100.199	Video Play: Can
2020/06/26 16:55:49		10.1.100.199	209.52.189.76 (r1---sn-uxa0n-t8gl.googlevideo.com)	YouTube_HD.Streaming	pass		HD Streaming
2020/06/26 16:55:49		10.1.100.199	209.52.189.76 (r1---sn-uxa0n-t8gl.googlevideo.com)	YouTube	pass		
2020/06/26 16:55:49		10.1.100.199	209.52.189.76 (r1---sn-uxa0n-t8gl.googlevideo.com)	YouTube	pass		
2020/06/26 16:55:49		10.1.100.199	216.58.193.78 (www.youtube.com)	YouTube_Video.Access	pass		Video Access
2020/06/26 16:55:33		10.1.100.199	172.217.14.225 (yt3.ggpht.com)	YouTube	pass		
2020/06/26 16:55:31		10.1.100.199	216.58.193.86 (i.ytimg.com)	YouTube	pass		
2020/06/26 16:55:31		10.1.100.199	216.58.193.78 (www.youtube.com)	YouTube	pass		

Learn mode in security policies in NGFW mode

In NGFW mode, administrators can configure a security policy in learn mode to monitor traffic that passes through the source and destination interfaces. The learn mode uses a special prefix in the `policymode` and `profile` fields in traffic and UTM logs for use by FortiAnalyzer and the Policy Analyzer Management Extension Application (MEA) that is available with FortiManager.

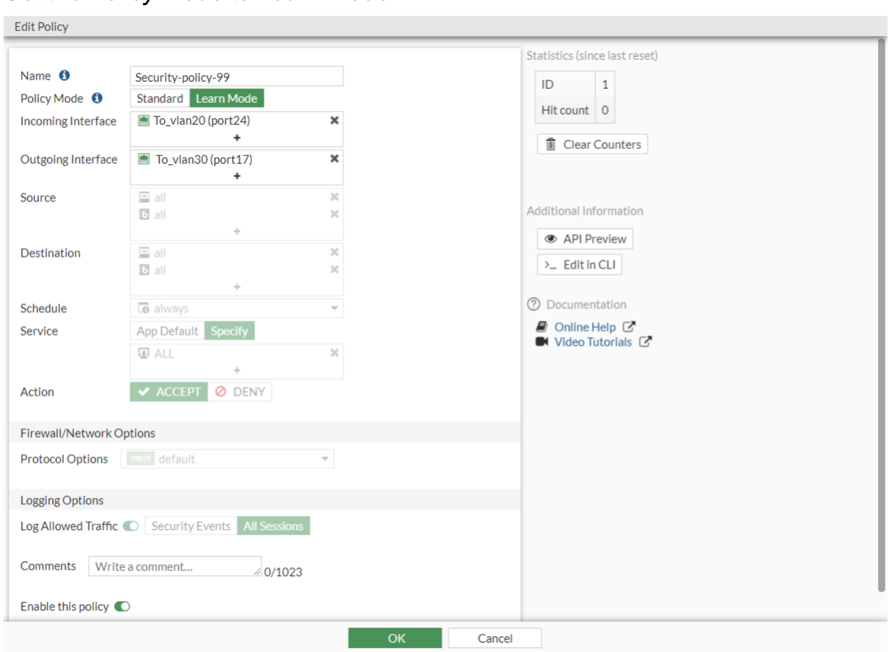
When enabled on FortiManager, Policy Analyzer MEA works with security policies in learning mode to analyze logs sent from a managed FortiGate to FortiAnalyzer. Based on the analyzed traffic, FortiManager administrators can choose to automatically create a policy in FortiManager for the managed FortiGate. For more information about Policy Analyzer MEA, see the Policy Analyzer Administration Guide.

The following limitations apply when learn mode is enabled in a security policy:

- Only interfaces with `device-identification enable` can be used as source interfaces in a security policy with learning mode enabled.
- Incoming and outgoing interfaces do not support `any`.
- Internet service is not supported.
- NAT46 and NAT64 are not supported.
- Users and groups are not supported.
- Some negate options are not supported.

To enable learn mode in the GUI:

1. Enable policy-based NGFW mode:
   a. Go to *System > Settings*.
   b. Set the *NGFW Mode* to *Policy-based* and click *Apply*.
2. Go to *Policy & Objects > Security Policy*, and open a security policy for editing.
3. Set the *Policy Mode* to *Learn Mode*.



4. Select an *Incoming Interface*.
5. Select an *Outgoing Interface*.
6. (Optional) Type a comment in the *Comments* box.
7. Toggle on *Enable this policy*.
8. Click *OK* to save the security policy.

To enable learn mode in the CLI:

1. Enable policy-based NGFW mode:

```
config system settings
    set ngfw-mode policy-based
end
```

2. Enable learn mode in a security policy:

```
config firewall security-policy
    edit <id>
        set learning-mode enable
    next
end
```

To view learn mode fields in logs in the CLI:

1. Filter and view fields in traffic logs:

```
# execute log filter category 0

# execute log display

1 logs found.

1 logs returned.

1: date=2022-03-21 time=10:21:11 eventtime=1647883271150012188 tz="-0700"
logid="0000000013" type="traffic" subtype="forward" level="notice" vd="root"
srcip=10.1.100.41 srcport=43296 srcintf="port24" srcintfrole="undefined"
dstip=172.16.200.55 dstport=80 dstintf="port17" dstintfrole="wan"
srccountry="Reserved" dstcountry="Reserved" sessionid=33934 proto=6
policymode="learn" action="accept" policyid=99 policytype="security-policy"
poluuid="6e3f7f54-a932-51ec-73ba-8282cfd0b73c" policyname="Security-policy-99"
centralnatid=3 service="HTTP" trandisp="snat" transip=172.16.200.9 transport=43296
duration=1 sentbyte=412 rcvdbyte=529 sentpkt=6 rcvdpkt=4 appid=15893
app="HTTP.BROWSER" appcat="Web.Client" apprisk="medium" utmaction="allow"
countweb=1 countav=1 countips=3 countapp=1 crscore=50 craction=2
srchwvendor="VMware" devtype="Computer" osname="Debian"
mastersrcmac="00:0c:29:b5:92:8d" srcmac="00:0c:29:b5:92:8d" srcserver=0
utmref=65534-0
```

2. Filter and view fields in UTM logs:

```
# execute log filter category 2

# execute log display

1 logs found.

1 logs returned.

1: date=2022-03-21 time=10:21:09 eventtime=1647883270101403283 tz="-0700"
logid="0211008193" type="utm" subtype="virus" eventtype="infected" level="notice"
vd="root" policyid=99 poluuid="6e3f7f54-a932-51ec-73ba-8282cfd0b73c"
policytype="security-policy" policymode="learn" msg="File is infected."
action="monitored" service="HTTP" sessionid=33934 srcip=10.1.100.41
dstip=172.16.200.55 srcport=43296 dstport=80 srccountry="Reserved"
dstcountry="Reserved" srcintf="port24" srcintfrole="undefined" dstintf="port17"
dstintfrole="wan" proto=6 direction="incoming" filename="eicar.com"
quarskip="Quarantine-disabled" virus="EICAR_TEST_FILE" viruscat="Virus" dtype="av-
engine" ref="http://www.fortinet.com/ve?vn=EICAR_TEST_FILE" virusid=2172
url="http://172.16.200.55/virus/eicar.com" profile="learn-av" agent="curl/7.35.0"
httpmethod="GET"
analyticscksum="275a021bbfb6489e54d471899f7db9d1663fc695ec2fe2a2c4538aabf651fd0f"
analyticssubmit="false" crscore=50 craction=2 crlevel="critical" rawdata="Response-
Content-Type=application/x-msdos-program"
```

3. Filter and view fields in UTM-IPS logs:

```
# execute log filter category 4

# execute log display

3 logs found.

3 logs returned.

1: date=2022-03-21 time=10:21:09 eventtime=1647883270101485354 tz="-0700"
logid="0419016384" type="utm" subtype="ips" eventtype="signature" level="alert"
vd="root" severity="info" srcip=10.1.100.41 srccountry="Reserved"
dstip=172.16.200.55 dstcountry="Reserved" srcintf="port24" srcintfrole="undefined"
dstintf="port17" dstintfrole="wan" sessionid=33934 action="detected" proto=6
service="HTTP" policyid=99 poluuid="6e3f7f54-a932-51ec-73ba-8282cfd0b73c"
policytype="security-policy" policymode="learn" attack="Eicar.Virus.Test.File"
srcport=43296 dstport=80 agent="curl/7.35.0" httpmethod="GET" direction="incoming"
attackid=29844 profile="learn-ips" ref="http://www.fortinet.com/ids/VID29844"
incidentserialno=158335134 attackcontextid="2/2"
attackcontext="YW0NCg0KWDVPIVAlQEFQWzRcUFpYNTQoUF4pN0NDKTd9JEVJQ0FSLVNUQU5EQVJELUFO
VElWSVJVUy1URVNULUZJTEUhJEgrSCo8L1BBQ0tFVD4="

2: date=2022-03-21 time=10:21:09 eventtime=1647883270101484791 tz="-0700"
logid="0419016384" type="utm" subtype="ips" eventtype="signature" level="alert"
```

```
vd="root" severity="info" srcip=10.1.100.41 srccountry="Reserved"
dstip=172.16.200.55 dstcountry="Reserved" srcintf="port24" srcintfrole="undefined"
dstintf="port17" dstintfrole="wan" sessionid=33934 action="detected" proto=6
service="HTTP" policyid=99 poluuid="6e3f7f54-a932-51ec-73ba-8282cfd0b73c"
policytype="security-policy" policymode="learn" attack="Eicar.Virus.Test.File"
srcport=43296 dstport=80 agent="curl/7.35.0" httpmethod="GET" direction="incoming"
attackid=29844 profile="learn-ips" ref="http://www.fortinet.com/ids/VID29844"
incidentserialno=158335134 attackcontextid="1/2"
attackcontext="PFBBVFRFUk5TPiBYNU8hUCVAQVBbNFxQWlg1NChQXik3Q0MpN30kRUlDQVItU1RBTkRB
UkQtQU5USVZJUlVTLVRFU1QtRklMRSEkSCtIKjtYNU8hUCVAQVBbNFxQWlg1NChQXik3Q0MpN30kRUlDQVI
tU1RBTkRBUkQtQU5USVZJUlVTLVRFU1QtRklMRSEkSCtIKjwvUEFUVEVSTlM+CjxVUkk+IDwvVVJJPgo8SE
VBREVSPiBIVFRQLzEuMSAyMDAgT0sNCkRhdGU6IE1vbiwgMjEgTWFyIDIwMjIgMTc6MjE6MTAgR01UDQpTZ
XJ2ZXI6IEFwYWNoZS8yLjQuMTggKFVidW50dSkNCkxhc3QtTW9kaWZpZWQ6IFRodSwgMDEgRGVjIDIwMTYg
MDE6MjY6MzUgR01UDQpFVGFnOiAiNDQtNTQyOGViNjU4MDk3YSINCkFjY2VwdC1SYW5nZXM6IGJ5dGVzDQp
Db250ZW50LUxlbmd0aDogNjgNCkNvbnRlbnQtVHlwZTogYXBwbGljYXRpb24veC1tc2Rvcy1wcm9ncmFtDQ
oNCjwvSEVBREVSPgo8Qk9EWT4gWDVPIVAlQEFQWzRcUFpYNTQoUF4pN0NDKTd9JEVJQ0FSLVNUQU5EQVJEL
UFOVElWSVJVUy1URVNULUZJTEUhJEgrSCo8L0JPRFk+CjxQQUNLRVQ+IEhUVFAvMS4xIDIwMCBPSw0KRGF0
ZTogTW9uLCAyMSBNYXIgMjAyMiAxNzoyMToxMCBHTVQNClNlcnZlcjogQXBhY2hlLzIuNC4xOCAoVWJ1bnR
1KQ0KTGFzdC1Nb2RpZmllZDogVGh1LCAwMSBEZWMgMjAxNiAwMToyNjozNSBHTVQNCkVUYWc6ICI0NC01ND
I4ZWI2NTgwOTdhIg0KQWNjZXB0LVJhbmdlczogYnl0ZXMNCkNvbnRlbnQtTGVuZ3RoOiA2OA0KQ29udGVud
C1UeXBlOiBhcHBsaWNhdGlvbi94LW1zZG9zLXByb2dy"

3: date=2022-03-21 time=10:21:09 eventtime=1647883270101483279 tz="-0700"
logid="0419016384" type="utm" subtype="ips" eventtype="signature" level="alert"
vd="root" severity="info" srcip=10.1.100.41 srccountry="Reserved"
dstip=172.16.200.55 dstcountry="Reserved" srcintf="port24" srcintfrole="undefined"
dstintf="port17" dstintfrole="wan" sessionid=33934 action="detected" proto=6
service="HTTP" policyid=99 poluuid="6e3f7f54-a932-51ec-73ba-8282cfd0b73c"
policytype="security-policy" policymode="learn" attack="Eicar.Virus.Test.File"
srcport=43296 dstport=80 hostname="172.16.200.55" url="/virus/eicar.com"
agent="curl/7.35.0" httpmethod="GET" direction="incoming" attackid=29844
profile="learn-ips" ref="http://www.fortinet.com/ids/VID29844"
incidentserialno=158335134 msg="file_transfer: Eicar.Virus.Test.File"
attackcontextid="0/2" rawdataid="1/1" rawdata="Response-Content-Type=application/x-
msdos-program"
```

4. Filter and view fields in UTM-webfilter logs:

```
# execute log filter category 3

# execute log display

2 logs found.

2 logs returned.

2: date=2022-03-21 time=10:21:09 eventtime=1647883270100329681 tz="-0700"
logid="0319013317" type="utm" subtype="webfilter" eventtype="urlmonitor"
level="notice" vd="root" policyid=99 poluuid="6e3f7f54-a932-51ec-73ba-8282cfd0b73c"
policytype="security-policy" policymode="learn" sessionid=33934 srcip=10.1.100.41
```

```
srcport=43296 srccountry="Reserved" srcintf="port24" srcintfrole="undefined"
dstip=172.16.200.55 dstport=80 dstcountry="Reserved" dstintf="port17"
dstintfrole="wan" proto=6 httpmethod="GET" service="HTTP" hostname="172.16.200.55"
agent="curl/7.35.0" profile="learn-webf" action="passthrough" reqtype="direct"
url="http://172.16.200.55/virus/eicar.com" sentbyte=92 rcvdbyte=0
direction="outgoing" msg="URL has been visited" ratemethod="domain" cat=255
catdesc="Unknown"
```

Local-in policy

While security profiles control traffic flowing through the FortiGate, local-in policies control inbound traffic that is going to a FortiGate interface.

Administrative access traffic (HTTPS, PING, SSH, and others) can be controlled by allowing or denying the service in the interface settings. Trusted hosts can be configured under an administrator to restrict the hosts that can access the administrative service.

Local-in policies allow administrators to granularly define the source and destination addresses, interface, and services. Traffic destined for the FortiGate interface specified in the policy that meets the other criteria is subject to the policies action.

Local-in policies can be used to restrict administrative access or other services, such as VPN, that can be specified as services. You can define source addresses or address groups to restrict access from. For example, by using a geographic type address you can restrict a certain geographic set of IP addresses from accessing the FortiGate.

By default, no local-in policies are defined, so there are no restrictions on local-in traffic.

Local-in policies can only be created or edited in the CLI. You can view the existing local-in policies in the GUI by enabling it in *System* > *Feature Visibility* under the *Additional Features* section. This page does not list the custom local-in policies.

To configure a local-in policy using the CLI:

```
config firewall {local-in-policy | local-in-policy6}
    edit <policy_number>
        set intf <interface>
        set srcaddr <source_address> [source_address] ...
        set dstaddr <destination_address> [destination_address] ...
        set action {accept | deny}
        set service <service_name> [service_name] ...
        set schedule <schedule_name>
        set comments <string>
    next
end
```

For example, to prevent the source subnet 10.10.10.0/24 from pinging port1, but allow administrative access for PING on port1:

```
config firewall address
    edit "10.10.10.0"
        set subnet 10.10.10.0 255.255.255.0
    next
end
```

```
config firewall local-in-policy
    edit 1
        set intf "port1"
        set srcaddr "10.10.10.0"
        set dstaddr "all"
        set service "PING"
        set schedule "always"
    next
end
```

To test the configuration:

1. From the PC at 10.10.10.12, start a continuous ping to port1:

   ```
   ping 192.168.2.5 -t
   ```

2. On the FortiGate, enable debug flow:

   ```
   # diagnose debug flow filter addr 10.10.10.12
   # diagnose debug flow filter proto 1
   # diagnose debug enable
   # diagnose debug flow trace start 10
   ```

3. The output of the debug flow shows that traffic is dropped by local-in policy 1:

   ```
   # id=20085 trace_id=1 func=print_pkt_detail line=5746 msg="vd-root:0 received a packet
   (proto=1, 10.10.10.12:1->192.168.2.5:2048) from port1. type=8, code=0, id=1, seq=128."
   id=20085 trace_id=1 func=init_ip_session_common line=5918 msg="allocate a new session-
   0017c5ad"
   id=20085 trace_id=1 func=vf_ip_route_input_common line=2615 msg="find a route:
   flag=80000000 gw-192.168.2.5 via root"
   id=20085 trace_id=1 func=fw_local_in_handler line=474 msg="iprope_in_check() check
   failed on policy 1, drop"
   ```

Additional options

To disable or re-enable the local-in policy, use the `set status {enable | disable}` command.

To dedicate the interface as an HA management interface, use the `set ha-mgmt-intf-only enable` command.

TTL policies

You can configure a time-to-live (TTL) policy to block attack traffic with high TTLs. This feature only applies to local-in traffic and does not apply to traffic passing through the FortiGate. You can use `srcintf` to set the interface that the local-in traffic hits. See config firewall ttl-policy.

To configure a TTL policy using the CLI:

```
config firewall ttl-policy
    edit <id>
        set status {enable | disable}
        set action {accept | deny}
        set srcintf <interface>
        set srcaddr <source_address> [source_address] ...
        set service <service_name> [service_name] ...
        set schedule <schedule_name>
```

```
        set ttl <value/range>
    next
end
```

DoS policy

A Denial of Service (DoS) policy examines network traffic arriving at a FortiGate interface for anomalous patterns, which usually indicates an attack.

A denial of service occurs when an attacking system starts an abnormally large number of sessions with a target system. The large number of sessions slows down or disables the target system, preventing legitimate users from using it.

DoS policies are checked before security policies, preventing attacks from triggering more resource intensive security protection and slowing down the FortiGate.

DoS anomalies

Predefined sensors are setup for specific anomalous traffic patterns. New DoS anomalies cannot be added by the user.

The predefined anomalies that can be used in DoS policies are:

Anomaly	Description	Recommended Threshold
tcp_syn_flood	If the SYN packet rate of new TCP connections, including retransmission, to one destination IP address exceeds the configured threshold value, the action is executed.	2000 packets per second.
tcp_port_scan	If the SYN packet rate of new TCP connections, including retransmission, from one source IP address exceeds the configured threshold value, the action is executed.	1000 packets per second.
tcp_src_session	If the number of concurrent TCP connections from one source IP address exceeds the configured threshold value, the action is executed.	5000 concurrent sessions.
tcp_dst_session	If the number of concurrent TCP connections to one destination IP address exceeds the configured threshold value, the action is executed.	5000 concurrent sessions.
udp_flood	If the UDP traffic to one destination IP address exceeds the configured threshold value, the action is executed.	2000 packets per second.
udp_scan	If the UDP sessions setup rate originating from one source IP address exceeds the configured threshold value, the action is executed.	2000 sessions per second.
udp_src_session	If the number of concurrent UDP connections from one source IP address exceeds the configured threshold value, the action is executed.	5000 concurrent sessions.
udp_dst_session	If the number of concurrent UDP connections to one destination IP address exceeds the configured threshold value, the action	5000 concurrent sessions.

Anomaly	Description	Recommended Threshold
	is executed.	
icmp_flood	If the number of ICMP packets sent to one destination IP address exceeds the configured threshold value, the action is executed.	250 packets per second.
icmp_sweep	If the ICMP sessions setup rate originating from one source IP address exceeds the configured threshold value, the action is executed.	100 sessions per second.
icmp_src_session	If the number of concurrent ICMP connections from one source IP address exceeds the configured threshold value, the action is executed.	300 concurrent sessions
icmp_dst_session	If the number of concurrent ICMP connections to one destination IP address exceeds the configured threshold value, the action is executed.	1000 concurrent sessions
ip_src_session	If the number of concurrent IP connections from one source IP address exceeds the configured threshold value, the action is executed.	5000 concurrent sessions.
ip_dst_session	If the number of concurrent IP connections to one destination IP address exceeds the configured threshold value, the action is executed.	5000 concurrent sessions.
sctp_flood	If the number of SCTP packets sent to one destination IP address exceeds the configured threshold value, the action is executed.	2000 packets per second
sctp_scan	If the number of SCTP sessions originating from one source IP address exceeds the configured threshold value, the action is executed.	1000 packets per second
sctp_src_session	If the number of concurrent SCTP connections from one source IP address exceeds the configured threshold value, the action is executed.	5000 concurrent sessions
sctp_dst_session	If the number of concurrent SCTP connections to one destination IP address exceeds the configured threshold value, the action is executed.	5000 concurrent sessions

For thresholds based on the number of concurrent sessions, blocking the anomaly will not allow more than the number of concurrent sessions to be set as the threshold.

For example, if the period for a particular anomaly is 60 seconds, such as those where the threshold is measured in concurrent sessions, after the 60 second timer has expired the number of allowed sessions that match the anomaly criteria is reset to zero. This means that, if you allow 10 sessions through before blocking, after the 60 seconds has elapsed, another 10 sessions will be allowed. The attrition of sessions from expiration should keep the allowed sessions from reaching the maximum.

For rate based thresholds, where the threshold is measured in packets per second, the *Block* action prevents anomalous traffic from overwhelming the firewall in two ways:

- continuous: Block packets once an anomaly is detected, and continue to block packets while the rate is above the threshold. This is the default setting.
- periodical: After an anomaly is detected, allow the configured number of packets per second.

For example, if a DoS policy is configured to block icmp_flood with a threshold of 10pps, and a continuous ping is started at a rate of 20pps for 1000 packets:

- In continuous mode, the first 10 packets are passed before the DoS sensor if triggered, and then the remaining 990 packets are blocked.
- In periodical mode, 10 packets are allowed to pass per second, so 500 packets are blocked in the 50 seconds during which the ping is occuring.

The actual numbers of passed and blocked packets may not be exact, as fluctuations in the rates can occur, but the numbers should be close to the defined threshold.

To configure the block action for rate based anomaly sensors:

```
config ips global
    set anomaly-mode {continuous | periodical}
end
```

DoS policies

A DoS policy can be configured to use one or more anomalies.

To configure a DoS policy in the GUI:

1. Go to *Policy & Objects > IPv4 DoS Policy* or *Policy & Objects > IPv6 DoS Policy* and click *Create New*.
   If the option is not visible, enable *DoS Policy* in *Feature Visibility*. See Feature visibility on page 2163 for details.

2. Configure the following:

Name	Enter a name for the policy.
Incoming Interface	Enter the interface that the policy applies to.
Source Address	Enter the source address.
Destination Address	Enter the destination address. This is the address that the traffic is addressed to. In this case, it must be an address that is associated with the firewall interface. For example, it could be an interface address, a secondary IP address, or the address assigned to a VIP address.
Service	Select the services or service groups. The ALL service can used or, to optimize the firewall resources, only the services that will be answered on an interface can be used.

L3 Anomalies
L4 Anomalies
	Configure the anomalies:
	• *Logging*: Enable/disable logging for specific anomalies or all of them. Anomalous traffic will be logged when the action is *Block* or *Monitor*.
	• *Action*: Select the action to take when the threshold is reached:
	  • *Disable*: Do not scan for the anomaly.
	  • *Block*: Block the anomalous traffic.
	  • *Monitor*: Allow the anomalous traffic, but record a log message if logging is enabled.
	• *Threshold*: The number of detected instances that triggers the anomaly action.

Comments	Optionally, enter a comment.

3. Enable the policy, then click *OK*.

The quarantine option is only available in the CLI. See Quarantine on page 794 for information.

To configure a DoS policy in the GUI:

```
config firewall DoS-policy
    edit 1
        set name "Flood"
        set interface "port1"
        set srcaddr "all"
        set dstaddr "all"
        set service "ALL"
        config anomaly
            edit "icmp_flood"
                set status enable
                set log enable
                set action block
                set quarantine attacker
                set quarantine-expiry 1d1h1m
```

```
                set quarantine-log enable
                set threshold 100
            next
        end
    next
end
```

name <string>	Enter a name for the policy.
interface <string>	Enter the interface that the policy applies to.
srcaddr <string>	Enter the source address.
dstaddr <string>	Enter the destination address. This is the address that the traffic is addressed to. In this case, it must be an address that is associated with the firewall interface. For example, it could be an interface address, a secondary IP address, or the address assigned to a VIP address.
service <string>	Enter the services or service groups. The ALL service can used or, to optimize the firewall resources, only the services that will be answered on an interface can be used.
status {enable | disable}	Enable/disable this anomaly.
log {enable | disable}	Enable/disable anomaly logging. When enabled, a log is generated whenever the anomaly action is triggered, regardless of which action is configured.
action {pass | block}	Set the action to take when the threshold is reached: • pass: Allow traffic, but record a log message if logging is enabled. • block: Block traffic if this anomaly is found.
quarantine {none | attacker}	Set the quarantine method (see Quarantine on page 794): • none: Disable quarantine. • attacker: Block all traffic from the attacker's IP address, and add the attacker's IP address to the banned user list.
quarantine-expiry <###d##h##m>	Set the duration of the quarantine, in days, hours, and minutes (###d##h##m) (1m - 364d23h59m, default = 5m). This option is available if quarantine is set attacker.
quarantine-log {enable | disable}	Enable/disable quarantine logging (default = disable). This option is available if quarantine is set attacker.
threshold <integer>	The number of detected instances - packets per second or concurrent session number - that triggers the anomaly action.

Quarantine

Quarantine is used to block any further traffic from a source IP address that is considered a malicious actor or a source of traffic that is dangerous to the network. Traffic from the source IP address is blocked for the duration of the quarantine, and the source IP address is added to the banned user list.

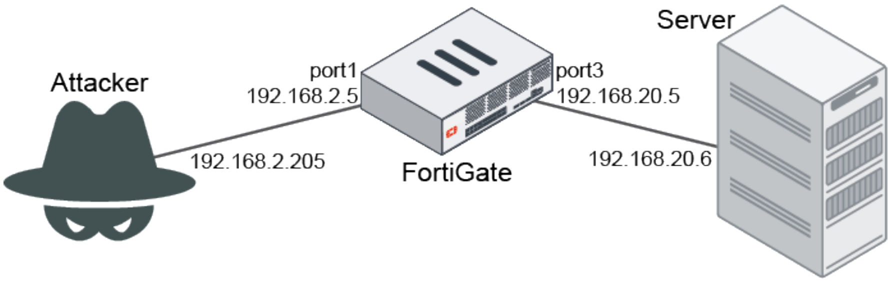
The banned user list is kept in the kernel, and used by Antivirus, Data Leak Prevention (DLP), DoS, and Intrusion Prevention System (IPS). Any policies that use any of these features will block traffic from the attacker's IP address.

To view the quarantined user list:

```
# diagnose user quarantine list
src-ip-addr       created                  expires                  cause
192.168.2.205     Wed Nov 25 12:47:54 2020 Wed Nov 25 12:57:54 2020 DOS
```

Troubleshooting DoS attacks

The best way to troubleshoot DoS attacks is with Anomaly logs and IPS anomaly debug messages.



To test an icmp_flood attack:

1. From the Attacker, launch an icmp_flood with 50pps lasting for 3000 packets.
2. On the FortiGate, configure continuous mode and create a DoS policy with an icmp_flood threshold of 30pps:

```
config firewall DoS-policy
    edit 1
        set name icmpFlood
        set interface "port1"
        set srcaddr "all"
        set dstaddr "all"
        set service "ALL"
        config anomaly
            edit "icmp_flood"
                set status enable
                set log enable
                set action block
                set threshold 30
            next
        end
    next
end
```

3. Configure the debugging filter:

```
# diagnose ips anomaly config
DoS sensors in kernel vd 0:
DoS id 1 proxy 0
  0 tcp_syn_flood status 0 log 0 nac 0 action 0 threshold 2000
  ...
  7 udp_dst_session status 0 log 0 nac 0 action 0 threshold 5000
  8 icmp_flood status 1 log 1 nac 0 action 7 threshold 30
  9 icmp_sweep status 0 log 0 nac 0 action 0 threshold 100
  ...
total # DoS sensors: 1.
```

```
# diagnose ips anomaly filter id 8
```

4. Launch the icmp_flood from a Linux machine. This example uses Nmap:

```
$ sudo nping --icmp --rate 50 -c 3000 192.168.2.50
SENT (0.0522s) ICMP [192.168.2.205 > 192.168.2.50 Echo request (type=8/code=0) id=8597
seq=1] IP [ttl=64 id=47459 iplen=28 ]
...
Max rtt: 11.096ms | Min rtt: 0.028ms | Avg rtt: 1.665ms
Raw packets sent: 3000 (84.000KB) | Rcvd: 30 (840B) | Lost: 2970 (99.00%)
Nping done: 1 IP address pinged in 60.35 seconds
```

5. During the attack, check the anomaly list on the FortiGate:

```
# diagnose ips anomaly list
list nids meter:
id=icmp_flood         ip=192.168.2.50 dos_id=1 exp=998 pps=46 freq=50

total # of nids meters: 1.
```

id=icmp_flood	The anomaly name.
ip=192.168.2.50	The IP address of the host that triggered the anomaly. It can be either the client or the server. For icmp_flood, the IP address is the destination IP address. For icmp_sweep, it would be the source IP address.
dos_id=1	The DoS policy ID.
exp=998	The time to be expired, in jiffies (one jiffy = 0.01 seconds).
pps=46	The number of packets that had been received when the diagnose command was executed.
freq=50	For session based anomalies, freq is the number of sessions. For packet rate based anomalies (flood, scan): • In continuous mode: freq is the greater of pps, or the number of packets received in the last second. • In periodic mode: freq is the pps.

6. Go to *Log & Report > Security Events* and download the *Anomaly* logs:

```
date=2020-11-20 time=14:38:39 eventtime=1605911919824184594 tz="-0800"
logid="0720018433" type="utm" subtype="anomaly" eventtype="anomaly" level="alert"
vd="root" severity="critical" srcip=192.168.2.205 srccountry="Reserved"
dstip=192.168.2.50 srcintf="port1" srcintfrole="undefined" sessionid=0 action="clear_
session" proto=1 service="PING" count=1307 attack="icmp_flood" icmpid="0x2195"
icmptype="0x08" icmpcode="0x00" attackid=16777316 policyid=1 policytype="DoS-policy"
ref="http://www.fortinet.com/ids/VID16777316" msg="anomaly: icmp_flood, 31 > threshold
30, repeats 28 times" crscore=50 craction=4096 crlevel="critical"

date=2020-11-20 time=14:39:09 eventtime=1605911949826224056 tz="-0800"
logid="0720018433" type="utm" subtype="anomaly" eventtype="anomaly" level="alert"
vd="root" severity="critical" srcip=192.168.2.205 srccountry="Reserved"
dstip=192.168.2.50 srcintf="port1" srcintfrole="undefined" sessionid=0 action="clear_
session" proto=1 service="PING" count=1497 attack="icmp_flood" icmpid="0x2195"
icmptype="0x08" icmpcode="0x00" attackid=16777316 policyid=1 policytype="DoS-policy"
```

```
ref="http://www.fortinet.com/ids/VID16777316" msg="anomaly: icmp_flood, 50 > threshold
30, repeats 1497 times" crscore=50 craction=4096 crlevel="critical"
```

Analysis

In the first log message:

msg="anomaly: icmp_flood, 31 > threshold 30	At the beginning of the attack, a log is recorded when the threshold of 30pps is broken.
repeats 28 times	The number of packets that has exceeded the threshold since the last time a log was recorded.
srcip=192.168.2.205 **dstip=192.168.2.50**	The source and destination IP addresses of the attack.
action="clear_session"	Equivalent to block. If `action` was set to `monitor` and logging was enabled, this would be `action="detected"`.

In the second log message:

- Because it is an ongoing attack, the FortiGate generates one log message for multiple packets every 30 seconds..
- It will not generate a log message if:
  - The same attack ID happened more than once in a five second period, or
  - The same attack ID happened more than once in a 30 second period and the actions are the same and have the same source and destination IP addresses.

msg="anomaly: icmp_flood, 50 > threshold 30	In the second before the log was recorded, 50 packets were detected, exceeding the configured threshold.
repeats 1497 times	The number of packets that has exceeded the threshold since the last time a log was recorded

Access control lists

An access control list (ACL) is a granular, targeted blocklist that is used to block IPv4 and IPv6 packets on a specified interface based on the criteria configured in the ACL policy.

On FortiGate models with ports that are connected through an internal switch fabric with TCAM capabilities, ACL processing is offloaded to the switch fabric and does not use CPU resources. VLAN interfaces that are based on physical switch fabric interfaces are also supported. Interfaces that are connected through an internal switch fabric usually have names prefixed with *port* or *lan*, such as *port1* or *lan2*; other interfaces are not supported.

The packets will be processed by the CPU when offloading is disabled or not possible, such as when a port on a supported model does not connect to the internal fabric switch.

ACL is supported on the following FortiGate models:

- 100D, 100E, 100EF, 101E
- 140D, 140D-POE, 140E, 140E-POE
- 1500D, 1500DT

- 3000D, 3100D, 3200D, 3700D
- All 300E and larger E-series models
- All 100F and larger F-series models

Example

To block all IPv4 and IPv6 telnet traffic from port2 to Company_Servers:

```
config firewall acl
    edit 1
       set interface "port2"
       set srcaddr "all"
       set dstaddr "Company_Servers"
       set service "TELNET"
    next
end
config firewall acl6
    edit 1
        set interface "port2"
        set srcaddr "all"
        set dstaddr "Company_Servers_v6"
        set service "TELNET"
    next
end
```

Diagnose commands

To check the number of packets dropped by an ACL:

```
# diagnose firewall acl counter
ACL id 1 dropped 0 packets
```

```
# diagnose firewall acl counter6
ACL id 2 dropped 0 packets
```

To clear the packet drop counters:

```
# diagnose firewall acl clearcounter
```

```
# diagnose firewall acl clearcounter6
```

Interface policies

Interface policies are implemented before the security policies and are only flow-based. They are configured in the CLI.

This feature allows you to attach a set of IPS policies with the interface instead of the forwarding path, so packets can be delivered to IPS before entering the firewall. This feature is used for following IPS deployments:

- One-Arm: By defining interface policies with IPS and DoS anomaly checks and enabling sniff-mode on the interface, the interface can be used for one-arm IDS.
- IPv6 IPS: IPS inspection can be enabled through interface IPv6 policy.

- Scan traffic that is destined to the FortiGate.
- Scan and log traffic that are silently dropped or flooded by Firewall or Multicast traffic.

IPS sensors can be assigned to an interface policy. Both incoming and outgoing packets are inspected by IPS sensor (signature).

To configure an interface policy:

```
config firewall interface-policy
    edit 1
        set status enable
        set comments 'test interface policy #1'
        set logtraffic utm
        set interface "port2"
        set srcaddr all
        set dstaddr all
        set service "ALL"
        set application-list-status disable
        set ips-sensor-status disable
        set dsri disable
        set av-profile-status enable
        set av-profile default
        set webfilter-profile-status disable
    next
end
```

Source NAT

The following topics provide instructions on configuring policies with source NAT:

- Static SNAT on page 799
- Dynamic SNAT on page 800
- Central SNAT on page 806
- Configuring an IPv6 SNAT policy on page 811
- SNAT policies with virtual wire pairs on page 813

Static SNAT

Network Address Translation (NAT) is the process that enables a single device such as a router or firewall to act as an agent between the Internet or Public Network and a local or private network. This agent acts in real time to translate the source or destination IP address of a client or server on the network interface. For the source IP translation, this enables a single public address to represent a significantly larger number of private addresses. For the destination IP translation, the firewall can translate a public destination address to a private address. So we don't have to configure a real public IP address for the server deployed in a private network.

We can subdivide NAT into two types: source NAT (SNAT) and destination NAT (DNAT). This topic is about SNAT, We support three NAT working modes: static SNAT, dynamic SNAT, and central SNAT.

In static SNAT all internal IP addresses are always mapped to the same public IP address. This is a port address translation, Since we have 60416 available port numbers, this one public IP address can handle the conversion of 60,416 internal IP addresses.

Internal Source IP	Source Port	Translated Source IP	Translated Source Port
10.1.100.1	11110	172.16.200.1	5117
10.1.100.1	11111	172.16.200.1	5118
10.1.100.2	11112	172.16.200.1	5119
......	......	172.16.200.1	......
......	......	172.16.200.1	65533

FortiGate firewall configurations commonly use the Outgoing Interface address.

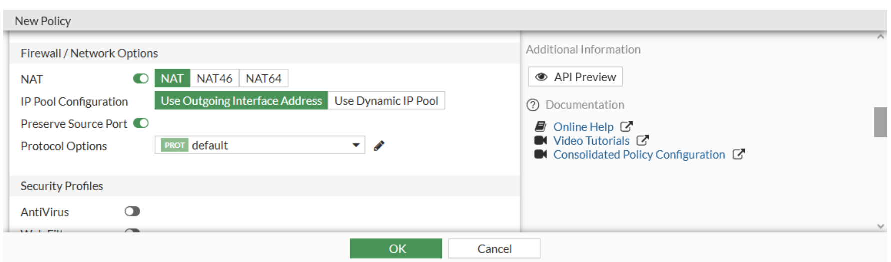
Sample configuration

The following example of static SNAT uses an internal network with subnet 10.1.100.0/24 (vlan20) and an external/ISP network with subnet 172.16.200.0/24 (vlan30).

When the clients in internal network need to access the servers in external network, We need to translate IP addresses from 10.1.100.0/24 to an IP address 172.16.200.0/24, In this example, we implement static SNAT by creating a firewall policy.

To configure static NAT:

1. Go to *Policy & Objects* > *Firewall Policy* and click *Create New*.
2. Configure the required policy parameters.
3. Enable *NAT* and select *Use Outgoing Interface Address*. For packets that match this policy, its source IP address is translated to the IP address of the outgoing interface.
4. If needed, enable *Preserve Source Port* to keep the same source port for services that expect traffic to come from a specific source port. Disable *Preserve Source Port* to allow more than one connection through the firewall for that service.



5. Click *OK*.

Dynamic SNAT

Dynamic SNAT maps the private IP addresses to the first available public address from a pool of addresses. In the FortiGate firewall, this can be done by using IP pools. IP pools is a mechanism that allows sessions leaving the FortiGate firewall to use NAT. An IP pool defines a single IP address or a range of IP addresses to be used as the source address for the duration of the session. These assigned addresses are used instead of the IP address assigned to that FortiGate interface.

IP pool types

FortiGate uses four types of IPv4 IP pools. This topic focuses on some of the differences between them.

Overload

This type of IP pool is similar to static SNAT mode. We need to define an external IP range that contains one or more IP addresses. When there is only one IP address it is almost the same as static SNAT, the outgoing interface address is used. When it contains multiple IP addresses, it is equivalent to an extended mode of static SNAT.

For instance, if we define an overload type IP pool with two external IP addresses (172.16.200.1—172.16.200.2), since there are 60,416 available port numbers per IP, this IP pool can handle 60,416*2 internal IP addresses.

Original Source IP	Original Source Port	Translated Source IP	Translated Source Port
10.1.100.1	11110	172.16.200.1	5117
10.1.100.2	11111	172.16.200.1	5118
......	......	172.16.200.1	......
......	......	172.16.200.1	65533
......	......	172.16.200.2	5117
......	......	......	......
......	......	172.16.200.2	65533

The mapped IP address can be calculated from the source IP address. The index number of the address in the pool is the remainder of the source IP address, in decimal, divided by the number addresses in the pool.

To calculate the decimal value of the source IP address, either use an online calculator, or use the following equation:

$$a.b.c.d = a * (256)^3 + b * (256)^2 + c * (256) + d$$

For example:

$$192.168.0.1 = 192 * (256)^3 + 168 * (256)^2 + 0 * (256) + 1 = 3232235521$$

If there is one IP pool, where:

- P_1 = the first address in the IP pool
- R_1 = the number of IP addresses in the IP pool
- X = the source IP address as a decimal number
- Y = the mapped IP address

Then the equation to determine the mapped address is:

$$Y = P_1 + X \bmod R_1$$

For example:

IP pool	Source IP address
172.26.73.20 to 172.26.73.90	192.168.1.200

1. Convert the source IP address to a decimal number:
$$192 * (256)^3 + 168 * (256)^2 + 1 * (256) + 200 = 3232235976$$
2. Determine the number of IP addresses in the pool:
$$172.26.73.90 - 172.26.73.20 = 71$$

3. Find the remainder of the source IP address divided by the number of addresses in the pool:
$$3232235976 \bmod 71 = 26$$
4. Add the remainder to the first IP address in the pool:
172.26.73.20 + 26 = 172.26.73.46

   So, the mapped IP address is **172.26.73.46**.

If there are multiple IP pools, the calculation is similar to when there is only one pool.

If there are two IP pools, where:

- P_1 = the first address in the first IP pool
- P_2 = the first address in the second IP pool
- R_1 = the number of IP addresses in the first IP pool
- R_2 = the number of IP addresses in the second IP pool
- X = the source IP address as a decimal number
- Y = the mapped IP address

Then the equations to determine the mapped address are:

$$\text{If } X \bmod (R_1 + R_2) >= R_1\text{, then } Y = P_2 + X \bmod R_2$$

$$\text{If } X \bmod (R_1 + R_2) < R_1\text{, then } Y = P_1 + X \bmod R_1$$

For example:

IP pools	Source IP address
pool01: 172.26.73.20 to 172.26.73.90	192.168.1.200
pool02: 172.26.75.50 to 172.26.75.150	

1. Convert the source IP address to a decimal number:
$$192 * (256)^3 + 168 * (256)^2 + 1 * (256) + 200 = 3232235976$$
2. Determine the total number of IP addresses in the pools:
(172.26.73.90 - 172.26.73.20) + (172.26.75.50 - 172.26.75.150) = 71 + 101 = 172
3. Find the remainder of the source IP address divided by the number of addresses in the pools:
$$3232235976 \bmod 172 = 108$$
4. The remainder is greater than the number of addresses in pool01, so the address is selected from pool02 and the remainder is recalculated based only on pool02:
$$3232235976 \bmod 101 = 40$$
5. Add the new remainder to the first IP address in pool02:
172.26.75.50 + 40 = 172.26.75.90

   So, the mapped IP address is **172.26.75.90**.

One-to-one

This type of IP pool means that the internal IP address and the external (translated) IP address match one-to-one. The port address translation (PAT) is disabled when using this type of IP pool. For example, if we define a one-to-one type IP pool with two external IP addresses (172.16.200.1 - 172.16.200.2), this IP pool only can handle two internal IP addresses.

Fixed port range

For the overload and one-to-one IP pool types, we do not need to define the internal IP range. For the fixed port range type of IP pool, we can define both internal IP range and external IP range. Since each external IP address and the number of available port numbers is a specific number, if the number of internal IP addresses is also determined, we can calculate the port range for each address translation combination. So we call this type fixed port range. This type of IP pool is a type of port address translation (PAT).

For instance, if we define one external IP address (172.16.200.1) and ten internal IP addresses (10.1.100.1-10.1.100.10), we have translation IP+Port combination like following table:

Original Source IP	Original Source Port	Translated Source IP	Translated Source Port Range
10.1.100.1	......	172.16.200.1	5117~11157
10.1.100.2	......	172.16.200.1	11158~17198
10.1.100.3	......	172.16.200.1	......
10.1.100.4	......	172.16.200.1	......
10.1.100.5	......	172.16.200.1	......
10.1.100.6	......	172.16.200.1	......
10.1.100.7	......	172.16.200.1	......
10.1.100.8	......	172.16.200.1	......
10.1.100.9	......	172.16.200.1	53445~59485
10.1.100.10	......	172.16.200.1	59486~65526

Port block allocation

This type of IP pool is also a type of port address translation (PAT). It gives users a more flexible way to control the way external IPs and ports are allocated. Users need to define *Block Size/Block Per User* and external IP range. *Block Size* means how many ports each Block contains. *Block per User* means how many blocks each user (internal IP) can use.

The following is a simple example:

- **External IP Range**: 172.16.200.1—172.16.200.1
- **Block Size**: 128
- **Block Per User**: 8

Result:

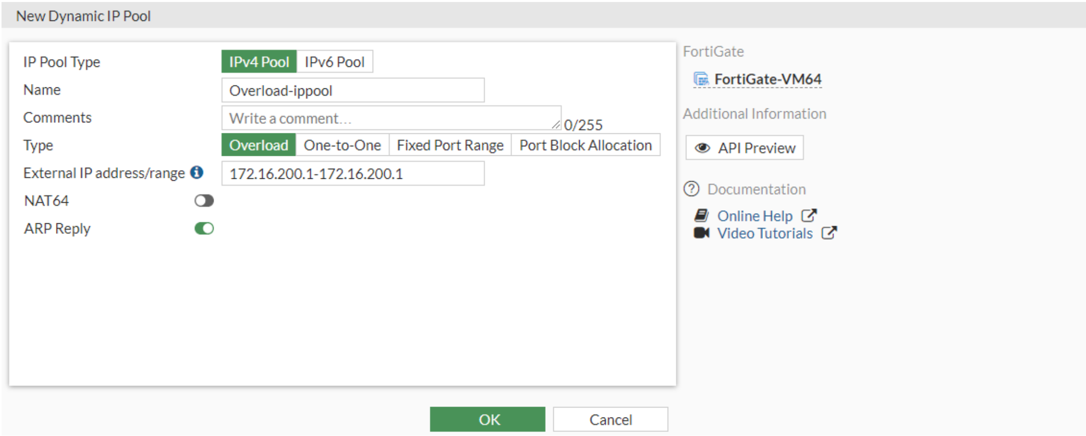
- **Total-PBAs**: 472 (60416/128)
- **Maximum ports can be used per User (Internal IP Address)**: 1024 (128*8)
- **How many Internal IP can be handled**: 59 (60416/1024 or 472/8)

Sample configuration

To configure overload IP pool in the GUI:

1. In *Policy & Objects > IP Pools*, click *Create New*.
2. Select *IPv4 Pool* and then select *Overload*.

3. Enter the external IP range separated by a hyphen (*172.16.200.1-172.16.200.1*).



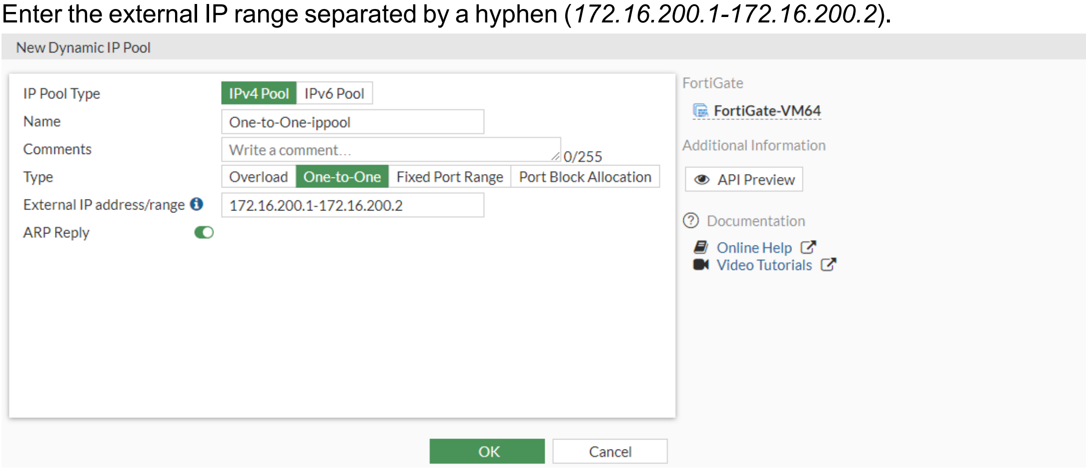
4. Click *OK*.

To configure overload IP pool in the CLI:

```
config firewall ippool
    edit "Overload-ippool"
        set startip 172.16.200.1
        set endip 172.16.200.1
    next
end
```

To configure one-to-one IP pool using the GUI:

1. In *Policy & Objects > IP Pools*, click *Create New*.
2. Select *IPv4 Pool* and then select *One-to-One*.
3. Enter the external IP range separated by a hyphen (*172.16.200.1-172.16.200.2*).



4. Click *OK*.

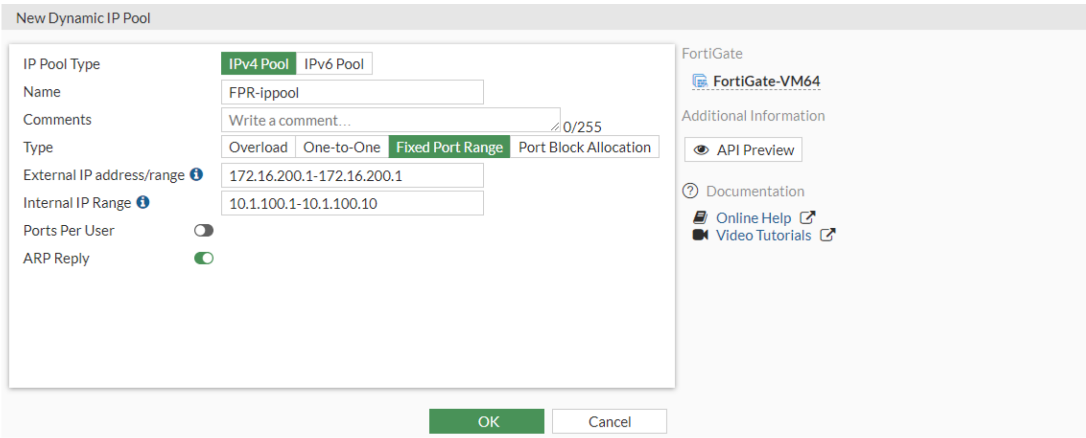
To configure one-to-one IP pool in the CLI:

```
config firewall ippool
    edit "One-to-One-ippool"
        set type one-to-one
```

```
        set startip 172.16.200.1
        set endip 172.16.200.2
    next
end
```

To configure fixed port range IP pool in the GUI:

1. In *Policy & Objects* > *IP Pools*, click *Create New*.
2. Select *IPv4 Pool* and then select *Fixed Port Range*.
3. Enter the external IP range separated by a hyphen *172.16.200.1-172.16.200.1*).
4. Enter the internal IP range separated by a hyphen *10.1.100.1-10.1.100.10*).



5. Click *OK*.

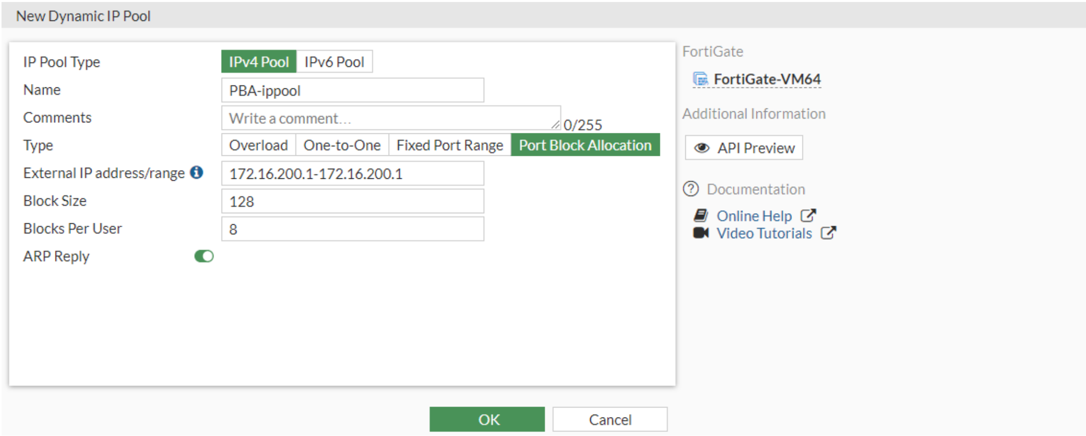
To configure fixed port range IP pool in the CLI:

```
config firewall ippool
    edit "FPR-ippool"
        set type fixed-port-range
        set startip 172.16.200.1
        set endip 172.16.200.1
        set source-startip 10.1.100.1
        set source-endip 10.1.100.10
   next
end
```

To configure port block allocation IP pool in the GUI:

1. In *Policy & Objects* > *IP Pools*, click *Create New*.
2. Select *IPv4 Pool* and then select *Port Block Allocation*.

3. Enter the external IP range separated by a hyphen *172.16.200.1-172.16.200.1*).



4. Click *OK*.

To configure port block allocation IP pool in the CLI:

```
config firewall ippool
    edit PBA-ippool
        set type port-block-allocation
        set startip 172.16.200.1
        set endip 172.16.200.1
        set block-size 128
        set num-blocks-per-user 8
    next
end
```

Central SNAT

The central SNAT table enables you to define and control (with more granularity) the address translation performed by FortiGate. With the NAT table, you can define the rules for the source address or address group, and which IP pool the destination address uses.

FortiGate reads the NAT rules from the top down until it hits a matching rule for the incoming address. This enables you to create multiple NAT policies that dictate which IP pool is used based on source address, destination address, and source port. NAT policies can be rearranged within the policy list. NAT policies are applied to network traffic after a security policy.

The central SNAT table allows you to create, edit, delete, and clone central SNAT entries.

Central SNAT notes

- The central NAT feature is not enabled by default.
- If central NAT is enabled, the NAT option under IPv4 policies is skipped and SNAT must be done via `central-snat-map`. The firewall policy list and dialog boxes have messages and redirection links to show this information.
- If NGFW mode is policy-based, then it is assumed that central NAT (specifically SNAT) is enabled implicitly.

Sample configuration

To enable central SNAT from the GUI:

1. In *System* > *Settings*, under *System Operations Settings*, enable *Central SNAT*.
2. Click *Apply*.

To enable or disable central SNAT using the CLI:

```
config system settings
   set central-nat {enable | disable}
end
```

When central NAT is enabled, *Policy & Objects* displays the Central SNAT section.

The Central SNAT policy has many options:

Field	Description
Type	Specify whether you are performing SNAT on IPv4 or IPv6. This option only appears when IPv6 is enabled under Feature Visibility.
Incoming Interface	Specify one or more interfaces for the ingress traffic.
Outgoing Interface	Specify one or more interfaces for the egress traffic.
Source Address	Specify the address or address group of the source.
Destination Address	Specify the address or address group of the destination.
NAT	Enable or disable to perform NAT. When disabled, no source address translation will occur.
IP Pool Configuration	Use outgoing interface address: • Use the address of the outgoing interfaces as source address. Use Dynamic IP Pool: • Choose an IP Pool to perform source NAT.
Protocol	Choose from any, TCP, UDP, SCTP, or specify the protocol number to match. For example, for ICMP, click *specify* with the protocol number 1.
Explicit port mapping	Enable in order to match this NAT policy only when the following ports are a match: • Choose an original source port from one to 65535. NAT'd port will be chosen by the FortiGate based on the IP Pool configuration. Explicit port mapping cannot apply to some protocols which do not use ports, such as ICMP. When enabling a NAT policy which uses Explicit port mapping, always consider that ICMP traffic will not match this policy. When using IP Pools, only the Overload type IP Pool allows Explicit port mapping. When Explicit port mapping is applied, you must define an original source port range and a translated sort port range. The source port will map one to one with the translated port. Refer to Dynamic SNAT to understand how each IP Pool type works.

Field	Description
Comments	Enter comments for this NAT policy.
Enable this policy	Enable or disable this policy.

To configure central SNAT using the CLI:

```
config firewall central-snat-map
edit <policyID number>
  set status {enable|disable}
  set orig-addr <valid address object preconfigured on the FortiGate>
  set srcintf <name of interface on the FortiGate>
  set dst-addr <valid address object preconfigured on the FortiGate>
  set dstintf <name of interface on the FortiGate>
  set protocol <integer for protocol number>
  set orig-port <integer for original port number>
  set nat-port <integer for translated port number>
  set comments <string>
end
```

Example one

Apply SNAT to all traffic from port2 to port3.

To configure from the CLI:

```
config firewall central-snat-map
    edit 1
        set srcintf "port3"
        set dstintf "port2"
        set orig-addr "all"
        set dst-addr "all"
    next
end
```

Example two

Apply an IP Pool to all traffic from port3 to port2 that are TCP. NAT all other traffic using the outgoing interface IP.

To configure from the CLI:

```
config firewall ippool
    edit "Overload-IPPOOL"
        set startip 192.168.2.201
        set endip 192.168.2.202
    next
end
config firewall central-snat-map
    edit 1
        set srcintf "port3"
        set dstintf "port2"
        set orig-addr "all"
        set dst-addr "all"
```

```
        set protocol 6
        set nat-ippool "Overload-IPPOOL"
    next
    edit 2
        set srcintf "port3"
        set dstintf "port2"
        set orig-addr "all"
        set dst-addr "all"
    next
end
```

To collect session table output from the CLI:

```
diagnose sys session list
```

The TCP session (protocol 6) is NAT'd with Overload-IPPOOL to 192.168.2.201:

```
session info: proto=6 proto_state=05 duration=14 expire=0 timeout=3600 flags=00000000
socktype=0 sockport=0 av_idx=0 use=3
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/255
state=may_dirty
statistic(bytes/packets/allow_err): org=860/7/1 reply=555/8/1 tuples=2
tx speed(Bps/kbps): 60/0 rx speed(Bps/kbps): 38/0
orgin->sink: org pre->post, reply pre->post dev=9->6/6->9 gwy=192.168.2.1/192.168.0.10
hook=post dir=org act=snat 192.168.0.10:49531->23.57.57.114:443(192.168.2.201:61776)
hook=pre dir=reply act=dnat 23.57.57.114:443->192.168.2.201:61776(192.168.0.10:49531)
pos/(before,after) 0/(0,0), 0/(0,0)
dst_mac=04:d5:90:5f:a2:2a
misc=0 policy_id=2 auth_info=0 chk_client_info=0 vd=0
serial=00011065 tos=ff/ff app_list=0 app=0 url_cat=0
sdwan_mbr_seq=0 sdwan_service_id=0
rpdb_link_id=00000000 rpdb_svc_id=0 ngfwid=n/a
npu_state=0x040000
```

A UDP session (protocol 17) is NAT'd to the outgoing interface IP address 192.168.2.86:

```
session info: proto=17 proto_state=01 duration=16 expire=163 timeout=0 flags=00000000
socktype=0 sockport=0 av_idx=0 use=3
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=/ helper=dns-udp vlan_cos=0/255
state=may_dirty
statistic(bytes/packets/allow_err): org=59/1/1 reply=187/1/1 tuples=2
tx speed(Bps/kbps): 3/0 rx speed(Bps/kbps): 11/0
orgin->sink: org pre->post, reply pre->post dev=9->6/6->9 gwy=192.168.2.1/192.168.0.10
hook=post dir=org act=snat 192.168.0.10:52177->4.2.2.1:53(192.168.2.86:61770)
hook=pre dir=reply act=dnat 4.2.2.1:53->192.168.2.86:61770(192.168.0.10:52177)
dst_mac=04:d5:90:5f:a2:2a
misc=0 policy_id=2 auth_info=0 chk_client_info=0 vd=0
serial=00011061 tos=ff/ff app_list=0 app=0 url_cat=0
sdwan_mbr_seq=0 sdwan_service_id=0
rpdb_link_id=00000000 rpdb_svc_id=0 ngfwid=n/a
npu_state=0x040000
```

Example three

Apply an IP Pool to all traffic from port3 to port2 that have a specific original port range, mapping the ports to the same NAT'd port range. Nat all other traffic using the outgoing interface IP.

To configure from the CLI:

```
config firewall central-snat-map
    edit 1
        set srcintf "port3"
        set dstintf "port2"
        set orig-addr "all"
        set dst-addr "all"
        set orig-port 50000-65535
        set nat-ippool "Overload-IPPOOL"
        set nat-port 50000-65535
    next
    edit 2
        set srcintf "port3"
        set dstintf "port2"
        set orig-addr "all"
        set dst-addr "all"
    next
end
```

To collect session table output from the CLI:

```
diagnose sys session list
```

Traffic with original port in the range between 50000-65535 will be NAT'd with the Overload type IP Pool. The mapped port is in the same port range:

```
session info: proto=17 proto_state=01 duration=3 expire=176 timeout=0 flags=00000000
socktype=0 sockport=0 av_idx=0 use=3
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=/ helper=dns-udp vlan_cos=0/255
state=may_dirty
statistic(bytes/packets/allow_err): org=71/1/1 reply=123/1/1 tuples=2
tx speed(Bps/kbps): 23/0 rx speed(Bps/kbps): 40/0
orgin->sink: org pre->post, reply pre->post dev=9->6/6->9 gwy=192.168.2.1/192.168.0.10
hook=post dir=org act=snat 192.168.0.10:52540->4.2.2.1:53(192.168.2.201:52540)
hook=pre dir=reply act=dnat 4.2.2.1:53->192.168.2.201:52540(192.168.0.10:52540)
dst_mac=04:d5:90:5f:a2:2a
misc=0 policy_id=2 auth_info=0 chk_client_info=0 vd=0
serial=00011399 tos=ff/ff app_list=0 app=0 url_cat=0
sdwan_mbr_seq=0 sdwan_service_id=0
rpdb_link_id=00000000 rpdb_svc_id=0 ngfwid=n/a
npu_state=0x040000
```

Traffic with original port outside the range of 50000-65535 will be NAT'd to the outgoing interface IP:

```
session info: proto=6 proto_state=01 duration=3 expire=3597 timeout=3600 flags=00000000
socktype=0 sockport=0 av_idx=0 use=3
origin-shaper=
reply-shaper=
```

```
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/255
state=may_dirty
statistic(bytes/packets/allow_err): org=2262/10/1 reply=2526/11/1 tuples=2
tx speed(Bps/kbps): 741/5 rx speed(Bps/kbps): 828/6
orgin->sink: org pre->post, reply pre->post dev=9->6/6->9 gwy=192.168.2.1/192.168.0.10
hook=post dir=org act=snat 192.168.0.10:49805->142.250.68.66:443(192.168.2.86:62214)
hook=pre dir=reply act=dnat 142.250.68.66:443->192.168.2.86:62214(192.168.0.10:49805)
pos/(before,after) 0/(0,0), 0/(0,0)
dst_mac=04:d5:90:5f:a2:2a
misc=0 policy_id=2 auth_info=0 chk_client_info=0 vd=0
serial=0001139a tos=ff/ff app_list=0 app=0 url_cat=0
sdwan_mbr_seq=0 sdwan_service_id=0
rpdb_link_id=00000000 rpdb_svc_id=0 ngfwid=n/a
npu_state=0x040000
```

Protocols which do not use ports, such as ICMP, will be NAT'd to the outgoing interface IP:

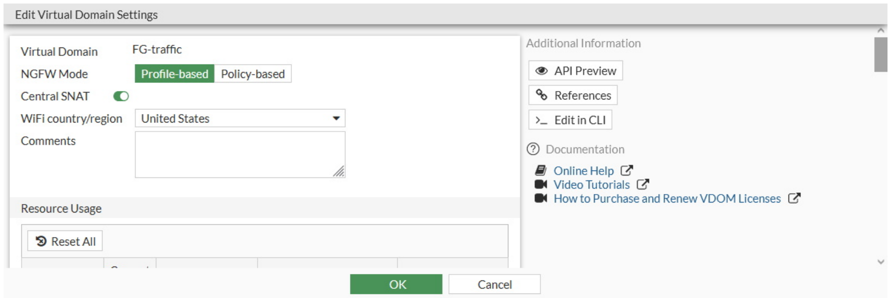
```
session info: proto=1 proto_state=00 duration=7 expire=59 timeout=0 flags=00000000
socktype=0 sockport=0 av_idx=0 use=3
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/255
state=may_dirty
statistic(bytes/packets/allow_err): org=480/8/1 reply=480/8/1 tuples=2
tx speed(Bps/kbps): 66/0 rx speed(Bps/kbps): 66/0
orgin->sink: org pre->post, reply pre->post dev=9->6/6->9 gwy=192.168.2.1/192.168.0.10
hook=post dir=org act=snat 192.168.0.10:1->4.2.2.1:8(192.168.2.86:62209)
hook=pre dir=reply act=dnat 4.2.2.1:62209->192.168.2.86:0(192.168.0.10:1)
dst_mac=04:d5:90:5f:a2:2a
misc=0 policy_id=2 auth_info=0 chk_client_info=0 vd=0
serial=0001138b tos=ff/ff app_list=0 app=0 url_cat=0
sdwan_mbr_seq=0 sdwan_service_id=0
rpdb_link_id=00000000 rpdb_svc_id=0 ngfwid=n/a
npu_state=0x040000
```

Configuring an IPv6 SNAT policy

IPv4 and IPv6 central SNAT maps are displayed in the same table.

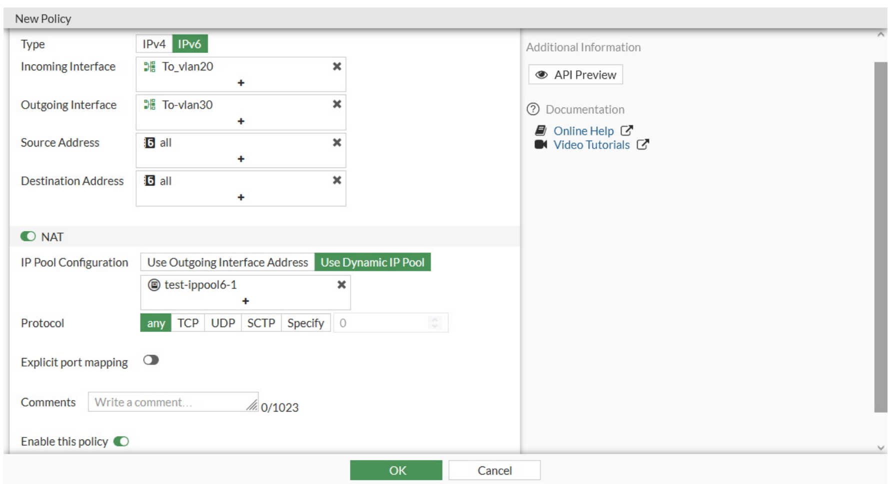
To configure an IPv6 policy with central SNAT in the GUI:

1. Enable central SNAT:
   a. In the Global VDOM, go to *System* > *VDOM*.
   b. Select a VDOM and click *Edit*. The *Edit Virtual Domain Settings* pane opens.
   c. Enable *Central SNAT*.



d. Click *OK*.

2. In the VDOM with central SNAT enabled (FG-traffic in this example), go to *Policy & Objects > Central SNAT* and click *Create New*.
3. Configure the policy settings:
   a. For *Type*, select *IPv6*.
   b. Enter the interface, address, and IP pool information.
   c. Configure the remaining settings as needed.



   d. Click *OK*.
   The matching SNAT traffic will be handled by the IPv6 central SNAT map.

To configure an IPv6 policy with central SNAT in the CLI:

1. Enable central SNAT:

```
config vdom
    edit FG-traffic
        config system settings
            set central-nat enable
        end
    next
end
```

2. Create an IPv6 central SNAT policy:

```
config vdom
    edit FG-traffic
        config firewall central-snat-map
```

```
            edit 2
                set type ipv6
                set srcintf "wan2"
                set dstintf "wan1"
                set orig-addr6 "all"
                set dst-addr6 "all"
                set nat-ippool6 "test-ippool6-1"
            next
        end
    next
end
```

3. Verify the SNAT traffic:

```
(FG-traffic) # diagnose sniffer packet any icmp6 4
interfaces=[any]
filters=[icmp6]
3.602891 wan2 in 2000:10:1:100::41 -> 2000:172:16:200::55: icmp6: echo request seq 0
3.602942 wan1 out 2000:172:16:200::199 -> 2000:172:16:200::55: icmp6: echo request seq 0
3.603236 wan1 in 2000:172:16:200::55 -> 2000:172:16:200::199: icmp6: echo reply seq 0
3.603249 wan2 out 2000:172:16:200::55 -> 2000:10:1:100::41: icmp6: echo reply seq 0
4.602559 wan2 in 2000:10:1:100::41 -> 2000:172:16:200::55: icmp6: echo request seq 1
4.602575 wan1 out 2000:172:16:200::199 -> 2000:172:16:200::55: icmp6: echo request seq 1
4.602956 wan1 in 2000:172:16:200::55 -> 2000:172:16:200::199: icmp6: echo reply seq 1
4.602964 wan2 out 2000:172:16:200::55 -> 2000:10:1:100::41: icmp6: echo reply seq 1
^C
8 packets received by filter
0 packets dropped by kernel
```

SNAT policies with virtual wire pairs

Source NAT (SNAT) can be configured in IPv4 and IPv6 policies with virtual wire pair (VWP) interfaces, and between VWP interfaces when central NAT is enabled.

To configure a policy using SNAT and a VWP interface when central NAT is disabled:

1. Create the VWP interface:

```
config system virtual-wire-pair
    edit "test-vw-1"
        set member "port1" "port4"
    next
end
```

2. Create the IP pool. The IP pool must have a different subnet than the VWP peers.

```
config firewall ippool
    edit "vwp-pool-1"
        set startip 172.16.222.99
        set endip 172.16.222.100
    next
end
```

3. Configure the firewall policy:

```
config firewall policy
    edit 88
```

```
        set srcintf "port4"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set logtraffic all
        set nat enable
        set ippool enable
        set poolname "vwp-pool-1"
    next
end
```

4. Verify the IP pool functions as expected and traffic passes through:

```
# diagnose sniffer packet any icmp 4
interfaces=[any]
filters=[icmp]
23.438095 port4 in 172.16.200.11 -> 172.16.200.156: icmp: echo request
23.438126 port1 out 172.16.222.100 -> 172.16.200.156: icmp: echo request
23.438492 port1 in 172.16.200.156 -> 172.16.222.100: icmp: echo reply
23.438501 port4 out 172.16.200.156 -> 172.16.200.11: icmp: echo reply
24.439305 port4 in 172.16.200.11 -> 172.16.200.156: icmp: echo request
24.439319 port1 out 172.16.222.100 -> 172.16.200.156: icmp: echo request
24.439684 port1 in 172.16.200.156 -> 172.16.222.100: icmp: echo reply
24.439692 port4 out 172.16.200.156 -> 172.16.200.11: icmp: echo reply

8 packets received by filter
0 packets dropped by kernel
```

To configure a SNAT between VWP interfaces when central NAT is enabled:

1. Enable central NAT:

```
config system settings
    set central-nat enable
end
```

2. Create the VWP interface:

```
config system virtual-wire-pair
    edit "test-vw-1"
        set member "port1" "port4"
    next
end
```

3. Create the IP pool. The IP pool must have a different subnet than the VWP peers.

```
config firewall ippool
    edit "vwp-pool-1"
        set startip 172.16.222.99
        set endip 172.16.222.100
    next
end
```

4. Configure the SNAT policy:

```
config firewall central-snat-map
    edit 2
        set srcintf "port4"
        set dstintf "port1"
        set orig-addr "all"
        set dst-addr "all"
        set nat-ippool "vwp-pool-1"
    next
end
```

5. Configure the firewall policy:

```
config firewall policy
    edit 90
        set srcintf "port4"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set logtraffic all
    next
end
```

Destination NAT

The following topics provide instructions on configuring policies with destination NAT:

- Static virtual IPs on page 815
- Virtual IP with services on page 818
- Virtual IPs with port forwarding on page 819
- Virtual server load balance on page 821

Static virtual IPs

Static Virtual IPs (VIP) are used to map external IP addresses to internal IP addresses. This is also called destination NAT, where a packet's destination is being NAT'd, or mapped, to a different address.

Static VIPs are commonly used to map public IP addresses to resources behind the FortiGate that use private IP addresses. A static one-to-one VIP is when the entire port range is mapped. A port forwarding VIP is when the mapping is configured on a specific port or port range.

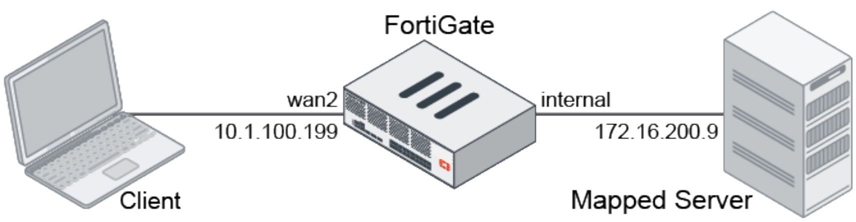
Some of the VIP configuration options are:

Setting	Description
VIP Type	• IPv4 (`config firewall vip`) - The source and destination are both IPv4. • IPv6 (`config firewall vip6`) - The source and destination are both IPv6. **Note:** *IPv6* is only available when *IPv6* is enabled in the *Feature Visibility*.

Setting	Description
Interface (`extintf`)	The external interface that the firewall policy source interface must match. For example, if the external interface is port1, then the VIP can be used in a policy from port1 to port3, but not in a policy from port2 to port3. If the external interface is *any*, then the VIP can be used in any firewall policy.
Type (`type`)	• Static NAT - Use an external IP address or address range. • FQDN - Use an external IP or FQDN address. • load-balance (CLI only) - Load balance traffic. • server-load-balance - Load balance traffic across multiple servers. SSL processing can be offloaded to the FortiGate. This type of VIP is configure from *Policy & Objects > Virtual Servers*. • dns-translation (CLI only) - DNS translation. • access-proxy - Used for ZTNA. See ZTNA HTTPS access proxy example on page 988 for details.
External IP address/range (`extip`)	In a static NAT VIP, the external IP address is the IP address that the FortiGate listens for traffic on. When the external interface is not*any*, 0.0.0.0 can be used to make the external IP address equivalent to the external interface's IP address. The external IP address is also used to perform SNAT for the mapped server when the server outbound traffic with a destination interface that matches the external interface. The firewall policy must also have NAT enabled.
IPv4 address/range (`mappedip`)	The IPv4 address or range that the internal resource is being mapped to.
IPv6 address/range (`ipv6-mappedip`)	The IPv6 address or range that the internal resource is being mapped to.
`srcintf-filter` (CLI only)	Listen for traffic to the external IP address only on the specified interface. While the external interface restricts the policies where the VIP can be used, it does not restrict listening to only the external interface. To restrict listening to only a specific interface, `srcint-filter` must be configured.
`nat-source-vip` (CLI only)	Force all of the traffic from the mapped server to perform SNAT with the external IP address, regardless of the destination interface. If `srcint-filter` is defined, then `nat-source-vip` only forces SNAT to be performed when the destination matches the `srcintf-filter` interface. In both cases, the firewall policy must have NAT enabled.
`arp-reply` (CLI only)	Enable/disable responding to ARP requests on the external IP address (default = enable).
Source address (`src-filter`)	Restrict the source IP address, address range, or subnet that is allowed to access the VIP.
Services (`service`)	Set the services that are allowed to be mapped.
Port Forwarding (`portforward`)	Enable port forwarding to specify the port (`mappedport`) to map to.

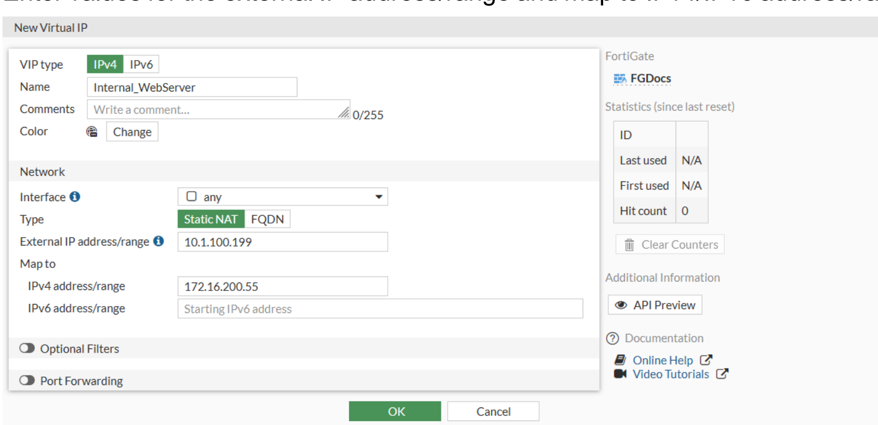
Setting	Description
	If no services are configured, you can configure the protocol (`protocol`) to use when forwarding packets, the external service port range (`extport`) to be mapped to a port range on the destination network, and the mapped port range (`mappedport` and `ipv6-mappedport`) on the destination network.
Port Mapping Type	• One to one - Each external service port is mapped to one port. A range is allowed, but the number of ports should be the same. • Many to Many - The port mapping can be one to one, one to many, or many to one. There are no restrictions on how many external ports must map to internal ports.

Sample configuration



To create a virtual IP in the GUI:

1. In *Policy & Objects* > *Virtual IPs* and click *Create New* > *Virtual IP*.
2. Select a *VIP Type* based on the IP versions used.
3. Enter a unique name for the virtual IP.
4. Enter values for the external IP address/range and map to IPv4/IPv6 address/range fields.



5. Click *OK*.

To create a virtual IP in the CLI:

```
config firewall vip
    edit "Internal_WebServer"
        set extip 10.1.100.199
        set extintf "any"
        set mappedip "172.16.200.55"
```

```
    next
end
```

To apply a virtual IP to policy in the CLI:

```
config firewall policy
    edit 8
        set name "Example_Virtual_IP_in_Policy"
        set srcintf "wan2"
        set dstintf "internal"
        set srcaddr "all"
        set dstaddr "Internal_WebServer"
        set action accept
        set schedule "always"
        set service "ALL"
        set nat enable
    next
end
```

Virtual IP with services

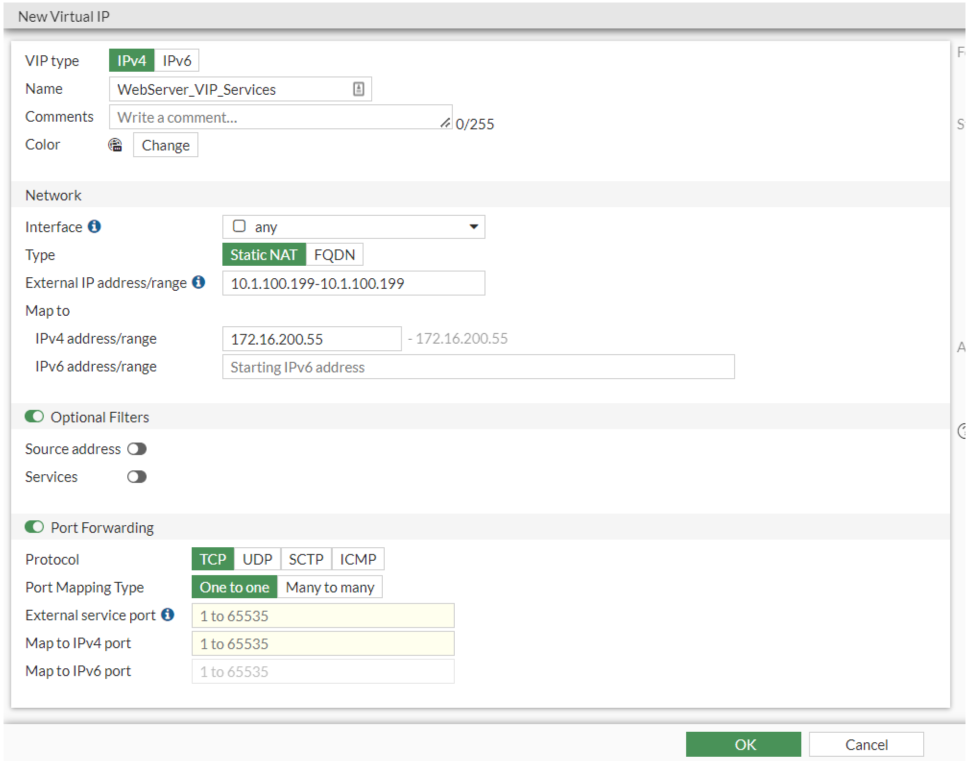
Virtual IP with services is a more flexible virtual IP mode. This mode allows users to define services to a single port number mapping.

This topic shows how to use virtual IP with services enabled. This example has one public external IP address. We map TCP ports 8080, 8081, and 8082 to an internal WebServer TCP port 80. This allows remote connections to communicate with a server behind the firewall.

Sample configuration

To create a virtual IP with services in the GUI:

1. In *Policy & Objects* > *Virtual IPs* and click *Create New* > *Virtual IP*.
2. Set *VIP Type* to *IPv4*.
3. Enter a unique name for the virtual IP and fill in the other fields.
4. Configure the fields in the *Network* section. For example:
   - Set *Interface* to *any*.
   - Set *External IP Address/Range* to *10.1.100.199*.
   - Set *Mapped IP Address/Range* to *172.16.200.55*.
5. Enable *Optional Filters* and then enable *Services*.
6. In the *Services* field click + to display the *Services* pane.
7. In the *Services* pane select *TCP_8080*, *TCP_8081*, and *TCP_8082*.
8. Enable *Port Forwarding* and set *Map to IPv4 port* to *80*.



9. Click *OK*.

To see the results:

1. Apply the above virtual IP to the firewall policy.
2. The results are:
   - Access 10.1.100.199:8080 from external network and FortiGate maps to 172.16.200.55:80 in internal network.
   - Access 10.1.100.199:8081 from external network and FortiGate maps to 172.16.200.55:80 in internal network.
   - Access 10.1.100.199:8082 from external network and FortiGate maps to 172.16.200.55:80 in internal network.

To create a virtual IP with services in the CLI:

```
config firewall vip
    edit "WebServer_VIP_Services"
        set service "TCP_8080" "TCP_8081" "TCP_8082"
        set extip 10.1.100.199
        set extintf "any"
        set portforward enable
        set mappedip "172.16.200.55"
        set mappedport 80
    next
end
```

Virtual IPs with port forwarding

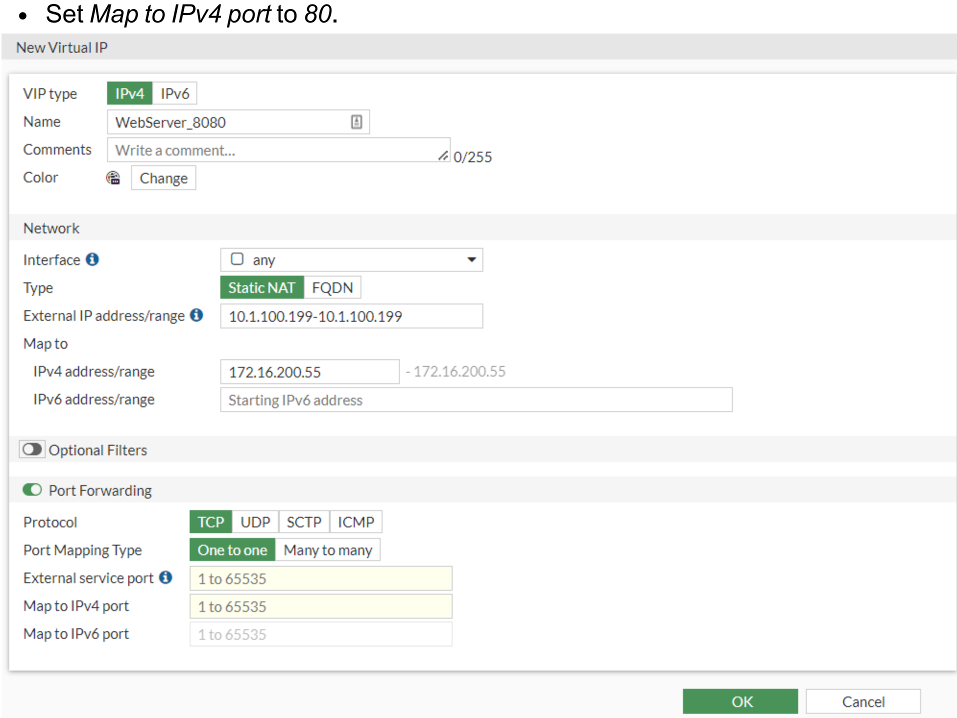
If you need to hide the internal server port number or need to map several internal servers to the same public IP address, enable port-forwarding for Virtual IP.

This topic shows how to use virtual IPs to configure port forwarding on a FortiGate unit. This example has one public external IP address. We map TCP ports 8080, 8081, and 8082 to different internal WebServers' TCP port 80. This allows remote connections to communicate with a server behind the firewall.

Sample configuration

To create a virtual IP with port forwarding in the GUI:

1. In *Policy & Objects > Virtual IPs*.
2. Click *Create New* and select *Virtual IP*.
3. For *VIP Type*, select *IPv4*.
4. Enter a unique name for the virtual IP and fill in the other fields.
5. Configure the fields in the *Network* section. For example:
   - Set *Interface* to *any*.
   - Set *External IP Address/Range* to *10.1.100.199*.
   - Set *Mapped IP Address/Range* to *172.16.200.55*.
6. Leave *Optional Filters* disabled.
7. Enable *Port Forwarding*.
8. Configure the fields in the *Port Forwarding* section. For example:
   - Set *Protocol* to *TCP*.
   - Set *External Service Port* to *8080*.
   - Set *Map to IPv4 port* to *80*.

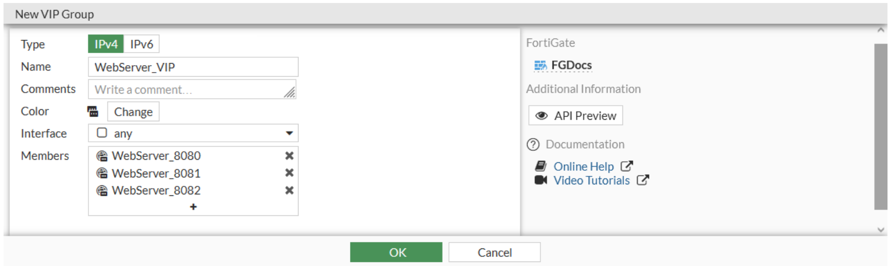


9. Click *OK*.
10. Follow the above steps to create two additional virtual IPs.
    a. For one virtual IP:
       - Use a different *Mapped IP Address/Range*, for example, *172.16.200.56*.
       - Set *External Service Port* to *8081*.

- Use the same *Map to IPv4 port* number: *80*.

b. For the other virtual IP:
   - Use a different *Mapped IP Address/Range*, for example, *172.16.200.57*.
   - Set *External Service Port* to *8082*.
   - Use the same *Map to IPv4 port* number: *80*.

11. Create a *Virtual IP Group* and put the above three virtual IPs into that group.



To see the results:

1. Apply the above virtual IP to the Firewall policy.
2. The results are:
   - Access 10.1.100.199:8080 from external network and FortiGate maps to 172.16.200.55:80 in internal network.
   - Access 10.1.100.199:8081 from external network and FortiGate maps to 172.16.200.56:80 in internal network.
   - Access 10.1.100.199:8082 from external network and FortiGate maps to 172.16.200.57:80 in internal network

Virtual server load balance

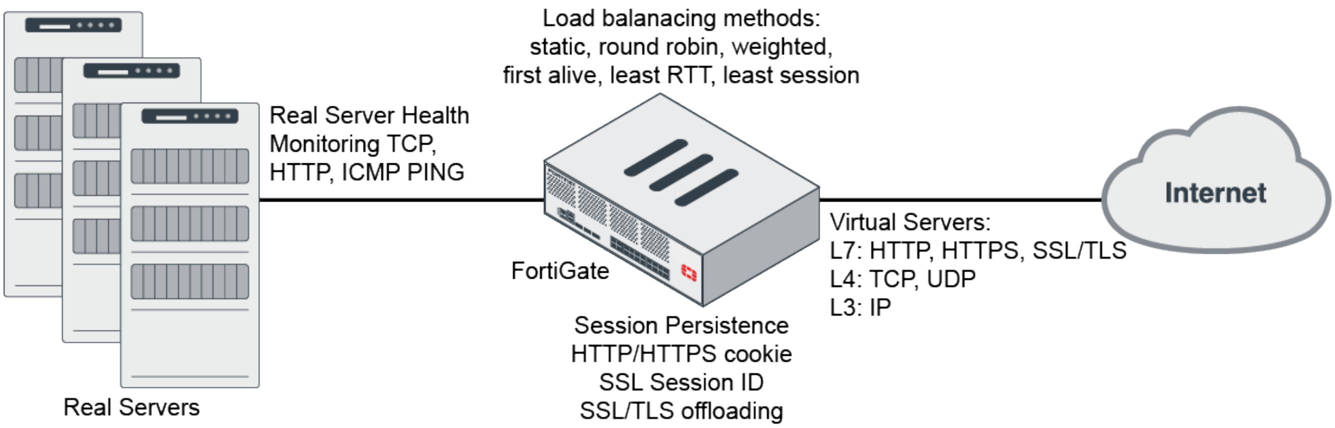
This topic shows a special virtual IP type: virtual server. Use this type of VIP to implement server load balancing.

The FortiOS server load balancing contains all the features of a server load balancing solution. You can balance traffic across multiple backend servers based on multiple load balancing schedules including:

- Static (failover)
- Round robin
- Weighted (to account for different sized servers or based on the health and performance of the server including round trip time and number of connections)

The load balancer supports HTTP, HTTPS, IMAPS, POP3S, SMTPS, SSL/TLS, and generic TCP/UDP and IP protocols. Session persistence is supported based on the SSL session ID based on an injected HTTP cookie, or based on the HTTP or HTTPS host. SSL/TLS load balancing includes protection from protocol downgrade attacks. Server load balancing is supported on most FortiGate devices and includes up to 10,000 virtual servers on high end systems.

Sample topology



SSL/TLS offloading

FortiGate SSL/TLS offloading is designed for the proliferation of SSL/TLS applications. The key exchange and encryption/decryption tasks are offloaded to the FortiGate unit where they are accelerated using FortiASIC technology which provides significantly more performance than a standard server or load balancer. This frees up valuable resources on the server farm to give better response to business operations. Server load balancing offloads most SSL/TLS versions including SSL 3.0, TLS 1.0, and TLS 1.2, and supports full mode or half mode SSL offloading with DH key sizes up to 4096 bits.

FortiGate SSL offloading allows the application payload to be inspected before it reaches your servers. This prevents intrusion attempts, blocks viruses, stops unwanted applications, and prevents data leakage. SSL/TLS content inspection supports TLS versions 1.0, 1.1, and 1.2 and SSL versions 1.0, 1.1, 1.2, and 3.0.

Virtual server requirements

When creating a new virtual server, you must configure the following options:

- Virtual Server Type.
- Load Balancing Methods.
- Health check monitoring (optional).
- Session persistence (optional).
- Virtual Server IP (External IP Address).
- Virtual Server Port (External Port).
- Real Servers (Mapped IP Address & Port).

Virtual server types

Select the protocol to be load balanced by the virtual server. If you select a general protocol such as IP, TCP, or UDP, the virtual server load balances all IP, TCP, or UDP sessions. If you select specific protocols such as HTTP, HTTPS, or SSL, you can apply additional server load balancing features such as *Persistence* and *HTTP Multiplexing*.

HTTP	Select *HTTP* to load balance only HTTP sessions with the destination port number that matches the *Virtual Server Port* setting. Change *Virtual Server Port* to match the destination port of the sessions to be load balanced (usually port 80 for HTTP sessions). You can enable *HTTP Multiplexing*. You can also set *Persistence* to *HTTP Cookie* to enable cookie-based persistence.
HTTPS	Select *HTTPS* to load balance only HTTPS sessions with the destination port number that matches the *Virtual Server Port* setting. Change *Virtual Server Port* to match the destination port of the sessions to be load balanced (usually port 443 for HTTPS sessions). You can enable *HTTP Multiplexing*. You can also set *Persistence* to *HTTP Cookie* to enable cookie-based persistence, or you can set *Persistence* to *SSL Session ID*.
IMAPS	Select *IMAPS* to load balance only IMAPS sessions with the destination port number that matches the *Virtual Server Port* setting. Change *Virtual Server Port* to match the destination port of the sessions to be load balanced (usually port 993 for IMAPS sessions). You can also set *Persistence* to *SSL Session ID*.
POP3S	Select *POP3S* to load balance only POP3S sessions with the destination port number that matches the *Virtual Server Port* setting. Change *Virtual Server Port* to match the destination port of the sessions to be load balanced (usually port 995 for POP3S sessions). You can also set *Persistence* to *SSL Session ID*.
SMTPS	Select *SMTPS* to load balance only SMTPS sessions with the destination port number that matches the *Virtual Server Port* setting. Change *Virtual Server Port* to match the destination port of the sessions to be load balanced (usually port 465 for SMTPS sessions). You can also set *Persistence* to *SSL Session ID*.
SSL	Select *SSL* to load balance only SSL sessions with the destination port number that matches the *Virtual Server Port* setting. Change *Virtual Server Port* to match the destination port of the sessions to be load balanced. You can also set *Persistence* to *SSL Session ID*.
TCP	Select *TCP* to load balance only TCP sessions with the destination port number that matches the *Virtual Server Port* setting. Change *Virtual Server Port* to match the destination port of the sessions to be load balanced.
UDP	Select *UDP* to load balance only UDP sessions with the destination port number that matches the *Virtual Server Port* setting. Change *Virtual Server Port* to match the destination port of the sessions to be load balanced.
IP	Select *IP* to load balance all sessions accepted by the security policy that contains this virtual server.

Load balancing methods

The load balancing method defines how sessions are load balanced to real servers.

All load balancing methods do not send traffic to real servers that are down or not responding. FortiGate can only determine if a real server is not responding by using a health check monitor. You should always add at least one health check monitor to a virtual server or to real servers; otherwise load balancing might try to distribute sessions to real servers that are not functioning.

Static	The traffic load is statically spread evenly across all real servers. Sessions are not assigned according to how busy individual real servers are. This load balancing method provides some persistence because all sessions from the same source address always go to the same real server. Because the distribution is stateless, so if a real server is added, removed, or goes up or down, the distribution is changed and persistence might be lost.
Round Robin	Directs new requests to the next real server. This method treats all real servers as equals regardless of response time or the number of connections. This method does not direct requests to real servers that down or non responsive.
Weighted	Real servers with a higher weight value receive a larger percentage of connections. Set the real server weight when adding a real server.
Least Session	Directs requests to the real server that has the least number of current connections. This method works best in environments where the real servers or other equipment you are load balancing all have similar capabilities. This load balancing method uses the FortiGate session table to track the number of sessions being processed by each real server. The FortiGate unit cannot detect the number of sessions actually being processed by a real server.
Least RTT	Directs sessions to the real server with the lowest round trip time. The round trip time is determined by a ping health check monitor. The default is 0 if no ping health check monitors are added to the virtual server.
First Alive	Directs sessions to the first live real server. This load balancing schedule provides real server failover protection by sending all sessions to the first live real server. If a real server fails, all sessions are sent to the next live real server. Sessions are not distributed to all real servers so all sessions are processed by the first real server only.
HTTP Host	Load balances HTTP host connections across multiple real servers using the host’s HTTP header to guide the connection to the correct real server.

Health check monitoring

In the FortiGate GUI, you can configure health check monitoring so that the FortiGate unit can verify that real servers are able respond to network connection attempts. If a real server responds to connection attempts, the load balancer continues to send sessions to it. If a real server stops responding to connection attempts, the load balancer assumes that the server is down and does not send sessions to it. The health check monitor configuration determines how the load balancer tests real servers. You can use a single health check monitor for multiple load balancing configurations. You can configure TCP, HTTP, DNS, and ping health check monitors. You usually set the health check monitor to use the same protocol as the traffic being load balanced to it. For example, for an HTTP load balancing configuration, you would normally use an HTTP health check monitor.

Session persistence

Use persistence to ensure a user is connected to the same real server every time the user makes an HTTP, HTTPS, or SSL request that is part of the same user session. For example, if you are load balancing HTTP and HTTPS sessions to a collection of eCommerce web servers, when users make a purchase, they will be starting multiple sessions as they navigate the eCommerce site. In most cases, all the sessions started by this user during one eCommerce session should be processed by the same real server. Typically, the HTTP protocol keeps track of these related sessions using cookies. HTTP cookie persistence ensure all sessions that are part of the same user session are processed by the same real server.

When you configure persistence, the FortiGate unit load balances a new session to a real server according to the load balance method. If the session has an HTTP cookie or an SSL session ID, the FortiGate unit sends all subsequent sessions with the same HTTP cookie or SSL session ID to the same real server.

Real servers

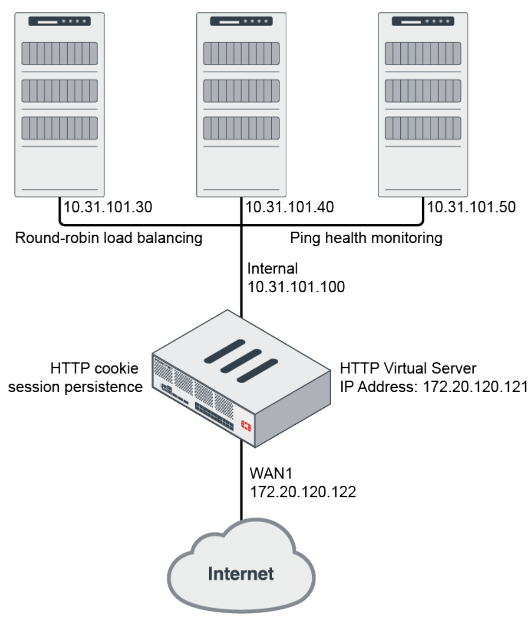
Add real servers to a load balancing virtual server to provide information the virtual server requires to send sessions to the server. A real server configuration includes the IP address of the real server and port number the real server receives sessions on. The FortiGate unit sends sessions to the real server's IP address using the destination port number in the real server configuration.

When configuring a real server, you can also specify the weight (if the load balance method is set to *Weighted*) and you can limit the maximum number of open connections between the FortiGate unit and the real server. If the maximum number of connections is reached for the real server, the FortiGate unit automatically switches all further connection requests to other real servers until the connection number drops below the limit. Setting *Maximum Connections* to 0 means that the FortiGate unit does not limit the number of connections to the real server.

Sample of HTTP load balancing to three real web servers

This example describes the steps to configure the load balancing configuration below. In this configuration, a FortiGate unit is load balancing HTTP traffic from the Internet to three HTTP servers on the internal network. HTTP sessions are accepted at the wan1 interface with destination IP address 172.20.120.121 on TCP port 8080, and forwarded from the internal interface to the web servers. When forwarded, the destination address of the session is translated to the IP address of one of the web servers.

This load balancing configuration also includes session persistence using HTTP cookies, round-robin load balancing, and TCP health monitoring for the real servers. Ping health monitoring consists of the FortiGate unit using ICMP ping to ensure the web servers can respond to network traffic.



General steps:

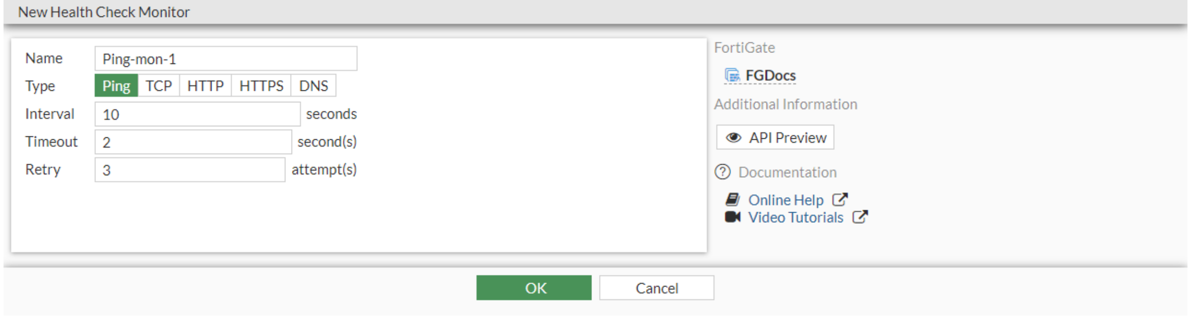
1. Create a health check monitor.
   A ping health check monitor causes the FortiGate to ping the real servers every 10 seconds. If one of the servers does not respond within 2 seconds, the FortiGate unit will retry the ping 3 times before assuming that the HTTP server is not responding.
2. Create a load balance virtual server with three real servers.
3. Add the load balancing virtual server to a policy as the destination address.

To see the virtual servers and health check monitors options in the GUI, *Load Balance* must be selected in *Feature Visibility > Additional Features*. See Feature visibility on page 2163 on page 1 for details.

Configure a load balancing virtual server in the GUI

To create a health check monitor:

1. Go to *Policy & Objects > Health Check*.
2. Click *Create New*.
3. Set the following:
   - *Name* to *Ping-mon-1*
   - *Type* to *Ping*
   - *Interval* to *10* seconds
   - *Timeout* to *2* seconds
   - *Retry* to *3* attempt(s)

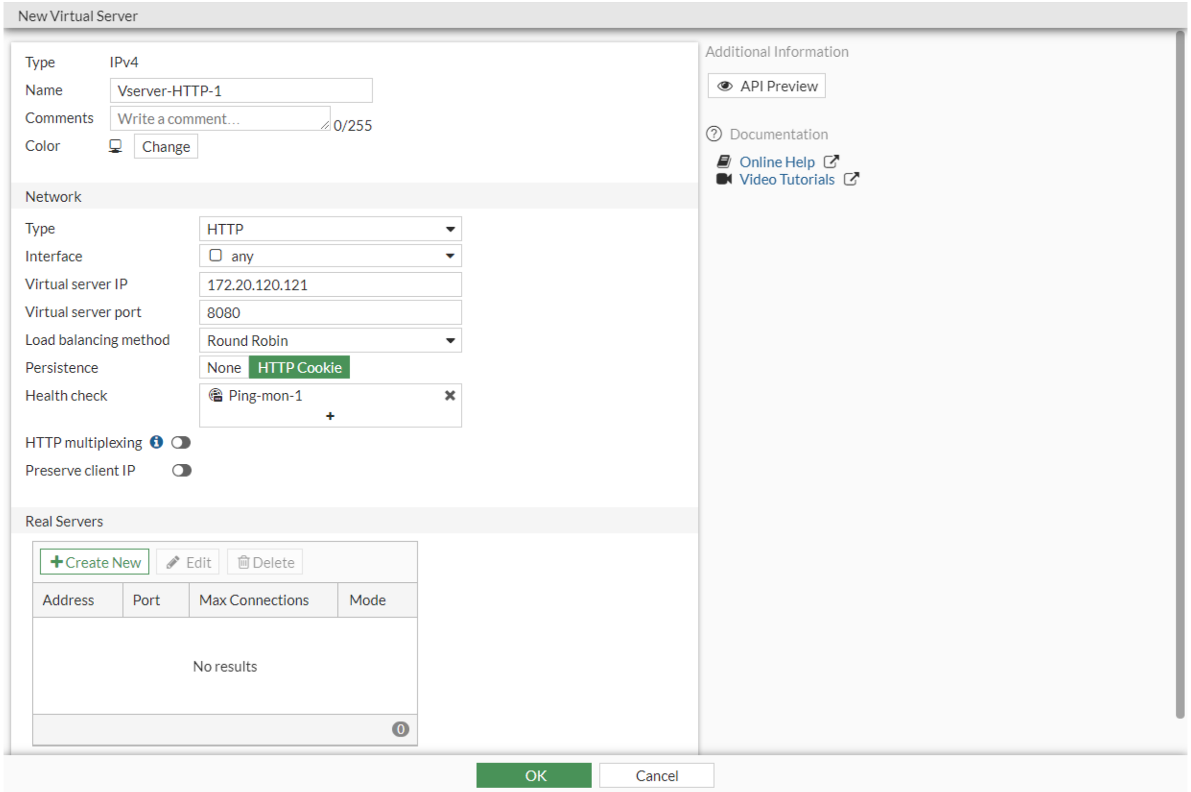


4. Click *OK*.

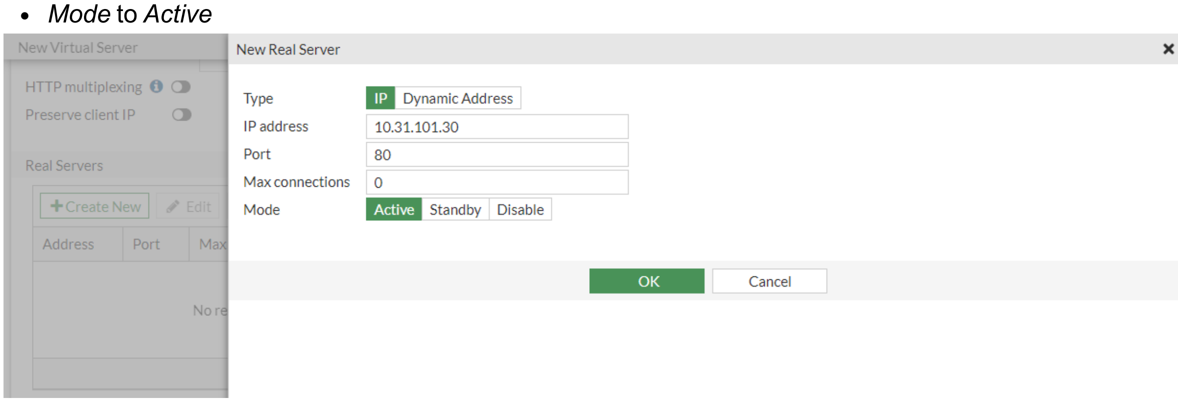
To create a virtual server:

1. Go to *Policy & Objects > Virtual Servers*.
2. Click *Create New*.
3. Set the following:
   - *Name* to *Vserver-HTTP-1*
   - *Type* to *HTTP*
   - *Interface* to *wan1*
   - *Virtual Server IP* to *172.20.120.121*
   - *Virtual Server Port* to *8080*

- *Load Balance Method* to *Round Robin*
- *Persistence* to *HTTP Cookie*
- *Health Check* to *Ping-mon-1*



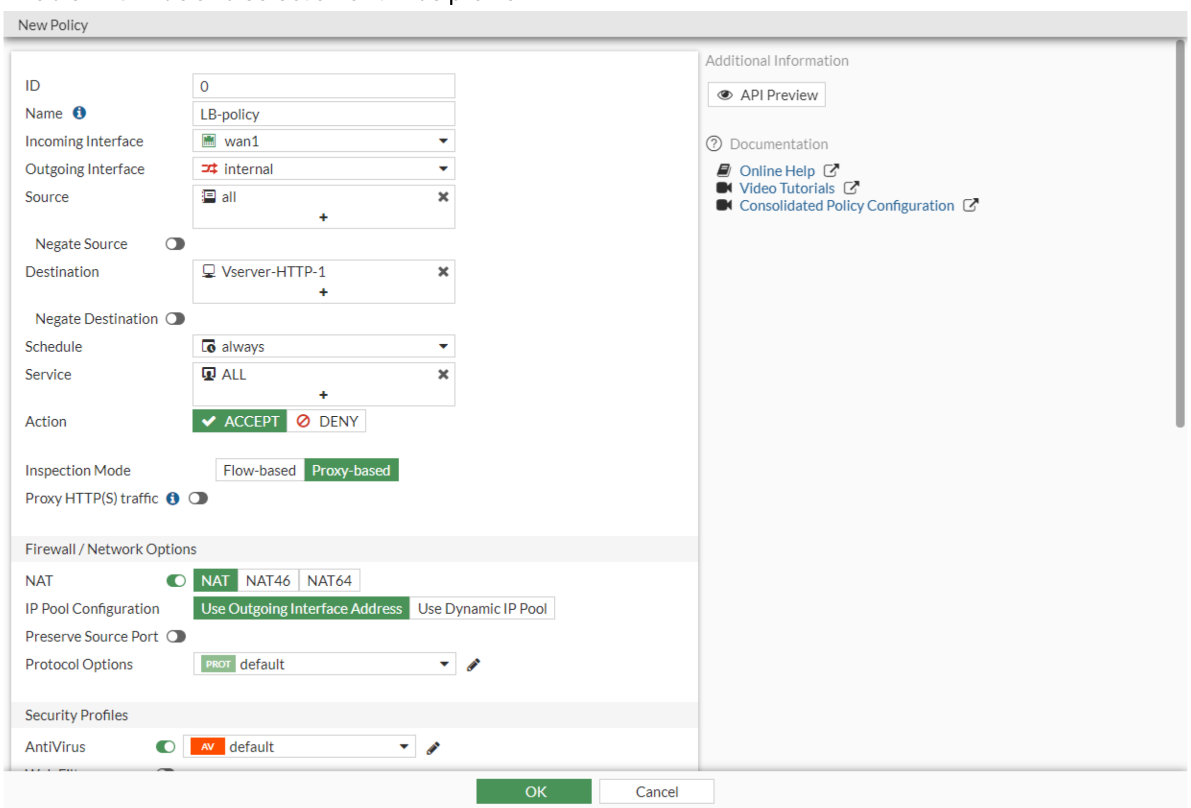
4. In the *Real Servers* table, click *Create New*.
5. Set the following for the first real server:
   - *Type* to *IP*
   - *IP Address* to *10.31.101.30*
   - *Port* to *80*
   - *Max Connections* to *0*
   - *Mode* to *Active*



6. Click *OK*. Configure two more real servers with IP addresses 10.31.101.40 and 10.31.101.50, and the same settings as the first real server.
7. Click *OK*.

To create a security policy that includes the load balance virtual server as the destination address:

1. Go to *Policy & Objects > Firewall Policy*.
2. Click *Create New*.
3. Set the *Inspection Mode* to *Proxy-based*. The new virtual server will not be available if the inspection mode is *Flow-based*.
4. Set the following:
   - *Name* to *LB-policy*
   - *Incoming Interface* to *wan1*
   - *Outgoing Interface* to *internal*
   - *Source* to *all*
   - *Destination* to *Vserver-HTTP-1*
   - *Schedule* to *always*
   - *Service* to *ALL*
   - *Action* to *ACCEPT*
5. Enable *NAT* and set *IP Pool Configuration* to *Use Outgoing Interface Address*.
6. Enable *AntiVirus* and select an antivirus profile.



7. Click *OK*.

Configure a load balancing virtual server in the CLI

To configure HTTP load balancing to three real web servers in the CLI:

1. Create a health check monitor:

```
config firewall ldb-monitor
    edit "Ping-mon-1"
        set type ping
        set interval 10
        set timeout 2
        set retry 3
    next
end
```

2. Create a virtual server:

```
config firewall vip
    edit "Vserver-HTTP-1"
        set type server-load-balance
        set extip 172.20.120.121
        set extintf "any"
        set server-type http
        set monitor "Ping-mon-1"
        set ldb-method round-robin
        set persistence http-cookie
        set extport 8080
        config realservers
            edit 1
                set type ip
                set ip 10.31.101.30
                set port 80
            next
            edit 2
                set type ip
                set ip 10.31.101.40
                set port 80
            next
            edit 3
                set type ip
                set ip 10.31.101.50
                set port 80
            next
        end
    next
end
```

3. Add the load balancing virtual server to a policy as the destination address:

```
config firewall policy
    edit 2
        set name "LB-policy"
        set inspection-mode proxy
        set srcintf "wan1"
        set dstintf "internal"
        set srcaddr "all"
        set dstaddr "Vserver-HTTP-1"
        set action accept
```

```
            set schedule "always"
            set service "ALL"
            set utm-status enable
            set ssl-ssh-profile "certificate-inspection"
            set av-profile "default"
            set fsso disable
            set nat enable
        next
    end
```

Results

Traffic accessing 172.20.120.121:8080 is forwarded in turn to the three real servers.

If the access request has an http-cookie, FortiGate forwards the access to the corresponding real server according to the cookie.

Examples and policy actions

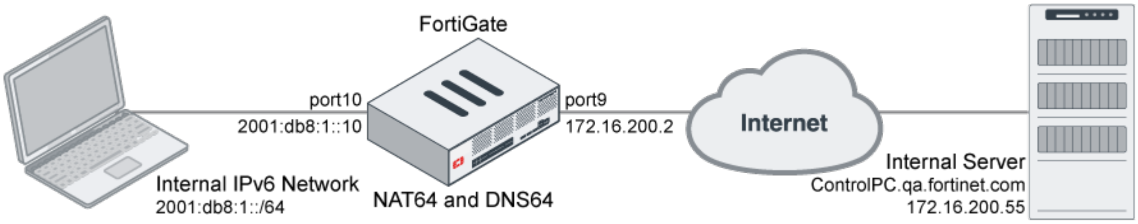
The following topics provide examples and instructions on policy actions:

NAT64 policy and DNS64 (DNS proxy)

NAT64 policy translates IPv6 addresses to IPv4 addresses so that a client on an IPv6 network can communicate transparently with a server on an IPv4 network.

NAT64 policy is usually implemented in combination with the DNS proxy called DNS64. DNS64 synthesizes AAAA records from A records and is used to synthesize IPv6 addresses for hosts that only have IPv4 addresses. DNS proxy and DNS64 are interchangeable terms.

Sample topology



In this example, a host on the internal IPv6 network communicates with `ControlPC.qa.fortinet.com` that only has IPv4 address on the Internet. Central NAT is disabled.

1. The host on the internal network does a DNS lookup for `ControlPC.qa.fortinet.com` by sending a DNS query for an AAAA record for `ControlPC.qa.fortinet.com`.
2. The DNS query is intercepted by the FortiGate DNS proxy. The DNS proxy performs an A-record query for `ControlPC.qa.fortinet.com` and gets back an RRSet containing a single A record with the IPv4 address `172.16.200.55`.
3. The DNS proxy then synthesizes an AAAA record. The IPv6 address in the AAAA record begins with the configured NAT64 prefix in the upper 96 bits and the received IPv4 address in the lower 32 bits. By default, the resulting IPv6 address is `64:ff9b::172.16.200.55`.
4. The host on the internal network receives the synthetic AAAA record and sends a packet to the destination address `64:ff9b::172.16.200.55`.
5. The packet is routed to the FortiGate internal interface (port10) where it is accepted by the NAT64 security policy.
6. The FortiGate translates the destination address of the packets from IPv6 address `64:ff9b::172.16.200.55` to IPv4 address `172.16.200.55` and translates the source address of the packets to `172.16.200.200` (or another address in the IP pool range) and forwards the packets out the port9 interface to the Internet.

Sample configuration

To configure a NAT64 policy with DNS64 in the GUI:

1. Enable IPv6 and DNS database:
   a. Go to *System* > *Feature Visibility*.
   b. In the *Core Features* section, enable *IPv6*.
   c. In the *Additional Features* section, enable *DNS Database*.
   d. Click *Apply*.
2. Enable DNS proxy on the IPv6 interface:
   a. Go to *Network* > *DNS Servers*.
   b. In the *DNS Service on Interface* table, click *Create New*.
   c. For *Interface*, select *port10*.
   d. For *Mode*, select *Forward to System DNS*.
   e. Click *OK*.
3. Configure the IPv6 DHCP server:
   a. Go to *Network* > *Interfaces* and edit port10.
   b. Enable *DHCPv6 Server* and enter the following:

IPv6 subnet	2001:db8:1::/64
DNS service	Specify
DNS server 1	2001:db8:1::10

c. Click *OK*.

4. Configure the IPv6 VIP for the destination IPv6 addresses:

   These are all of the IPv6 addresses that the FortiGate DNS proxy synthesizes when an IPv6 device performs a DNS query that resolves to an IPv4 Address. In this example, the synthesized IPv6 address in the AAAA record begins with the configured NAT64 prefix in the upper 96 bits, so the VIP is for all the IPv6 addresses that begin with 64:ff9b.

   a. Go to *Policy & Objects* > *Virtual IPs* and click *Create New* > *Virtual IP*.

   b. Enter the following:

VIP type	IPv6
Name	vip6
Eternal IP address/range	64:ff9b::-64:ff9b::ffff:ffff
Map to IPv4 address/range	Use Embedded

   c. Click *OK*.

5. Configure the IPv6 firewall address for the internal network:

   a. Click *Create New* > *Address*.

   b. Enter the following:

Category	IPv6 Address
Name	internal-net6
Type	IPv6 Subnet
IP/Netmask	2001:db8:1::/48

   c. Click *OK*.

6. Configure the IP pool containing the IPv4 address that is used as the source address of the packets exiting port9:

   a. Go to *Policy & Objects* > *IP Pools* and click *Create New*.

   b. Enter the following:

IP Pool Type	IPv4 Pool
Name	exit-pool4
Type	Overload
External IP address/range	172.16.200.200-172.16.200.207
NAT64	Enable

External IP address/range must start and end on the boundaries of a valid subnet. For example, 172.16.200.0-172.16.200.7 and 172.16.200.16-172.16.200.31 are a valid subnets (/29 and /28 respectively).

c. Click *OK*.

7. Configure the NAT64 policy:
   a. Go to *Policy & Objects* > *Firewall Policy* and click *Create New*.
   b. Enter the following:

Name	policy64-1
Incoming Interface	port10
Outgoing Interface	port9
Source	internal-net6
Destination	vip6
Schedule	always
Service	ALL
Action	ACCEPT
NAT	NAT64
IP Pool Configuration	exit-pool4

   c. Click *OK*.

To configure a NAT64 policy with DNS64 in the CLI:

1. Enable IPv6 and DNS database:

```
config system global
   set gui-ipv6 enable
end

config system settings
   set gui-dns-database enable
end
```

2. Enable DNS proxy on the IPv6 interface:

```
config system dns-server
   edit "port10"
       set mode forward-only
   next
end
```

3. Configure the IPv6 DHCP server:

```
config system dhcp6 server
   edit 1
       set subnet 2001:db8:1::/64
       set interface "port10"
       set dns-server1 2001:db8:1::10
   next
end
```

4. Configure the IPv6 VIP for the destination IPv6 addresses:

```
config firewall vip6
    edit "vip6"
        set extip 64:ff9b::-64:ff9b::ffff:ffff
        set embedded-ipv4-address enable
    next
end
```

5. Configure the IPv6 firewall address for the internal network:

```
config firewall address6
    edit "internal-net6"
        set ip6 2001:db8:1::/48
    next
end
```

6. Configure the IP pool containing the IPv4 address that is used as the source address of the packets exiting port9:

```
config firewall ippool
    edit "exit-pool4"
        set startip 172.16.200.200
        set endip 172.16.200.207
        set nat64 enable
    next
end
```

External IP address/range must start and end on the boundaries of a valid subnet. For example, 172.16.200.0-172.16.200.7 and 172.16.200.16-172.16.200.31 are a valid subnets (/29 and /28 respectively).

7. Configure the NAT64 policy:

```
config firewall policy
    edit 1
        set name "policy64-1"
        set srcintf "port10"
        set dstintf "port9"
        set action accept
        set nat64 enable
        set srcaddr "all"
        set dstaddr "all"
        set srcaddr6 internal-net6
        set dstaddr6 vip6
        set schedule "always"
        set service "ALL"
        set ippool enable
        set poolname "exit-pool4"
    next
end
```

To enable DNS64 and related settings using the CLI:

Enabling DNS64 means that all IPv6 traffic received by the current VDOM can be subject to NAT64 if the source and destination address matches an NAT64 security policy.

By default, the setting `always-synthesize-aaaa-record` is enabled. If you disable this setting, the DNS proxy (DNS64) will attempt to find an AAAA records for queries to domain names and therefore resolve the host names to IPv6 addresses. If the DNS proxy cannot find an AAAA record, it synthesizes one by adding the NAT64 prefix to the A record.

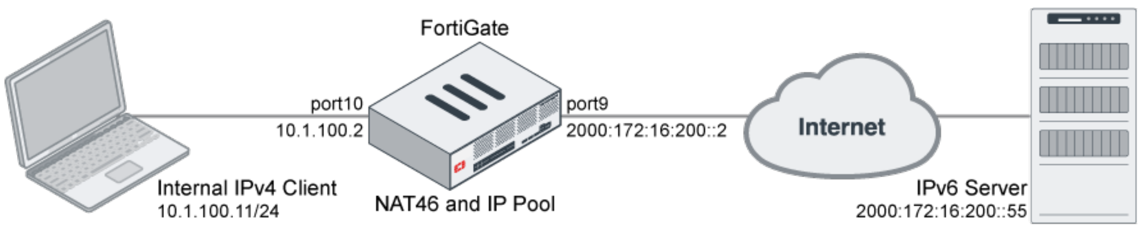
```
config system dns64
    set status {enable | disable}
    set dns64-prefix <ipv6-prefix>
    set always-synthesize-aaaa-record {enable | disable}
end
```

By default, the `dns64-prefix` is `64:ff9b::/96`.

NAT46 policy

NAT46 refers to the mechanism that allows IPv4 addressed hosts to communicate with IPv6 hosts. Without such a mechanism, IPv4 environments cannot connect to IPv6 networks.

Sample topology



In this example, an IPv4 client tries to connect to an IPv6 server. A VIP is configured on FortiGate to map the server IPv6 IP address `2000:172:16:200:55` to an IPv4 address `10.1.100.55`. On the other side, an IPv6 IP pool is configured and the source address of packets from client are changed to the defined IPv6 address. In this setup, the client PC can access the server by using IP address `10.1.100.55`.

Sample configuration

To configure NAT46 in the GUI:

1. Enable IPv6:
   a. Go to *System* > *Feature Visibility*.
   b. In the *Core Features* section, enable *IPv6*.
   c. Click *Apply*.
2. Configure the VIP:
   a. Go to *Policy & Objects* > *Virtual IPs* and click *Create New* > *Virtual IP*.
   b. Enter the following:

VIP type	IPv4
Name	vip46_server
Interface	port2
Type	Static NAT
External IP address/range	10.1.100.55
Map to IPv6 address/range	2000:172:16:200::55

   c. Click *OK*.

3. Configure the IPv6 IP pool:
   a. Go to *Policy & Objects > IP Pools* and click *Create New*.
   b. Enter the following:

IP Pool Type	IPv6 Pool
Name	client_external
External IP address/range	2000:172:16:201::-2000:172:16:201::7
NAT46	Enable

   c. Click *OK*.
4. Configure the firewall policy:
   a. Go to *Policy & Objects > Firewall Policy* and click *Create New*.
   b. Enter the following:

Name	policy46-1
Incoming Interface	port10
Outgoing Interface	port9
Source	all
Destination	vip46_server
Schedule	always
Service	ALL
Action	ACCEPT
NAT	NAT46
IP Pool Configuration	client_external

   c. Configure the other settings as needed.
   d. Click *OK*.

To configure NAT46 in the CLI:

1. Enable IPv6:

```
config system global
    set gui-ipv6 enable
end
```

2. Configure the VIP:

```
config firewall vip
    edit "vip46_server"
        set extip 10.1.100.55
        set nat44 disable
        set nat46 enable
        set extintf "port2"
        set ipv6-mappedip 2000:172:16:200::55
```

```
    next
end
```

3. Configure the IPv6 IP pool:

```
config firewall ippool6
    edit "client_external"
        set startip 2000:172:16:201::
        set endip 2000:172:16:201::7
        set nat46 enable
    next
end
```

4. Configure the firewall policy:

```
config firewall policy
    edit 2
        set name "policy46-1"
        set srcintf "port10"
        set dstintf "port9"
        set action accept
        set nat46 enable
        set srcaddr "all"
        set dstaddr "vip46_server"
        set srcaddr6 "all"
        set dstaddr6 "all"
        set schedule "always"
        set service "ALL"
        set logtraffic all
        set auto-asic-offload disable
        set ippool enable
        set poolname6 "client_external"
    next
end
```

Sample troubleshooting

To trace the flow and troubleshoot:

```
# diagnose debug flow filter saddr 10.1.100.11
# diagnose debug flow show function-name enable
show function name
# diagnose debug flow show iprope enable
show trace messages about iprope
# diagnose debug flow trace start 5

id=20085 trace_id=1 func=print_pkt_detail line=5401 msg="vd-root:0 received a packet
(proto=1, 10.1.100.11:27592->10.1.100.55:2048) from port10. type=8, code=0, id=27592,
seq=1."
id=20085 trace_id=1 func=init_ip_session_common line=5561 msg="allocate a new session-
000003b9"
id=20085 trace_id=1 func=iprope_dnat_check line=4948 msg="in-[port10], out-[]"
id=20085 trace_id=1 func=iprope_dnat_tree_check line=822 msg="len=1"
```

```
id=20085 trace_id=1 func=__iprope_check_one_dnat_policy line=4822 msg="checking gnum-100000
policy-1"
id=20085 trace_id=1 func=get_vip46_addr line=998 msg="find DNAT46: IP-2000:172:16:200::55,
port-27592"
id=20085 trace_id=1 func=__iprope_check_one_dnat_policy line=4904 msg="matched policy-1,
act=accept, vip=1, flag=100, sflag=2000000"
id=20085 trace_id=1 func=iprope_dnat_check line=4961 msg="result: skb_flags-02000000, vid-1,
ret-matched, act-accept, flag-00000100"
id=20085 trace_id=1 func=fw_pre_route_handler line=183 msg="VIP-10.1.100.55:27592, outdev-
unkown"
id=20085 trace_id=1 func=__ip_session_run_tuple line=3220 msg="DNAT 10.1.100.55:8-
>10.1.100.55:27592"
id=20085 trace_id=1 func=vf_ip_route_input_common line=2594 msg="find a route: flag=80000000
gw-10.1.100.55 via root"
id=20085 trace_id=1 func=ip4_nat_af_input line=601 msg="nat64 ipv4 received a packet
proto=1"
id=20085 trace_id=1 func=__iprope_check line=2112 msg="gnum-100012, check-ffffffffa0024ebe"
id=20085 trace_id=1 func=__iprope_check_one_policy line=1873 msg="checked gnum-100012
policy-1, ret-matched, act-accept"
id=20085 trace_id=1 func=__iprope_user_identity_check line=1677 msg="ret-matched"
id=20085 trace_id=1 func=get_new_addr46 line=1047 msg="find SNAT46: IP-2000:172:16:201::13
(from IPPOOL), port-27592"
id=20085 trace_id=1 func=__iprope_check_one_policy line=2083 msg="policy-1 is matched, act-
accept"
id=20085 trace_id=1 func=__iprope_check line=2131 msg="gnum-100012 check result: ret-
matched, act-accept, flag-08050500, flag2-00200000"
id=20085 trace_id=1 func=iprope_policy_group_check line=4358 msg="after check: ret-matched,
act-accept, flag-08050500, flag2-00200000"
id=20085 trace_id=1 func=resolve_ip6_tuple line=4389 msg="allocate a new session-00000081"
```

NAT46 and NAT64 policy and routing configurations

Multiple NAT46 and NAT64 related objects are consolidated into regular objects. A per-VDOM virtual interface, naf.<vdom>, is automatically added to process NAT46/NAT64 traffic. The features include:

- `vip46` and `vip64` settings are consolidated in `vip` and `vip6` configurations.
- `policy46` and `policy64` settings are consolidated in `firewall policy` settings.
- `nat46`/`nat64` are included in `firewall policy` settings.
- `ippool` and `ippool6` support NAT46 and NAT64 (when enabled, the IP pool should match a subnet).
- Central SNAT supports NAT46 and NAT64.
- `add-nat46-route` in `ippool6` and `add-nat64-route` in `ippool` are enabled by default. The FortiGate generates a static route that matches the IP range in `ippool6` or `ippool` for the naf tunnel interface.

Automatic processing of the naf tunnel interface is not supported in security policies.

To configure NAT46/NAT64 translation, use the standard `vip`/`vip6` setting, apply it in a firewall policy, enable NAT46/NAT64, and enter the IP pool to complete the configuration.

The external IP address cannot be the same as the external interface IP address.

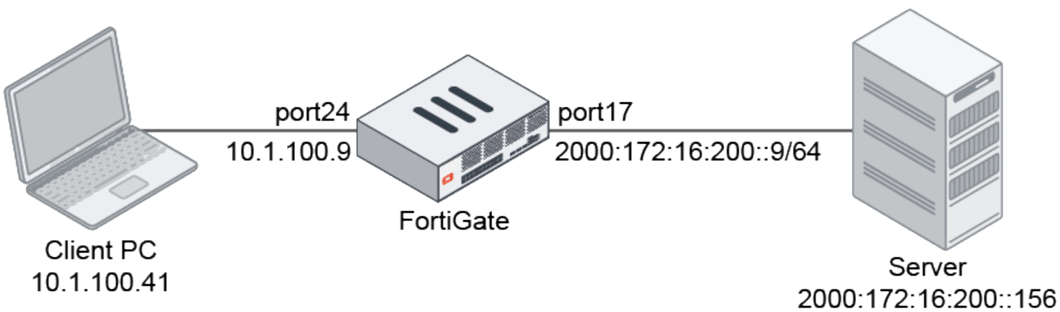
Examples

IPv6 must be enabled to configure these examples. In the GUI, so go to *System* > *Feature Visibility* and enable *IPv6*. In the CLI, enter the following:

```
config system global
    set gui-ipv6 enable
end
```

NAT46 policy

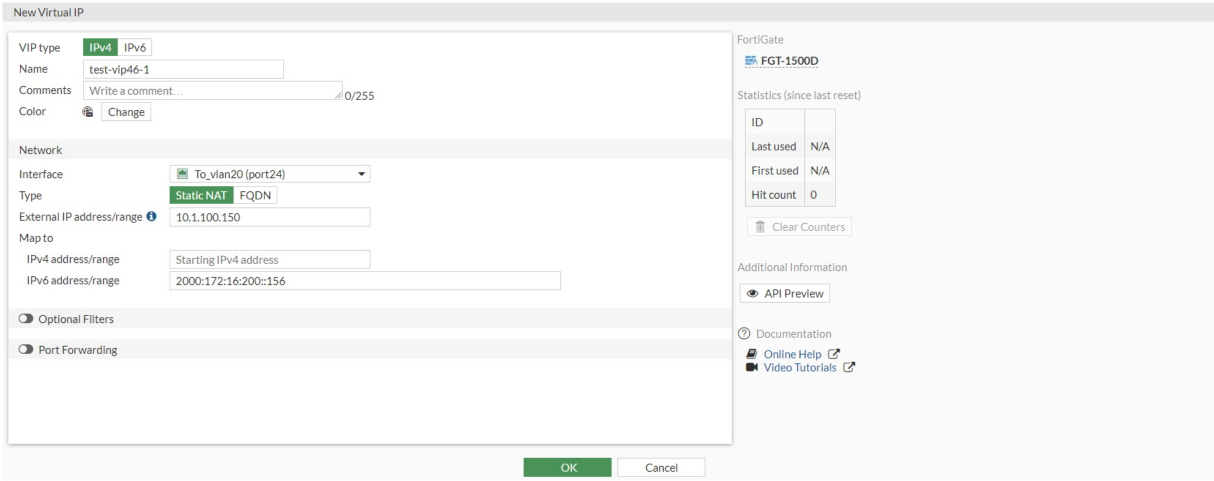
In this example, a client PC is using IPv4 and an IPv4 VIP to access a server that is using IPv6. The FortiGate uses NAT46 to translate the request from IPv4 to IPv6 using the virtual interface naf.root. An `ippool6` is applied so that the request is SNATed to the `ippool6` address (2000:172:16:101::1 - 2000:172:16:101::1).



To create a NAT46 policy in the GUI:

1. Configure the VIP:
   a. Go to *Policy & Objects* > *Virtual IPs* and click *Create New* > *VIP*.
   b. Enter the following:

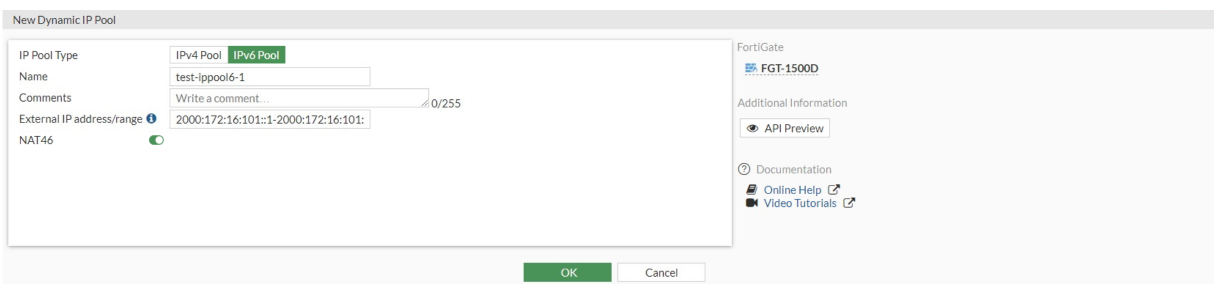
VIP type	IPv4
Name	test-vip46-1
Interface	To_vlan20
Type	Static NAT
External IP address/range	10.1.100.150
Map to IPv6 address/range	2000:172:16:200::156



c. Click *OK*.

2. Configure the IPv6 pool:
   a. Go to *Policy & Objects* > *IP Pools* and click *Create New*.
   b. Enter the following:

IP Pool Type	IPv6 Pool
Name	test-ippool6-1
External IP address/range	2000:172:16:101::1-2000:172:16:101::1
NAT46	Enable



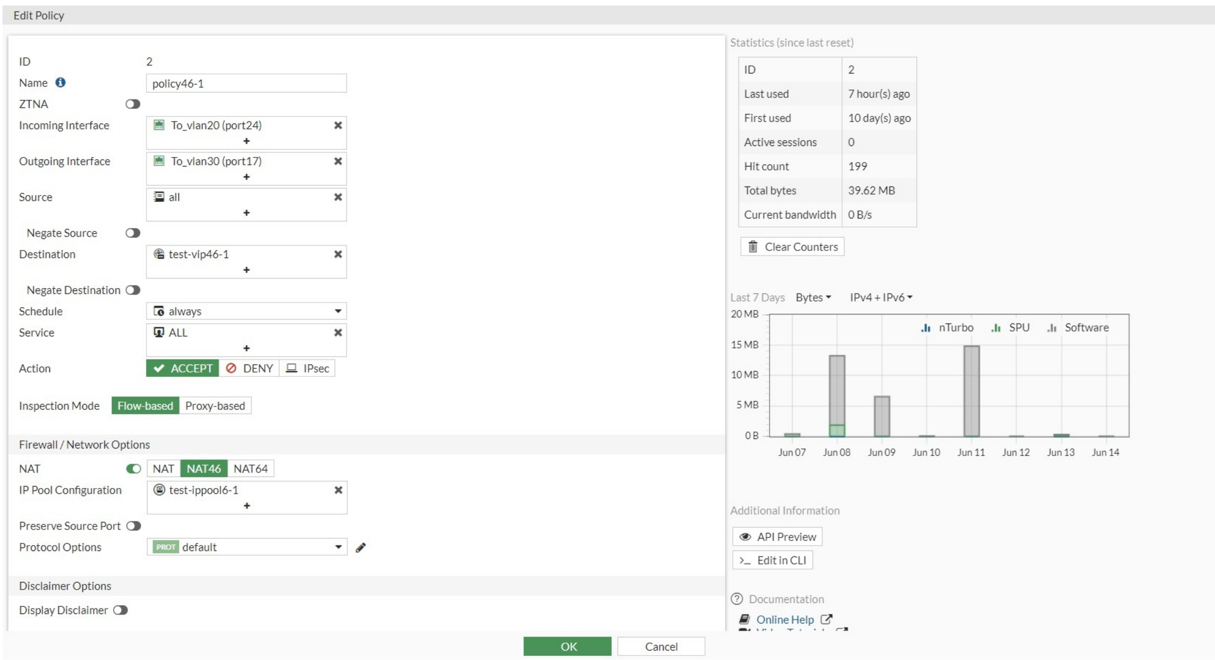
c. Click *OK*.

3. Configure the firewall policy:
   a. Go to *Policy & Objects* > *Firewall Policy* and click *Create New* or edit an existing policy.
   b. Enter the following:

Name	policy46-1
Incoming Interface	To_vlan20
Outgoing Interface	To_vlan30
Source	all
Destination	test-vip46-1
Schedule	always

Service	ALL
Action	ACCEPT
NAT	NAT46
IP Pool Configuration	test-ippool6-1

c. Configure the other settings as needed.



d. Click *OK*.

To create a NAT46 policy in the CLI:

1. Configure the VIP:

```
config firewall vip
    edit "test-vip46-1"
        set extip 10.1.100.150
        set nat44 disable
        set nat46 enable
        set extintf "port24"
        set arp-reply enable
        set ipv6-mappedip 2000:172:16:200::156
    next
end
```

2. Configure the IPv6 pool:

```
config firewall ippool6
    edit "test-ippool6-1"
        set startip 2000:172:16:101::1
        set endip 2000:172:16:101::1
        set nat46 enable
        set add-nat46-route enable
```

```
    next
end
```

3. Configure the firewall policy:

```
config firewall policy
    edit 2
        set name "policy46-1"
        set srcintf "port24"
        set dstintf "port17"
        set action accept
        set nat46 enable
        set srcaddr "all"
        set dstaddr "test-vip46-1"
        set srcaddr6 "all"
        set dstaddr6 "all"
        set schedule "always"
        set service "ALL"
        set logtraffic all
        set auto-asic-offload disable
        set ippool enable
        set poolname6 "test-ippool6-1"
    next
end
```

To verify the traffic and session tables:

1. Verify the traffic by the sniffer packets:

```
(root) # diagnose sniffer packet any 'icmp or icmp6' 4
interfaces=[any]
filters=[icmp or icmp6]
2.593302 port24 in 10.1.100.41 -> 10.1.100.150: icmp: echo request
2.593344 naf.root out 10.1.100.41 -> 10.1.100.150: icmp: echo request
2.593347 naf.root in 2000:172:16:101::1 -> 2000:172:16:200::156: icmp6: echo request seq
1
2.593383 port17 out 2000:172:16:101::1 -> 2000:172:16:200::156: icmp6: echo request seq
1
2.593772 port17 in 2000:172:16:200::156 -> 2000:172:16:101::1: icmp6: echo reply seq 1
2.593788 naf.root out 2000:172:16:200::156 -> 2000:172:16:101::1: icmp6: echo reply seq
1
2.593790 naf.root in 10.1.100.150 -> 10.1.100.41: icmp: echo reply
2.593804 port24 out 10.1.100.150 -> 10.1.100.41: icmp: echo reply
11 packets received by filter
0 packets dropped by kernel
```

2. Verify the session tables for IPv4 and IPv6:

```
(root) # diagnose sys session list
session info: proto=1 proto_state=00 duration=2 expire=59 timeout=0 flags=00000000
socktype=0 sockport=0 av_idx=0 use=3
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/255
state=log may_dirty f00 netflow-origin netflow-reply
statistic(bytes/packets/allow_err): org=252/3/1 reply=252/3/1 tuples=2
tx speed(Bps/kbps): 106/0 rx speed(Bps/kbps): 106/0
```

```
orgin->sink: org pre->post, reply pre->post dev=24->53/53->24
gwy=10.1.100.150/10.1.100.41
hook=pre dir=org act=noop 10.1.100.41:29388->10.1.100.150:8(0.0.0.0:0)
hook=post dir=reply act=noop 10.1.100.150:29388->10.1.100.41:0(0.0.0.0:0)
peer=2000:172:16:101::1:29388->2000:172:16:200::156:128 naf=1
hook=pre dir=org act=noop 2000:172:16:101::1:29388->2000:172:16:200::156:128(:::0)
hook=post dir=reply act=noop 2000:172:16:200::156:29388->2000:172:16:101::1:129(:::0)
misc=0 policy_id=2 auth_info=0 chk_client_info=0 vd=0
serial=00012b77 tos=ff/ff app_list=0 app=0 url_cat=0
sdwan_mbr_seq=0 sdwan_service_id=0
rpdb_link_id=00000000 rpdb_svc_id=0 ngfwid=n/a
npu_state=0x040001 no_offload
no_ofld_reason:  disabled-by-policy non-npu-intf
total session 1
```

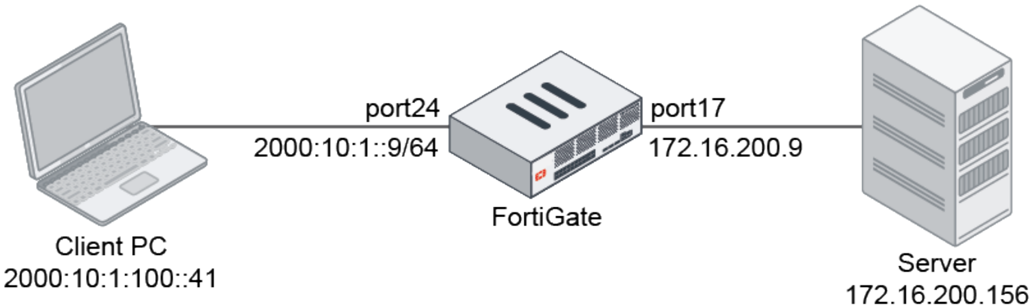
```
(root) # diagnose sys session6 list
session6 info: proto=58 proto_state=00 duration=5 expire=56 timeout=0 flags=00000000
sockport=0 socktype=0 use=3
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/0
state=log may_dirty
statistic(bytes/packets/allow_err): org=312/3/0 reply=312/3/0 tuples=2
tx speed(Bps/kbps): 0/0 rx speed(Bps/kbps): 0/0
orgin->sink: org pre->post, reply pre->post dev=53->17/17->53
hook=pre dir=org act=noop 2000:172:16:101::1:29388->2000:172:16:200::156:128(:::0)
hook=post dir=reply act=noop 2000:172:16:200::156:29388->2000:172:16:101::1:129(:::0)
peer=10.1.100.150:29388->10.1.100.41:0 naf=2
hook=pre dir=org act=noop 10.1.100.41:29388->10.1.100.150:8(0.0.0.0:0)
hook=post dir=reply act=noop 10.1.100.150:29388->10.1.100.41:0(0.0.0.0:0)
misc=0 policy_id=2 auth_info=0 chk_client_info=0 vd=0
serial=00001bbc tos=ff/ff ips_view=1024 app_list=0 app=0 url_cat=0
rpdb_link_id = 00000000 ngfwid=n/a
npu_state=0x000001 no_offload
no_ofld_reason:  disabled-by-policy
total session 1
```

The IPv4 session is between the incoming physical interface port24 and naf.root. The IPv6 session is between the naf.root and the outgoing physical interface port17.

NAT64 policy

In this example, a client PC is using IPv6 and an IPv6 VIP to access a server that is using IPv4. The FortiGate uses NAT64 to translate the request from IPv6 to IPv4 using the virtual interface naf.root. An `ippool` is applied so that the request is SNATed to the `ippool` address (172.16.101.2 - 172.16.101.3).

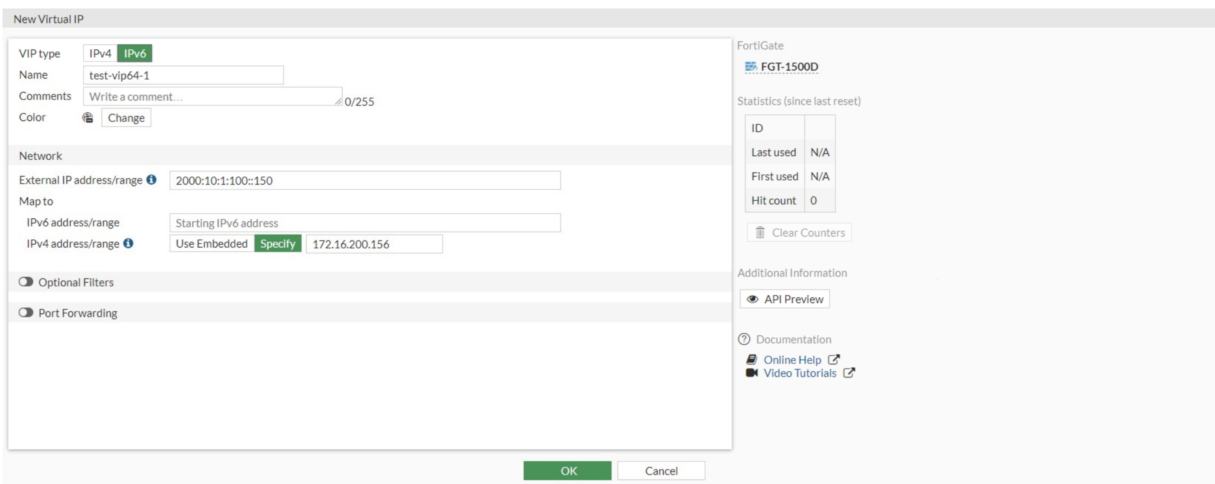
An embedded VIP64 object is used in this configuration so a specific IPv4 mapped IP does not need to be set. The lower 32 bits of the external IPv6 address are used to map to the IPv4 address. Only an IPv6 prefix is defined. In this example, the IPv6 prefix is 2001:10:1:100::, so the IPv6 address 2001:10:1:100::ac10:c89c will be translated to 172.16.200.156.



To create a NAT64 policy in the GUI:

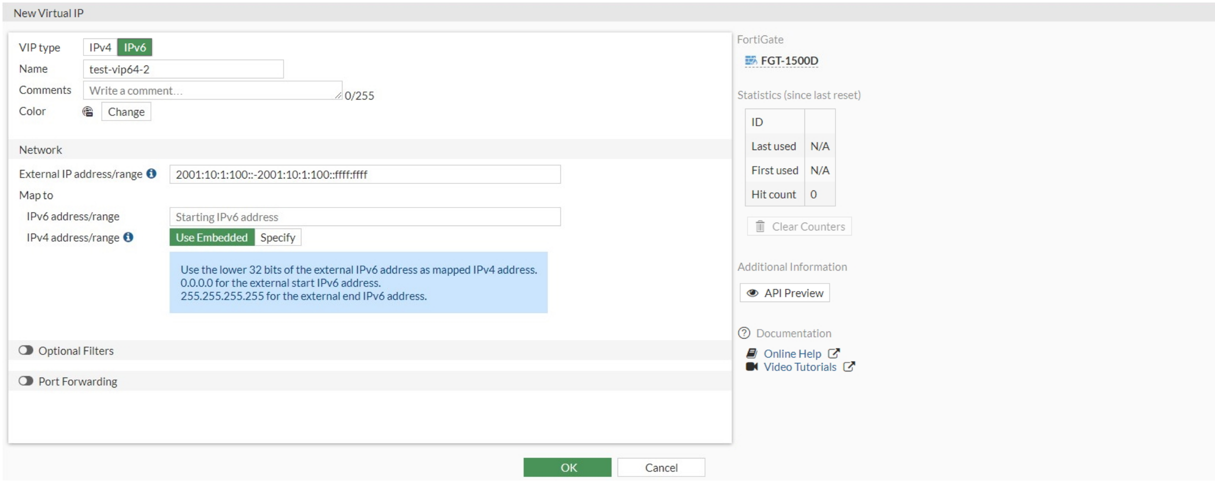
1. Configure the VIP:
   a. Go to *Policy & Objects* > *Virtual IPs* and click *Create New* > *VIP*.
   b. Enter the following:

VIP type	IPv6
Name	test-vip64-1
External IP address/range	2000:10:1:100::150
Map to IPv4 address/range	**Specify**: 172.16.200.156



   c. Click *OK*.
2. Configure the VIP with the embedded IPv4 address enabled:
   a. Go to *Policy & Objects* > *Virtual IPs* and click *Create New* > *VIP*.
   b. Enter the following:

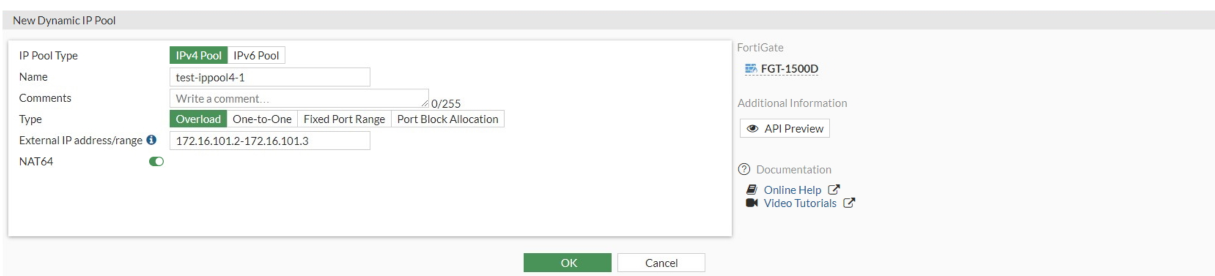
VIP type	IPv6
Name	test-vip64-2
External IP address/range	2001:10:1:100::-2001:10:1:100::ffff:ffff
Map to IPv4 address/range	Use Embedded



c. Click *OK*.

3. Configure the IP pool:
   a. Go to *Policy & Objects > IP Pools* and click *Create New*.
   b. Enter the following:

IP Pool Type	IPv4 Pool
Name	test-ippool4-1
Type	Overload
External IP address/range	172.16.101.2-172.16.101.3
NAT64	Enable



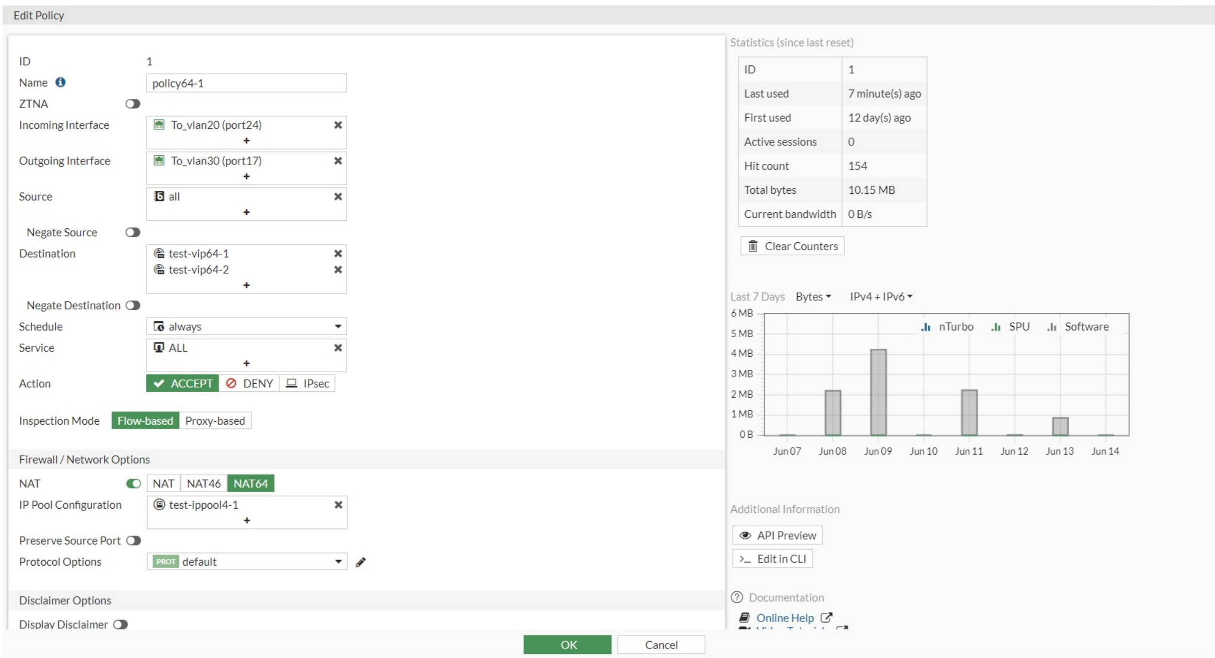
   c. Click *OK*.

4. Configure the firewall policy:
   a. Go to *Policy & Objects > Firewall Policy* and click *Create New* or edit an existing policy.
   b. Enter the following:
   c.

Name	policy64-1
Incoming Interface	To_vlan20
Outgoing Interface	To_vlan30
Source	all
Destination	test-vip64-1 test-vip64-2

Schedule	always
Service	ALL
Action	ACCEPT
NAT	NAT64
IP Pool Configuration	test-ippool4-1

d. Configure the other settings as needed.



e. Click *OK*.

To create a NAT64 policy in the CLI:

1. Configure the VIP:

```
config firewall vip6
    edit "test-vip64-1"
        set extip 2000:10:1:100::150
        set nat66 disable
        set nat64 enable
        set ipv4-mappedip 172.16.200.156
    next
end
```

2. Configure the VIP with the embedded IPv4 address enabled:

```
config firewall vip6
    edit "test-vip64-2"
        set extip 2001:10:1:100::-2001:10:1:100::ffff:ffff
        set nat66 disable
        set nat64 enable
        set embedded-ipv4-address enable
```

```
    next
end
```

3. Configure the IP pool:

```
config firewall ippool
    edit "test-ippool4-1"
        set startip 172.16.101.2
        set endip 172.16.101.3
        set nat64 enable
        set add-nat64-route enable
    next
end
```

4. Configure the firewall policy:

```
config firewall policy
    edit 1
        set name "policy64-1"
        set srcintf "port24"
        set dstintf "port17"
        set action accept
        set nat64 enable
        set srcaddr "all"
        set dstaddr "all"
        set srcaddr6 "all"
        set dstaddr6 "test-vip64-1" "test-vip64-2"
        set schedule "always"
        set service "ALL"
        set logtraffic all
        set auto-asic-offload disable
        set ippool enable
        set poolname "test-ippool4-1"
    next
end
```

To verify the traffic and session tables:

1. Verify the VIP64 traffic by the sniffer packets:

```
(root) # diagnose sniffer packet any 'icmp or icmp6' 4
interfaces=[any]
filters=[icmp or icmp6]
20.578417 port24 in 2000:10:1:100::41 -> 2000:10:1:100::150: icmp6: echo request seq 1
20.578495 naf.root out 2000:10:1:100::41 -> 2000:10:1:100::150: icmp6: echo request seq
1
20.578497 naf.root in 172.16.101.2 -> 172.16.200.156: icmp: echo request
20.578854 port17 out 172.16.101.2 -> 172.16.200.156: icmp: echo request
20.579083 port17 in 172.16.200.156 -> 172.16.101.2: icmp: echo reply
20.579093 naf.root out 172.16.200.156 -> 172.16.101.2: icmp: echo reply
20.579095 naf.root in 2000:10:1:100::150 -> 2000:10:1:100::41: icmp6: echo reply seq 1
20.579377 port24 out 2000:10:1:100::150 -> 2000:10:1:100::41: icmp6: echo reply seq 1
11 packets received by filter
0 packets dropped by kernel
```

2. Verify the session tables for IPv6 and IPv4:

```
(root) # diagnose sys session6 list
session6 info: proto=58 proto_state=00 duration=5 expire=56 timeout=0 flags=00000000
sockport=0 socktype=0 use=3
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/0
state=log may_dirty
statistic(bytes/packets/allow_err): org=312/3/0 reply=312/3/0 tuples=2
tx speed(Bps/kbps): 55/0 rx speed(Bps/kbps): 55/0
orgin->sink: org pre->post, reply pre->post dev=24->53/53->24
hook=pre dir=org act=noop 2000:10:1:100::41:29949->2000:10:1:100::150:128(:::0)
hook=post dir=reply act=noop 2000:10:1:100::150:29949->2000:10:1:100::41:129(:::0)
peer=172.16.101.2:45392->172.16.200.156:8 naf=1
hook=pre dir=org act=noop 172.16.101.2:45392->172.16.200.156:8(0.0.0.0:0)
hook=post dir=reply act=noop 172.16.200.156:45392->172.16.101.2:0(0.0.0.0:0)
misc=0 policy_id=1 auth_info=0 chk_client_info=0 vd=0
serial=000021ec tos=ff/ff ips_view=1024 app_list=0 app=0 url_cat=0
rpdb_link_id = 00000000 ngfwid=n/a
npu_state=0x040001 no_offload
no_ofld_reason:  disabled-by-policy non-npu-intf
total session 1
```

```
(root) # diagnose sys session list
session info: proto=1 proto_state=00 duration=7 expire=54 timeout=0 flags=00000000
socktype=0 sockport=0 av_idx=0 use=3
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/255
state=log may_dirty f00
statistic(bytes/packets/allow_err): org=252/3/1 reply=252/3/1 tuples=2
tx speed(Bps/kbps): 0/0 rx speed(Bps/kbps): 0/0
orgin->sink: org pre->post, reply pre->post dev=53->17/17->53
gwy=172.16.200.156/172.16.101.2
hook=pre dir=org act=noop 172.16.101.2:45392->172.16.200.156:8(0.0.0.0:0)
hook=post dir=reply act=noop 172.16.200.156:45392->172.16.101.2:0(0.0.0.0:0)
peer=2000:10:1:100::150:29949->2000:10:1:100::41:129 naf=2
hook=pre dir=org act=noop 2000:10:1:100::41:29949->2000:10:1:100::150:128(:::0)
hook=post dir=reply act=noop 2000:10:1:100::150:29949->2000:10:1:100::41:129(:::0)
misc=0 policy_id=1 auth_info=0 chk_client_info=0 vd=0
serial=0001347f tos=ff/ff app_list=0 app=0 url_cat=0
sdwan_mbr_seq=0 sdwan_service_id=0
rpdb_link_id=00000000 rpdb_svc_id=0 ngfwid=n/a
npu_state=0x000001 no_offload
no_ofld_reason:  disabled-by-policy
total session 1
```

The IPv6 session is between the incoming physical interface port24 and naf.root. The IPv4 session is between the naf.root and the outgoing physical interface port17.

3. Verify the embedded VIP64 traffic by the sniffer packets:

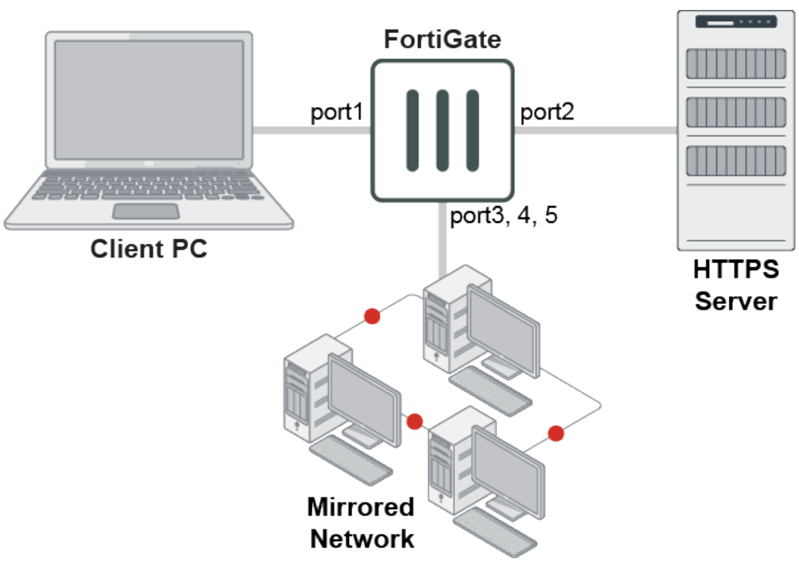
```
(root) # diagnose sniffer packet any 'icmp or icmp6' 4
interfaces=[any]
filters=[icmp or icmp6]
```

```
7.696010 port24 in 2000:10:1:100::41 -> 2001:10:1:100::ac10:c89c: icmp6: echo request
seq 1
7.696057 naf.root out 2000:10:1:100::41 -> 2001:10:1:100::ac10:c89c: icmp6: echo request
seq 1
7.696060 naf.root in 172.16.101.2 -> 172.16.200.156: icmp: echo request
7.696544 port17 out 172.16.101.2 -> 172.16.200.156: icmp: echo request
7.696821 port17 in 172.16.200.156 -> 172.16.101.2: icmp: echo reply
7.696839 naf.root out 172.16.200.156 -> 172.16.101.2: icmp: echo reply
7.696841 naf.root in 2001:10:1:100::ac10:c89c -> 2000:10:1:100::41: icmp6: echo reply
seq 1
7.697167 port24 out 2001:10:1:100::ac10:c89c -> 2000:10:1:100::41: icmp6: echo reply seq
1
11 packets received by filter
0 packets dropped by kernel
```

Mirroring SSL traffic in policies

SSL mirroring allows the FortiGate to decrypt and mirror traffic to a designated port. A new decrypted traffic mirror profile can be applied to IPv4, IPv6, and explicit proxy firewall policies in both flow and proxy mode. Full SSL inspection must be used in the policy for the traffic mirroring to occur.

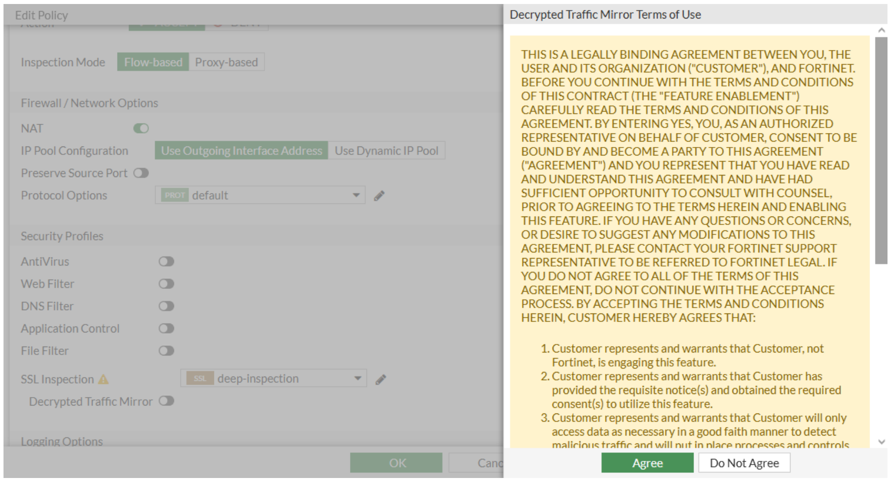
SSL inspection is automatically enabled when you enable a security profile on the policy configuration page.



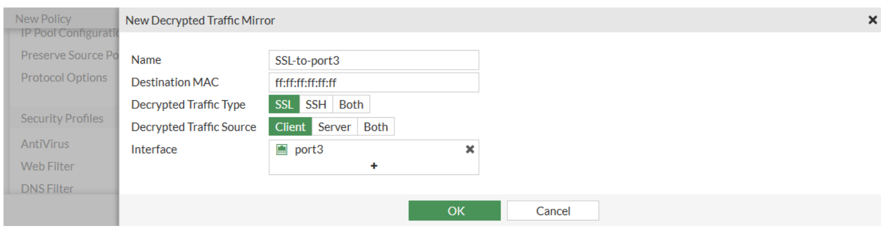
To configure SSL mirroring in a policy in the GUI:

1. Go to *Policy & Objects* > *Firewall Policy*.
2. Create a new policy, or edit an existing one.
3. Configure the interfaces, sources, and other required information.
4. In the *Security Profiles* section, for *SSL Inspection*, select *deep-inspection*, or another profile that uses *Full SSL Inspection*.

5. Enable *Decrypted Traffic Mirror* . The terms of use agreement opens.



6. Click *Agree* to accept the terms.
7. In the drop-down list, select a decrypted traffic mirror, or click *Create* to create a new one.
   In this example, a new decrypted traffic mirror is created using the port3 interface.



8. Click *OK* to save the policy.

To configure SSL mirroring in proxy mode in the CLI:

1. Create the decrypted traffic mirror profile:

```
config firewall decrypted-traffic-mirror
    edit SSL-to-port3
        set dstmac ff:ff:ff:ff:ff:ff
        set traffic-type ssl
        set traffic-source client
        set interface port3
    next
end
```

2. Configure the policy to enable SSL traffic mirroring:

```
config firewall policy
    edit 1
        set name "mirror-policy"
        set srcintf "port1"
        set dstintf "port2"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set nat enable
```

```
        set ssl-ssh-profile "deep-inspection"
        set decrypted-traffic-mirror "SSL-to-port3"

THIS IS A LEGALLY BINDING AGREEMENT BETWEEN YOU, THE USER AND ITS ORGANIZATION
("CUSTOMER"), AND FORTINET. BEFORE YOU CONTINUE WITH THE TERMS AND CONDITIONS OF THIS
CONTRACT (THE "FEATURE ENABLEMENT") CAREFULLY READ THE TERMS AND CONDITIONS OF THIS
AGREEMENT. BY ENTERING YES, YOU, AS AN AUTHORIZED REPRESENTATIVE ON BEHALF OF CUSTOMER,
CONSENT TO BE BOUND BY AND BECOME A PARTY TO THIS AGREEMENT ("AGREEMENT") AND YOU
REPRESENT THAT YOU HAVE READ AND UNDERSTAND THIS AGREEMENT AND HAVE HAD SUFFICIENT
OPPORTUNITY TO CONSULT WITH COUNSEL, PRIOR TO AGREEING TO THE TERMS HEREIN AND ENABLING
THIS FEATURE. IF YOU HAVE ANY QUESTIONS OR CONCERNS, OR DESIRE TO SUGGEST ANY
MODIFICATIONS TO THIS AGREEMENT, PLEASE CONTACT YOUR FORTINET SUPPORT REPRESENTATIVE TO
BE REFERRED TO FORTINET LEGAL. IF YOU DO NOT AGREE TO ALL OF THE TERMS OF THIS
AGREEMENT, DO NOT CONTINUE WITH THE ACCEPTANCE PROCESS. BY ACCEPTING THE TERMS AND
CONDITIONS HEREIN, CUSTOMER HEREBY AGREES THAT:

1. Customer represents and warrants that Customer, not Fortinet, is engaging this
feature.

2. Customer represents and warrants that Customer has provided the requisite notice(s)
and obtained the required consent(s) to utilize this feature.

3. Customer represents and warrants that Customer will only access data as necessary in
a good faith manner to detect malicious traffic and will put in place processes and
controls to ensure this occurs.

4. Customer represents and warrants that Customer has the right to enable and utilize
this feature, and Customer is fully in compliance with all applicable laws in so doing.

5. Customer shall indemnify Fortinet in full for any of the above certifications being
untrue.

6. Customer shall promptly notify Fortinet Legal in writing of any breach of these Terms
and Conditions and shall indemnify Fortinet in full for any failure by Customer or any
of its employees or representatives to abide in full by the Terms and Conditions above.

7. Customer agrees that these Terms and Conditions shall be governed by the laws of the
State of California, without regards to the choice of laws provisions thereof and
Customer hereby agrees that any dispute related to these Terms and Conditions shall be
resolved in Santa Clara County, California, USA, and Customer hereby consents to
personal jurisdiction in Santa Clara County, California, USA.

Do you want to continue? (y/n) y
    next
end
```

Recognize anycast addresses in geo-IP blocking

An anycast IP can be advertised from multiple locations and the router selects a path based on latency, distance, cost, number of hops, and so on. This technique is widely used by providers to route users to the closest server. Since the IP is hosted in multiple geographic locations, there is no way to specify one single location to that IP.

Anycast IP address ranges can be bypassed in geo-IP blocking. The ISDB contains a list of confirmed anycast IP ranges that can be used for this purpose.

When the source or destination is set to `geoip`, you can enable the `geoip-anycast` option. Once enabled, IPs where the anycast option is set to `1` in `geoip_db` are bypassed in country matching and blocking.

You can only use the CLI to configure this feature.

To enable the geoip-anycast option using the CLI:

```
config firewall policy
   edit 1
      set name "policyid-1"
      set srcintf "wan2"
      set dstintf "wan1"
      set srcaddr "all"
      set dstaddr "test-geoip-CA_1"
      set action accept
      set schedule "always"
      set service "ALL"
      set geoip-anycast enable
      set logtraffic all
      set nat enable
   next
end
```

To check the geoip-anycast option for an IP address using the CLI:

```
diagnose geoip ip2country 1.0.0.1
   1.0.0.1 - Australia, is anycast ip
```

The anycast IP is `1.0.0.1`.

Matching GeoIP by registered and physical location

IP addresses have both a physical and registered location in the geography IP database. Sometimes these two locations are different. The `geoip-match` command allows users to match an IPv4 address in an firewall policy to its physical or registered location when a GeoIP is used as a source or destination address. IPv6 policies currently support geography address objects but do not support `geoip-match`.

In the following example, the physical location of 220.243.219.10 is CA (Canada), the registered location is CN (China), and it is not an anycast IP.

To configure GeoIP matching based on registered location:

1. Create a firewall policy to match the IP:

```
config firewall policy
    edit 1
        set name "policy_id_1"
        set srcintf "wan2"
        set dstintf "wan1"
        set srcaddr "all"
```

```
        set dstaddr "test-geoip-CA"
        set action accept
        set schedule "always"
        set service "ALL"
        set geoip-match registered-location
        set logtraffic all
        set auto-asic-offload disable
        set nat enable
    next
end
```

Since CA is applied as a destination address and registered location IP matching is enabled, if the destination IP of the traffic is 220.243.219.10, then the traffic will be blocked because the registered location is CN.

2. Verify that the policy is blocking traffic from the IP address:

```
# diagnose sniffer packet any icmp 4
interfaces=[any]
filters=[icmp]
5.383798 wan2 in 10.1.100.41 -> 220.243.219.10: icmp: echo request
6.381982 wan2 in 10.1.100.41 -> 220.243.219.10: icmp: echo request
7.382608 wan2 in 10.1.100.41 -> 220.243.219.10: icmp: echo request
^C
3 packets received by filter
0 packets dropped by kernel
```

To configure GeoIP matching based on physical location:

1. Create a firewall policy to match the IP:

```
config firewall policy
    edit 1
        set name "policy_id_1"
        set srcintf "wan2"
        set dstintf "wan1"
        set srcaddr "all"
        set dstaddr "test-geoip-CA"
        set action accept
        set schedule "always"
        set service "ALL"
        set geoip-match physical-location
        set logtraffic all
        set auto-asic-offload disable
        set nat enable
    next
end
```

Since CA is applied as a destination address and physical location IP matching is enabled, if the destination IP of the traffic is 220.243.219.10, then the traffic will pass through.

2. Verify that the policy is allowing traffic from the IP address:

```
# diagnose sniffer packet any icmp 4
interfaces=[any]
filters=[icmp]
5.273985 wan2 in 10.1.100.41 -> 220.243.219.10: icmp: echo request
5.274176 wan1 out 172.16.200.10 -> 220.243.219.10: icmp: echo request
6.274426 wan2 in 10.1.100.41 -> 220.243.219.10: icmp: echo request
6.274438 wan1 out 172.16.200.10 -> 220.243.219.10: icmp: echo request
```

```
7.273978 wan2 in 10.1.100.41 -> 220.243.219.10: icmp: echo request
7.273987 wan1 out 172.16.200.10 -> 220.243.219.10: icmp: echo request
^C
6 packets received by filter
0 packets dropped by kernel
```

HTTP to HTTPS redirect for load balancing

You can configure a virtual server with HTTP to HTTPS redirect enabled. When enabled, a virtual server can convert a client's HTTP requests to HTTPS requests. Through this mandatory conversion, HTTP traffic is converted to HTTPS traffic. This conversion improves the security of the user network.

You can only enable this feature by using the CLI. After you enable this feature, traffic flows as follows:

- When FortiGate receives an HTTP request for an external IP, such as 10.1.100.201 in the following example, FortiGate sends an HTTP 303 response back to the original client and redirects HTTP to HTTPS, instead of forwarding the HTTP request to the real backend servers.
- The client browser restarts the TCP session to HTTPS.
- The HTTPS session comes to the FortiGate where a matching firewall policy allows the HTTPS traffic and establishes a secure SSL connection, and then forwards the request to the real backend servers.

To configure virtual server with HTTPS redirect enabled:

1. Create a virtual server with `server-type` set to `http`:
```
config firewall vip
   edit "virtual-server-http"
      set type server-load-balance
      set extip 10.1.100.201
      set extintf "wan2"
      set server-type http
      set ldb-method round-robin
      set extport 80
      config realservers
         edit 1
            set ip 172.16.200.44
            set port 80
         next
         edit 2
            set ip 172.16.200.55
            set port 80
         next
      end
   next
end
```
2. Create a virtual server with `server-type` set to `https` and with the same external IP address:
```
config firewall vip
   edit "virtual-server-https"
      set type server-load-balance
      set extip 10.1.100.201
      set extintf "wan2"
      set server-type https
      set ldb-method round-robin
      set extport 443
         config realservers
            edit 1 set ip 172.16.200.44
```

```
            set port 443
         next
         edit 2
            set ip 172.16.200.55
            set port 443
         next
      end
      set ssl-certificate "Fortinet_CA_SSL"
   next
end
```

3. Enable the `http-redirect` option for the virtual server with `server-type` set to `http`:

```
config firewall vip
   edit "virtual-server-http"
      set http-redirect enable
   next
end
```

4. Add the two virtual servers to a policy:

```
config firewall policy
   edit 9
      set srcintf "wan2"
      set dstintf "wan1"
      set srcaddr "all"
      set dstaddr "virtual-server-http" "virtual-server-https"
      set action accept
      set schedule "always"
      set service "ALL"
      set inspection-mode proxy set logtraffic all
      set auto-asic-offload disable
      set nat enable
   next
end
```

Use Active Directory objects directly in policies

Active Directory (AD) groups can be used directly in identity-based firewall policies. You do not need to add remote AD groups to local FSSO groups before using them in policies.

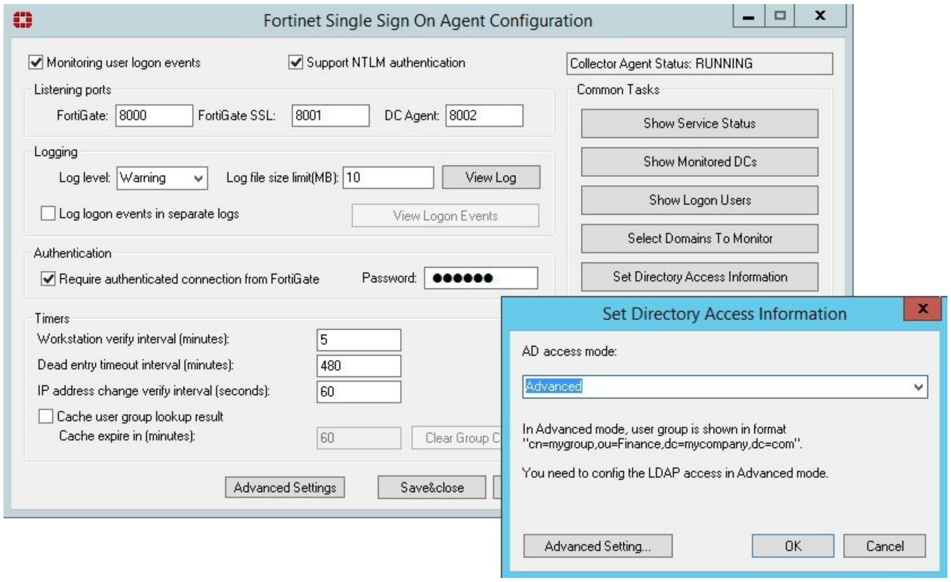
FortiGate administrators can define how often group information is updated from AD LDAP servers.

To retrieve and use AD user groups in policies:

1. Set the FSSO Collector Agent AD access mode on page 855
2. Add an LDAP server on page 856
3. Create the FSSO collector that updates the AD user groups list on page 857
4. Use the AD user groups in a policy on page 858

Set the FSSO Collector Agent AD access mode

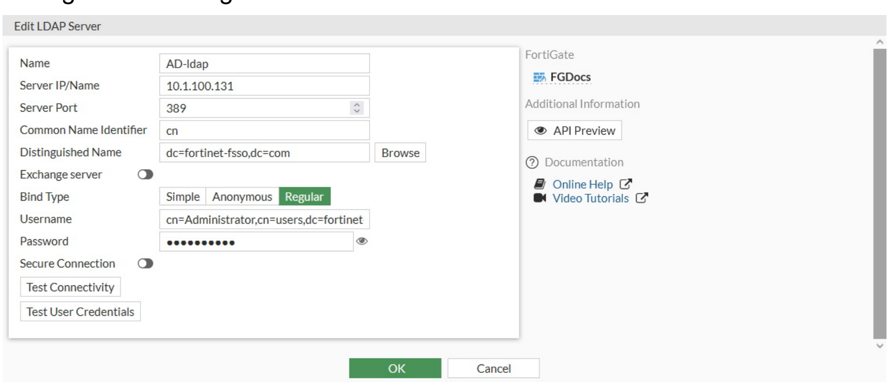
To use this feature, you must set FSSO Collector Agent to *Advanced* AD access mode. If the FSSO Collector Agent is running in the default mode, FortiGate cannot correctly match user group memberships.



Add an LDAP server

To add an LDAP server in the GUI:

1. Go to *User & Authentication > LDAP Servers*.
2. Click *Create New*.
3. Configure the settings as needed.



4. If secure communication over TLS is supported by the remote AD LDAP server:
   a. Enable *Secure Connection* .
   b. Select the protocol.
   c. Select the certificate from the CA that issued the AD LDAP server certificate.
      If the protocol is LDAPS, the port will automatically change to 636.
5. Click *OK*.

To add an LDAP server in the CLI:

```
config user ldap
    edit "AD-ldap"
        set server "10.1.100.131"
```

```
        set cnid "cn"
        set dn "dc=fortinet-fsso,dc=com"
        set type regular
        set username "cn=Administrator,cn=users,dc=fortinet-fsso,dc=com"
        set password XXXXXXXXXXXXXXXXXXXXXXXX
    next
end
```

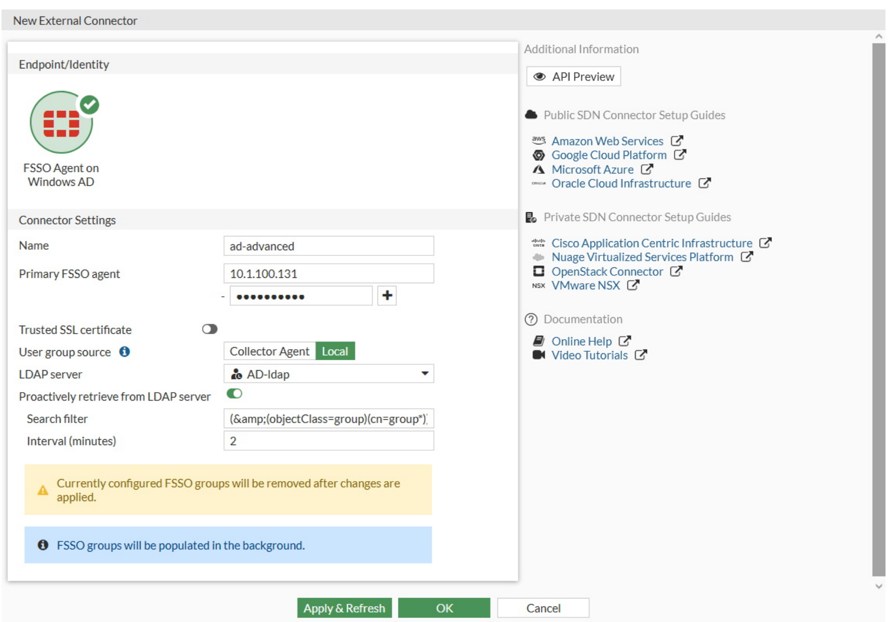
Create the FSSO collector that updates the AD user groups list

To create an FSSO agent connector in the GUI:

1. Go to *Security Fabric > External Connectors*.
2. Click *Create New*.
3. In the *Endpoint/Identity* section, click *FSSO Agent on Windows AD*.
4. Fill in the *Name*
5. Set the *Primary FSSO Agent* to the IP address of the FSSO Collector Agent, and enter its password.
6. Set the *User Group Source* to *Local*.
7. Set the *LDAP Server* to the just created *AD-ldap* server.
8. Enable *Proactively Retrieve from LDAP Server*.
9. Set the *Search Filter* to *(&(objectClass=group)(cn=group*))*.

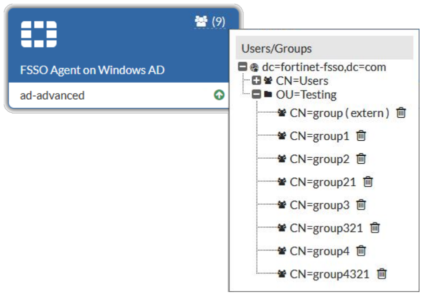
   The default search filter retrieves all groups, including Microsoft system groups. In this example, the filter is configured to retrieve *group1*, *group2*, etc, and not groups like *grp199*.

   The filter syntax is not automatically checked; if it is incorrect, the FortiGate might not retrieve any groups.
10. Set the *Interval (minutes)* to configure how often the FortiGate contacts the remote AD LDAP server to update the group information.



11. Click *OK*.
12. To view the AD user groups that are retrieved by the FSSO agent, hover the cursor over the group icon on the fabric

connector listing.



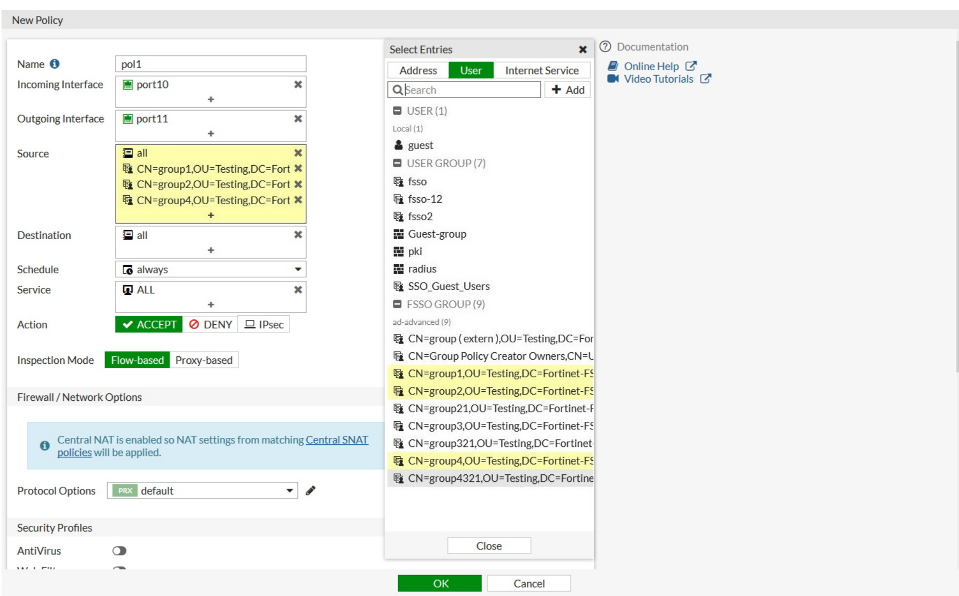
To create an FSSO agent connector in the CLI:

```
config user fsso
    edit "ad-advanced"
        set server "10.1.100.131"
        set password XXXXXXXXXXXXXX
        set ldap-server "AD-ldap"
        set ldap-poll enable
        set ldap-poll-interval 2
        set ldap-poll-filter "(&amp;(objectClass=group)(cn=group*))"
    next
end
```

You can view the retrieved AD user groups with the `show user adgrp` command.

Use the AD user groups in a policy

The AD user groups retrieved by the FortiGate can be used directly in firewall policies.



No session timeout

To allow clients to permanently connect with legacy medical applications and systems that do not have keepalive or auto-reconnect features, the session timeout can be set to never for firewall services, policies, and VDOMs.

The options to disable session timeout are hidden in the CLI.

To set the session TTL value of a custom service to never:

```
config firewall service custom
    edit "tcp_23"
        set tcp-portrange 23
        set session-ttl never
    next
end
```

To set the session TTL value of a policy to never:

```
config firewall policy
    edit 201
        set srcintf "wan1"
        set dstintf "wan2"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "TCP_8080"
        set logtraffic disable
        set session-ttl never
        set nat enable
    next
end
```

To set the session TTL value of a VDOM to never:

```
config system session-ttl
    set default never
    config port
        edit 1
            set protocol 6
            set timeout never
            set start-port 8080
            set end-port 8080
        next
    end
end
```

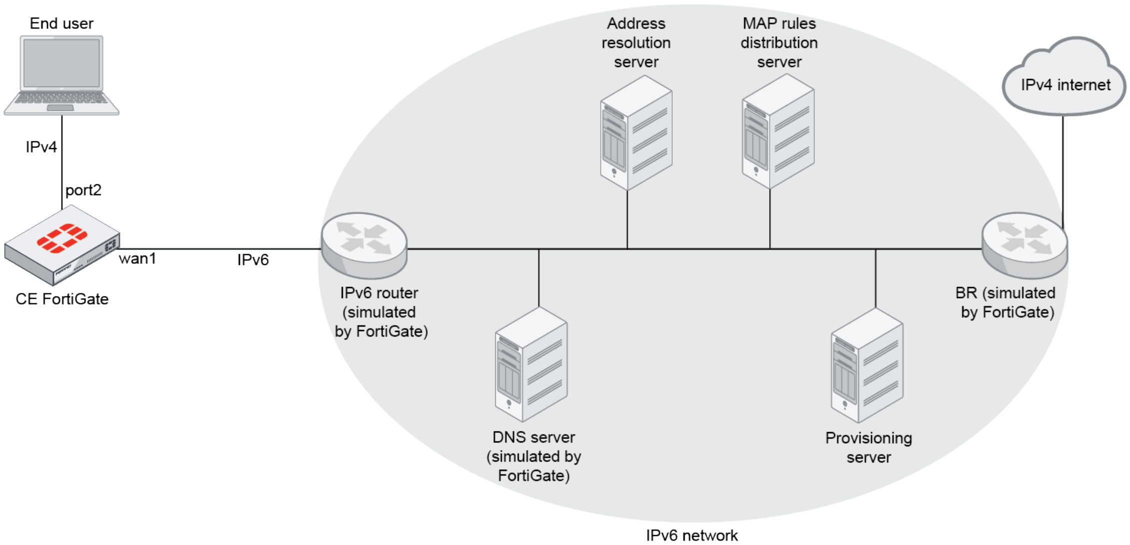
To view a session list with the timeout set to never:

```
# diagnose sys session list

session info: proto=6 proto_state=01 duration=9 expire=never timeout=never flags=00000000
sockflag=00000000 sockport=0 av_idx=0 use=3
origin-shaper=
reply-shaper=
```

```
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/255
state=log may_dirty f00
statistic(bytes/packets/allow_err): org=2290/42/1 reply=2895/34/1 tuples=2
tx speed(Bps/kbps): 238/1 rx speed(Bps/kbps): 301/2
orgin->sink: org pre->post, reply pre->post dev=18->17/17->18 gwy=172.16.200.55/10.1.100.41
hook=post dir=org act=snat 10.1.100.41:34256->172.16.200.55:23(172.16.200.10:34256)
hook=pre dir=reply act=dnat 172.16.200.55:23->172.16.200.10:34256(10.1.100.41:34256)
pos/(before,after) 0/(0,0), 0/(0,0)
misc=0 policy_id=9 auth_info=0 chk_client_info=0 vd=1
serial=00000b27 tos=ff/ff app_list=0 app=0 url_cat=0
sdwan_mbr_seq=0 sdwan_service_id=0
rpdb_link_id = 00000000 ngfwid=n/a
dd_type=0 dd_mode=0
npu_state=0x000001 no_offload
no_ofld_reason:  disabled-by-policy
total session 1
```

MAP-E support

On a customer edge (CE) FortiGate, an IPv4-over-IPv6 (MAP-E) tunnel can be created between the FortiGate and the border relay (BR) operating in an IPv6 network. A tunnel interface is created between the FortiGate and BR, which can be applied to firewall policies and IPsec VPN.



To configure a MAP-E tunnel between the FortiGate and the BR:

1. Configure fixed IP mode.
   a. Configure IPv6 on the interface:

```
config system interface
    edit "wan1"
        config ipv6
            set autoconf enable
            set unique-autoconf-addr enable
            set interface-identifier ::6f:6c1f:3400:0
```

```
        end
    next
end
```

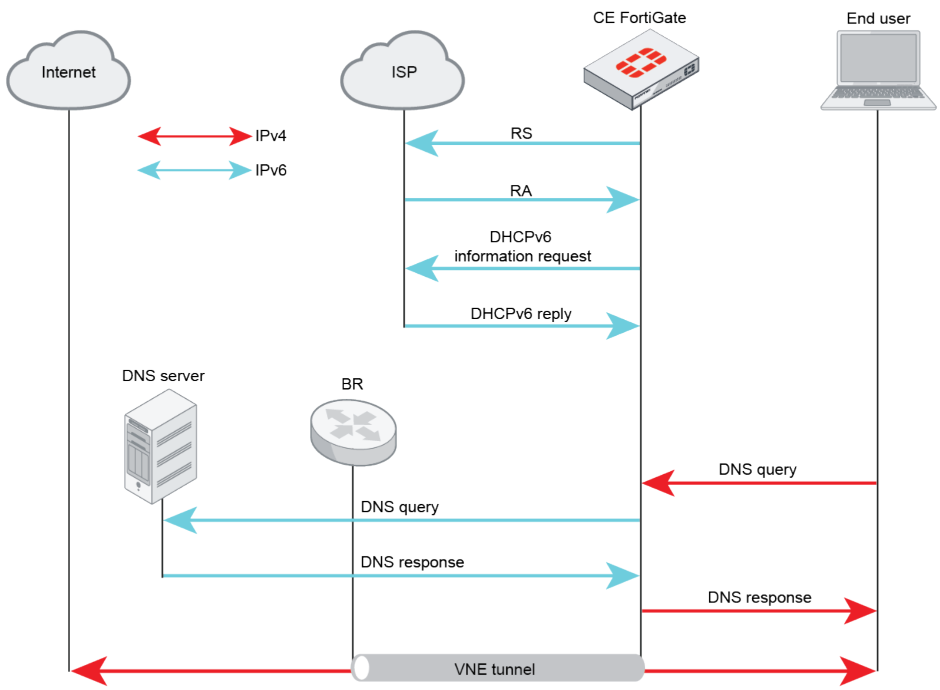
The `interface-identifier` is an IPv6 address. Its last 64-bit will be kept and the rest will be cleared automatically. It will combine with the IPv6 prefix it gets from the IPv6 router to generate the IPv6 address of the interface.

By default, `unique-autoconf-addr` is disabled. It must be enabled so it can handle IPv6 prefix changing.

b. Configure the VNE tunnel:

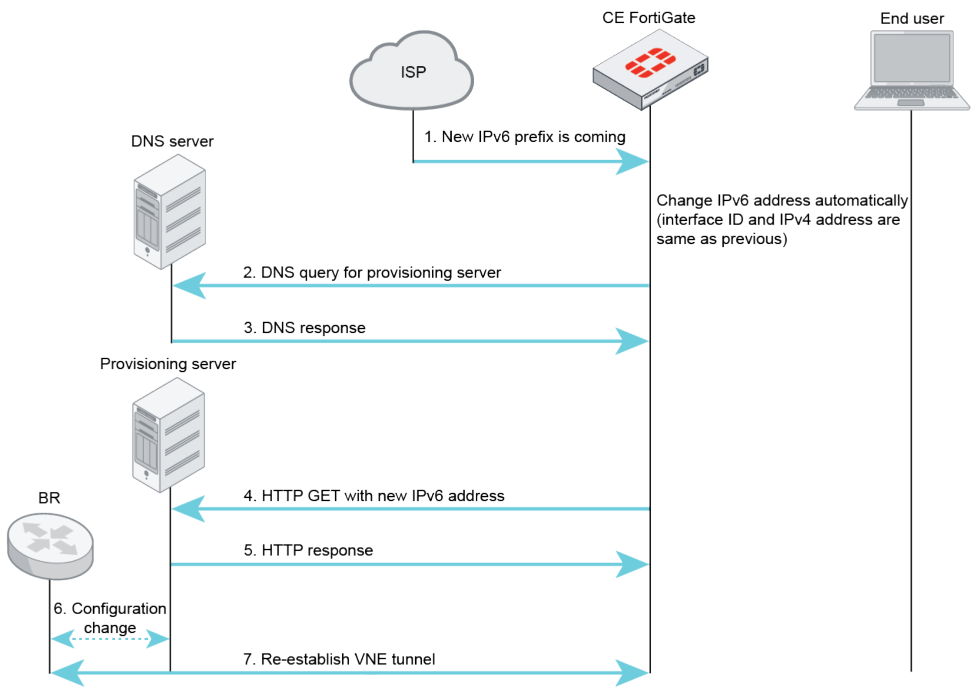
```
config system vne-tunnel
    set status enable
    set interface "wan1"
    set mode fixed-ip
    set ipv4-address 10.10.81.81 255.255.255.0
    set br 2001:160::82
    set update-url "http://qa.forosqa.com/update?user=xxxx&pass=yyyy"
end
```

Initial sequence overview of VNE tunnel under fixed IP mode:



Once the IPv6 address of the FortiGate changes, the tunnel will be down because the BR does not know the FortiGate's new IPv6 address. The FortiGate uses `update-url` to update the new IPv6 address to the provisioning server. The provisioning server updates the FortiGate’s IPv6 address to the BR so the VNE tunnel can be re-established.

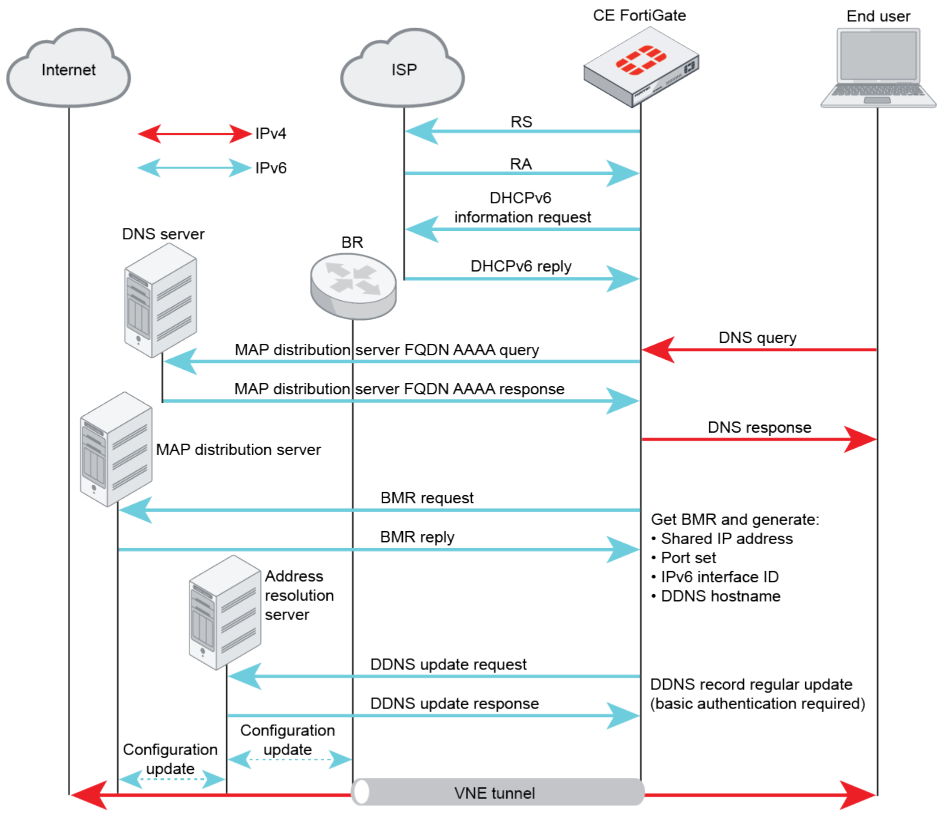
Communication sequence overview of re-establishing VNE tunnel:



2. Configure the VNE tunnel to use MAP-E mode:

```
config system vne-tunnel
    set status enable
    set interface 'wan1'
    set ssl-certificate "Fortinet_Factory"
    set bmr-hostname ********
    set auto-asic-offload enable
    set mode map-e
end
```

Initial sequence overview of VNE tunnel under MAP-E mode:



The FortiGate sends a MAP rule request to the MAP distribution server once the IPv6 address is configured on the FortiGate by RS/RA. Next, the FortiGate will send an AAAA query to get the IPv6 address of the MAP distribution server. After sending the BMR request to the MAP distribution server, the FortiGate will get the IPv4 address, port set, BR IPv6 address, and hostname of the address resolution server from the BMR reply. The VNE tunnel between the FortiGate and BR is now established.

The address resolution server is actually a dynamic DNS. The hostname is used for the FortiGate to maintain an IPv6 address when it changes.

The FortiGate updates the DDNS server with its IPv6 address whenever it updates, which in turn provides the update to the MAP distribution server and BR so they know how to resolve the FortiGate by hostname.

Once the VNE tunnel is established, a tunnel interface is created (`vne.root`), and an IPv4-over-IPv6 tunnel is set up between the FortiGate and BR. The route, firewall policy, and DNS server can now be configured to let the traffic go through the VNE tunnel and the and protect the end-user. The VNE tunnel can also be used in IPsec phase 1.

3. Configure the route:

```
config router static
    edit 1
        set device "vne.root"
    next
end
```

4. Configure the firewall policy:

```
config firewall policy
    edit 111
        set name "ff"
        set srcintf "port2"
        set dstintf "vne.root"
```

```
            set srcaddr "all"
            set dstaddr "all"
            set action accept
            set schedule "always"
            set service "ALL"
            set utm-status enable
            set ssl-ssh-profile "certificate-inspection"
            set av-profile "default"
            set nat enable
        next
    end
```

5. Configure the DNS server:

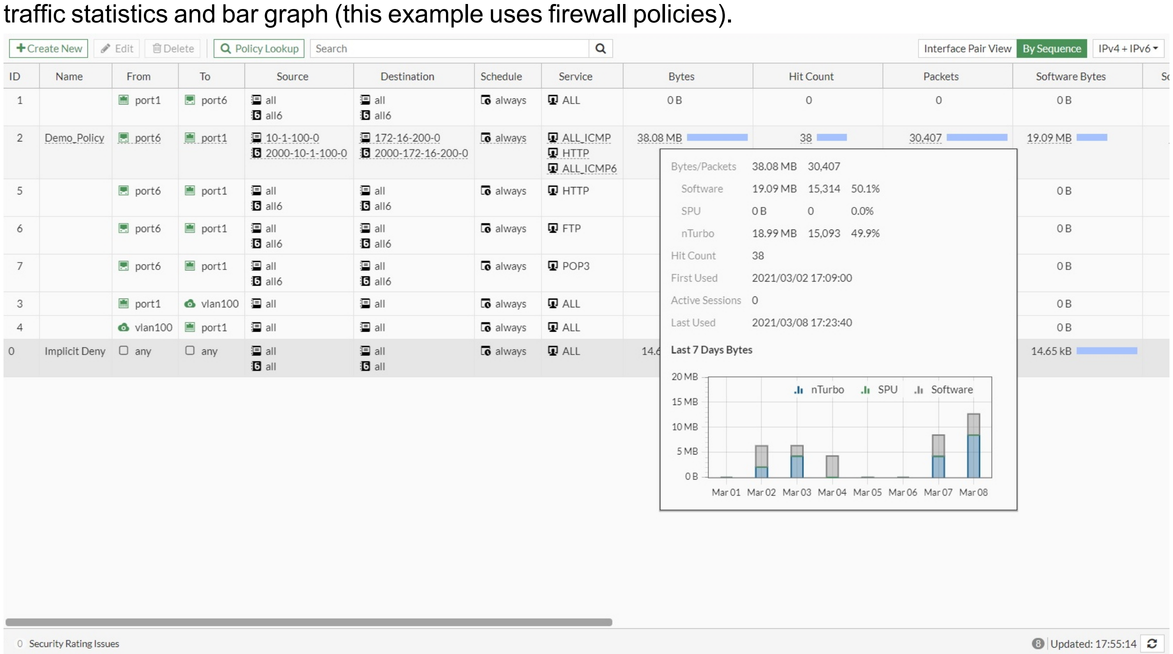
```
config system dns-server
    edit "port2"
    next
end
```

Seven-day rolling counter for policy hit counters

Instead of storing a single number for the hit count and byte count collected since the inception of each policy, seven numbers for the last seven days and an active counter for the current day are stored. The past seven-day hit count is displayed in the policy list and policy pages. A seven-day bar chart shows statistics on each policy page. This feature is currently supported in firewall and multicast policies, but not security policies.

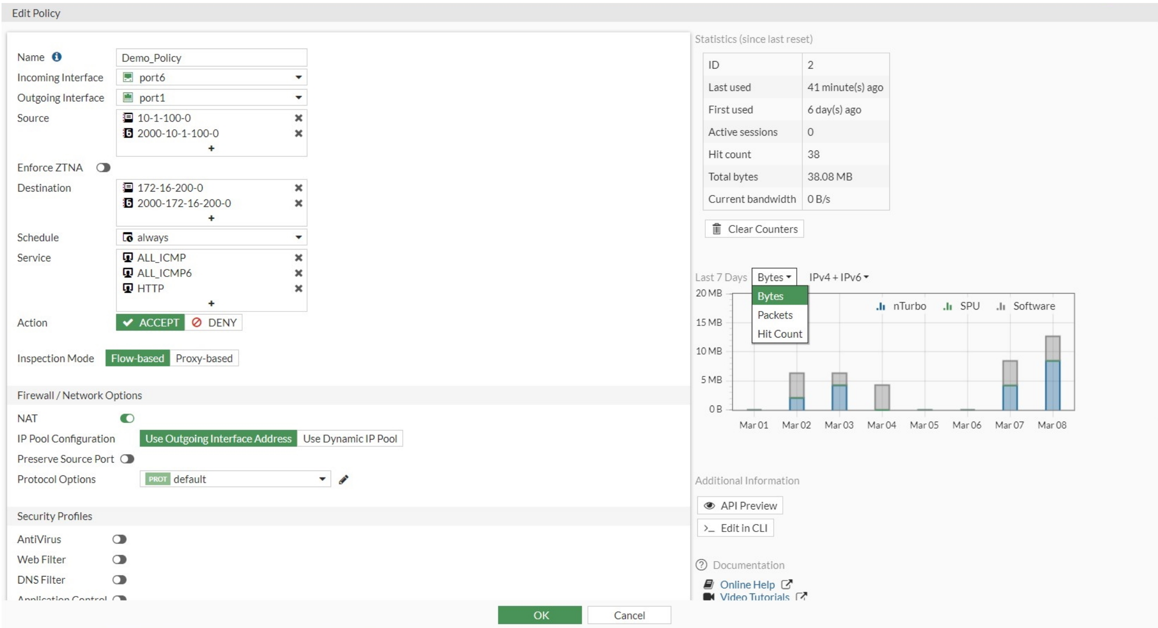
To view the rolling counter information in the GUI:

1. Go to *Policy & Objects > Firewall Policy* or *Policy & Objects > Multicast Policy*.
2. Select a policy and hover over the *Bytes*, *Packets*, or *Hit Count* values to view the tooltip with the corresponding traffic statistics and bar graph (this example uses firewall policies).



3. Click *Edit*. The policy traffic statistics appear in the right-hand side of the page.

4. Use the dropdowns to filter the bar graph data by counter (*Bytes*, *Packets*, or *Hit Count*) and policy type (*IPv4*, *IPv6*, or *IPv4 + IPv6*).



5. Optionally, click *Clear Counters* to delete the traffic statistics for the policy.
6. Click *OK*.

To view the rolling counter information in the CLI:

```
# diagnose firewall iprope show 100004 2
idx=2 pkts/bytes=14709/18777329 asic_pkts/asic_bytes=8087/10413737 nturbo_pkts/nturbo_
bytes=8087/10413737 flag=0x0 hit count:19 (4 7 0 1 1 3 3 0)
    first:2021-03-02 17:09:00 last:2021-03-08 17:23:40
 established session count:0
    first est:2021-03-02 17:11:20 last est:2021-03-08 17:23:40
```

```
# diagnose firewall iprope6 show 100004 2
idx=2 pkts/bytes=15698/19307164 asic_pkts/asic_bytes=7006/8578911 nturbo_pkts/nturbo_
bytes=7006/8578911 flag=0x0 hit count:19 (4 7 0 1 3 2 2 0)
    first:2021-03-02 17:10:32 last:2021-03-08 17:23:33
 established session count:0
    first est:2021-03-02 17:11:43 last est:2021-03-08 17:23:33
```

Cisco Security Group Tag as policy matching criteria

The FortiGate can read the Cisco Security Group Tag (SGT) in Ethernet frames, and use them as matching criteria in firewall policies. A policy can match based on the presence of an SGT, or the detection of a specific ID or IDs.

When a packet with a SGT passes through and a session is established, the `ext_header_type=0xc5:0xc5` flag is included in the session table.

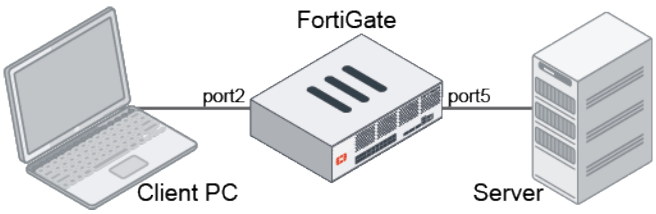
This feature is available in flow mode policies for virtual wire pair policies or policies in transparent mode VDOMs.

To configure a firewall policy to detect SGTs in Ethernet frames:

```
config firewall policy
    edit 1
        set sgt-check {enable | disable}
        set sgt <ID numbers>
    next
end
```

Examples

In these examples, port2 and port5 are in a virtual wire pair. Firewall policies are created that pass traffic with SGTs with a specific ID number, any ID number, or either of two specific ID numbers.



To configure the virtual wire pair:

```
config system virtual-wire-pair
    edit "test-vwp-1"
        set member "port5" "port2"
        set wildcard-vlan enable
    next
end
```

To configure a firewall policy to match frames that have an SGT with ID 20 and allow them through:

```
config firewall policy
    edit 1
        set srcintf "port2"
        set dstintf "port5"
        set action accept
        set srcaddr "all"
        set dstaddr "all"
        set schedule "always"
        set service "ALL"
        set sgt-check enable
        set sgt 20
    next
end
```

To configure a firewall policy to match frames that have an SGT with any ID:

```
config firewall policy
    edit 1
        set srcintf "port2"
        set dstintf "port5"
        set action accept
        set srcaddr "all"
```

```
        set dstaddr "all"
        set schedule "always"
        set service "ALL"
        set sgt-check enable
    next
end
```

To configure a firewall policy to match frames that have the SGT with IDs 20 or 21:

```
config firewall policy
    edit 1
        set srcintf "port2"
        set dstintf "port5"
        set action accept
        set srcaddr "all"
        set dstaddr "all"
        set schedule "always"
        set service "ALL"
        set sgt-check enable
        set sgt 20 21
    next
end
```

To check the session list:

```
# diagnose sys session list

session info: proto=6 proto_state=01 duration=10 expire=3593 timeout=3600 flags=00000000
socktype=0 sockport=0 av_idx=0 use=3
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/0
state=log may_dirty br dst-vis f00
statistic(bytes/packets/allow_err): org=112/2/1 reply=60/1/1 tuples=2
tx speed(Bps/kbps): 10/0 rx speed(Bps/kbps): 5/0
orgin->sink: org pre->post, reply pre->post dev=13->10/10->13 gwy=0.0.0.0/0.0.0.0
hook=pre dir=org act=noop 10.1.1.11:36970->10.1.2.11:80(0.0.0.0:0)
hook=post dir=reply act=noop 10.1.2.11:80->10.1.1.11:36970(0.0.0.0:0)
pos/(before,after) 0/(0,0), 0/(0,0)
dst_mac=00:b0:e1:22:cf:e4
misc=0 policy_id=1 auth_info=0 chk_client_info=0 vd=1
serial=0000183c tos=ff/ff app_list=0 app=0 url_cat=0
sdwan_mbr_seq=0 sdwan_service_id=0
rpdb_link_id=00000000 rpdb_svc_id=0 ngfwid=n/a
npu_state=0x000001 no_offload
no_ofld_reason:  disabled-by-policy
ext_header_type=0xc5:0xc5
total session 1
```

Objects

The following topics provide information about objects:

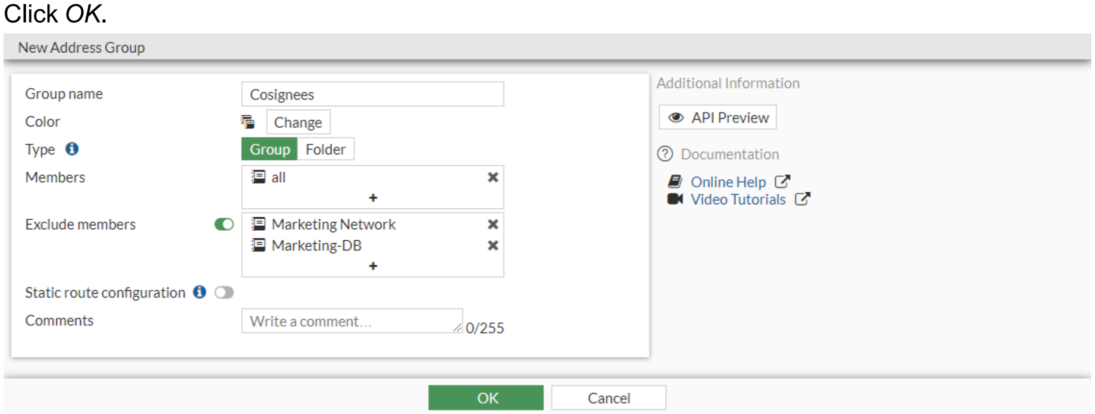
Address group exclusions

Specific IP addresses or ranges can be subtracted from the address group with the *Exclude Members* setting in IPv4 address groups.

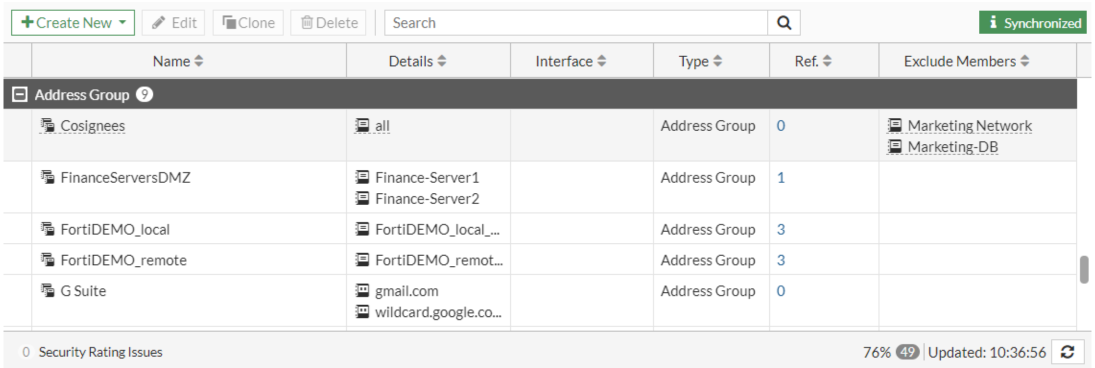
This feature is only supported for IPv4 address groups, and only for addresses with a *Type* of *IP Range* or *Subnet*.

To exclude addresses from an address group using the GUI:

1. Go to *Policy & Objects > Addresses*.
2. Create a new address group, or edit an existing address group.
3. Enable *Exclude Members* and click the + to add entries.
4. Configure the other settings as needed.
5. Click *OK*.



The excluded members are listed in the *Exclude Members* column.



To exclude addresses from an address group using the CLI:

```
config firewall addrgrp
    edit <address group>
        set exclude enable
        set exclude-member <address> <address> ... <address>
    next
end
```

MAC addressed-based policies

MAC addresses can be added to the following IPv4 policies:

- Firewall
- Virtual wire pair
- ACL
- Central SNAT
- DoS

A MAC address is a link layer-based address type and it cannot be forwarded across different IP segments. In FortiOS, you can configure a firewall address object with a singular MAC, wildcard MAC, multiple MACs, or a MAC range.

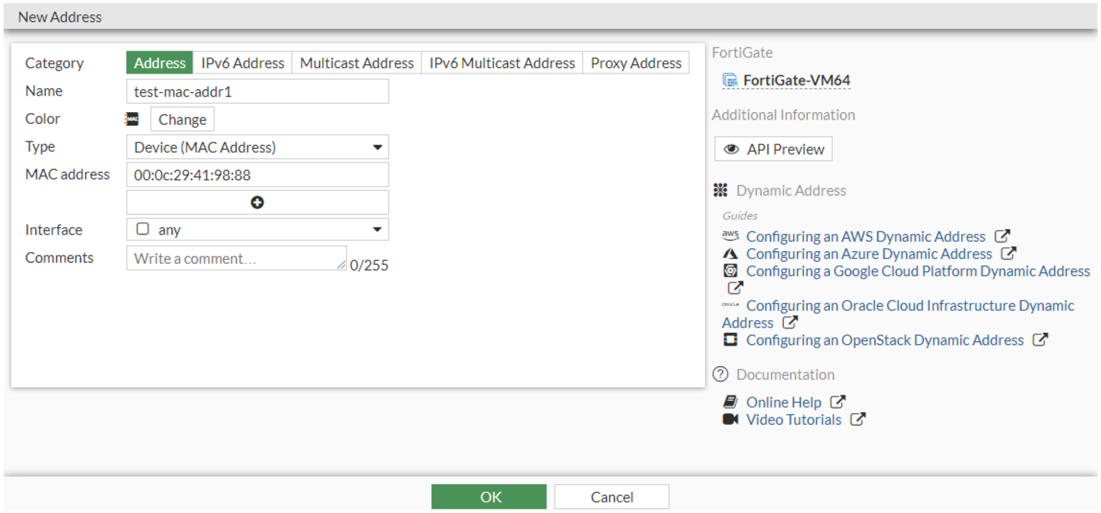
FortiOS only supports the MAC address type as source address for policies in NAT mode VDOM. When you use the MAC address type in a policy as source address in NAT mode VDOM, IP address translation (NAT) is still performed according to the rules defined in the policy. The MAC address type only works for source address matching. It does not have any association with NAT actions.

For policies in transparent mode or the virtual wire pair interface, you can use the MAC address type as source or destination address.

To configure a MAC address using the GUI:

1. Go to *Policy & Objects > Addresses* and click *Create New > Address*.
2. Enter a name.
3. For *Category*, select *Address*.
4. For *Type*, select *Device (MAC Address)*.

5. Enter the MAC address.



6. Click *OK*.
7. Go to *Policy & Objects > Firewall Policy* to apply the address type to a policy in NAT mode VDOM:
   a. For *Source*, select the MAC address you just configured.
   b. For *Destination*, select an address.

   In NAT mode VDOM, this address type cannot be used as destination address.

   c. Configure the other settings as needed.
   d. Click *OK*.

To configure a MAC address using the CLI:

1. Create a new MAC address:

```
config firewall address
    edit "test-mac-addr1"
        set type mac
        set macaddr 00:0c:29:41:98:88
    next
end
```

2. Apply the address type to a policy. In transparent mode or the virtual wire pair interface, this address type can be mixed with other address types in the policy:

```
config firewall policy
    edit 1
        set srcintf "port2"
        set dstintf "port1"
        set srcaddr "test-mac-addr1" "10-1-100-42"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set logtraffic all
```

```
        set nat enable
    next
end
```

ISDB well-known MAC address list

The Internet Service Database (ISDB) includes well-known vendor MAC address range lists. The lists can only be used for source MAC addresses in IPv4 policies, and include the vendor name and the MAC address ranges that the vendor belongs to.

To view the vendor list:

```
# diagnose vendor-mac id
Please input Vendor MAC ID.
ID: 1 name: "Asus"
ID: 2 name: "Acer"
ID: 3 name: "Amazon"
ID: 4 name: "Apple"
ID: 5 name: "Xiaomi"
ID: 6 name: "BlackBerry"
ID: 7 name: "Canon"
ID: 8 name: "Cisco"
ID: 9 name: "Linksys"
ID: 10 name: "D-Link"
ID: 11 name: "Dell"
ID: 12 name: "Ericsson"
ID: 13 name: "LG"
ID: 14 name: "Fujitsu"
ID: 15 name: "Fitbit"
ID: 16 name: "Fortinet"
ID: 17 name: "OPPO"
ID: 18 name: "Hitachi"
ID: 19 name: "HTC"
ID: 20 name: "Huawei"
ID: 21 name: "HP"
ID: 22 name: "IBM"
ID: 23 name: "Juniper"
ID: 24 name: "Lenovo"
ID: 25 name: "Microsoft"
ID: 26 name: "Motorola"
ID: 27 name: "Netgear"
ID: 28 name: "Nokia"
ID: 29 name: "Nintendo"
ID: 30 name: "PaloAltoNetworks"
ID: 31 name: "Polycom"
ID: 32 name: "Samsung"
ID: 33 name: "Sharp"
ID: 34 name: "Sony"
ID: 35 name: "Toshiba"
ID: 36 name: "VMware"
ID: 37 name: "Vivo"
ID: 38 name: "Zyxel"
ID: 39 name: "ZTE"
```

To view the MAC address ranges for a vendor:

```
# diagnose vendor-mac id 16
Vendor MAC: 16(Fortinet)
Version: 0000700021
Timestamp: 201908081432
Number of MAC ranges: 6
00:09:0f:00:00:00 - 00:09:0f:ff:ff:ff
04:d5:90:00:00:00 - 04:d5:90:ff:ff:ff
08:5b:0e:00:00:00 - 08:5b:0e:ff:ff:ff
70:4c:a5:00:00:00 - 70:4c:a5:ff:ff:ff
90:6c:ac:00:00:00 - 90:6c:ac:ff:ff:ff
e8:1c:ba:00:00:00 - e8:1c:ba:ff:ff:ff
```

To query the vendor of a specific MAC address or range:

```
# diagnose vendor-mac match 00:09:0f:ff:ff:ff 48
Vendor MAC: 16(Fortinet), matched num: 1
```

To use the vendor ID in a firewall policy:

```
config firewall policy
    edit 9
        set name "policy_id_9"
        set uuid 6150cf30-308d-51e9-a7a3-bcbd05d61f93
        set srcintf "wan2"
        set dstintf "wan1"
        set srcaddr "all"
        set dstaddr "all"
        set vendor-mac 36 16
        set action accept
        set schedule "always"
        set service "ALL"
        set logtraffic all
        set auto-asic-offload disable
        set nat enable
    next
end
```

Only packets whose source MAC address belong to Fortinet or VMware are passed by the policy.

Dynamic policy — fabric devices

The dynamic address group represents the configured IP addresses of all Fortinet devices connected to the Security Fabric. It currently includes FortiManager, FortiAnalyzer, FortiClient EMS, FortiMail, FortiAP(s), and FortiSwitch(es). Like other dynamic address groups for fabric connectors, it can be used as an IPv4 address in firewall policies and objects.

The list of firewall addresses includes a default address object called `FABRIC_DEVICE`. You can apply the `FABRIC_DEVICE` object to the following types of policies:

- Firewall policy, including virtual wire pairs, NAT 46, and NAT 64 (IPv4 only)
- IPv4 shaping policy
- IPv4 ACL policy

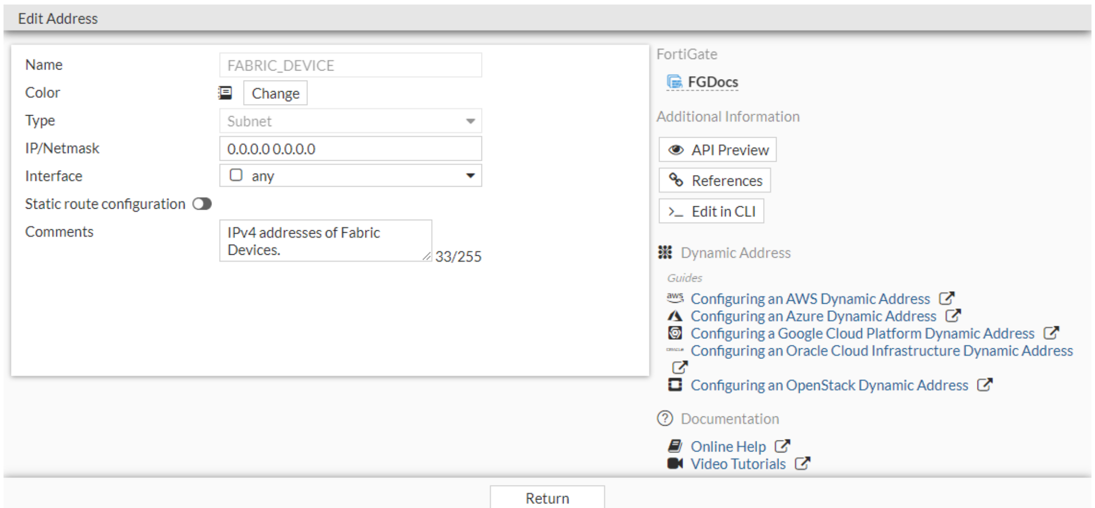
You cannot apply the `FABRIC_DEVICE` object to the following types of policies:

- IPv4 explicit proxy policy

You also cannot use the `FABRIC_DEVICE` object with the following settings:

- Custom extension on `internet-service`
- Exclusion of `addrgrp`

Initially the `FABRIC_DEVICE` object does not have an address value. The address value is populated dynamically as things change. As a result, you cannot edit the `FABRIC_DEVICE` object, add any addresses to the object, or remove any addresses from the object. The *Edit Address* pane in the GUI only has a *Return* button because the object is read-only:



The `FABRIC_DEVICE` object address values are populated based on:

- FortiAnalyzer IP (from the *Fabric Settings* pane)
- FortiManager IP (from the *Fabric Settings* pane)
- FortiMail IP (from the *Fabric Settings* pane)
- FortiClient EMS IP (from the *Fabric Settings* pane)
- FortiAP IPs (from the *FortiAP Setup* pane or DHCP)
- FortiSwitch IPs (from the *FortiSwitch Setup* page or DHCP)

To apply the FABRIC_DEVICE object to a firewall policy using the GUI:

1. Go to *Policy & Objects > Firewall Policy*.
2. Create a new policy or edit an existing policy.
3. For the *Destination* field, select *FABRIC_DEVICE* from the list of address entries.
4. Configure the rest of the policy as needed.
5. Click *OK*.

To apply the FABRIC_DEVICE object to a firewall policy using the CLI:

```
config firewall address
    edit "FABRIC_DEVICE"
        set type ipmask
        set comment "IPv4 addresses of Fabric Devices."
        set visibility enable
```

```
        set associated-interface ''
        set color 0
        set allow-routing disable
        set subnet 0.0.0.0 0.0.0.0
    next
end
```

```
config firewall policy
    edit 1
        set srcintf "port2"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "FABRIC_DEVICE"
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set fsso disable
        set nat enable
    next
end
```

Diagnose command

You can use the diagnose command to list IP addresses of Fortinet devices that are configured in the Security Fabric.

To run the diagnose command using the CLI:

```
(root) # diagnose firewall sf-addresses list

FabricDevices: 172.18.64.48
FortiAnalyzer: 172.18.60.25
FortiSandbox: 172.18.52.154
FortiManager: 172.18.28.31
FortiClientEMS: 172.18.62.6
FortiAP:
FortiSwitch:
FortiAP/SW-DHCP:
```

FSSO dynamic address subtype

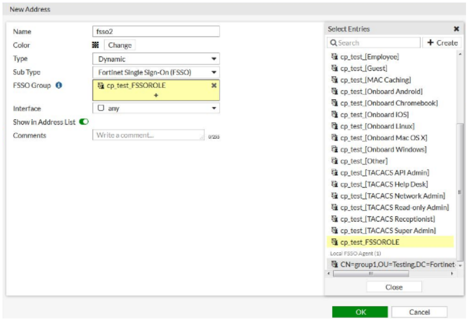
The Fortinet Single Sign-ON (FSSO) dynamic firewall address subtype can be used in policies that support dynamic address types. The FortiGate will update the dynamic address used in firewall policies based on the source IP information for the authenticated FSSO users.

It can also be used with FSSO group information that is forwarded by ClearPass Policy Manager (CPPM) via FortiManager, and other FSSO groups provided by the FSSO collector agent or FortiNAC.

To configure FSSO dynamic addresses with CPPM and FortiManager in the GUI:

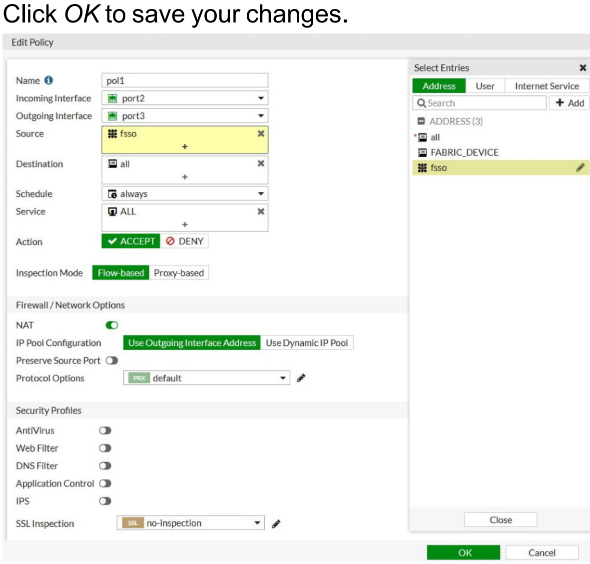
1. Create the dynamic address object:
   a. Go to *Policy & Objects > Addresses*, and click *Create New > Address*.
   b. For *Type*, select *Dynamic*.

c. For *Sub Type*, select *Fortinet Single Sign-On (FSSO)*. The *Select Entries* pane opens and displays all available FSSO groups.
d. Select one or more groups.
e. Click *OK* to save the configuration.



In the address table, there will be an error message for the address you just created (*Unresolved dynamic address: fsso*). This is expected because there are currently no authenticated FSSO users (based on source IP) in the local FSSO user list.

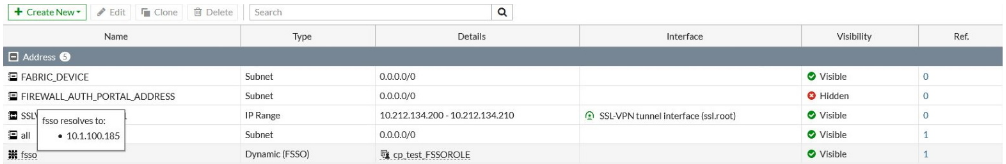
2. Add the dynamic address object to a firewall policy:
   a. Go to *Policy & Objects > Firewall Policy*.
   b. Create a new policy or edit an existing policy.
   c. For *Source*, add the dynamic FSSO address object you just created.
   d. Configure the rest of the policy as needed.
   e. Click *OK* to save your changes.



3. Test the authentication to add a source IP address to the FSSO user list:
   a. Log in as user and use CPPM for user authentication to connect to an external web server. After successful authentication, CPPM forwards the user name, source IP address, and group membership to the FortiGate via FortiManager.
   b. Go to *Monitor > Firewall User Monitor* to view the user name (*fsso1*) and IP address.

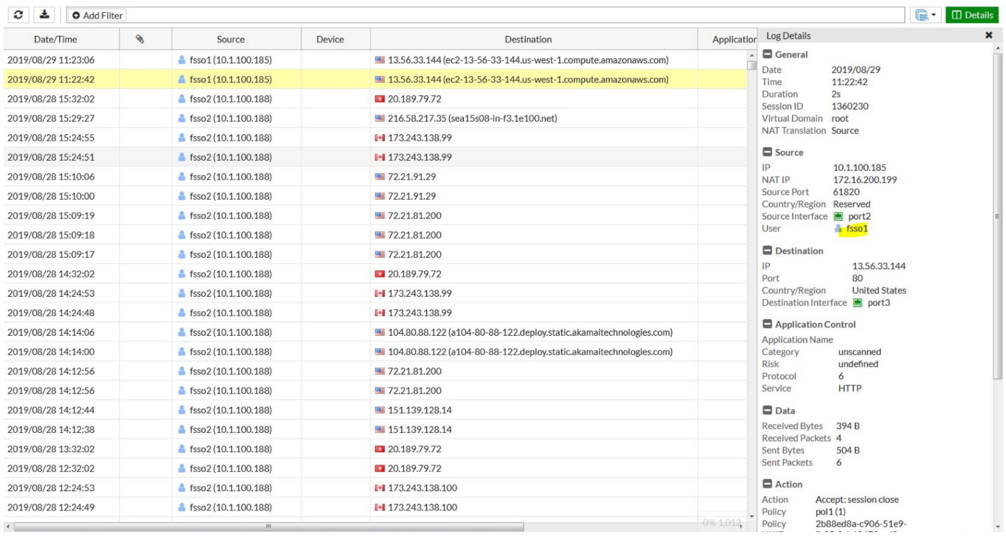
   c. Go to *Policy & Objects > Addresses* to view the updated address table. The error message no longer appears.

d. Hover over the dynamic FSSO address to view the IP address (*fsso resolves to: 10.1.100.185*).



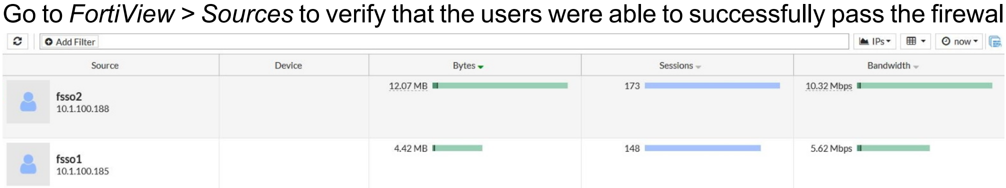
To verify user traffic in the GUI:

1. Go to *Log & Report > Forward Traffic*.
Details for the user *fsso1* are visible in the traffic log:



- If another user is authenticated by CPPM, then the dynamic address *fsso* entry in the address table will be updated. The IP address for user *fsso2* (*10.1.100.188*) is now visible:

2. Go to *FortiView > Sources* to verify that the users were able to successfully pass the firewall policy.



If a user logs off and CPPM receives log off confirmation, then CPPS updates the FortiGate FSSO user list via FortiManager. The user IP address is deleted from the dynamic FSSO address, and the user is no longer be able to pass the firewall policy.

To configure FSSO dynamic addresses with CPPM and FortiManager in the CLI:

1. Create the dynamic address object:

```
config firewall address
    edit "fsso"
        set type dynamic
        set sub-type fsso
        set fsso-group "cp_test_FSSOROLE"
    next
end
```

2. Add the dynamic address object to a policy:

```
config firewall policy
    edit 1
        set name "pol1"
        set srcintf "port2"
        set dstintf "port3"
        set srcaddr "fsso"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set logtraffic all
        set fsso disable
        set nat enable
    next
end
```

To verify user traffic in the CLI:

1. Check the FSSO user list:

```
diagnose debug authd fsso list
----FSSO logons----
IP: 10.1.100.185  User: fsso1  Groups: cp_test_FSSOROLE  Workstation:  MemberOf: FSSO-
CPPM cp_test_FSSOROLE
Total number of logons listed: 1, filtered: 0
----end of FSSO logons----
```

2. Check the authenticated firewall users list:

```
diagnose firewall auth list
10.1.100.185, fsso1
type: fsso, id: 0, duration: 2928, idled: 2928
server: FortiManager
packets: in 0 out 0, bytes: in 0 out 0
group_id: 2 33554433
group_name: FSSO-CPPM cp_test_FSSOROLE
----- 1 listed, 0 filtered ------
```

After user traffic passes through the firewall, the nu

```
diagnose firewall auth list
10.1.100.185, fsso1
type: fsso, id: 0, duration: 3802, idled: 143
server: FortiManager
packets: in 1629 out 1817, bytes: in 2203319 out 133312
```

```
group_id: 2 33554433
group_name: FSSO-CPPM cp_test_FSSOROLE
----- 1 listed, 0 filtered ------
```

ClearPass integration for dynamic address objects

ClearPass Policy Manager (CPPM) can gather information about the statuses of network hosts, for example, the latest patches or virus infections. Based on this information, CPPM send the IP addresses and current states, such as Healthy or Infected, to the FortiGate.

On the FortiGate, the IP addresses received from CPPM are added to a dynamic firewall address with the *clearpass-spt* subtype. This address can be used in any policy that supports dynamic addresses, such as Firewall or SSL-VPN policies.

In this example, you create two dynamic IP addresses that are used in two firewall policies (deny and allow). One policy allows traffic (host state = Healthy), and the other denies traffic (host state = Infected). When CPPM sends the information, the IP addresses are assigned according to their host state: Healthy or Infected.

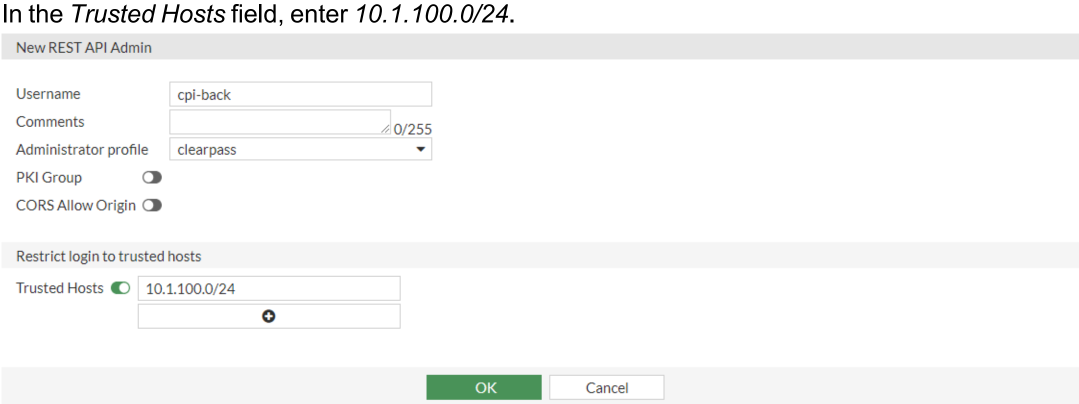
You can then verify that traffic from the Infected host is denied access by the deny policy, and traffic from the Healthy host is allowed access by the allow policy.

Create a REST API administrator

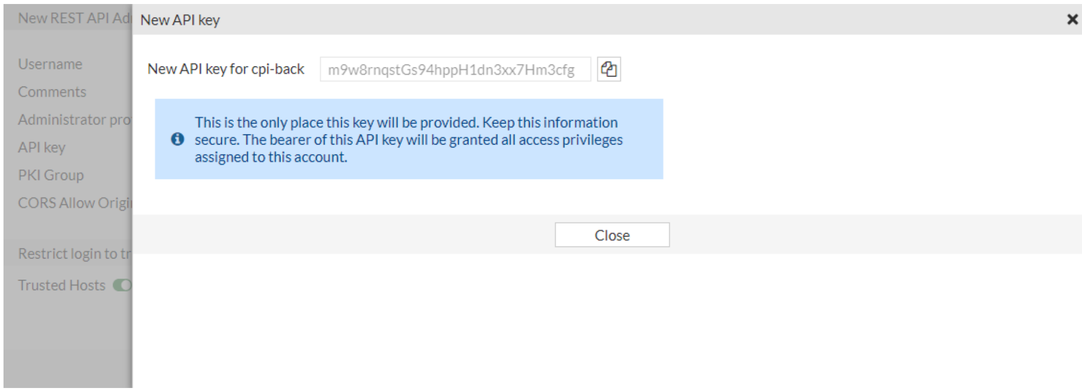
A REST API administrator is required to generate an authorization token for REST API messages, and to limit hosts that can send REST API messages to the FortiGate.

To create a REST API administrator in the GUI:

1. Go to *System > Administrators*.
2. Click *Create New > REST API Admin*.
3. Configure the *Username* and other information as needed.
4. Disable *PKI Group*.
5. In the *Trusted Hosts* field, enter *10.1.100.0/24*.



   For this example, an administrator profile called *clearpass* was created with full read/write access. See Administrator profiles on page 1949 for details.
6. Click *OK*.
   The *New API key* pane opens.



The API key is the REST API authorization token that is used in REST API messages sent by CPPM to the FortiGate.

7. Copy the API key to a secure location. A new key can be generated if this one is lost or compromised.
8. Click *Close*.

To create a REST API administrator in the CLI:

```
config system api-user
    edit "cpi-back"
        set accprofile "clearpass"
        config trusthost
            edit 1
                set ipv4-trusthost 10.1.100.0 255.255.255.0
            next
        end
    next
end
```

```
execute api-user generate-key cp-api
    New API key: 0f1HxGHh9r9p74k7qgfHNH40p51bjs
    NOTE: The bearer of this API key will be granted all access privileges assigned to the
api-user cp-api.
```

Create dynamic IP addresses with the clearpass subtype

Two dynamic IP addresses are required, one for the allow policy, and the other for the deny policy.

To create the dynamic IP addresses:

```
config firewall address
    edit "cppm"
        set type dynamic
        set sub-type clearpass-spt
        set clearpass-spt healthy
        set comment ''
        set visibility enable
        set associated-interface ''
        set color 0
    next
    edit "cppm-deny"
        set type dynamic
```

```
        set sub-type clearpass-spt
        set clearpass-spt infected
        set comment ''
        set visibility enable
        set associated-interface ''
        set color 0
    next
end
```

Create firewall policies

Two firewall policies are required, one to accept traffic (*cppm-allow*), and the other to deny traffic (*cppm-deny*).

To create the firewall policies in the GUI:

1. Go to *Policy & Objects > Firewall Policy*.
2. Configure the allow policy:
   a. Click *Create New*.
   b. Enter a name for the policy.
   c. Set *Source* set to *cppm*.
   d. Set *Action* to *ACCEPT*.
   e. Configure the remaining settings as needed.
   f. Click *OK*.
3. Configure the deny policy:
   a. Click *Create New*.
   b. Enter a name for the policy.
   c. Set *Source* set to *cppm-deny*.
   d. Set *Action* to *DENY*.
   e. Configure the remaining settings as needed.
   f. Click *OK*.

To create the firewall policies in the CLI:

```
config firewall address
    edit "cppm"
        set type dynamic
        set sub-type clearpass-spt
        set clearpass-spt healthy
        set comment ''
        set visibility enable
        set associated-interface ''
        set color 0
    next
    edit "cppm-deny"
        set type dynamic
        set sub-type clearpass-spt
        set clearpass-spt infected
        set comment ''
        set visibility enable
        set associated-interface ''
        set color 0
```

```
    next
end
```

Verification

Go to *Log & Report > Forward Traffic* to review traffic logs and ensure that traffic is allowed or denied as expected.

To verify that FortiGate addresses are assigned correctly, enter the following:

```
# diagnose firewall dynamic list
List all dynamic addresses:
cppm-deny: ID(141)
        ADDR(10.1.100.188)

cppm: ID(176)
        ADDR(10.1.100.185)
        ADDR(10.1.100.186)
```

Group address objects synchronized from FortiManager

Address objects from external connectors that are learned by FortiManager are synchronized to FortiGate. These objects can be grouped together with the FortiGate CLI to simplify selecting connector objects in the FortiGate GUI. Multiple groups can be created.

This option is only available for objects that are synchronized from FortiManager.

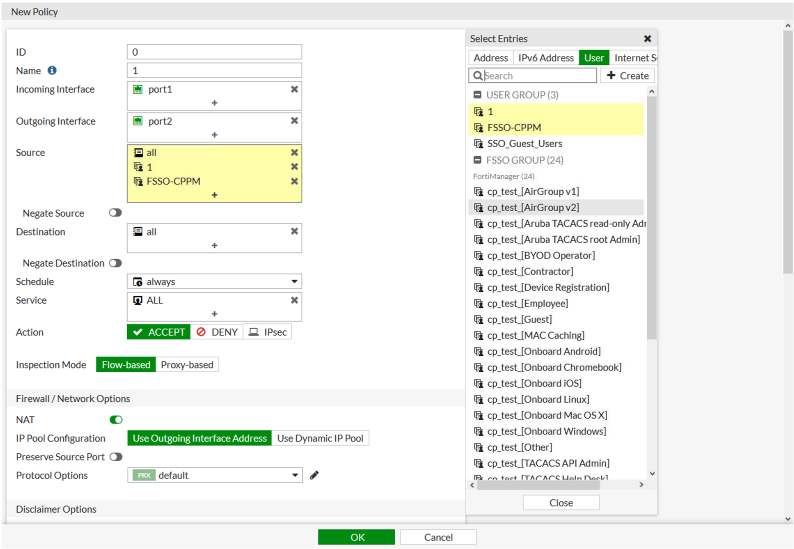
To add an object to a connector group:

```
config user adgrp
    edit <object_name>
        set server-name "FortiManager"
        set connector-source <group_name>
    next
end
```

Example

In this example, objects learned by the FortiManager from an Aruba ClearPass device are synchronized to the FortiGate. Some of the objects are then added to a group called *ClearPass* to make them easier to find in the object list when creating a firewall policy.

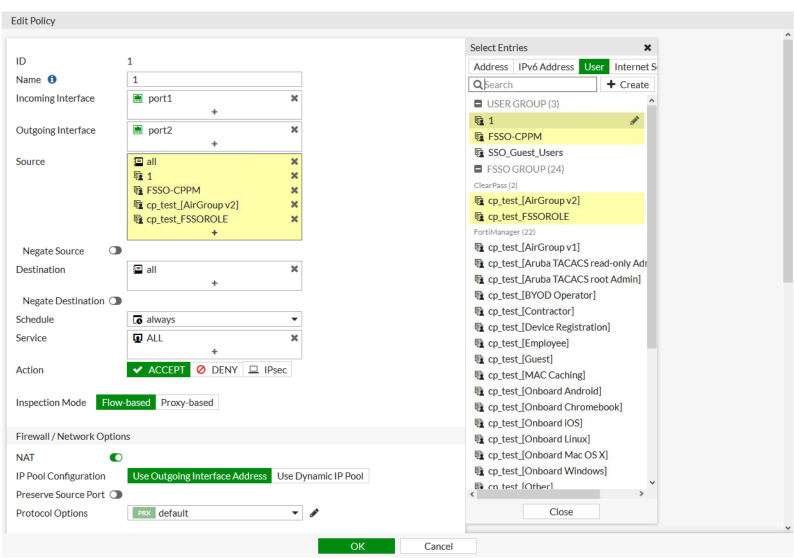
Prior to being grouped, the synchronized objects are listed under the FortiManager heading in the object lists.



To add some of the objects to a group:

```
config user adgrp
    edit "cp_test_FSSOROLE"
        set server-name "FortiManager"
        set connector-source "ClearPass"
    next
    edit "cp_test_[AirGroup v2]"
        set server-name "FortiManager"
        set connector-source "ClearPass"
    next
end
```

The objects are now listed under the *ClearPass* heading.



Using wildcard FQDN addresses in firewall policies

You can use wildcard FQDN addresses in firewall policies. IPv4, IPv6, ACL, local, shaping, NAT64, NAT46, and NGFW policy types support wildcard FQDN addresses.

For wildcard FQDN addresses to work, the FortiGate should allow DNS traffic to pass through.

Initially, the wildcard FQDN object is empty and contains no addresses. When the client tries to resolve a FQDN address, the FortiGate will analyze the DNS response. The IP address(es) contained in the answer section of the DNS response will be added to the corresponding wildcard FQDN object. It is therefore necessary to have the DNS session-helpers defined in the `config system session-helper` setting.

Since FortiGate must analyze the DNS response, it does not work with DNS over HTTPS.

In FortiOS 7.0 and later, FortiGate supports DNS over TLS. It is possible to analyze DNS responses sent over DoT, as long as there is a firewall policy that allows the DNS traffic from the client and is configured with a DNS filter that supports DoT. For information on configuring this, see DNS inspection with DoT and DoH on page 1184.

When the wildcard FQDN gets the resolved IP addresses, FortiOS loads the addresses into the firewall policy for traffic matching.

The FortiGate will keep the IP addresses in the FQDN object table as long as the DNS entry itself has not expired. Once it expires, the IP address is removed from the wildcard FQDN object until another query is made. At any given time, a single wildcard FQDN object may have up to 1000 IP addresses.

The DNS expiry TTL value is set by the authoritative name server for that DNS record. If the TTL for a specific DNS record is very short and you would like to cache the IP address longer, then you can extend it with the CLI. See To extend the TTL for a DNS record in the CLI: on page 885

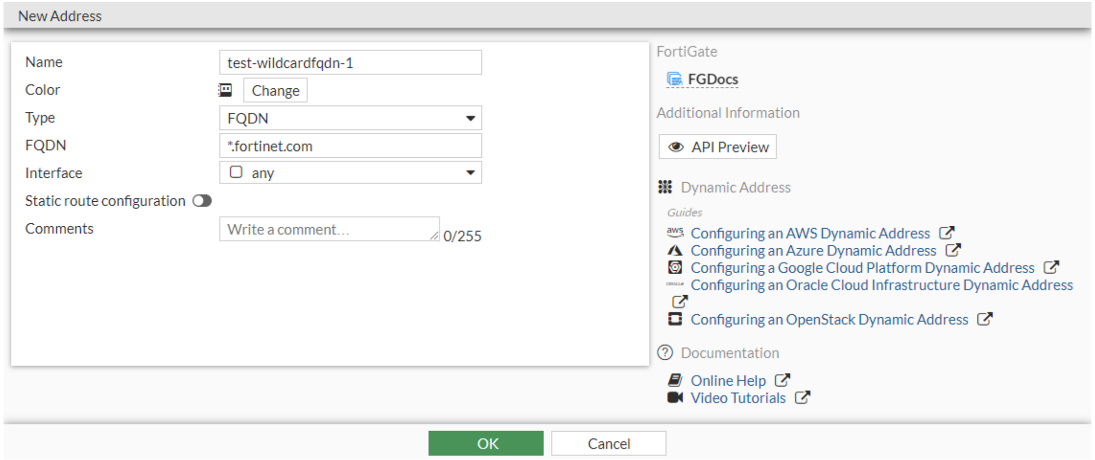
For more information, see FQDN address firewall object type.

Wildcard FQDN IPs are synchronized to other autoscale members whenever a peer learns of a wildcard FQDN address.

To create a wildcard FQDN using the GUI:

1. Go to *Policy & Objects* > *Addresses* and click *Create New* > *Address*.
2. Specify a *Name*.
3. For *Type*, select *FQDN*.

4. For *FQDN*, enter a wildcard FQDN address, for example, `*.fortinet.com`.



5. Click *OK*.

To use a wildcard FQDN in a firewall policy using the GUI:

1. Go to *Policy & Objects* > *Firewall Policy* and click *Create New*.
2. For *Destination*, select the wildcard FQDN.
3. Configure the rest of the policy as needed.
4. Click *OK*.

To create a wildcard FQDN using the CLI:

```
config firewall address
    edit "test-wildcardfqdn-1"
        set type fqdn
        set fqdn "*.fortinet.com"
    next
end
```

To use wildcard FQDN in a firewall policy using the CLI:

```
config firewall policy
    edit 2
        set srcintf "port3"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "test-wildcardfqdn-1"
        set action accept
        set schedule "always"
        set service "ALL"
        set auto-asic-offload disable
        set nat enable
    next
end
```

To use the diagnose command to list resolved IP addresses of wildcard FQDN objects:

```
# diagnose firewall fqdn list
```

```
List all FQDN:
```

```
*.fortinet.com: ID(48) ADDR(96.45.36.159) ADDR(192.168.100.161) ADDR(65.39.139.161)
```

Alternatively:

```
# diagnose test application dnsproxy 6

worker idx: 0

vfid=0 name=*.fortinet.com ver=IPv4  min_ttl=3266:0, cache_ttl=0 , slot=-1, num=3,
wildcard=1

       96.45.36.159 (ttl=68862:68311:68311) 192.168.100.161 (ttl=3600:3146:3146)
65.39.139.161

(ttl=3600:3481:3481)
```

To use the diagnose command for firewall policies which use wildcard FQDN:

```
# diagnose firewall iprope list 100004
...
destination fqdn or dynamic address (1):*.fortinet.com ID(48) uuid_idx=57 ADDR
(208.91.114.104) ADDR(208.91.114.142) ADDR(173.243.137.143) ADDR(65.104.9.196) ADDR
(96.45.36.210)
...
```

To extend the TTL for a DNS record in the CLI:

In this the example the `set cache-ttl` value has been extended to 3600 seconds.

```
config firewall address
   edit "fortinet.com"
      set type fqdn
      set fqdn "www.fortinet.com”
      set cache-ttl 3600
   next
end
```

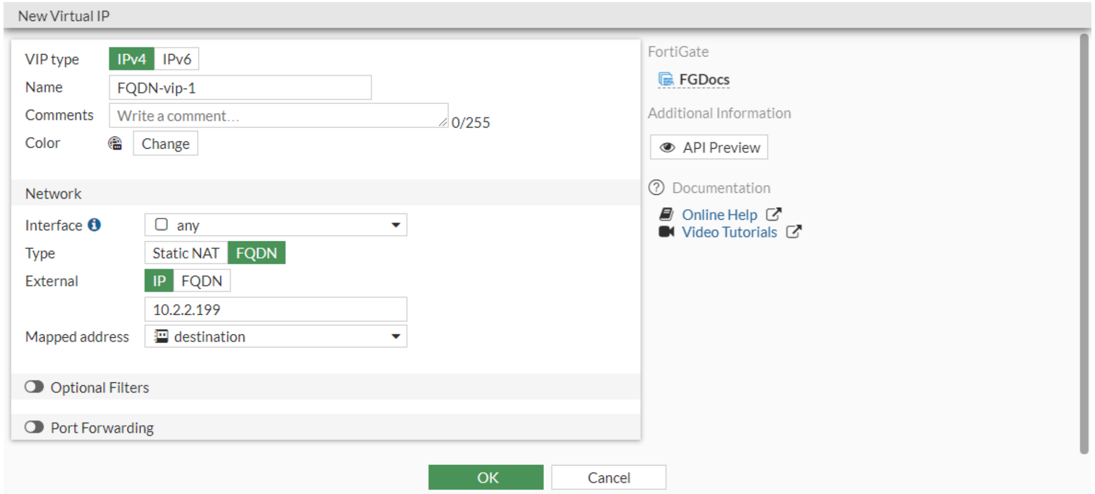
Configure FQDN-based VIPs

In public cloud environments, sometimes it is necessary to map a VIP to an FQDN address.

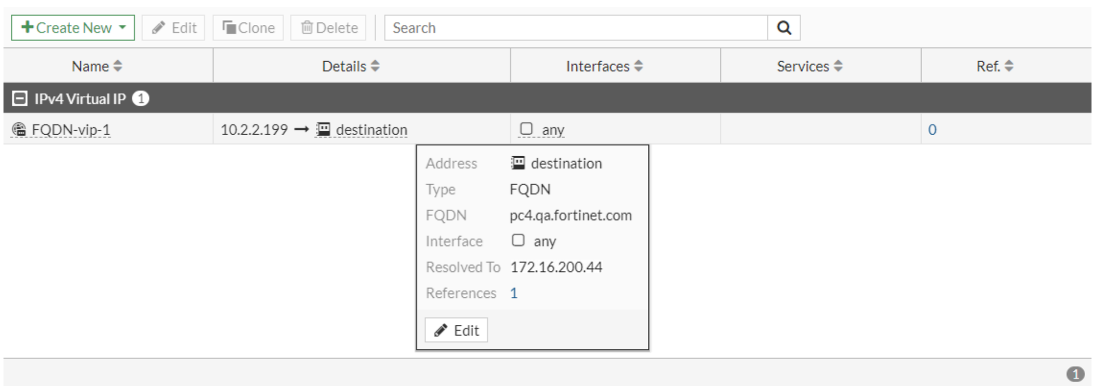
To configure an FQDN-based VIP in the GUI:

1. Go to *Policy & Objects* > *Virtual IPs* and click *Create New* > *Virtual IP*.
2. Enter a name for the VIP.
3. Select an interface.
4. For *Type*, select *FQDN*.
5. For *External*, select *IP* and enter the external IP address.

6. For *Mapped address*, select an FQDN address.



7. Click *OK*.
   In the virtual IP list, hover over the address to view more information.



To configure an FQDN-based VIP in the CLI:

```
config firewall vip
    edit "FQDN-vip-1"
        set type fqdn
        set extip 10.2.2.199
        set extintf "any"
        set mapped-addr "destination"
    next
end
```

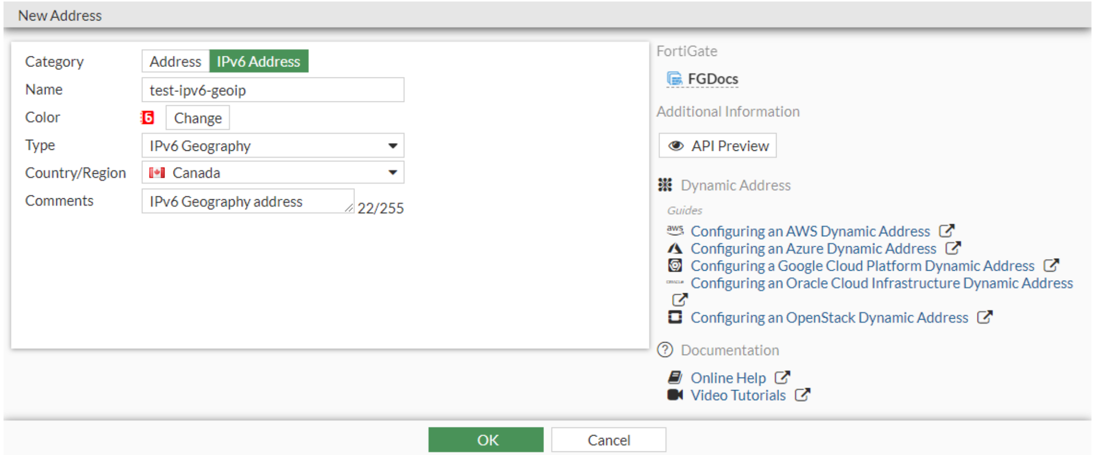
IPv6 geography-based addresses

Geography-based IPv6 addresses can be created and applied to IPv6 firewall policies.

IPv6 geography-based addresses do not support `geoip-override` or `geoip-anycast`.

To create an IPv6 geography-based address in the GUI:

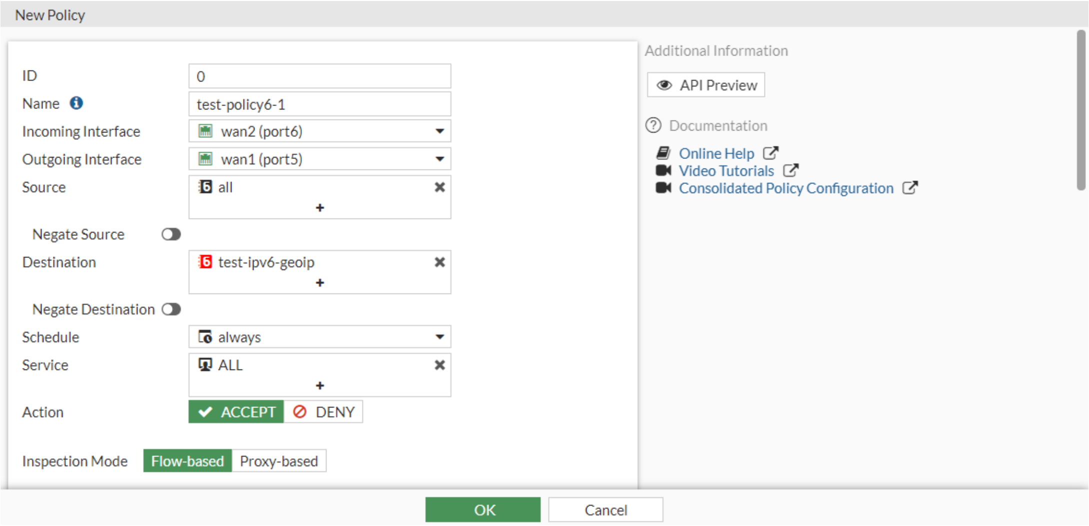
1. Go to *Policy and Objects > Addresses*.
2. Click *Create New > Address*.
3. Set *Category* to *IPv6 Address*.
4. Enter a name for the address.
5. Set *Type* to *IPv6 Geography*.
6. Select the *Country/Region* from the list.
7. Optionally, enter comments.



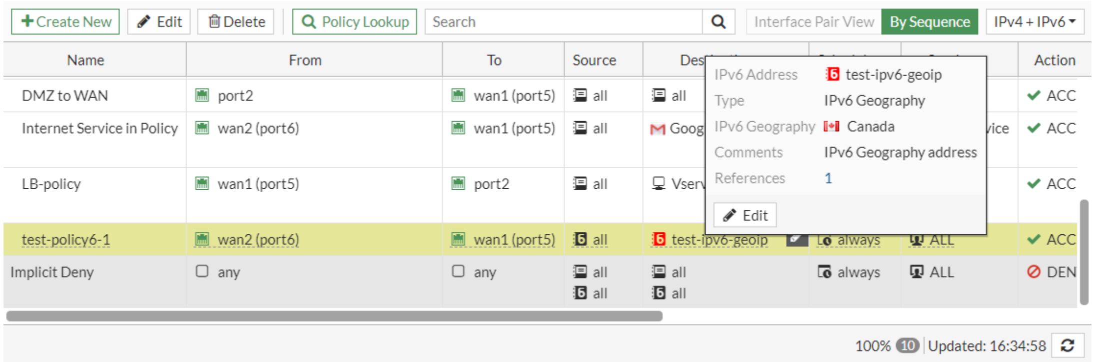
8. Click *OK*.

To use the IPv6 geography address in a policy:

1. Go to *Policy & Objects > Firewall Policy*.
2. Edit an existing policy, or create a new one, using the IPv6 geography address as the *Source* or *Destination Address*.



3. In the policy list, hover over the address to view details.



To configure an IPv6 geography-based address in the CLI:

1. Create an IPv6 geography-based address:

```
config firewall address6
    edit "test-ipv6-geoip"
        set type geography
        set color 6
        set comment "IPv6 Geography address"
        set country "CA"
    next
end
```

2. Use the IPv6 geography-based address in a policy:

```
config firewall policy
    edit 1
        set name "test-policy6-1"
        set srcintf "port6"
        set dstintf "port5"
        set srcaddr6 "all"
        set dstaddr6 "test-ipv6-geoip"
        set action accept
        set schedule "always"
        set service "ALL"
        set nat enable
    next
end
```

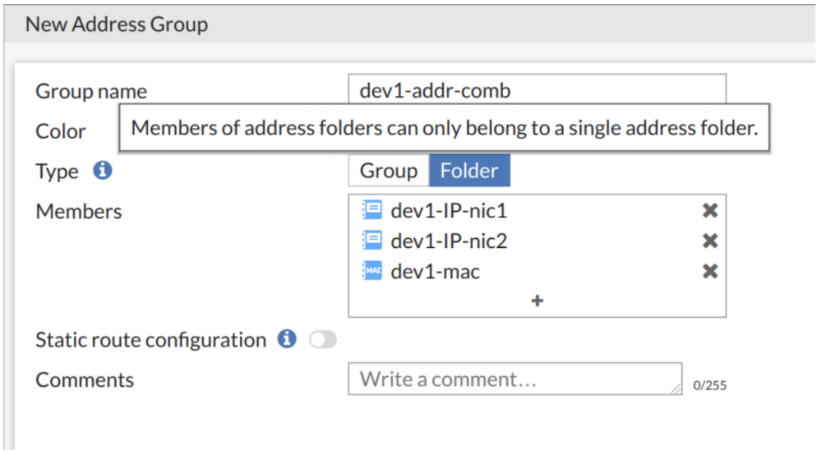
Array structure for address objects

Some address objects logically belong to the same device, such as two IPs from the same computer. These address objects can be grouped into an address folder, which is an exclusive list of address objects that do not appear in other address groups or folders.

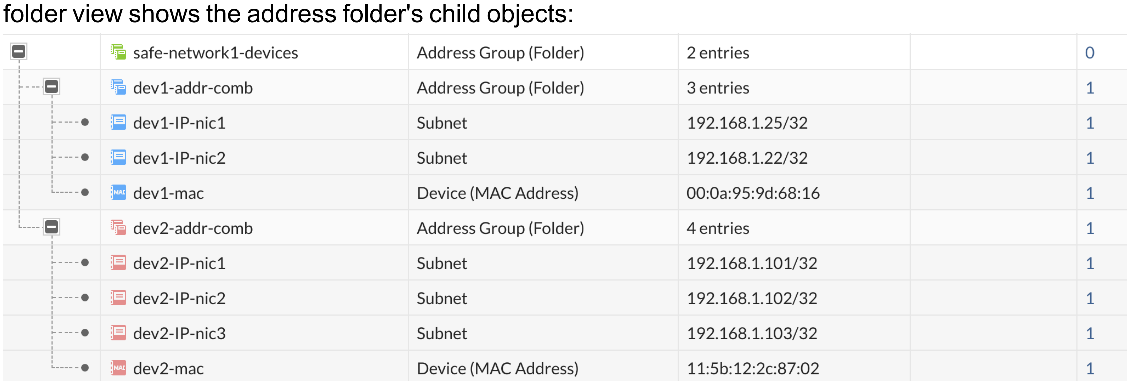
In the CLI, the folder type can be set after the member list is already populated. If the member list contains an incompatible entry, then the setting will be discarded when the `next`/`end` command is issued. If the folder type is set before the member list is populated, then the possible member entry list will be filtered according to the selected type.

To create an address folder in the GUI:

1. Go to *Policy & Objects > Addresses*.
2. Click *Create New > Address Group* and enter a name.
3. For *Type*, select *Folder*.
4. For *Members*, click the + to add the addresses. Address folders and groups are exclusive, so the *Select Entries* window filters out address objects that are a member of an existing group or folder.



5. Click *OK*.
6. In the address table, expand the *Address Group* section to view the folder (*dev1-addr-comb*). The expandable folder view shows the address folder's child objects:



safe-network1-devices	Address Group (Folder)	2 entries		0
dev1-addr-comb	Address Group (Folder)	3 entries		1
dev1-IP-nic1	Subnet	192.168.1.25/32		1
dev1-IP-nic2	Subnet	192.168.1.22/32		1
dev1-mac	Device (MAC Address)	00:0a:95:9d:68:16		1
dev2-addr-comb	Address Group (Folder)	4 entries		1
dev2-IP-nic1	Subnet	192.168.1.101/32		1
dev2-IP-nic2	Subnet	192.168.1.102/32		1
dev2-IP-nic3	Subnet	192.168.1.103/32		1
dev2-mac	Device (MAC Address)	11:5b:12:2c:87:02		1

To configure an address folder in the CLI:

```
config firewall addrgrp
    edit "safe-network1-devices"
        set type folder
        set member "dev1-addr-comb" "dev2-addr-comb"
        set comment ''
        set exclude disable
        set color 13
    next
end

config firewall addrgrp
    edit "dev1-addr-comb"
        set type folder
        set member "dev1-IP-nic1" "dev1-IP-nic2" "dev1-mac"
```

```
        set comment ''
        set exclude disable
        set color 18
    next
end
```

```
config firewall addrgrp
    edit "dev2-addr-comb"
        set type folder
        set member "dev2-IP-nic1" "dev2-IP-nic2" "dev2-IP-nic3" "dev2-mac"
        set comment ''
        set exclude disable
        set color 5
    next
end
```

IPv6 MAC addresses and usage in firewall policies

Users can define IPv6 MAC addresses that can be applied to the following policies:

- Firewall
- Virtual wire pair
- ACL/DoS
- Central NAT
- NAT64
- Local-in

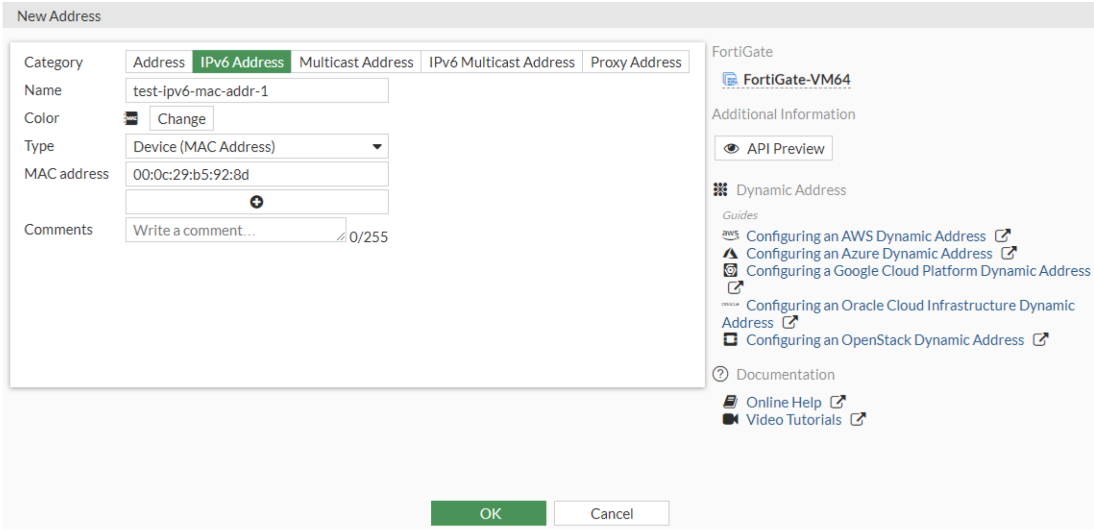
In FortiOS, you can configure a firewall address object with a singular MAC, wildcard MAC, multiple MACs, or a MAC range. In this example, a firewall policy is configured in a NAT mode VDOM with the IPv6 MAC address as a source address.

IPv6 MAC addresses cannot be used as destination addresses in VDOMs when in NAT operation mode.

To configure IPv6 MAC addresses in a policy in the GUI:

1. Create the MAC address:
   a. Go to *Policy & Objects* > *Addresses* and click *Create New* > *Address*.
   b. For *Category*, select *IPv6 Address*.
   c. Enter an address name.
   d. For *Type*, select *Device (MAC Address)*.

e. Enter the the MAC address.



f. Click *OK*.

2. Configure the policy:
   a. Go to *Policy & Objects* > *Firewall Policy* and click *Create New*.
   b. For *Source*, select the IPv6 MAC address object.
   c. Configure the other settings as needed.
   d. Click *OK*.

To configure IPv6 MAC addresses in a policy in the CLI:

1. Create the MAC address:

```
config firewall address6
    edit "test-ipv6-mac-addr-1"
        set type mac
        set macaddr 00:0c:29:b5:92:8d
    next
end
```

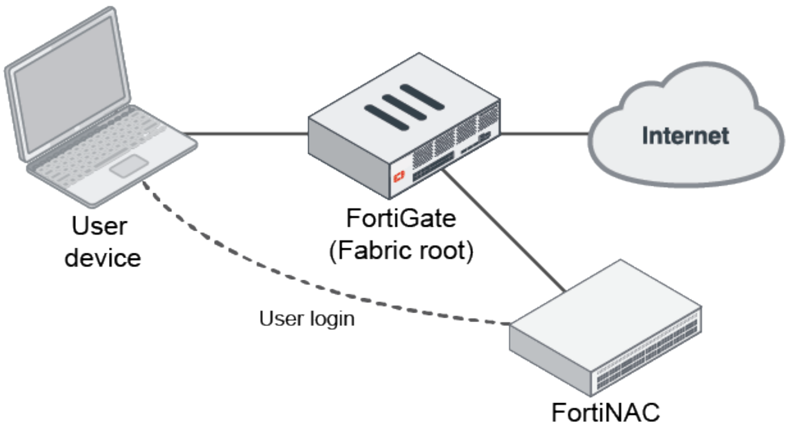
2. Configure the policy:

```
config firewall policy
    edit 2
        set srcintf "wan2"
        set dstintf "wan1"
        set srcaddr "all"
        set dstaddr "all"
        set srcaddr6 "test-ipv6-mac-addr-1" "2000-10-1-100-0"
        set dstaddr6 "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set logtraffic all
        set auto-asic-offload disable
        set nat enable
    next
end
```

FortiNAC tag dynamic address

The FortiNAC tag dynamic firewall address type is used to store the device IP, FortiNAC firewall tags, and FortiNAC group information sent from FortiNAC by the REST API when user logon and logoff events are registered.

In the following example, the user connecting to the network will be required to first log on to the FortiNAC. When the login succeeds, the logon information is synchronized to the FortiGate using the REST API. The FortiGate updates the dynamic firewall address object with the user and IP information of the user device. This firewall address is used in firewall policies to dynamically allow network access for authenticated users, thereby allowing SSO for the end user.

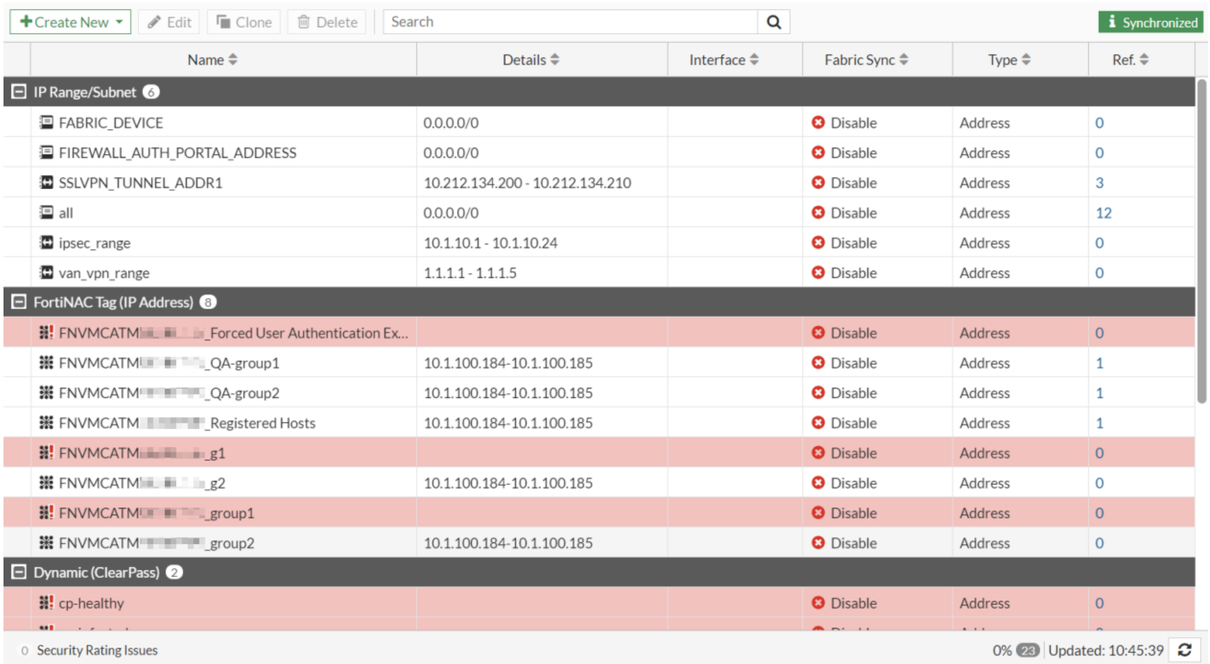


This example assumes the following:

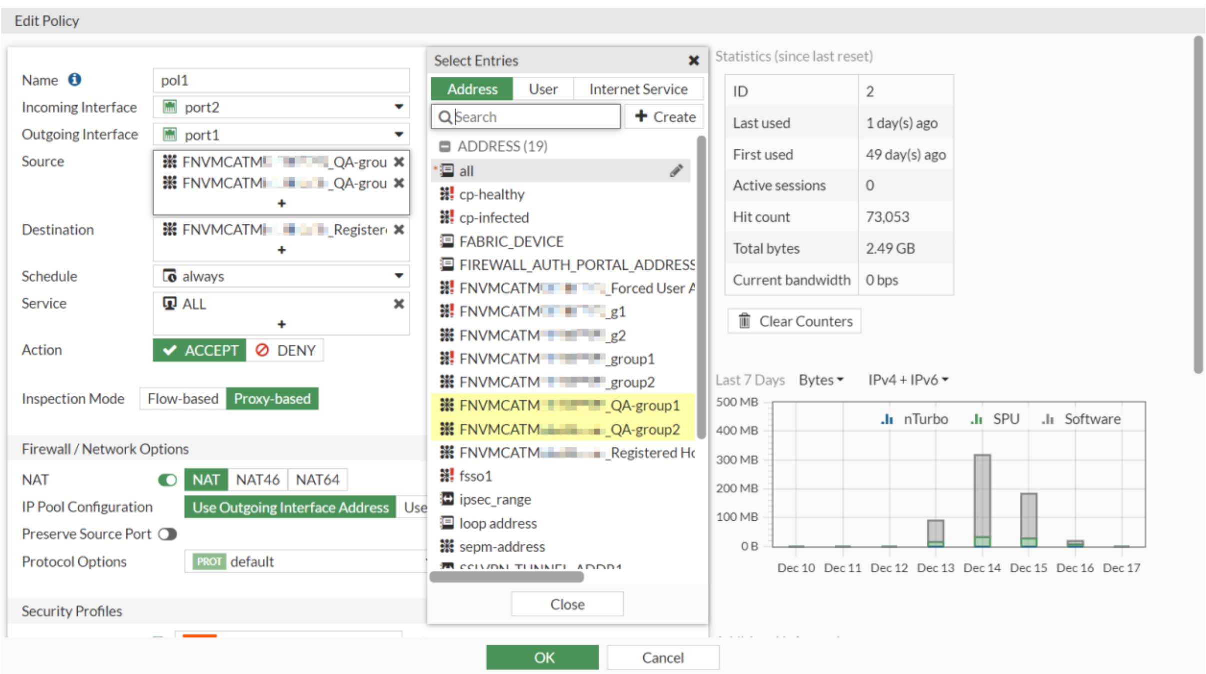
- The FortiGate is the Security Fabric root device (refer to Configuring the root FortiGate and downstream FortiGates on page 2208 for more information).
- The FortiNAC is running version 9.2.2 (or later), and it is connected to the Security Fabric (refer to Configuring FortiNAC on page 2250 for more information).
- Firewall tags and groups have been assigned in FortiNAC to the registered FortiGate (refer to Virtualized Devices for more information). Unlike firewall tags, which are simple labels that can be configured on FortiNAC, firewall groups can be local, built-in, user-defined, or remote user groups imported from a remote server used for user authentication. Only groups that the user of the current logon event belongs to are sent to the FortiGate. Firewall tags are sent for all user authentication.

To use a FortiNAC tag dynamic firewall address in a policy:

1. Trigger two user logon events on the FortiNAC.
2. In FortiOS, go to *Policy & Objects > Addresses*, and expand the *FortiNAC Tag (IP Address)* section to view the newly created dynamic firewall address objects. The dynamic firewall addresses matching the current user logon status on FortiNAC have the current IP address of user devices. The addresses without matching user logons are marked with a red exclamation mark (!).

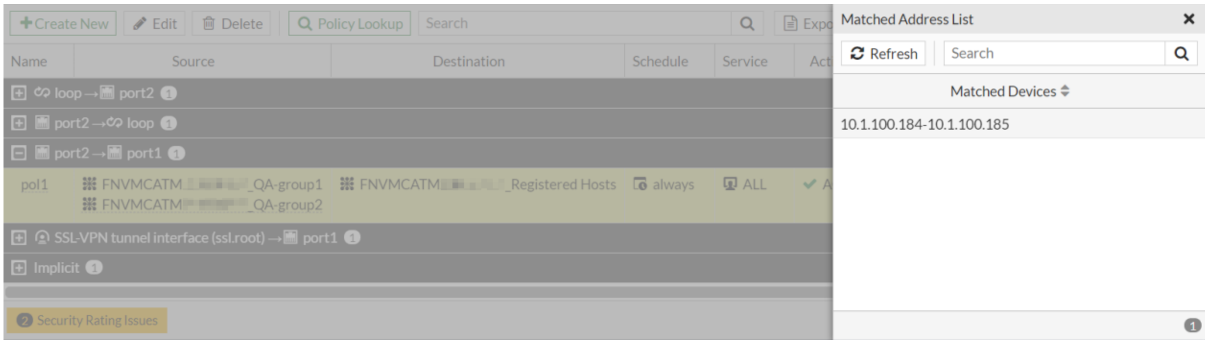


3. Go to *Policy & Objects > Firewall Policy* and click *Create New* or edit an existing policy. FortiNAC tag dynamic firewall address an be used as source or destination addresses.

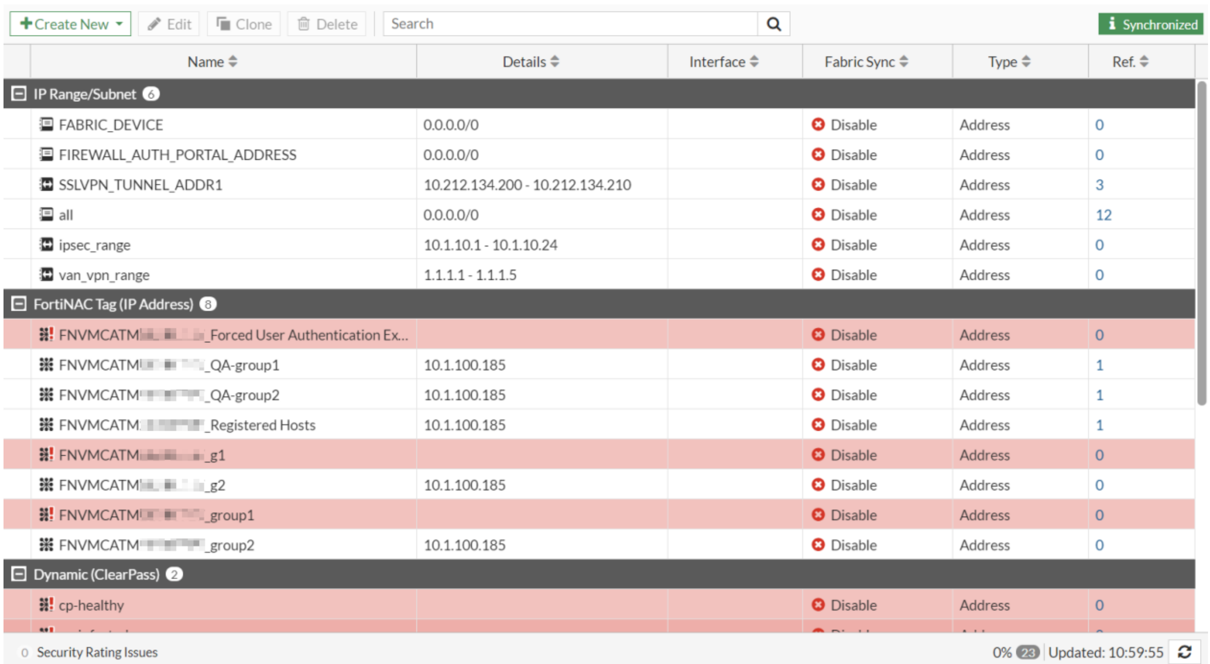


4. Configure the settings as needed, then click *OK*. In this policy, traffic can only pass if it originates from any of the mapped IP addresses (10.1.100.184 and 10.1.100.185); other traffic cannot pass.

5. Hover over the address in the policy, then in the tooltip, click *View Matched Addresses*.

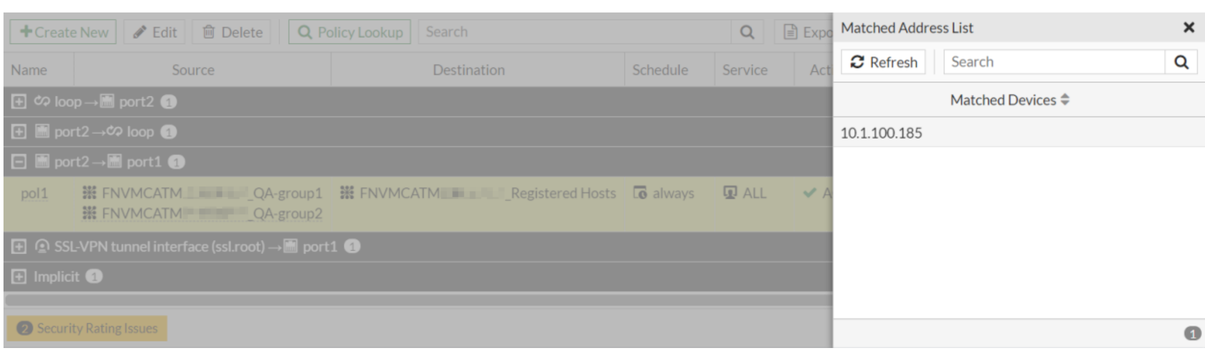


6. Have one of the users log off from the FortiNAC.
7. In FortiOS, go to *Policy & Objects > Addresses* and verify the *FortiNAC Tag* addresses. A user logged off from 10.1.100.184, so now only 10.1.100.185 is mapped to the dynamic firewall objects.



All firewall policies using those objects are automatically updated.

8. Go to *Policy & Objects > Firewall Policy* . Hover over the address in the policy, then in the tooltip, click *View Matched Addresses*.



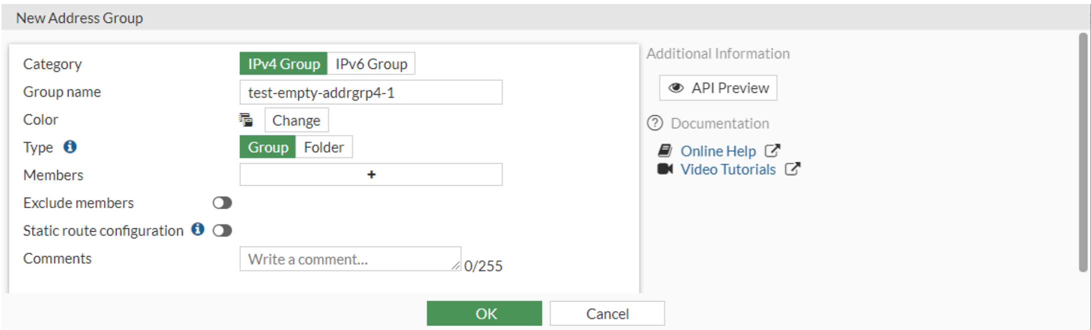
The firewall policy was automatically updated so that traffic from 10.1.100.184 can no longer pass, and only traffic from 10.1.100.185 can pass.

Allow empty address groups

Address groups with no members can be configured in the GUI, CLI, and through the API. In previous versions of FortiOS, error messages appear for empty address groups and they cannot be configured.

To create an empty address group in the GUI:

1. Go to *Policy & Objects > Addresses* and click *Create New > Address Group*.
2. Enter a name.



3. Click *OK*. The *This field is required.* error is not displayed under the *Members* field.

To create an empty address group in the CLI:

```
config firewall addrgrp
    edit "test-empty-addrgrp4-1"
    next
end
```

No error message is returned in the console.

Remove overlap check for VIPs

There is no overlap check for VIPs, so there are no constraints when configuring multiple VIPs with the same external interface and IP. A new security rating report alerts users of any VIP overlaps.

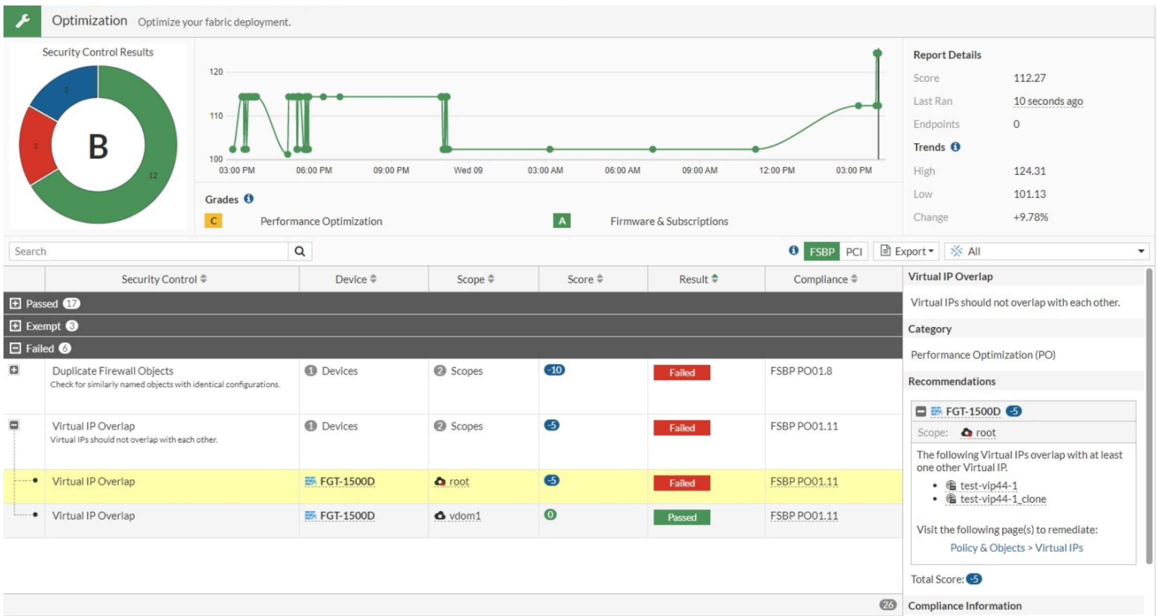
To configure two VIPs with the same external interface and IP:

```
config firewall vip
    edit "test-vip44-1"
        set extip 10.1.100.154
        set mappedip "172.16.200.156"
        set extintf "port24"
    next
    edit "test-vip44-1_clone"
        set extip 10.1.100.154
        set mappedip "172.16.200.156"
        set extintf "port24"
        set src-filter 10.1.100.11
    next
end
```

No error message appears regarding the overlapping VIPs.

To view the security rating report:

1. Go to *Security Fabric > Security Rating* and click the *Optimization* scorecard.
2. Expand the *Failed* section. The *Virtual IP Overlap* results show an overlap (*test-vip44-1* and *test-vip44-1_clone*) on the root FortiGate.



VIP groups

Virtual IP addresses (VIPs) can be organized into groups. This is useful in scenarios where there are multiple VIPs that are used together in firewall policies. If the VIP group members change, or a group member's settings change (such as the IP address, port, or port mapping type), then those changes are automatically updated in the corresponding firewall policies.

The following table summarizes which VIP types are allowed and not allowed to be members of a VIP group:

Group type	VIP types allowed as members	VIP types not allowed as members
IPv4	• Static NAT • Load balance • DNS translation • FQDN	• Access proxy • Server load balance
IPv6	• Static NAT	• Access proxy • Server load balance

Different VIP types can be added to the same group.

To configure a VIP group in the GUI:

1. Go to *Policy & Objects* > *Virtual IPs* and click *Create New* > *Virtual IP Group*.
2. Set the *Type* to *IPv4* or *IPv6*.
3. Enter a name.
4. Optionally, enter additional information in the *Comments* field.
5. For IPv4 groups, select the *Interface*. Select a specific interface if all of the VIPs are on the same interface; otherwise, select *any*.
6. Click the + in the *Members* field and select the members to add to the group.
7. Click *OK*.

To configure an IPv4 VIP group in the CLI:

```
config firewall vipgrp
    edit <name>
        set interface <name>
        set member <vip1> <vip2> ...
    next
end
```

To configure an IPv6 VIP group in the CLI:

```
config firewall vipgrp6
    edit <name>
        set member <vip1> <vip2> ...
    next
end
```

Internet Services

The following topics provide instructions on configuring policies with Internet Service:

Using Internet Service in policy

This topic shows how to apply a predefined Internet Service entry into a policy.

The Internet Service Database is a comprehensive public IP address database that combines IP address range, IP owner, service port number, and IP security credibility. The data comes from the FortiGuard service system. Information is regularly added to this database, for example, geographic location, IP reputation, popularity & DNS, and so on. All this information helps users define Internet security more effectively. You can use the contents of the database as criteria for inclusion or exclusion in a policy.

From FortiOS version 5.6, Internet Service is included in the firewall policy. It can be applied to a policy only as a destination object. From version 6.0, Internet Service can be applied both as source and destination objects in a policy. You can also apply Internet Services to shaping policy.

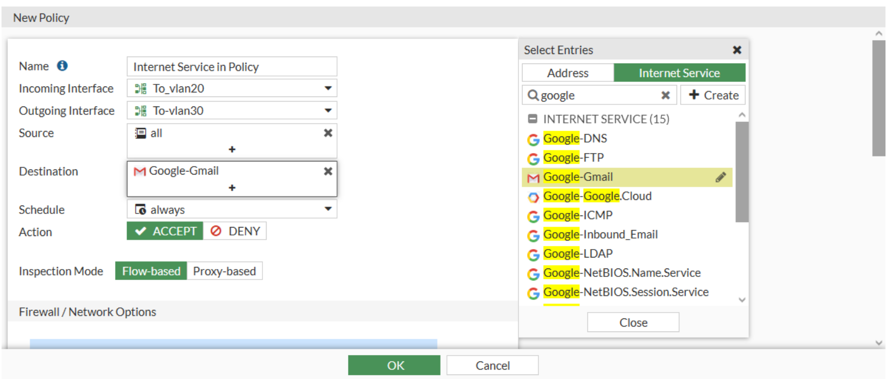
There are three types of Internet Services you can apply to a firewall policy:

- Predefined Internet Services
- Custom Internet Services
- Extension Internet Services

Sample configuration

To apply a predefined Internet Service entry to a policy using the GUI:

1. Go to *Policy & Objects* > *Firewall Policy* and click *Create New*.
2. Click in the *Destination* field.
3. In the *Select Entries* pane, click *Internet Service* and select *Google-Gmail*.



4. Configure the remaining fields as needed.
5. Click *OK*.

To apply a predefined Internet Service entry to a policy in the CLI:

In the CLI, enable the `internet-service` first and then use its ID to apply the policy.

This example uses Google Gmail and its ID is 65646. Each Internet Service has a unique ID.

```
config firewall policy
    edit 9
        set name "Internet Service in Policy"
        set srcintf "wan2"
        set dstintf "wan1"
        set srcaddr "all"
        set internet-service enable
        set internet-service-id 65646
        set action accept
        set schedule "always"
        set utm-status enable
        set av-profile "g-default"
```

```
        set ssl-ssh-profile "certificate-inspection"
        set nat enable
    next
end
```

To diagnose an Internet Service entry in the CLI:

```
# diagnose internet-service id-summary 65646
Version: 0000600096
Timestamp: 201902111802
Total number of IP ranges: 444727
Number of Groups: 7
Group(0), Singularity(20), Number of IP ranges(142740)
Group(1), Singularity(19), Number of IP ranges(1210)
Group(2), Singularity(16), Number of IP ranges(241)
Group(3), Singularity(15), Number of IP ranges(38723)
Group(4), Singularity(10), Number of IP ranges(142586)
Group(5), Singularity(8), Number of IP ranges(5336)
Group(6), Singularity(6), Number of IP ranges(113891)
Internet Service: 65646(Google.Gmail)
Number of IP range: 60
Number of IP numbers: 322845
Singularity: 15
Reputation: 5(Known and verified safe sites such as Gmail, Amazon, eBay, etc.)
Icon Id: 510
Second Level Domain: 53(gmail.com)
Direction: dst
Data source: isdb
```

Result

Because the IP and services related to Google Gmail on the Internet are included in this Internet Service (65646), all traffic to Google Gmail is forwarded by this policy.

Using custom Internet Service in policy

Custom Internet Services can be created and used in firewall policies.

When creating a custom Internet Service, you must set following elements:

- IP or IP ranges
- Protocol number
- Port or port ranges
- Reputation

You must use CLI to create a custom Internet Service, except for geographic based services (see Allow creation of ISDB objects with regional information on page 911).

CLI syntax

```
config firewall internet-service-custom
    edit <name>
       set comment <comment>
       set reputation {1 | 2 | 3 | 4 | 5}
```

```
        config entry
            edit <ID>
                set protocol <protocol #>
                set dst <object_name>
                config port-range
                    edit <ID>
                        set start-port <port #>
                        set end-port <port #>
                    next
                end
            next
        end
    end
end
```

Sample configuration

To configure a custom Internet Service:

```
config firewall internet-service-custom
   edit "test-isdb-1"
       set comment "Test Custom Internet Service"
       set reputation 4
       config entry
          edit 1
              set protocol 6
              config port-range
                  edit 1
                      set start-port 80
                      set end-port 443
                  next
              end
              set dst "10-1-100-0"
          next
          edit 2
              set protocol 6
              config port-range
                  edit 1
                      set start-port 80
                      set end-port 80
                  next
              end
              set dst "172-16-200-0"
          next
       end
   next
end
```

To apply a custom Internet Service into a policy:

```
config firewall policy
    edit 1
        set name "Internet Service in Policy"
        set srcintf "wan2"
        set dstintf "wan1"
```

```
        set srcaddr "all"
        set internet-service enable
        set internet-service-id 65646
        set internet-service-custom "test-isdb-1"
        set action accept
        set schedule "always"
        set utm-status enable
        set av-profile "g-default"
        set ssl-ssh-profile "certificate-inspection"
        set nat enable
    next
end
```

Result

In addition to the IP address, IP address ranges, and services allowed by Google.Gmail, this policy also allows the traffic which access to 10.1.100.0/24 and TCP/80-443 and 172.16.200.0/24 and TCP/80.

Using extension Internet Service in policy

Extension Internet Service lets you add custom or remove existing IP address and port ranges to an existing predefined Internet Service entries. Using an extension type Internet Service is actually editing a predefined type Internet Service entry and adding IP address and port ranges to it.

When creating an extension Internet Service and adding custom ranges, you must set following elements:

- IP or IP ranges
- Protocol number
- Port or port ranges

You must use CLI to add custom IP address and port entries into a predefined Internet Service.

You must use GUI to remove entries from a predefined Internet Service.

Custom extension Internet Service CLI syntax

```
config firewall internet-service-extension
    edit <ID #>
        set comment <comment>
        config entry
            edit <ID #>
                set protocol <number #>
                set dst <object_name>
                config port-range
                    edit <ID #>
                         set start-port <number #>
                         set end-port <number #>
                    next
                end
            next
        end
    end
end
```

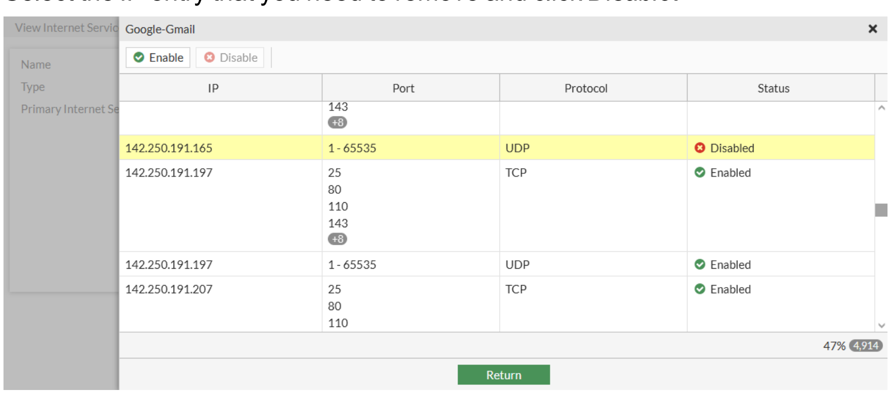
Sample configuration

To configure an extension Internet Service in the CLI:

```
config firewall internet-service-extension
    edit 65646
        set comment "Test Extension Internet Service 65646"
        config entry
            edit 1
                set protocol 6
                config port-range
                    edit 1
                        set start-port 80
                        set end-port 443
                    next
                end
                set dst "172-16-200-0"
            next
            edit 2
                set protocol 17
                config port-range
                    edit 1
                        set start-port 53
                        set end-port 53
                    next
                end
                set dst "10-1-100-0"
            next
        end
    next
end
```

To remove IP address and port entries from an existing Internet Service in the GUI:

1. Go to *Policy & Objects > Internet Service Database*.
2. Search for *Google-Gmail*.
3. Select *Google-Gmail* and click *Edit*.
4. In the gutter, click *View/Edit Entries*.
5. Select the *IP* entry that you need to remove and click *Disable*.



6. Click *Return* twice.

To remove IP address and port entries from an existing Internet Service in the CLI:

```
config firewall internet-service-extension
    edit 65646
        config disable-entry
            edit 1
                set protocol 17
                config port-range
                    edit 1
                    next
                end
                config ip-range
                    edit 1
                        set start-ip 142.250.191.165
                        set end-ip 142.250.191.165
                    next
                end
            next
        end
    next
end
```

To apply an extension Internet Service into policy in the CLI:

```
config firewall policy
    edit 9
        set name "Internet Service in Policy"
        set srcintf "wan2"
        set dstintf "wan1"
        set srcaddr "all"
        set internet-service enable
        set internet-service-id 65646
        set action accept
        set schedule "always"
        set utm-status enable
        set av-profile "g-default"
        set ssl-ssh-profile "certificate-inspection"
        set nat enable
    next
end
```

Result

In addition to the IP addresses, IP address ranges, and services allowed by Google.Gmail, this policy also allows the traffic which accesses 10.1.100.0/24 and UDP/53 and 172.16.200.0/24 and TCP/80-443. At the same time, the traffic that accesses 2.20.183.160 is dropped because this IP address and port is disabled from Google.Gmail.

Global IP address information database

The Internet Service and IP Reputation databases download details about public IP address, including: ownership, known services, geographic location, blocklisting information, and more. The details are available in drilldown information, tooltips, and other mechanisms in the FortiView and other pages.

The global IP address database is an integrated database containing all public IP addresses, and is implemented in the Internet Service Database.

To view the owner of the IP address:

```
(global) # get firewall internet-service-owner ?
    id    Internet Service owner ID.
    1  Google
    2  Facebook
    3  Apple
    4  Yahoo
    5  Microsoft
    ......
    115  Cybozu
    116  VNC
```

To check for any known service running on an IP address:

```
(global) # diagnose internet-service info FG-traffic 6 80 8.8.8.8
    Internet Service: 65537(Google.Web)
```

To check GeoIP location and blocklist information:

```
(global) # diagnose internet-service id 65537 | grep 8.8.8.8
    8.8.8.8-8.8.8.8 geo_id(11337) block list(0x0) proto(6) port(80 443)
    8.8.8.8-8.8.8.8 geo_id(11337) block list(0x0) proto(17) port(443)
```

To check a known malicious server:

```
(global) # diagnose internet-service id-summary 3080383
    Version: 0000600096
    Timestamp: 201902111802
    Total number of IP ranges: 444727
    Number of Groups: 7
    Group(0), Singularity(20), Number of IP ranges(142740)
    Group(1), Singularity(19), Number of IP ranges(1210)
    Group(2), Singularity(16), Number of IP ranges(241)
    Group(3), Singularity(15), Number of IP ranges(38723)
    Group(4), Singularity(10), Number of IP ranges(142586)
    Group(5), Singularity(8), Number of IP ranges(5336)
    Group(6), Singularity(6), Number of IP ranges(113891)
    Internet Service: 3080383(Botnet.C&C.Server)
    Number of IP range: 111486
    Number of IP numbers: 111486
    Singularity: 20
    Reputation: 1(Known malicious sites related to botnet servers, phishing sites, etc.)
    Icon Id: 591
    Second Level Domain: 1(other)
    Direction: dst
    Data source: irdb
```

To check questionable usage:

```
(global) # diagnose internet-service id-summary 2818238
    Version: 0000600096
```

```
    Timestamp: 201902111802
    Total number of IP ranges: 444727
    Number of Groups: 7
    Group(0), Singularity(20), Number of IP ranges(142740)
    Group(1), Singularity(19), Number of IP ranges(1210)
    Group(2), Singularity(16), Number of IP ranges(241)
    Group(3), Singularity(15), Number of IP ranges(38723)
    Group(4), Singularity(10), Number of IP ranges(142586)
    Group(5), Singularity(8), Number of IP ranges(5336)
    Group(6), Singularity(6), Number of IP ranges(113891)
    Internet Service: 2818238(Tor.Relay.Node)
    Number of IP range: 13718
    Number of IP numbers: 13718
    Singularity: 20
    Reputation: 2(Sites providing high risk services such as TOR, proxy, P2P, etc.)
    Icon Id: 43
    Second Level Domain: 1(other)
    Direction: dst
    Data source: irdb

(global) # diagnose internet-service id-summary 2818243
    Version: 0000600096
    Timestamp: 201902111802
    Total number of IP ranges: 444727
    Number of Groups: 7
    Group(0), Singularity(20), Number of IP ranges(142740)
    Group(1), Singularity(19), Number of IP ranges(1210)
    Group(2), Singularity(16), Number of IP ranges(241)
    Group(3), Singularity(15), Number of IP ranges(38723)
    Group(4), Singularity(10), Number of IP ranges(142586)
    Group(5), Singularity(8), Number of IP ranges(5336)
    Group(6), Singularity(6), Number of IP ranges(113891)
    Internet Service: 2818243(Tor.Exit.Node)
    Number of IP range: 1210
    Number of IP numbers: 1210
    Singularity: 19
    Reputation: 2(Sites providing high risk services such as TOR, proxy, P2P, etc.)
    Icon Id: 43
    Second Level Domain: 1(other)
    Direction: src
    Data source: irdb
```

IP reputation filtering

There are currently five reputation levels in the Internet Service Database (ISDB), and custom reputation levels can be defined in a custom internet service. You can configure firewall policies to filter traffic according to the desired reputation level. If the reputation level of either the source or destination IP address is equal to or greater than the level set in the policy, then the packet is forwarded, otherwise, the packet is dropped.

The five default reputation levels are:

1	Known malicious sites, such as phishing sites or sites related to botnet servers
2	High risk services sites, such as TOR, proxy, and P2P

3	Unverified sites
4	Reputable social media sites, such as Facebook and Twitter
5	Known and verified safe sites, such as Gmail, Amazon, and eBay

The default minimum reputation level in a policy is zero, meaning that the reputation filter is disabled.

For IP addresses that are not included in the ISDB, the default reputation level is three.

The default reputation direction is `destination`.

Example 1

Packets from the source IP address with reputation levels three, four, or five will be forwarded by this policy.

To set the reputation level and direction in a policy using the CLI:

```
config firewall policy
    edit 1
        set srcintf "wan2"
        set dstintf "port1"
        set srcaddr “all”
        set dstaddr "all"
        set reputation-minimum 3
        set reputation-direction source
        set action accept
        set schedule "always"
        set service "ALL"
        set logtraffic all
        set auto-asic-offload disable
        set nat enable
    next
end
```

Packets from the source IP address with reputation levels three, four, or five will be forwarded by this policy.

Example 2

This policy allows only outbound FTP traffic, if the destination server has a minimum reputation of 4.

To set the reputation level and direction in a policy using the CLI:

```
config firewall policy
    edit 1
        set srcintf "port1"
        set dstintf "wan2"
        set srcaddr “all”
        set dstaddr "all"
        set reputation-minimum 4
        set reputation-direction destination
        set action accept
```

```
        set schedule "always"
        set service "FTP"
        set logtraffic all
        set auto-asic-offload disable
        set nat enable
    next
end
```

Internet service groups in policies

This feature provides support for Internet Service Groups in traffic shaping and firewall policies. Service groups can be used as the source and destination of the policy. Internet Service Groups are used as criteria to match traffic; the shaper will be applied when the traffic matches.

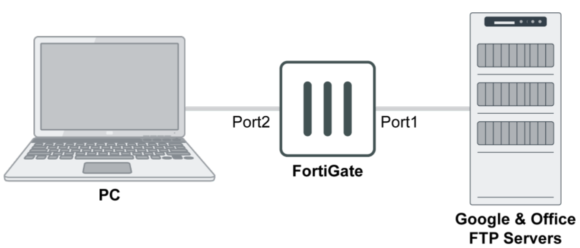
To use a group as a destination, `internet-service` must be enabled. To use a group as a source, `internet-service-src` must be enabled.

The following CLI variables are available in the `firewall policy` and `firewall shaping-policy` commands:

Variable	Description
`internet-service-group <string>`	Internet Service group name.
`internet-service-custom-group <string>`	Custom Internet Service group name.
`internet-service-src-group <string>`	Internet Service source group name.
`internet-service-src-custom-group <string>`	Custom Internet Service source group name.

Examples

The following examples use the below topology.



Example 1

In this example, the PC is allowed to access Google, so all Google services are put into an Internet Service Group.

To configure access to Google services using an Internet Service Group using the CLI:

1. Create a Service Group:

```
config firewall internet-service-group
    edit "Google_Group"
        set direction destination
```

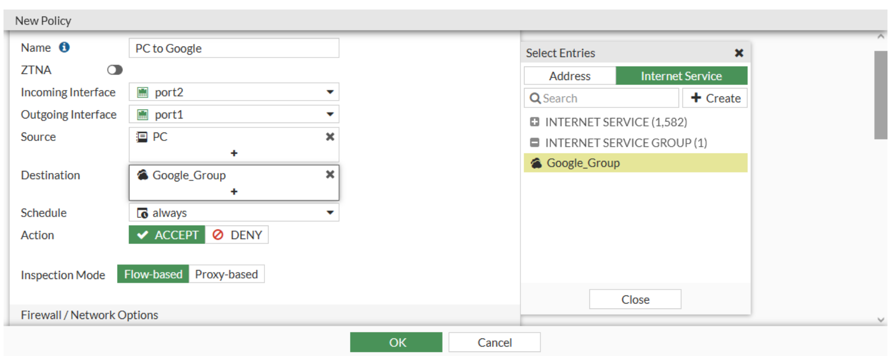
```
        set member Google-Other Google-Web Google-ICMP Google-DNS Google-Outbound_Email
Google-SSH Google-FTP Google-NTP Google-Inbound_Email Google-LDAP Google-
NetBIOS.Session.Service Google-RTMP Google-NetBIOS.Name.Service Google-Google.Cloud
Google-Gmail
    next
end
```

2. Create a firewall policy to allow access to all Google Services from the PC:

```
config firewall policy
    edit 1
        set name "PC to Google"
        set srcintf "port2"
        set dstintf "port1"
        set srcaddr "all"
        set internet-service enable
        set internet-service-group "Google_Group"
        set action accept
        set schedule "always"
        set fsso disable
        set nat enable
    next
end
```

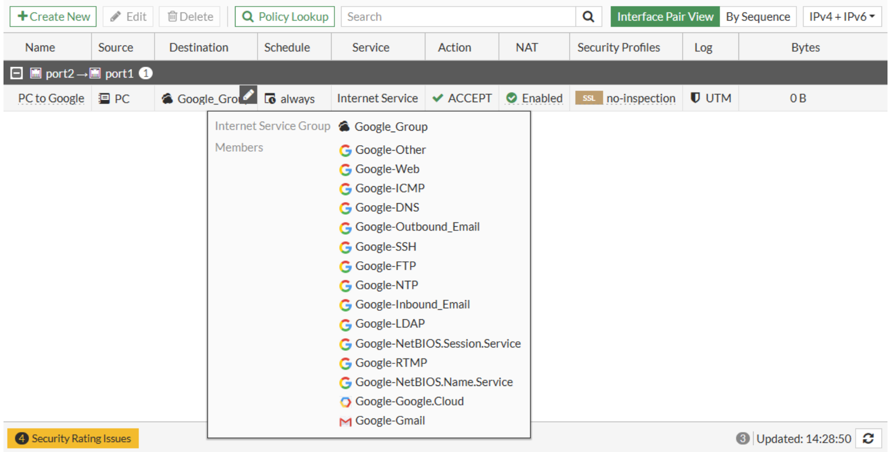
To configure access to Google services using an Internet Service Group in the GUI:

1. On the FortiGate, create a Service Group using the CLI.
2. Go to *Policy & Objects* > *Firewall Policy*, and create a new policy.
3. Set the *Destination* as the just created Internet Service Group.



4. Configure the remaining options, then click *OK*.

5. Go to *Policy & Objects > Firewall Policy* and hover over the group to view a list of its members.



Example 2

In this example, two office FTP servers are put into an Internet Custom Service Group, and the PC connection to the FTP servers is limited to 1Mbps.

To put two FTP servers into a custom service group and limit the PC connection speed to them in the CLI:

1. Create custom internet services for the internal FTP servers:

```
config firewall internet-service-custom
    edit "FTP_PM"
        config entry
            edit 1
                config port-range
                    edit 1
                        set start-port 21
                        set end-port 21
                    next
                end
                set dst "PM_Server"
            next
        end
    next
    edit "FTP_QA"
        config entry
            edit 1
                config port-range
                    edit 1
                        set start-port 21
                        set end-port 21
                    next
                end
                set dst "QA_Server"
            next
        end
```

```
    next
end
```

2. Create a custom internet server group and add the just created custom internet services to it:

```
config firewall internet-service-custom-group
    edit "Internal_FTP"
        set member "FTP_QA" "FTP_PM"
    next
end
```

3. Create a traffic shaper to limit the maximum bandwidth:

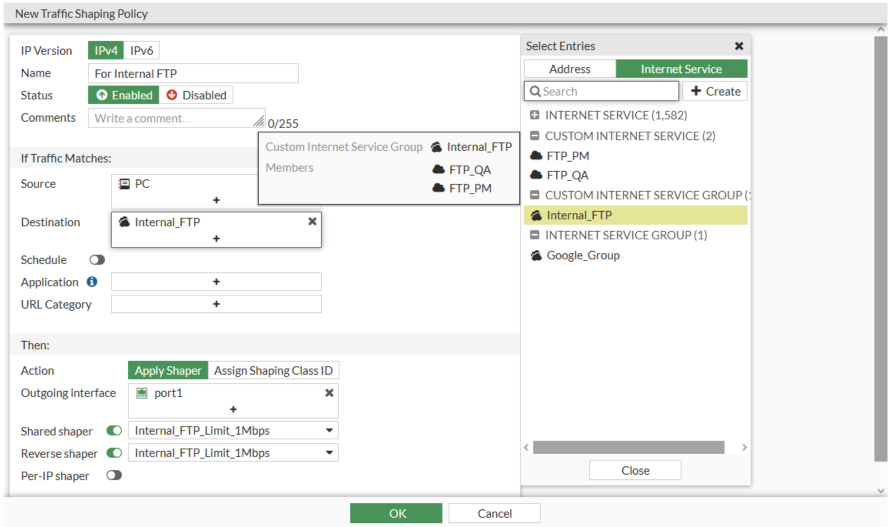
```
config firewall shaper traffic-shaper
    edit "Internal_FTP_Limit_1Mbps"
        set guaranteed-bandwidth 500
        set maximum-bandwidth 1000
        set priority medium
    next
end
```

4. Create a firewall shaping policy to limit the speed from the PC to the internal FTP servers:

```
config firewall shaping-policy
    edit 1
        set name "For Internal FTP"
        set internet-service enable
        set internet-service-custom-group "Internal_FTP"
        set dstintf "port1"
        set traffic-shaper "Internal_FTP_Limit_1Mbps"
        set traffic-shaper-reverse "Internal_FTP_Limit_1Mbps"
        set srcaddr "PC"
    next
end
```

To put two FTP servers into a custom service group and limit the PC connection speed to the in the GUI:

1. Create custom internet services for the internal FTP servers using the CLI.
2. Create a custom internet server group and add the just created custom internet services to it using the CLI.
3. Create a traffic shaper to limit the maximum bandwidth:
   a. Go to *Policy & Objects > Traffic Shaping*, select the *Traffic Shapers* tab, and click *Create New*.
   b. Enter a *Name* for the shaper, such as *Internal_FTP_Limit_1Mbps*.
   c. Set the *Traffic Priority* to *Medium*.
   d. Enable *Max Bandwidth* and set it to *1000*.
   e. Enable *Guaranteed Bandwidth* and set it to *500*.
   f. Click *OK*.
4. Create a firewall shaping policy to limit the speed from the PC to the internal FTP servers:
   a. Go to *Policy & Objects > Traffic Shaping*, select the *Traffic Shaping Policy* tab, and click *Create New*.
   b. Set the *Destination* to the just created custom internet service group, and apply the just create traffic shaper.



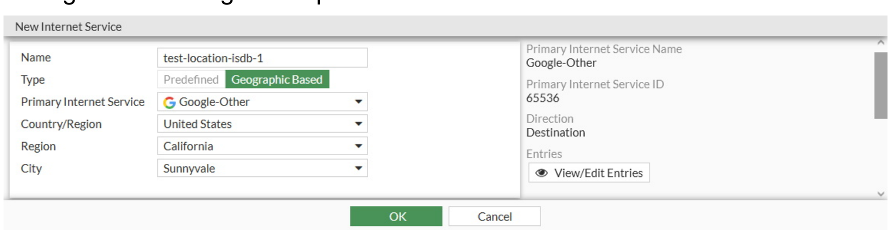
c. Configure the remaining options as shown, then click *OK*.

Allow creation of ISDB objects with regional information

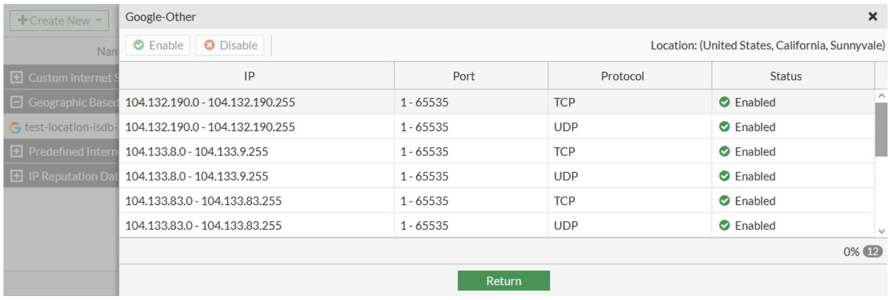
Geographic-based Internet Service Database (ISDB) objects allow users to define a country, region, and city. These objects can be used in firewall policies for more granular control over the location of the parent ISDB object. ISDB objects are now referenced in policies by name instead of ID.

To apply a location-based ISDB object to a policy in the GUI:

1. Create the ISDB object:
   - **a.** Go to *Policy & Objects* > *Internet Service Database* and click *Create New* > *Geographic Based Internet Service*.
   - **b.** Configure the settings as required.

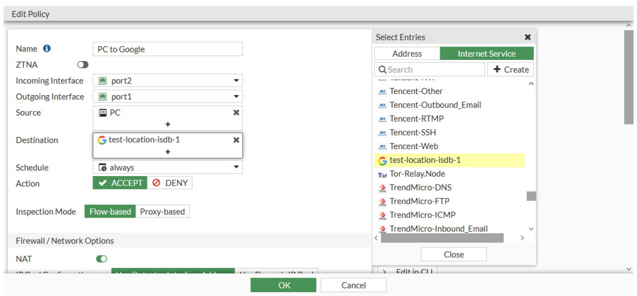


   - **c.** Click *OK*.
2. View the IP ranges in the location-based internet service:
   - **a.** Go to *Policy & Objects* > *Internet Service Database* .
   - **b.** In the table, hover over the object created in step 1 and click *View/Edit Entries*. The list of IPs is displayed:



c. Click *Return*.

3. Add the ISDB object to a policy:
   a. Go to *Policy & Objects > Firewall Policy* and create a new policy or edit an existing one.
   b. For *Destination*, click *Internet Service* and select the ISDB object created in step 1.
   c. Configure the other settings as needed.



   d. Click *OK*.

To apply a location-based ISDB object to a policy in the CLI:

1. Create the ISDB object:

```
config firewall internet-service-name
    edit "test-location-isdb-1"
        set type location
        set internet-service-id 65536
        set country-id 840
        set region-id 283
        set city-id 23352
    next
end
```

2. View the IP ranges in the location-based internet service:

```
# diagnose internet-service id 65536 | grep "country(840) region(283) city(23352)"
96.45.33.73-96.45.33.73 country(840) region(283) city(23352) blocklist(0x0) reputation
(4), domain(5) popularity(0) botnet(0) proto(6) port(1-65535)
96.45.33.73-96.45.33.73 country(840) region(283) city(23352) blocklist(0x0) reputation
(4), domain(5) popularity(0) botnet(0) proto(17) port(1-65535)
198.94.221.56-198.94.221.56 country(840) region(283) city(23352) blocklist(0x0)
reputation(4), domain(5) popularity(4) botnet(0) proto(6) port(1-65535)
198.94.221.56-198.94.221.56 country(840) region(283) city(23352) blocklist(0x0)
reputation(4), domain(5) popularity(4) botnet(0) proto(17) port(1-65535)
```

3. Add the ISDB object to a policy:

```
config firewall policy
    edit 3
        set name "PC to Google"
        set srcintf "port2"
        set dstintf "port1"
        set srcaddr "PC"
        set internet-service enable
        set internet-service-name "test-location-isdb-1"
        set action accept
        set schedule "always"
        set logtraffic all
        set logtraffic-start enable
        set auto-asic-offload disable
        set nat enable
    next
end
```

Internet service customization

Internet Service Database (ISDB) entries can be tuned for their environments by adding custom ports and port ranges, as well as port mapping.

To add a custom port range:

```
config firewall internet-service-addition
    edit 65646
        set comment "Add custom port-range:tcp/8080-8090 into 65646"
        config entry
            edit 1
                set protocol 6
                config port-range
                    edit 1
                        set start-port 8080
                        set end-port 8090
                    next
                end
            next
        end
    next
end
Warning: Configuration will only be applied after rebooting or using the 'execute internet-
service refresh' command.
```

To verify that the change was applied:

```
# diagnose internet-service info FG-traffic 6 8080 2.20.183.160
Internet Service: 65646(Google.Gmail)
```

To configure additional port mapping:

```
config firewall internet-service-append
    set match-port 10
```

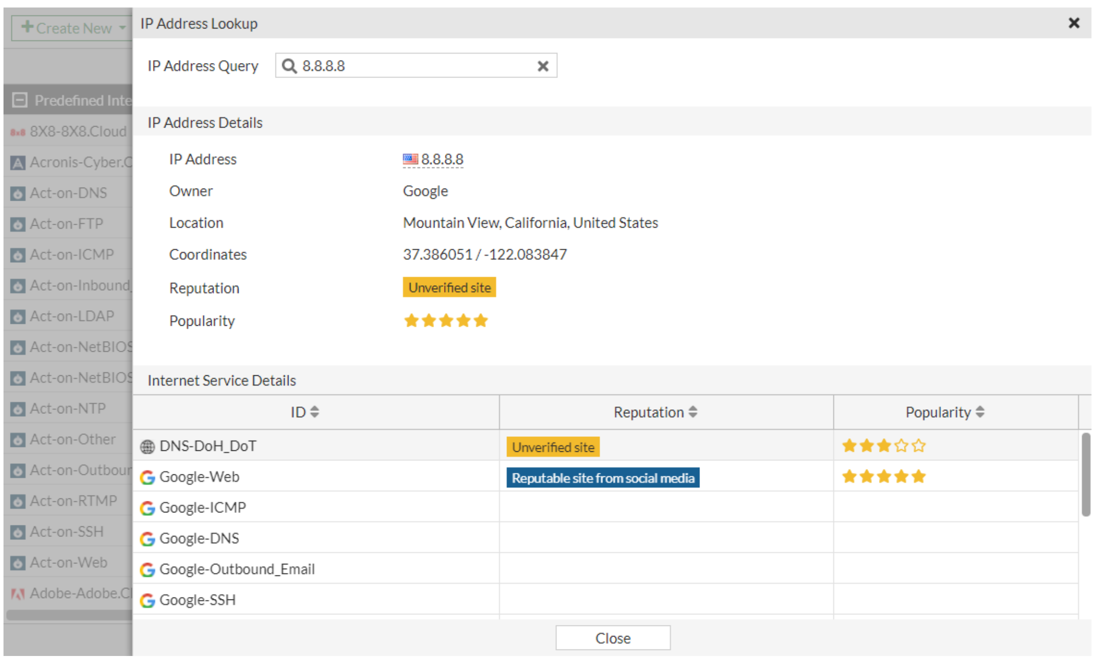
```
    set append-port 20
end
Warning: Configuration will only be applied after rebooting or using the 'execute internet-
service refresh' command.
```

Look up IP address information from the Internet Service Database page

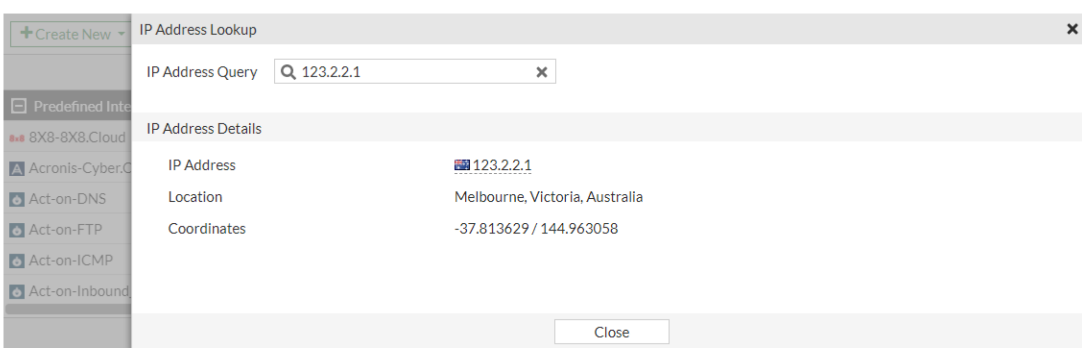
The *IP Address Lookup* button allows users to look up IP address information from the Internet Service Database and GeoIP Database. Returned IP address information includes the reverse IP address/domain lookup, location, reputation, and other internet service information.

To look up IP address information:

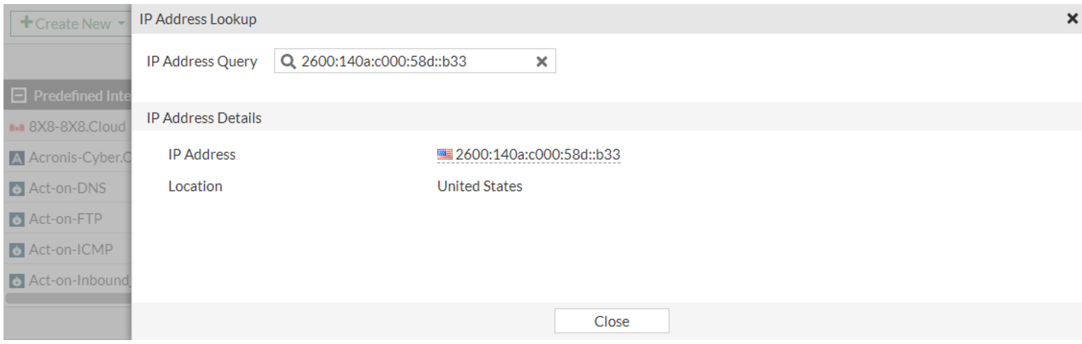
1. Go to *Policy & Objects > Internet Service Database*.
2. Click *IP Address Lookup*. The *IP Address Lookup* pane opens.
3. In the *IP Address Query* field, enter the IP address and press `Enter`.
   Results of an IP address from the Internet Service Database:



   Results of an IP address from the GeoIP Database:



   Results of an IPv6 address from the GeoIP Database:



4. Click *Close*.

Protocol options

Firewall policies contain a *Protocol Options* field that defines the parameters for handling protocol-specific traffic. Multiple protocol options profiles can be configured in FortiOS since the requirements may differ between policies. A single protocol options profile is applied per policy, but the profile can be used in multiple policies.

To create a protocol options profile, go to *Policy & Objects > Protocol Options*. The following settings can be configured.

Log oversized files

Enable this option to log the occurrence of oversized files being processed. This does not change how they are processed. It only allows the FortiGate to log that they were either blocked or allowed through.

It is common practice to allow larger files through without antivirus processing. Monitor the logs for the frequency of oversized file processing to determine whether or not to alter the settings for treating oversized files. The threshold setting for oversized files and emails is located in the *Common Options* section.

RPC over HTTP

This protocol is used by Microsoft Exchange Servers to perform virus scanning on emails that use RPC over HTTP.

Protocol port mapping

To optimize the FortiGate's resources, the mapping and inspection of the following protocols can be enabled or disabled:

- HTTP
- SMTP
- POP3
- IMAP
- FTP
- NNTP
- MAPI
- DNS
- CIFS

Each protocol has a default TCP port. The ports can be modified to inspect any port with flowing traffic. The packet headers indicate which protocol generated the packet.

Common options

The *Comfort Clients* and *Block Oversized File/Email* options apply to multiple protocols.

Comfort clients

When proxy-based antivirus scanning is enabled, the FortiGate buffers files as they are downloaded. Once the entire file is captured, the FortiGate begins scanning the file. The user must wait during the buffering and scanning procedure. After the scan is completed and if no infection is found, the file is sent to the next step in the process flow. If the file is large, this part of the process can take some time. In some cases, enough time that some users may get impatient and cancel the download.

The *Comfort Clients* option mitigates this potential issue by feeding a trickle of data while waiting for the scan to complete. The user is aware that processing is taking place, and that there has not been a failure in the transmission. The slow transfer rate continues until the antivirus scan is complete. The transfer will proceed at full speed once the file is scanned successfully and does not contain any viruses.

If there is evidence of an infection, the FortiGate caches the URL and drops the connection. The client does not receive any notification of what happened because the download to the client has already started. Instead, the download stops and the user is left with a partially downloaded file. If the user tries to download the same file again within a short period of time, the cached URL is matched and the download is blocked. A notification is displayed that the download was blocked. The number of URLs in the cache is limited by the size of the cache.

Client comforting is available for HTTP and FTP traffic. If the FortiGate supports SSL content scanning and inspection, client comforting can be configured for HTTPS and FTPS traffic.

Buffering the entire file allows the FortiGate to eliminate the danger of missing an infection due to fragmentation because the file is reassembled before examination. This buffering is performed whenever the *Comfort Clients* option is disabled.

Client comforting can send unscanned and potentially infected content to the client, so only enable this option if you are prepared to accept this risk. Keeping the client comforting interval high and the amount low will reduce the amount of potentially infected data that is downloaded.

Block oversized files and emails

This option is related to antivirus scanning. The FortiGate has a finite amount of resources to buffer and scan a file. If a large file (such as an ISO image or video file) is downloaded, this could overwhelm or exceed the FortiGate's memory, especially if other large files are being downloaded at the same time.

A threshold is assigned to identify an oversize file or email. The default is 10 MB. The range varies per model, and the minimum is 1 MB. Any file or email over this threshold will not be processed by policies applying the antivirus security profile.

If the FortiGate enters conserve mode on a regular basis, lowering the threshold can lessen the impact of processing the files on memory. This can increase risk, even though malware is more likely to be in smaller files.

Web options

The *Chunked Bypass* option applies to traffic containing web protocols.

Chunked bypass

Chunked bypass is a mechanism in HTTP 1.1 that allows a web server to start sending chunks of dynamically generated output in response to a request before actually knowing the actual size of the content. For dynamically generated content, enabling chunked bypass speeds up the initial response to HTTP requests, but the content is not held in the proxy as an entire file before proceeding.

Email options

The *Allow Fragmented Messages* and *Append Signature (SMTP)* options apply to email protocols.

Allow fragmented messages

The specifications of RFC 2046 allow for the breaking up of emails and sending the fragments in parallel to be rebuilt and read at the other end by the mail server. It was originally designed to increase the performance over slower connections where larger email messages were involved. Feasibility of using this function depends on the mail configuration. Outside of Microsoft Outlook, not many email clients are set up to break up messages like this. The drawback of this feature is that if malware is broken up between multiple fragments of the message, there is a risk that it will not be detected by some antivirus configurations because all the code may not be present at the same time to identify the malware.

Append signature

This option adds a plain text email signature to SMTP email messages as they pass through the FortiGate. The message maximum is 1023 characters.

This feature works best in an environment where there is some standardization of what goes into the senders' personal signatures so that there is no duplication or contradiction of information. For example:

- *This email should not be forwarded without prior approval.*
- *Please consider the environment before printing this email.*
- *For questions regarding purchasing our products, please call ...*

Traffic shaping

A FortiGate provides quality of service (QoS) by applying bandwidth limits and prioritization to network traffic. Traffic shaping is one technique used by the FortiGate to provide QoS. A basic approach to traffic shaping is to prioritize higher priority traffic over lower priority traffic during periods of traffic congestion. This provides a stabilizing effect for important traffic while throttling less important traffic.

The FortiGate can be configured to deliver traffic shaping with policing or traffic shaping with queuing. The general difference between the two is as follows:

Technique	Description
Traffic shaping with policing	When traffic exceeds the configured bandwidth limits, traffic is dropped.
Traffic shaping with queuing	When traffic exceeds the configured bandwidth limits, traffic is delayed for transport until bandwidth frees up. Traffic may be dropped if the queues are full.

Policing and queuing can both prioritize traffic and deliver guaranteed bandwidth and maximum bandwidth by setting bandwidth limits. The implementation differs though, since queuing uses queues, and policing does not. In queuing, before a packet egresses an interface, it is first enqueued to a queue using an algorithm such as RED or FIFO. The kernel dequeues the packet based on the HTB algorithm before sending it out. In policing, traffic simply drops if it is over the allocated bandwidth.

The following topics provide information about configuring traffic shaping:

- Traffic shaping policies on page 920
- Traffic shaping profiles on page 924
- Traffic shapers on page 934
- Global traffic prioritization on page 946
- DSCP matching and DSCP marking on page 949
- Examples on page 953

Configuration methods

There are different methods to configure traffic shaping on the FortiGate. The following table lists the methods and their capabilities in order of preference. If all three methods are configured, the first will be preferred over the second, which is preferred over the third.

Method	Policing		Queuing
	Traffic prioritization	Guaranteed and maximum bandwidth limits	Traffic queuing
Traffic shaping profile*	Yes	Yes, based on percentage of outbandwidth	Yes
Traffic shaper	Yes	Yes, based on rate	No
Global traffic prioritization	Yes	No	No

* Traffic shaping profiles are configured as either policing or queuing types. Queuing allows for additional options when configuring a shaping class entry.

The features of each method's implementation are slightly different. The following is a brief summary of the traffic policing features and the approach each method takes.

Traffic prioritization

The FortiGate can place packets into different priority levels in order to prioritize certain traffic over others.

Method	Description
Traffic shaping profile	Traffic is placed into classes. A total of 30 classes are available. For each class, traffic can be configured into five priority levels.
Traffic shaper	Traffic can be prioritized into the high (2), medium (3), or low (4) levels. When traffic is below the guaranteed bandwidth of the shaper, the traffic is automatically applied the critical level (1).
Global traffic prioritization	Traffic is prioritized into high (2), medium (3), or low (4) based on ToS (type of service) or DSCP.

Guaranteed and maximum bandwidth limits

The general purpose for configuring guaranteed bandwidth is to allocate a certain proportion of the total outbandwidth to guarantee transport for a certain type of traffic. This is configured and handled differently in each method.

A traffic shaping profile, when applied to an interface’s egress shaping profile, can be configured to use up to 100% of the interface’s configured bandwidth between all the classes. It does not matter what priority is configured in each class. The guaranteed bandwidth is always honored.

Traffic shapers, however, do not have a hard limit on the guaranteed bandwidth. Administrators need to be aware how much guaranteed bandwidth has been allocated to all their traffic shapers, so that they do not exceed the total outbandwidth of an interface. Traffic under the guaranteed bandwidth of a traffic shaper is given a priority of one. If the total traffic with priority one exceeds the total outbandwidth, traffic can be dropped.

The maximum bandwidth limit caps the maximum bandwidth that can be used. This is configured as a percentage of the outbandwidth in a traffic shaping profile. It is configured as a rate for traffic shapers.

Configuring outbandwidth

Traffic shaping is generally configured for egress traffic leaving the FortiGate. Therefore, it is necessary for the interface outbandwidth to be defined for traffic prioritization to take place in all of the traffic shaping configuration methods. Interface outbandwidth is also needed when defining the guaranteed and maximum bandwidth in a traffic shaping profile.

For traffic shapers, configuring outbandwidth is not necessary to apply maximum bandwidth limits; however, outbandwidth is necessary for guaranteed bandwidth. Traffic under the guaranteed bandwidth limit on a traffic shaper is given priority 1. If outbandwidth is not configured, traffic prioritization does not take place and the priority is meaningless.

Traffic shaping policy

Traffic shaping profiles and traffic shapers are methods of policing traffic. Traffic shaping policies are used to map traffic to a traffic shaper or assign them to a class.

A traffic shaping policy is a rule that matches traffic based on certain IP header fields and/or upper layer criteria. For example, it can match traffic based on source and destination IP, service, application, and URL category. One common use case is to match traffic based on the ToS or DS (differentiated services) field in the IP header. This allows Type of Service or Differentiated Services (DiffServ) tags to be read from traffic from a downstream device and prioritized accordingly on the FortiGate.

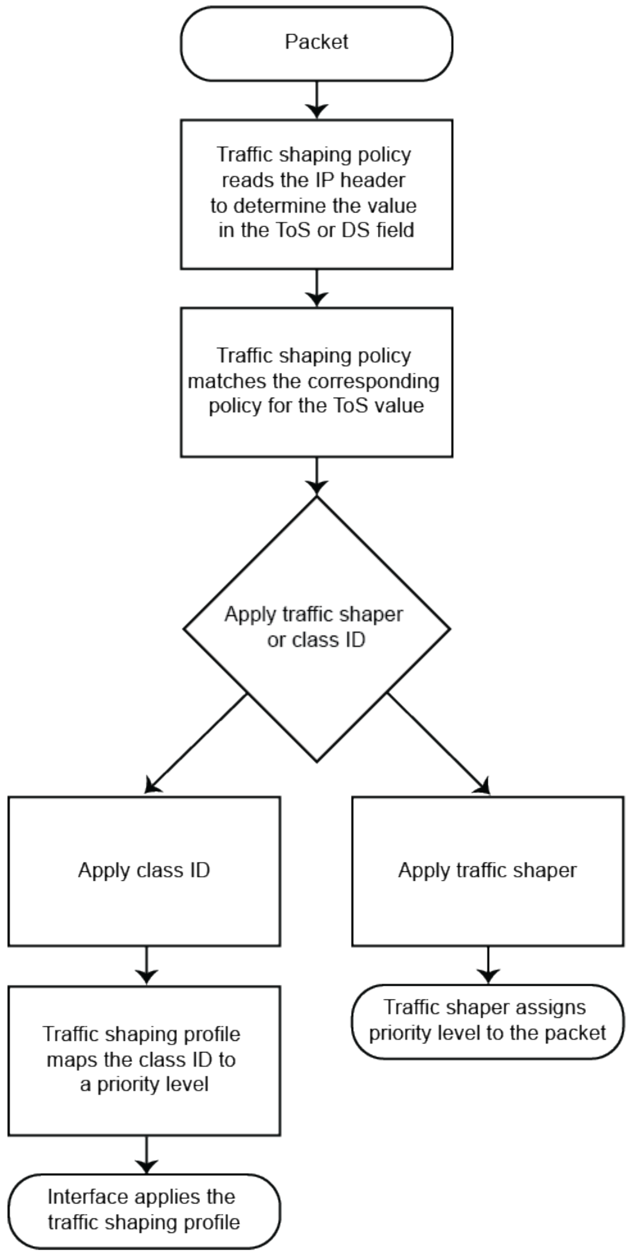
DSCP matching and DSCP marking

DSCP matching and DSCP marking can be performed on a firewall shaping policy and a regular firewall policy. DSCP matching is used to match DSCP tags from ingress traffic, and DSCP marking is used to change the DSCP tag on egress traffic.

In a firewall shaping policy and regular firewall policy, use the `tos` and `tos-mask` fields to perform DSCP matching. Use the `diffserv-forward` and `diffserv-reverse` fields to perform DSCP marking.

Traffic shaping policies

As mentioned in Traffic shaping on page 917, traffic shaping starts with the traffic shaping policy. Traffic shaping policies are used to map traffic to a traffic shaper or assign them to a class. Traffic is then shaped by the shaper or the shaping profile that is applied on an interface.



Traffic can also be shaped by applying traffic shapers directly on a firewall policy. However, this legacy approach can only be configured from the CLI, and is not a preferred method for applying traffic shaping. As the number of firewall policies increases, managing shaping on each individual policy becomes increasingly difficult. For the same reason, it is also not recommended to mix the legacy approach with traffic shaping policies to avoid the added complexity.

Overview

A traffic shaping policy is a rule that matches traffic based on certain IP header fields and/or upper layer criteria. When traffic hits the firewall, the FortiGate will first look up a firewall policy, and then match a shaping policy. The matching traffic will apply a traffic shaper, class ID, or assign a DSCP DiffServ tag to the outgoing traffic.

The traffic shaping policies must be placed in the correct order in the traffic shaping policy list page to obtain the desired results. Policies are matched from top-down, so the traffic shaping policies should be arranged in a sequence that places the more granular policies above general policies.

The policy can be configured by going to *Policy & Objects* > *Traffic Shaping* and selecting the *Traffic Shaping Policies* tab. If the menu does not display the traffic shaping settings, go to *System* > *Feature Visibility* and enable *Traffic Shaping*.

Configuring traffic shaping policies

A traffic shaping policy can be split into two parts:

- Options used to match the traffic
- Options used to apply actions to the matched traffic

In the GUI, the options are configured in the *If Traffic Matches* and *Then* sections. In the CLI, all options are configured under `config firewall shaping-policy`. Some options can only be configured from the CLI.

The following options can be configured for traffic matching criteria:

GUI option	CLI option	Description
Source		
Address	`set srcaddr <address_object>`	Select the address object to match the source IP.
User	`set users <user_object>`	Select the user object to match the user authenticated for the session.

Internet Service	`set internet-service-src enable`
	`set internet-service-src-name <name>`
	`set internet-service-src-group <group>`
	`set internet-service-src-custom <custom>`
	`set internet-service-src-custom-group <custom_group>`
		Select the internet service to match the source of the incoming traffic. Internet service currently cannot be used with source address.

Destination		
Address	`set dstaddr <address_object>`	Select the address object to match the destination IP.

Internet Service	`set internet-service enable`
	`set internet-service-name <name>`
	`set internet-service-group <group>`
	`set internet-service-custom <custom>`
		Select the internet service to match the destination of the incoming traffic. Internet service currently cannot be used with destination address and service.

GUI option	CLI option	Description
	`set internet-service-custom-group <custom_group>`	
Schedule	`set schedule <schedule>`	Enable to select a schedule (one-time, recurring, or group).
Service	`set service <service>`	Select the service or service group for the traffic.
Application		Application control must be enabled in the related firewall policy to learn the application of the traffic.
Application	`set application <application>`	Select the application to match the application of the traffic.
Category	`set app-category <category>`	Select the application category to match the application of the traffic.
Group	`set app-group <groups>`	Select the application group to match the application of the traffic.
URL Category	`set url-category <category>`	Select the URL category to match the URL of the traffic. A web filter profile must be enabled in the related firewall policy to know the URL of the traffic (see Web filter on page 1122).

n/a	`set tos-mask <hexadecimal_mask>`
	`set tos <value>`
	`set tos-negate {enable | disable}`
		Specify the type of service (ToS) and mask to match.
		These options can only be configured in the CLI.

The following options can be configured for actions to apply to the matched traffic:

GUI option	CLI option	Description
Outgoing interface	`set dstintf <interface>`	Select the destination interface that the traffic shaping applies to (required).
Apply shaper		
Shared shaper	`set traffic-shaper <shaper>`	Select the shared shaper to be applied to traffic in the ingress-to-egress direction. For example, on traffic that egresses on the wan interface, the shaper is applied to upload or outbound traffic.

GUI option	CLI option	Description
Reverse shaper	`set traffic-shaper-reverse <shaper>`	Select the reverse shaper to be applied to traffic in the egress-to-ingress direction. For example, on traffic that egresses on the wan interface, the shaper is applied to download or inbound traffic.
Per-IP shaper	`set per-ip-shaper <shaper>`	Select the per-IP shaper. Per-IP shapers affect downloads and uploads. The allotted bandwidth applies to each individual IP. In a shared shaper, the allotted bandwidth applies to all IPs.
Assign shaping class ID		
Traffic shaping class ID	`set class-id <class>`	Set the class ID to apply the matching traffic. Class IDs are further prioritized within a traffic shaping profile and applied to an interface.
n/a	`set diffserv-forward {enable | disable}` `set diffservcode-forward <code>` `set diffserv-reverse {enable | disable}` `set diffservcode-reverse <code>`	Specify the settings to apply a DSCP tag to the forward or reverse traffic. The DiffServ code is in 6-bit binary format. These options can only be configured in the CLI.

Traffic shapers and class IDs can be applied at the same time when configuring traffic shaping policies. However, to reduce the complexity, it is recommended to use one method over the other.

The following topics include examples with traffic shaping policies:

- Interface-based traffic shaping profile on page 953
- Shared traffic shaper on page 934
- Per-IP traffic shaper on page 939

Traffic shaping profiles

As mentioned in Traffic shaping on page 917, the three main methods of configuring traffic shaping are:

- Traffic shaping profiles
- Traffic shapers
- Global traffic prioritization

A traffic shaping profile allows traffic shaping to be configured with policing or queuing. Up to 30 classes can be defined, with prioritization and bandwidth limits configured for each class. When queuing is enabled, metrics can be configured for traffic queuing in each class.

Traffic shaping with policing

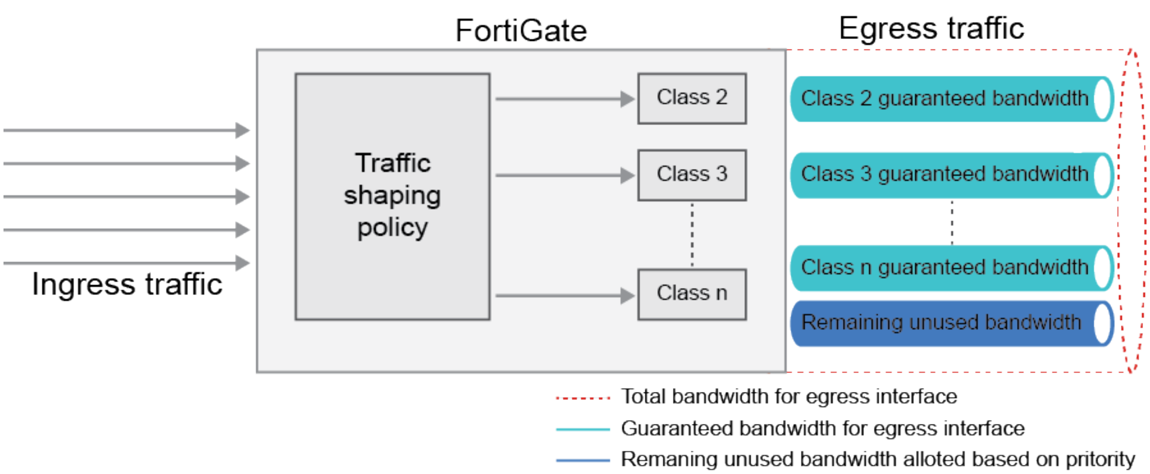
At the most basic level, policing involves traffic prioritization and bandwidth limits. Traffic prioritization helps categorize traffic into different priority levels: low, medium, high, critical, and top. When bandwidth is limited, traffic with higher priority levels will take precedence over lower priority traffic. Traffic with lower priority levels that exceeds available bandwidth will be dropped. These levels are only applicable in the context of traffic shaping profiles and should not be confused with global traffic prioritization levels.

Bandwidth limits define the guaranteed and maximum bandwidth allotted to each traffic class. These limits are configured as a percentage of the outbandwidth, which is the outbound bandwidth configured on an interface.

Guaranteed bandwidth limits guarantee the minimum bandwidth that is allotted to a given class of traffic. The sum of all guaranteed bandwidth of all classes within a traffic shaping profile cannot exceed 100%. However, the sum of all guaranteed bandwidth does not need to add up to 100%. The guaranteed bandwidth is always respected, even if one class has lower priority than another.

Maximum bandwidth limits define the maximum percentage of the outbandwidth that a traffic class can use up. This value often will be 100%, given that when there is no other traffic going through other classes, you would want to fully utilize the bandwidth of the outbound link. Traffic throughput exceeding the maximum bandwidth will be dropped.

The following diagram illustrates ingress traffic and how the FortiGate assigns classes and bandwidth to each class.



When comparing traffic shaping profiles and traffic shapers, it is important to remember that guaranteed and maximum bandwidth in a traffic shaping profile is a percentage of the outbandwidth, while guaranteed and maximum bandwidth in a traffic shaper is a rate (Kbps, Mbps, and so on). As long as the outbandwidth is true to its measurement, the bandwidth usage should not exceed the available bandwidth of a link when using a traffic shaping profile.

Congestion occurs when actual traffic surpasses the outbandwidth limit. At this point, traffic prioritization helps determine which traffic will be prioritized over others. First, the guaranteed bandwidth limit is allocated for each class. The left over bandwidth is allocated to traffic classes based on priority. The traffic classes with the highest priority can use as much of the remaining bandwidth as needed. Then, the remaining bandwidth can be allocated to classes at the next priority level, and so forth.

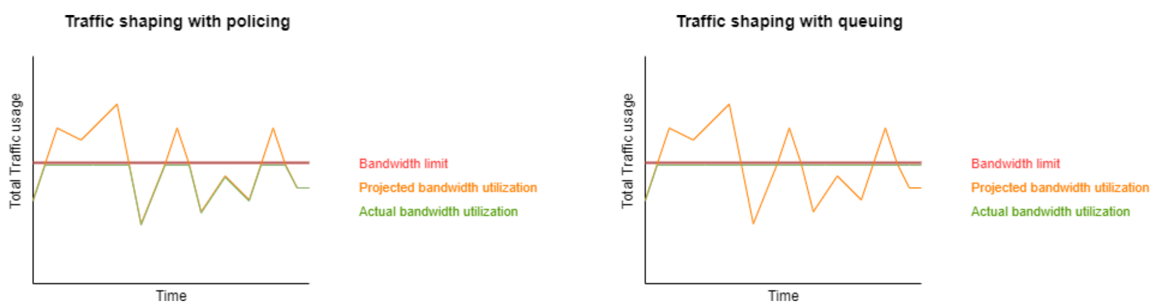
To see examples of applied traffic prioritization and bandwidth limits, see the debugs in Verifying that the traffic is being shaped on page 929.

Traffic shaping with queuing

When traffic congestion occurs and if there is no queuing, then the excess packets are dropped. With queuing, when traffic exceeds the configured bandwidth limits, the traffic is delayed for transport until bandwidth frees up. Traffic may still be dropped if the queues are full.

In queuing, before a packet egresses an interface, it is first enqueued using an algorithm, such as random early detection (RED) or first in, first out (FIFO). The kernel then dequeues the packet based on the HTB algorithm before sending it out. Queuing can be configured per shaping profile, and it can be customized per class.

The following diagram shows how traffic policing differs from traffic queuing by comparing the bandwidth limit, projected bandwidth utilization, and actual bandwidth utilization.



For more information about traffic shaping with queuing, see Traffic shaping with queuing using a traffic shaping profile on page 930.

Configuring traffic shaping profiles

The main steps to configure traffic shaping are:

1. Configure the traffic shaping policy, and assign matched traffic to a class (see Traffic shaping policies on page 920).
2. Configure the traffic shaping profile and apply traffic bandwidth, prioritization and/or queuing per class.
3. Configure the interface outbandwidth and apply an egress shaping profile to the interface.

Configuring the traffic shaping profile

A traffic shaping profile consists of the class ID and the settings per class ID. It also defines the type of traffic shaping to apply (policing or queuing) and the default class ID for traffic that does not match any traffic shaping policies.

A class can be configured in the GUI as part of a traffic shaping profile or policy. In the CLI, a traffic class must be defined before it can be assigned within a traffic shaping profile. Class IDs range from 2 - 31, and they can be reused between different traffic shaping profiles.

When configuring a traffic shaping profile, the settings can be defined per class.

The following options can be configured for traffic shaping classes:

GUI option	CLI option	Description
Default	`set default-class-id <class-id>`	Set the default class ID. Each profile must have one default class ID. The default class ID can be changed at any time.

GUI option	CLI option	Description
Traffic shaping class ID	`set class-id <integer>`	Set the class ID (2 - 31).
Guaranteed bandwidth	`set guaranteed-bandwidth-percentage <integer>`	Set the percentage of the outbandwidth that will be guaranteed for the class ID.
Maximum bandwidth	`set maximum-bandwidth-percentage <integer>`	Set the percentage of the outbandwidth that will be the maximum bandwidth for the class ID.
Priority	`set priority {top | critical | high | medium | low}`	Select the priority level for the class ID.

To configure a traffic shaping profile in the GUI:

1. Go to *Policy & Objects* > *Traffic Shaping*, select the *Traffic Shaping Profiles* tab, and click *Create New*.
2. Enter the profile name, and optionally enter a comment.
3. In the *Traffic Shaping Classes* section, click *Create New*.
4. Configure the traffic shaping class ID settings (*Traffic shaping class ID*, *Guaranteed bandwidth*, *Maximum bandwidth*, and *Priority*).
5. Click *OK*.
6. Create more shaping classes as needed (the total guaranteed bandwidth of all classes cannot exceed 100%).
7. Click *OK*.

To configure a traffic shaping profile in the CLI:

1. Configure the shaping class:

```
config firewall traffic-class
    edit <integer>
        set class-name <string>
    next
end
```

2. Configure the shaping profile:

```
config firewall shaping-profile
    edit <name>
        set type {policing | queuing}
        set default-class-id <class-id>
        config shaping-entries
            edit <id>
                set class-id <integer>
                set priority {top | critical | high | medium | low}
                set guaranteed-bandwidth-percentage <integer>
                set maximum-bandwidth-percentage <integer>
            next
        end
    next
end
```

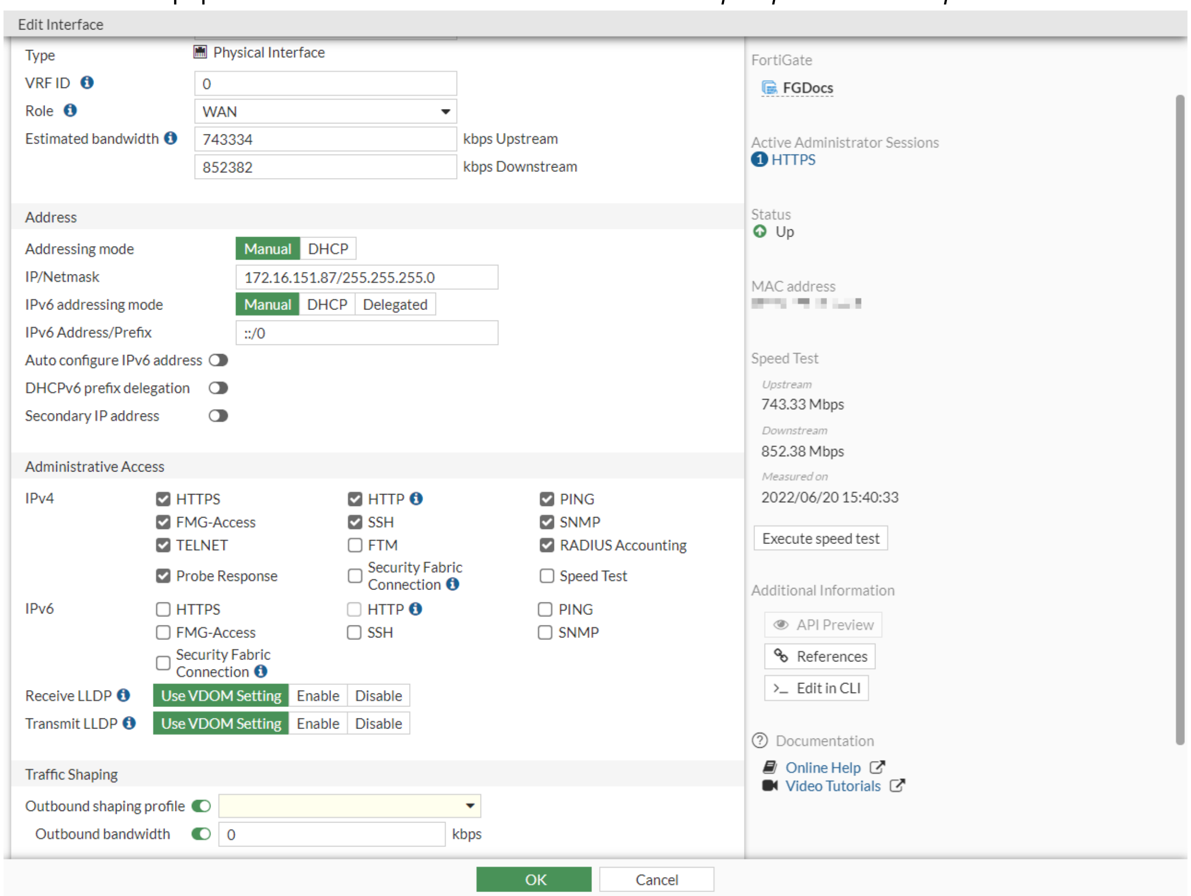
Configuring the interface outbandwidth

There are two settings that must be configured on an interface that has traffic shaping applied to egressing traffic: a traffic shaping profile must be assigned, and the outbound bandwidth must be configured.

Since traffic shaping is often configured on the WAN interface for egressing traffic, the outbound bandwidth is effectively the upstream bandwidth allowed by your ISP. On the FortiGate, it is possible to perform a speed test on interfaces are assigned a WAN role assigned (see Manual interface speedtest on page 526). The speed test performs measurements against public cloud servers, and provides an accurate measurement of the upstream bandwidth. After the test is complete, the results can be used to populate the *Outbound bandwidth* field.

To configure traffic shaping on an interface:

1. Go to *Network > Interfaces* and double-click an interface to edit it.
2. For interfaces assigned a *WAN* role, in the right-side of the screen, click *Execute speed test*.
3. When the test completes, click *OK* in the *Confirm* pane to apply the results to the estimated bandwidth. The speed test results are populated in the *Estimated bandwidth* fields for *kbps Upstream* and *kbps Downstream*.



4. In the *Traffic Shaping* section, enable *Outbound shaping profile* and select a profile.
5. Enable *Outbound bandwidth* and copy the *kbps Upstream* value from the speed test, or enter a custom value.
6. Click *OK*.

Verifying that the traffic is being shaped

In this example, three traffic classes are defined in the traffic shaping profile assigned to port1. The outbandwidth configured on port1 is 1000 Kbps. Each class has an `allocated-bandwidth`, `guaranteed-bandwidth`, `max-bandwidth`, and `current-bandwidth` value.

- The `guaranteed-bandwidth` and `max-bandwidth` are rates that are converted from the percentage of outbandwidth configured for each class. For example, `class-id` 2 has 10% `guaranteed-bandwidth`, equivalent to 100 Kbps, and 100% `max-bandwidth` equivalent to 1000 Kbps.
- The `allocated-bandwidth` displays the real-time bandwidth allocation for the traffic class based on all available factors. This value changes as traffic demand changes.
- The `current-bandwidth` displays the real-time bandwidth usage detected for the traffic class.

To verify that traffic is being shaped by the traffic shaping profile:

1. Enable debug flow to view the live traffic as it matches a traffic shaping policy:
   ```
   # diagnose debug flow show function-name enable
   # diagnose debug flow show iprope enable
   # diagnose debug flow filter <filters>
   # diagnose debug flow trace start <repeat_number>
   # diagnose debug enable
   ```
   The `iprope_shaping_check` function outputs the shaping policy matched for any given traffic:
   ```
   ...
   id=20085 trace_id=21 func=iprope_shaping_check line=934 msg="in-[port3], out-[port1],
   skb_flags-02000000, vid-0"
   id=20085 trace_id=21 func=__iprope_check line=2277 msg="gnum-100015, check-
   ffffffffa002a8fe"
   id=20085 trace_id=21 func=__iprope_check_one_policy line=2029 msg="checked gnum-100015
   policy-3, ret-matched, act-accept"
   id=20085 trace_id=21 func=__iprope_check_one_policy line=2247 msg="policy-3 is matched,
   act-accept"
   id=20085 trace_id=21 func=__iprope_check line=2294 msg="gnum-100015 check result: ret-
   matched, act-accept, flag-00000000, flag2-00000000"
   ```
2. Display the session list:
   ```
   # diagnose sys session filter <filters>
   # diagnose sys session list
   ```
   Sessions that match a shaping policy will display `class_id` and `shaping_policy_id` fields:
   ```
   ...
   session info: proto=6 proto_state=05 duration=32 expire=0 timeout=3600 flags=00000000
   socktype=0 sockport=0 av_idx=0 use=4
   origin-shaper=
   reply-shaper=
   per_ip_shaper=
   class_id=4 shaping_policy_id=3 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/255
   ```
3. Display the interface statistics:
   ```
   # diagnose netlink interface list port1
   if=port1 family=00 type=1 index=3 mtu=1500 link=0 master=0
   ref=95 state=start present fw_flags=2001b800 flags=up broadcast run allmulti multicast
   Qdisc=pfifo_fast hw_addr=52:54:00:7e:af:a6 broadcast_addr=ff:ff:ff:ff:ff:ff
   inbandwidth=10000(kbps)         total_bytes=2098887K    drop_bytes=7854K
   ```

```
egress traffic control:
        bandwidth=1000(kbps) lock_hit=241 default_class=3 n_active_class=3
        class-id=2      allocated-bandwidth=140(kbps)   guaranteed-bandwidth=100(kbps)
                        max-bandwidth=1000(kbps)        current-bandwidth=147(kbps)
                        priority=low    forwarded_bytes=8161K
                        dropped_packets=2032    dropped_bytes=3074K
        class-id=3      allocated-bandwidth=30(kbps)    guaranteed-bandwidth=300(kbps)
                        max-bandwidth=1000(kbps)        current-bandwidth=10(kbps)
                        priority=medium         forwarded_bytes=501K
                        dropped_packets=1       dropped_bytes=1195
        class-id=4      allocated-bandwidth=830(kbps)   guaranteed-bandwidth=500(kbps)
                        max-bandwidth=1000(kbps)        current-bandwidth=810(kbps)
                        priority=high   forwarded_bytes=1393K
                        dropped_packets=379     dropped_bytes=572K
stat: rxp=8349728 txp=11101735 rxb=2216101183 txb=1394077978 rxe=0 txe=0 rxd=0 txd=0
mc=0 collision=0 @ time=1654202868
re: rxl=0 rxo=0 rxc=0 rxf=0 rxfi=0 rxm=0
te: txa=0 txc=0 txfi=0 txh=0 txw=0
misc rxc=0 txc=0
input_type=0 state=3 arp_entry=0 refcnt=95
```

If the debug output does not display egress traffic control by class and displays them by priority, it is likely that global traffic prioritization is configured. The global traffic prioritization settings must be disabled to view the preceding debug output (see Global traffic prioritization on page 946).

Traffic shaping with queuing using a traffic shaping profile

You can use the weighted random early detection (WRED) queuing function within traffic shaping.

This topic includes two parts:

- Traffic shaping with queuing on page 930
- Burst control in queuing mode on page 931

You cannot configure or view WRED in the GUI; you must use the CLI.

WRED is not supported when traffic is offloaded to an NPU.

Traffic shaping with queuing

Traffic shaping has a queuing option. Use this option to fine-tune the queue by setting the profile queue size or performing random early drop (RED) according to queue usage.

This example shows setting the profile queue size limit to 5 so that the queue can contain a maximum of five packets and more packets are dropped.

To set the profile queue size limit:

```
config firewall shaping-profile
    edit "profile"
```

```
        set type queuing
        set default-class-id 31
        config shaping-entries
            edit 31
                set class-id 31
                set guaranteed-bandwidth-percentage 5
                set maximum-bandwidth-percentage 10
                set limit 5  <range from 5 to 10000; default: 1000>
            next
        end
    next
end
```

This example shows performing RED according to queue usage by setting `red-probability`, `min`, and `max`. Setting `red-probability` to `10` means start to drop packets when queue usage reaches the `min` setting. When queue usage reaches the `max` setting, drop 10% of the packets.

- Level 1: when queue is less than `min` packets, drop 0% of packets.
- Level 2: when queue reaches `min` packets, start to drop packets.
- Level 3: when queue usage is between `min` and `max` packets, drop 0–10% of packets by proportion.
- Level 4: when queue (average queue size) is more than `max` packets, drop 100% of packets.

To set RED according to queue usage:

```
config firewall shaping-profile
    edit "profile"
        set type queuing
        set default-class-id 31
        config shaping-entries
            edit 31
                set class-id 31
                set guaranteed-bandwidth-percentage 5
                set maximum-bandwidth-percentage 10
                set red-probability 10 <range from 0 to 20; default: 0 no drop>
                set min 100 <range from 3 to 3000>
                set max 300 <range from 3 to 3000>
            next
        end
    next
end
```

To troubleshoot this function, use the following diagnose commands:

```
diagnose netlink intf-class list <intf>
diagnose netlink intf-qdisc list <intf>
```

Burst control in queuing mode

In a hierarchical token bucket (HTB) algorithm, each traffic class has buckets to allow a burst of traffic. The maximum burst is determined by the bucket size `burst` (for guaranteed bandwidth) and `cburst` (for maximum bandwidth). The shaping profile has `burst-in-msec` and `cburst-in-msec` parameters for each shaping entry (`class id`) to control the bucket size.

This example uses the outbandwidth of the interface as 1 Mbps and the maximum bandwidth of class is 50%.

burst = burst-in-msec * guaranteed bandwidth = 100 ms × 1 Mbps x 50% = 50000 b = 6250 B

cburst = cburst-in-msec * maximum bandwidth = 200 ms × 1 Mbps x 50% = 100000 b = 12500 B

The following example sets `burst-in-msec` to 100 and `cburst-in-msec` to 200.

To set burst control in queuing mode:

```
config firewall shaping-profile
    edit "profile"
        set type queuing
        set default-class-id 31
        config shaping-entries
            edit 31
                set class-id 31
                set guaranteed-bandwidth-percentage 5
                set maximum-bandwidth-percentage 50
                set burst-in-msec 100 <range from 0 to 2000>
                set cburst-in-msec 200 <range from 0 to 2000>
            next
        end
    next
end
```

Example

Enabling RED for FTP traffic from QA

This example shows how to enable RED for FTP traffic from QA. This example sets a maximum of 10% of the packets to be dropped when queue usage reaches the maximum value.

To configure the firewall address:

```
config firewall address
    edit QA_team
        set subnet 10.1.100.0/24
    next
end
```

To set the shaping policy to classify traffic into different class IDs:

```
config firewall shaping-policy
    edit 1
        set service HTTPS HTTP
        set dstintf port1
        set srcaddr QA_team
        set dstaddr all
        set class-id 10
    next
    edit 2
        set service FTP
        set dstintf port1
        set srcaddr QA_team
        set dstaddr all
        set class-id 20
    next
end
```

To set the shaping policy to define the speed of each class ID:

```
config firewall shaping-profile
    edit QA_team_profile
       set type queuing
       set default-class-id 30
            config shaping-entries
                edit 1
                    set class-id 10
                    set guaranteed-bandwidth-percentage 50
                    set maximum-bandwidth-percentage 100
                next
                edit 2
                    set class-id 20
                    set guaranteed-bandwidth-percentage 30
                    set maximum-bandwidth-percentage 60
                    set red-probability 10
                next
                edit 3
                    set class-id 30
                    set guaranteed-bandwidth-percentage 20
                    set maximum-bandwidth-percentage 50
                next
            end
    next
end
```

To apply the shaping policy to the interface:

```
config sys interface
    edit port1
        set outbandwidth 10000
        set egress-shaping-profile QA_team_profile
    next
end
```

To use diagnose commands to troubleshoot:

```
# diagnose netlink intf-class list port1
class htb 1:1 root rate 1250000Bps ceil 1250000Bps burst 1600B/8 mpu 0B overhead 0B cburst
1600B/8 mpu 0B overhead 0B level 7 buffer [00004e20] cbuffer [00004e20]
 Sent 11709 bytes 69 pkt (dropped 0, overlimits 0 requeues 0)
 rate 226Bps 2pps backlog 0B 0p
 lended: 3 borrowed: 0 giants: 0
 tokens: 18500 ctokens: 18500
class htb 1:10 parent 1:1 leaf 10: prio 1 quantum 62500 rate 625000Bps ceil 1250000Bps burst
1600B/8 mpu 0B overhead 0B cburst 1600B/8 mpu 0B overhead 0B level 0 buffer [00009c40]
cbuffer [00004e20]
 Sent 0 bytes 0 pkt (dropped 0, overlimits 0 requeues 0)
 rate 0Bps 0pps backlog 0B 0p
 lended: 0 borrowed: 0 giants: 0
 tokens: 40000 ctokens: 20000
class htb 1:20 parent 1:1 leaf 20: prio 1 quantum 37500 rate 375000Bps ceil 750000Bps burst
1599B/8 mpu 0B overhead 0B cburst 1599B/8 mpu 0B overhead 0B level 0 buffer [0001046a]
cbuffer [00008235]
 Sent 0 bytes 0 pkt (dropped 0, overlimits 0 requeues 0)
```

```
 rate 0Bps 0pps backlog 0B 0p
 lended: 0 borrowed: 0 giants: 0
 tokens: 66666 ctokens: 33333
class htb 1:30 parent 1:1 leaf 30: prio 1 quantum 25000 rate 250000Bps ceil 625000Bps burst
1600B/8 mpu 0B overhead 0B cburst 1600B/8 mpu 0B overhead 0B level 0 buffer [000186a0]
cbuffer [00009c40]
 Sent 11709 bytes 69 pkt (dropped 0, overlimits 0 requeues 0)
 rate 226Bps 2pps backlog 0B 0p
 lended: 66 borrowed: 3 giants: 0
 tokens: 92500 ctokens: 37000
class red 20:1 parent 20:0
```

```
 # diagnose netlink intf-qdisc list port1
qdisc htb 1: root refcnt 5 r2q 10 default 30 direct_packets_stat 0 ver 3.17
 Sent 18874 bytes 109 pkt (dropped 0, overlimits 5 requeues 0)
 backlog 0B 0p
qdisc pfifo 10: parent 1:10 refcnt 1 limit 1000p
 Sent 0 bytes 0 pkt (dropped 0, overlimits 0 requeues 0)
 backlog 0B 0p
qdisc red 20: parent 1:20 refcnt 1 limit 4000000B min 300000B max 1000000B ewma 9 Plog 23
Scell_log 20 flags 0
 Sent 0 bytes 0 pkt (dropped 0, overlimits 0 requeues 0)
 backlog 0B 0p
  marked 0 early 0 pdrop 0 other 0
qdisc pfifo 30: parent 1:30 refcnt 1 limit 1000p
 Sent 18874 bytes 109 pkt (dropped 0, overlimits 0 requeues 0)
 backlog 0B 0p
```

Traffic shapers

The following topics provide more information about traffic shapers:

- Shared traffic shaper on page 934
- Per-IP traffic shaper on page 939
- Changing traffic shaper bandwidth unit of measurement on page 942
- Multi-stage DSCP marking and class ID in traffic shapers on page 942
- Adding traffic shapers to multicast policies on page 944

Shared traffic shaper

Shared traffic shaper is used in a firewall shaping policy to indicate the priority and guaranteed and maximum bandwidth for a specified type of traffic use.

The maximum bandwidth indicates the largest amount of traffic allowed when using the policy. You can set the maximum bandwidth to a value between 1 and 16776000 Kbps. The GUI displays an error if any value outside this range is used. If you want to allow unlimited bandwidth, use the CLI to enter a value of 0.

The guaranteed bandwidth ensures that there is a consistent reserved bandwidth available. When setting the guaranteed bandwidth, ensure that the value is significantly less than the interface's bandwidth capacity. Otherwise, the interface will allow very little or no other traffic to pass through, potentially causing unwanted latency.

In a shared traffic shaper, the administrator can prioritize certain traffic as high, medium, or low. FortiOS provides bandwidth to low priority connections only when high priority connections do not need the bandwidth. For example, you should assign a high traffic priority to a policy for connecting a secure web server that needs to support e-commerce traffic. You should assign less important services a low priority.

When you configure a shared traffic shaper, you can apply bandwidth shaping per policy or for all policies. By default, a shared traffic shaper applies traffic shaping evenly to all policies that use the shared traffic shaper.

When configuring a per-policy traffic shaper, FortiOS applies the traffic shaping rules defined for each security policy individually. For example, if a per-policy traffic shaper is configured with a maximum bandwidth of 1000 Kbps, any security policies that have that traffic shaper enabled get 1000 Kbps of bandwidth each.

If a traffic shaper for all policies is configured with a maximum bandwidth of 1000 Kbps, all policies share the 1000 Kbps on a first-come, first-served basis.

The configuration is as follows:

```
config firewall shaper traffic-shaper
   edit "traffic_shaper_name"
      set per-policy enable
   next
end
```

The shared traffic shaper selected in the traffic shaping policy affects traffic in the direction defined in the policy. For example, if the source port is LAN and the destination is WAN1, the traffic shaping affects the flow in this direction only, affecting the outbound traffic's upload speed. You can define the traffic shaper for the policy in the opposite direction (reverse shaper) to affect the inbound traffic's download speed. In this example, that would be from WAN1 to LAN.

Only traffic through forward traffic shapers will be included in FortiView; reverse and per-IP shapers are not included.

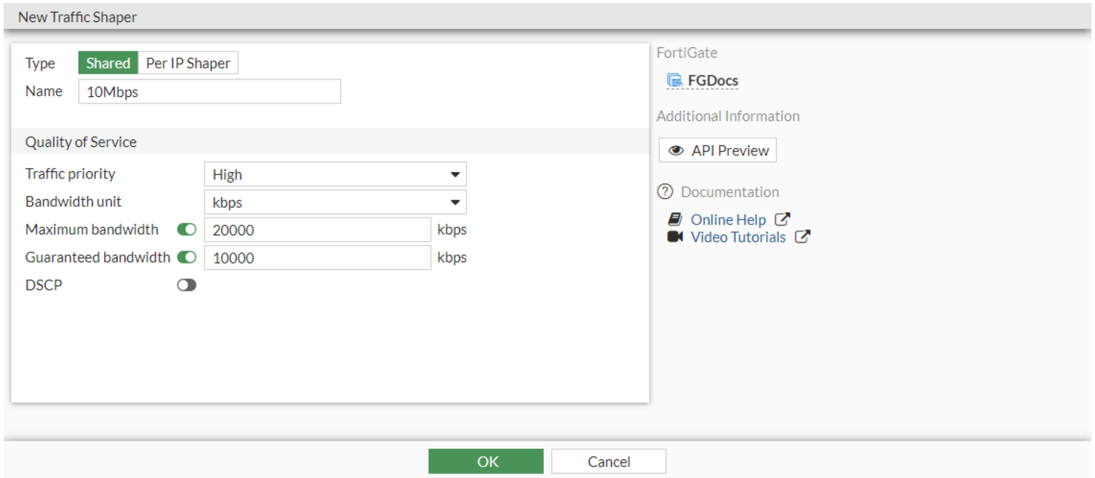
Traffic shapers can be added to a multicast policy when multicast routing is enabled.

The following example shows how to apply different speeds to different types of service. The example configures two shared traffic shapers to use in two firewall shaping policies. One policy guarantees a speed of 10 Mbps for VoIP traffic. The other policy guarantees a speed of 1 Mbps for other traffic. In the example, FortiOS communicates with a PC using port10 and the Internet using port9.

To configure shared traffic shapers in the GUI:

1. Create a firewall policy:
   a. Go to *Policy & Objects > Firewall Policy* and click *Create New*.
   b. Set the *Name* to *Internet Access*.
   c. Set the *Incoming Interface* to *port10*.
   d. Set the *Outgoing Interface* to *port9*.
   e. Set the *Source* and *Destination* to *all*.
   f. Set the *Schedule* to *always*.
   g. Set the *Service* to *ALL*.
   h. Click *OK*.
2. Create the shared traffic shapers:
   a. Go to *Policy & Objects > Traffic Shaping*, select the *Traffic Shapers* tab, and click *Create New*.
   b. Set the *Name* to *10Mbps*. This shaper is for VoIP traffic.
   c. Set the *Traffic Priority* to *High*.
   d. Enable *Max Bandwidth* and enter *20000*.

   e. Enable *Guaranteed Bandwidth* and enter *10000*.



   f. Click *OK*.
   g. Repeat the above steps to create another traffic shaper named *1Mbps* with the *Traffic Priority* set to *Low*, the *Max Bandwidth* set to *10000*, and the *Guaranteed Bandwidth* set to *1000*.
3. Create a firewall shaping policy:
   a. Go to *Policy & Objects > Traffic Shaping*, select the *Traffic Shaping Policies* tab, and click *Create New*.
   b. Set the *Name* to *VoIP_10Mbps_High*. This policy is for VoIP traffic.
   c. Set the *Source* and *Destination* to *all*.
   d. Set the *Service* to all VoIP services.
   e. Set the *Outgoing Interface* to *port9*.
   f. Enable *Shared shaper* and select *10Mbps*.
   g. Enable *Reverse shaper* and select *10Mbps*.
   h. Click *OK*.
   i. Repeat the above steps to create another firewall shaping policy named *Other_1Mbps_Low* for other traffic, with the *Source* and *Destination* set to *all*, *Service* set to *ALL*, *Outgoing Interface* set to *port9*, and *Shared shaper* and *Reverse shaper* set to *1Mbps*.

To configure shared traffic shapers in the CLI:

1. Create a firewall policy:
```
config firewall policy
   edit 1
      set name "Internet Access"
      set srcintf "port10"
      set dstintf "port9"
      set srcaddr "all"
      set dstaddr "all"
      set action accept
      set schedule "always"
      set service "ALL"
      set fsso disable
      set nat enable
   next
end
```
2. Create the shared traffic shapers:
```
config firewall shaper traffic-shaper
```

```
    edit "10Mbps"
        set guaranteed-bandwidth 10000
        set maximum-bandwidth 20000
    next
    edit "1Mbps"
        set guaranteed-bandwidth 1000
        set maximum-bandwidth 10000
        set priority low
    next
end
```

3. Create a firewall shaping policy:

```
config firewall shaping-policy
    edit 1
        set name "VOIP_10Mbps_High"
        set service "H323" "IRC" "MS-SQL" "MYSQL" "RTSP" "SCCP" "SIP" "SIP-MSNmessenger"
        set dstintf "port9"
        set traffic-shaper "10Mbps"
        set traffic-shaper-reverse "10Mbps"
        set srcaddr "all"
        set dstaddr "all"
    next
    edit 2
        set name "Other_1Mbps_Low"
        set service "ALL"
        set dstintf "port9"
        set traffic-shaper "1Mbps"
        set traffic-shaper-reverse "1Mbps"
        set srcaddr "all"
        set dstaddr "all"
    next
end
```

To troubleshoot shared traffic shapers:

1. Check if specific traffic is attached to the correct traffic shaper. The example output shows the traffic attached to the 10Mbps and 1Mbps shapers:

```
# diagnose firewall iprope list 100015
policy index=1 uuid_idx=0 action=accept
flag (0):
shapers: orig=10Mbps(2/1280000/2560000)
cos_fwd=0 cos_rev=0
group=00100015 av=00000000 au=00000000 split=00000000
host=4 chk_client_info=0x0 app_list=0 ips_view=0
misc=0 dd_type=0 dd_mode=0
zone(1): 0 -> zone(1): 38
source(1): 0.0.0.0-255.255.255.255, uuid_idx=0,
dest(1): 0.0.0.0-255.255.255.255, uuid_idx=0,
service(15):
    [6:0x0:0/(1,65535)->(1720,1720)] helper:auto
    [6:0x0:0/(1,65535)->(1503,1503)] helper:auto
    [17:0x0:0/(1,65535)->(1719,1719)] helper:auto
    [6:0x0:0/(1,65535)->(6660,6669)] helper:auto
    [6:0x0:0/(1,65535)->(1433,1433)] helper:auto
    [6:0x0:0/(1,65535)->(1434,1434)] helper:auto
    [6:0x0:0/(1,65535)->(3306,3306)] helper:auto
    [6:0x0:0/(1,65535)->(554,554)] helper:auto
```

```
   [6:0x0:0/(1,65535)->(7070,7070)] helper:auto
   [6:0x0:0/(1,65535)->(8554,8554)] helper:auto
   [17:0x0:0/(1,65535)->(554,554)] helper:auto
   [6:0x0:0/(1,65535)->(2000,2000)] helper:auto
   [6:0x0:0/(1,65535)->(5060,5060)] helper:auto
   [17:0x0:0/(1,65535)->(5060,5060)] helper:auto
   [6:0x0:0/(1,65535)->(1863,1863)] helper:auto

policy index=2 uuid_idx=0 action=accept
flag (0):
shapers: orig=1Mbps(4/128000/1280000)
cos_fwd=0 cos_rev=0
group=00100015 av=00000000 au=00000000 split=00000000
host=4 chk_client_info=0x0 app_list=0 ips_view=0
misc=0 dd_type=0 dd_mode=0
zone(1): 0 -> zone(1): 38
source(1): 0.0.0.0-255.255.255.255, uuid_idx=0,
dest(1): 0.0.0.0-255.255.255.255, uuid_idx=0,
service(1):
   [0:0x0:0/(0,0)->(0,0)] helper:auto
```

2. Check if the correct traffic shaper is applied to the session. The example output shows that the 1Mbps shaper is applied to the session:

```
# diagnose sys session list
session info: proto=6 proto_state=01 duration=11 expire=3599 timeout=3600 flags=00000000
     sockflag=00000000 sockport=0 av_idx=0 use=5
origin-shaper=1Mbps prio=4 guarantee 128000Bps max 1280000Bps traffic 1050Bps drops 0B
reply-shaper=
per_ip_shaper=
class_id=0 shaping_policy_id=2 ha_id=0 policy_dir=0 tunnel=/ helper=ftp vlan_cos=0/255
state=may_dirty npu npd os mif route_preserve
statistic(bytes/packets/allow_err): org=868/15/1 reply=752/10/1 tuples=2
tx speed(Bps/kbps): 76/0 rx speed(Bps/kbps): 66/0
orgin->sink: org pre->post, reply pre->post dev=39->38/38->39 gwy=172.16.200.55/0.0.0.0
hook=post dir=org act=snat 10.1.100.11:58241->172.16.200.55:21(172.16.200.1:58241)
hook=pre dir=reply act=dnat 172.16.200.55:21->172.16.200.1:58241(10.1.100.11:58241)
pos/(before,after) 0/(0,0), 0/(0,0)
misc=0 policy_id=1 auth_info=0 chk_client_info=0 vd=4
serial=0003255f tos=ff/ff app_list=0 app=0 url_cat=0
sdwan_mbr_seq=0 sdwan_service_id=0
rpdb_link_id = 00000000
dd_type=0 dd_mode=0
npu_state=0x100000
npu info: flag=0x00/0x00, offload=0/0, ips_offload=0/0, epid=0/0, ipid=0/0,
     vlan=0x0000/0x0000
vlifid=0/0, vtag_in=0x0000/0x0000 in_npu=0/0, out_npu=0/0, fwd_en=0/0, qid=0/0
no_ofld_reason: offload-denied helper
total session 1
```

3. Check the statuses of shared traffic shapers:

```
# diagnose firewall shaper traffic-shaper list
name 10Mbps
maximum-bandwidth 2500 KB/sec
guaranteed-bandwidth 1250 KB/sec
current-bandwidth 0 B/sec
priority 2
tos ff
packets dropped 0
```

```
bytes dropped 0

name 1Mbps
maximum-bandwidth 1250 KB/sec
guaranteed-bandwidth 125 KB/sec
current-bandwidth 0 B/sec
priority 4
tos ff
packets dropped 0
bytes dropped 0
```

Per-IP traffic shaper

With per-IP traffic shaping, you can limit each IP address's behavior to avoid a situation where one user uses all of the available bandwidth. In addition to controlling the maximum bandwidth used per IP address, you can also define the maximum number of concurrent sessions for an IP address. For example, if you apply a per-IP shaper of 1 Mbps to your entire network, FortiOS allocates each user/IP address 1 Mbps of bandwidth. Even if the network consists of a single user, FortiOS allocates them 1 Mbps. If there are ten users, each user gets 1 Mbps of bandwidth, totaling 10 Mbps of outgoing traffic.

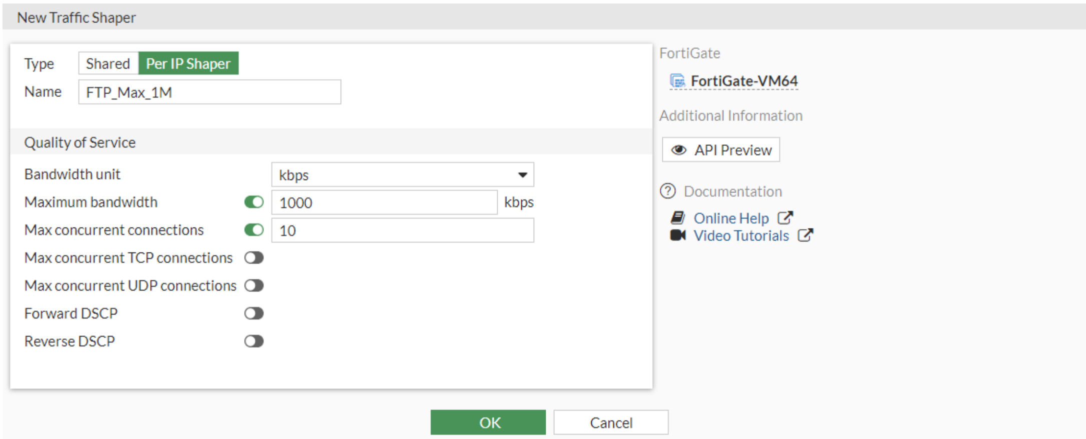
For shared shapers, all users share the set guaranteed and maximum bandwidths. For example, if you set a shared shaper for all PCs using an FTP service to 10 Mbps, all users uploading to the FTP server share the 10 Mbps.

Shared shapers affect upload speed. If you want to limit the download speed from the FTP server in the example, you must configure the shared shaper as a reverse shaper. Per-IP shapers apply the speed limit on both upload and download operations. Only traffic through forward traffic shapers will be included in FortiView; reverse and per-IP shapers are not included.

The following example shows how to apply a per-IP shaper to a traffic shaping policy. This shaper assigns each user a maximum bandwidth of 1 Mbps and allows each user to have a maximum of ten concurrent connections to the FTP server. In the example, FortiOS communicates with users using port10 and the FTP server using port9.

To configure a per-IP traffic shaper in the GUI:

1. Create a firewall policy:
   a. Go to *Policy & Objects* > *IPv4 Policy* and click *Create New*.
   b. Set the *Name* to *FTP Access*.
   c. Set the *Incoming Interface* to *port10*.
   d. Set the *Outgoing Interface* to *port9*.
   e. Set the *Source* to *all*.
   f. Set the *Destination* to *FTP_Server*.
   g. Set the *Schedule* to *always*.
   h. Set the *Service* to *ALL*.
   i. Click *OK*.
2. Create the per-IP traffic shaper:
   a. Go to *Policy & Objects* > *Traffic Shaping*, select the *Traffic Shapers* tab, and click *Create New*.
   b. Set *Type* to *Per IP Shaper*.
   c. Enter the *Name* (*FTP_Max_1M*). This shaper is for VoIP traffic.
   d. Enable *Max Bandwidth* and enter *1000*.
   e. Enable *Max Concurrent Connections* and enter *10*. This means that each user can have up to ten concurrent connections to the FTP server.



   f. Click *OK*.

3. Create a firewall shaping policy:
   a. Go to *Policy & Objects* > *Traffic Shaping*, select the *Traffic Shaping Policies* tab, and click *Create New*.
   b. Enter the *Name* (*FTP speed 1M*).
   c. Set the *Source* to the addresses and users that require access to the FTP server.
   d. Set the *Destination* to *FTP_Server*.
   e. Set the *Service* to *ALL*.
   f. Set the *Outgoing Interface* to *port9*.
   g. Enable *Per-IP shaper* and select *FTP_Max_1M*.
   h. Click *OK*.

To configure a per-IP traffic shaper in the CLI:

1. Create a firewall policy:
```
config firewall policy
    edit 1
        set name "FTP Access"
        set srcintf "port10"
        set dstintf "port9"
        set srcaddr "all"
        set dstaddr "FTP_Server"
        set action accept
        set schedule "always"
        set service "ALL"
        set fsso disable
        set nat enable
    next
end
```
2. Create the per-IP traffic shaper:
```
config firewall shaper per-ip-shaper
    edit "FTP_Max_1M"
        set max-bandwidth 1000
        set max-concurrent-session 10
    next
end
```
3. Create a firewall shaping policy:
```
config firewall shaping-policy
    edit 1
```

```
        set name "FTP speed 1M"
        set service "ALL"
        set dstintf "port9"
        set per-ip-shaper "FTP_Max_1M"
        set srcaddr "PC1" "WinPC" "PC2"
        set dstaddr "FTP_Server"
    next
end
```

To troubleshoot per-IP traffic shapers:

1. Check if specific traffic is attached to the correct traffic shaper. The example output shows the traffic attached to the FTP_Max_1M shaper:
```
# diagnose firewall iprope list 100015
policy index=3 uuid_idx=0 action=accept
flag (0):
shapers: per-ip=FTP_Max_1M
cos_fwd=0 cos_rev=0
group=00100015 av=00000000 au=00000000 split=00000000
host=2 chk_client_info=0x0 app_list=0 ips_view=0
misc=0 dd_type=0 dd_mode=0
zone(1): 0 -> zone(1): 38
source(3): 10.1.100.11-10.1.100.11, uuid_idx=30, 10.1.100.143-10.1.100.143, uuid_idx=32,
        10.1.100.22-10.1.100.22, uuid_idx=31,
dest(1): 172.16.200.55-172.16.200.55, uuid_idx=89,
service(1):
    [0:0x0:0/(0,65535)->(0,65535)] helper:auto
```
2. Check if the correct traffic shaper is applied to the session. The example output shows that the FTP_Max_1M shaper is applied to the session:
```
# diagnose sys session list
session info: proto=6 proto_state=01 duration=36 expire=3567 timeout=3600 flags=00000000
        sockflag=00000000 sockport=0 av_idx=0 use=4
origin-shaper=
reply-shaper=
per_ip_shaper=FTP_Max_1M
class_id=0 shaping_policy_id=3 ha_id=0 policy_dir=0 tunnel=/ helper=ftp vlan_cos=0/255
state=may_dirty per_ip npu npd mif route_preserve
statistic(bytes/packets/allow_err): org=506/9/1 reply=416/6/1 tuples=2
tx speed(Bps/kbps): 0/0 rx speed(Bps/kbps): 0/0
orgin->sink: org pre->post, reply pre->post dev=39->38/38->39 gwy=172.16.200.55/0.0.0.0
hook=post dir=org act=snat 10.1.100.11:58275->172.16.200.55:21(172.16.200.1:58275)
hook=pre dir=reply act=dnat 172.16.200.55:21->172.16.200.1:58275(10.1.100.11:58275)
pos/(before,after) 0/(0,0), 0/(0,0)
misc=0 policy_id=1 auth_info=0 chk_client_info=0 vd=2
serial=0000211a tos=ff/ff app_list=0 app=0 url_cat=0
sdwan_mbr_seq=0 sdwan_service_id=0
rpdb_link_id = 00000000
dd_type=0 dd_mode=0
npu_state=0x100000
npu info: flag=0x00/0x00, offload=0/0, ips_offload=0/0, epid=0/0, ipid=0/0,
        vlan=0x0000/0x0000
vlifid=0/0, vtag_in=0x0000/0x0000 in_npu=0/0, out_npu=0/0, fwd_en=0/0, qid=0/0
no_ofld_reason: offload-denied helper
```
3. Check the statuses of per-IP traffic shapers. The output should resemble the following:
```
# diagnose firewall shaper per-ip-shaper list
name FTP_Max_1M
```

```
maximum-bandwidth 125 KB/sec
maximum-concurrent-session 10
tos ff/ff
packets dropped 0
bytes dropped 0
addr=10.1.100.11 status: bps=0 ses=3
```

Changing traffic shaper bandwidth unit of measurement

Bandwidth speeds are measured in kilobits per second (Kbps), and bytes that are sent and received are measured in megabytes (MB). In some cases, this can cause confusion depending on whether your ISP uses kilobits per second (Kbps), kilobytes per second (KBps), megabits per second (Mbps), or gigabits per second (Gbps).

You can change the unit of measurement for traffic shapers in the CLI.

To change the bandwidth unit of measurement for a shared traffic shaper:

```
config firewall shaper traffic-shaper
    edit <traffic_shaper_name>
        set bandwidth-unit {kbps | mbps | gbps}
    next
end
```

To change the bandwidth unit of measurement for a per-IP traffic shaper:

```
config firewall shaper per-ip-shaper
    edit <traffic_shaper_name>
        set bandwidth-unit {kbps | mbps | gbps}
    next
end
```

Multi-stage DSCP marking and class ID in traffic shapers

Traffic shapers have a multi-stage method so that packets are marked with a different differentiated services code point (DSCP) and `class id` at different traffic speeds. Marking packets with a different DSCP code is for the next hop to classify the packets. The FortiGate benefits by marking packets with a different `class id`. Combined with the egress interface shaping profile, the FortiGate can handle the traffic differently according to its `class id`.

Rule	DSCP code	Class ID
speed < guarantee bandwidth	`diffservcode`	`class id` in shaping policy
guarantee bandwidth < speed < exceed bandwidth	`exceed-dscp`	`exceed-class-id`
exceed bandwidth < speed	`maximum-dscp`	`exceed-class-id`

This example sets the following parameters:

- When the current bandwidth is less than 50 Kbps, mark packets with `diffservcode` 100000 and set `class id` to 10.
- When the current bandwidth is between 50 Kbps and 100 Kbps, mark packets with `exceed-dscp` 111000 and set `exceed-class-id` to 20.

- When the current bandwidth is more than 100 Kbps, mark packets with `maximum-dscp` 111111 and set `exceed-class-id` to 20.

To set multi-stage DSCP marking and class ID in a traffic shaper:

```
config firewall shaper traffic-shaper
    edit "50k-100k-150k"
        set guaranteed-bandwidth 50
        set maximum-bandwidth 150
        set diffserv enable
        set dscp-marking-method multi-stage
        set exceed-bandwidth 100
        set exceed-dscp 111000
        set exceed-class-id 20
        set maximum-dscp 111111
        set diffservcode 100000
    next
end
```

```
config firewall shaping-policy
     edit 1
         set service "ALL"
         set dstintf PORT2
         set srcaddr "all"
         set dstaddr "all"
         set class-id 10
     next
end
```

Traffic shapers also have an `overhead` option that defines the per-packet size overhead used in rate computation.

To set the traffic shaper overhead option:

```
config firewall shaper traffic-shaper
    edit "testing"
        set guaranteed-bandwidth 50
        set maximum-bandwidth 150
        set overhead 14 <range from 0 to 100>
    next
end
```

Example

This example shows how to mark QA traffic with a different DSCP according to real-time traffic speed.

To configure the firewall address:

```
config firewall address
    edit QA_team
        set subnet 10.1.100.0/24
    next
end
```

To configure the firewall shaper traffic shaper:

```
config firewall shaper traffic-shaper
    edit "500k-1000k-1500k"
        set guaranteed-bandwidth 500
        set maximum-bandwidth 1500
        set diffserv enable
        set dscp-marking-method multi-stage
        set exceed-bandwidth 1000
        set exceed-dscp 111000
        set maximum-dscp 111111
        set diffservcode 100000
    next
end
```

```
config firewall shaping-policy
    edit QA_team
        set service "ALL"
        set dstintf port1
        set traffic-shaper "500k-1000k-1500k"
        set traffic-shaper-reverse "500k-1000k-1500k"
        set srcaddr "QA_team"
        set dstaddr "all"
    next
end
```

Adding traffic shapers to multicast policies

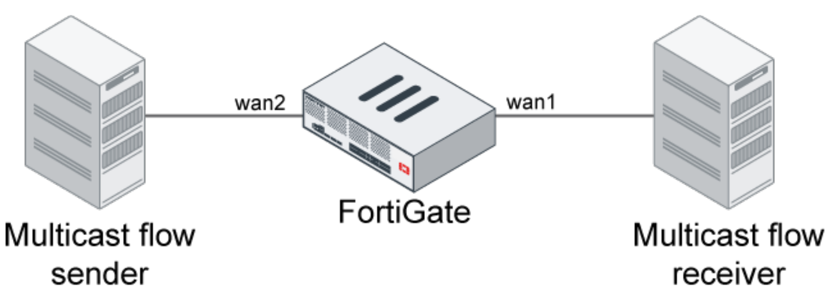
When multicast routing is enabled, a traffic shaper can be added to a multicast policy.

Only a shared traffic shaper with the `per-policy` option disabled can be used. This is the default state of the `per-policy` option. The `auto-asic-offload` option must also be disabled on the multicast policy.

This feature is currently not supported on IPv6 multicast policies or on transparent mode VDOMs.

Example

In this example, a traffic shaper is applied to the multicast policy. A multicast flow sender sends the multicast data stream. The shaper attached to the multicast session is checked, and the shaping of the data stream is confirmed in the multicast session.



To apply traffic shaping to a multicast policy:

1. Enable multicast routing on the VDOM:

```
config router multicast
    set multicast-routing enable
    config pim-sm-global
        config rp-address
            edit 1
                set ip-address 10.1.100.10
            next
        end
    end
    config interface
        edit "wan2"
            set pim-mode sparse-mode
        next
        edit "wan1"
            set pim-mode sparse-mode
        next
    end
end
```

2. Create a traffic shaper:

```
config firewall shaper traffic-shaper
    edit "shaper128kbps-high"
        set guaranteed-bandwidth 128
        set maximum-bandwidth 128
        set per-policy disable
        set diffserv enable
        set diffservcode 010101
    next
end
```

3. Apply the traffic shaper to the multicast policy and disable NPU offloading:

```
config firewall multicast-policy
    edit 1
        set name "test_multicast-policy"
        set logtraffic enable
        set srcintf "wan2"
        set dstintf "wan1"
        set srcaddr "all"
        set dstaddr "all"
        set snat enable
        set auto-asic-offload disable
        set traffic-shaper "shaper128kbps-high"
    next
end
```

4. Check the shaper and DSCP in the multicast session:

```
# diagnose sys mcast-session list
    session info: id=26 vf=0 proto=17 10.1.100.41.35537->230.0.0.1.7878
    used=2 path=1 duration=118 expire=179 indev=18 pkts=119 bytes=64260
    state=00000000:
    session-npu-info: ipid/vlifid=0/0 vlanid/vtag_in=0/0 in_npuid=0 tae_index=0 qid=0
```

```
    fwd_map=0x00000000
        path: log snat npu-deny nsaddr=172.16.200.10 policy=1, outdev=17, tos=0x15
                origin-shaper=shaper128kbps-high prio=2 tos=0x15 guarantee 16000Bps max
    16000Bps traffic 620Bps drops 0pkt/0B
        Total 1 sessions
```

Global traffic prioritization

Global traffic prioritization allows your traffic to be prioritized as high (2), medium (3), or low (4) based on ToS (type of service) or DSCP. When using ToS-based priority, integers 0 to 15 can be used, which correspond to the definitions of the ToS field values in RFC 1349. When using DSCP, values 0 to 63 can be used, which correspond to the six bits in the DSCP value.

The outbandwidth must be defined in order for global prioritization to take effect. When the outbandwidth is defined on an interface without an applied `egress-shaping-profile`, the interface has a total of five priority levels:

Priority level	Description
0	Top
1	Critical
2	High
3	Medium
4	Low

Priority level 0 is reserved for administrative and local out traffic. Priority level 1 is used for traffic that is below guaranteed bandwidth when using a traffic shaper.

Traffic shaper and traffic shaping profile configurations take precedence over global traffic prioritization.

CLI commands

The following commands are used to configure the prioritization either by ToS or DSCP.

To configure the traffic prioritization type and level:

```
config system global
    set traffic-priority {tos | dscp}
    set traffic-priority-level {high | medium | low}
end
```

To configure the ToS-based priority table:

```
config system tos-based-priority
    edit <id>
        set tos <0-15>
        set priority (high | medium | low)
```

```
    next
end
```

To configure the DSCP-based priority table:

```
config system dscp-based-priority
    edit <id>
        set ds <0-63>
        set priority (high | medium | low)
    next
end
```

To configure the interface outbandwidth:

```
config system interface
    edit <name>
        set outbandwidth <bandwidth in kbps>
    next
end
```

Example

In the following configuration, packets with DSCP markings of 1 are prioritized as high, and packets with DSCP markings of 2 are prioritized as medium. All the other traffic is prioritized as low. The outbandwidth on interface port3 is set to 1000 kbps.

To configure DSCP-based traffic prioritization:

1. Configure DSCP-based prioritization in the global settings:

```
config system global
    set traffic-priority dscp
    set traffic-priority-level low
end
```

2. Configure the DSCP-based priority table:

```
config system dscp-based-priority
    edit 1
        set ds 1
        set priority high
    next
    edit 2
        set ds 2
        set priority medium
    next
end
```

3. Configure the outbandwidth on port3:

```
config system interface
    edit "port3"
        set outbandwidth 1000
    next
end
```

Verifying the traffic prioritization

When traffic exceeds the outbandwidth of 1000 kbps, traffic prioritization will take effect. Since the form of traffic shaping applied here is policing, excess packets above the outbandwidth are dropped.

In scenario 1, approximately 300 kbps of high priority traffic and 300 kbps of medium priority traffic passes through the FortiGate on port3.

To debug the bandwidth allocation:

```
# diagnose netlink interface list port3
if=port3 family=00 type=1 index=5 mtu=1500 link=0 master=0
ref=35 state=start present fw_flags=3800 flags=up broadcast run allmulti multicast
Qdisc=pfifo_fast hw_addr=52:54:00:fb:81:0c broadcast_addr=ff:ff:ff:ff:ff:ff
outbandwidth=1000(kbps)
        priority=0      allocated-bandwidth=0(kbps)     total_bytes=9311K       drop_
bytes=197K
        priority=1      allocated-bandwidth=0(kbps)     total_bytes=0   drop_bytes=0
        priority=2      allocated-bandwidth=354(kbps)   total_bytes=20407K      drop_
bytes=48K
        priority=3      allocated-bandwidth=354(kbps)   total_bytes=7093K       drop_
bytes=1262K
        priority=4      allocated-bandwidth=290(kbps)   total_bytes=266018K     drop_
bytes=7743K
stat: rxp=15450901 txp=25933756 rxb=5456860515 txb=17257309292 rxe=0 txe=0 rxd=0 txd=0 mc=0
collision=0 @ time=1629439926
re: rxl=0 rxo=0 rxc=0 rxf=0 rxfi=0 rxm=0
te: txa=0 txc=0 txfi=0 txh=0 txw=0
misc rxc=0 txc=0
input_type=0 state=3 arp_entry=0 refcnt=35
```

High priority (2) traffic is allocated 354 kbps of bandwidth. Medium priority (3) traffic is also allocated 354 kbps of bandwidth. The remaining bandwidth is allocated to low priority (4) traffic.

In scenario 2, approximately 400 kbps of high priority traffic and 800 kbps of medium priority traffic passes through the FortiGate on port3.

To debug the bandwidth allocation:

```
# diagnose netlink interface list port3
if=port3 family=00 type=1 index=5 mtu=1500 link=0 master=0
ref=36 state=start present fw_flags=3800 flags=up broadcast run allmulti multicast
Qdisc=pfifo_fast hw_addr=52:54:00:fb:81:0c broadcast_addr=ff:ff:ff:ff:ff:ff
outbandwidth=1000(kbps)
        priority=0      allocated-bandwidth=7(kbps)     total_bytes=9981K       drop_
bytes=240K
        priority=1      allocated-bandwidth=0(kbps)     total_bytes=0   drop_bytes=0
        priority=2      allocated-bandwidth=425(kbps)   total_bytes=31478K      drop_
bytes=101K
        priority=3      allocated-bandwidth=567(kbps)   total_bytes=12056K      drop_
bytes=1984K
        priority=4      allocated-bandwidth=0(kbps)     total_bytes=266795K     drop_
bytes=7771K
stat: rxp=15461740 txp=25950805 rxb=5459688950 txb=17273940560 rxe=0 txe=0 rxd=0 txd=0 mc=0
collision=0 @ time=1629440553
re: rxl=0 rxo=0 rxc=0 rxf=0 rxfi=0 rxm=0
```

```
te: txa=0 txc=0 txfi=0 txh=0 txw=0
misc rxc=0 txc=0
input_type=0 state=3 arp_entry=0 refcnt=36
```

High priority (2) traffic is allocated 425 kbps of bandwidth. Medium priority (3) traffic is allocated 567 kbps of bandwidth. Since the total bandwidth required exceeds 1000 kbps, the remaining medium priority (3) traffic is dropped. In comparing the successive debug outputs, the `drop_bytes` counter for medium priority (3) traffic gets bigger.

DSCP matching and DSCP marking

This feature has three parts:

- DSCP matching in firewall policies
- DSCP matching in firewall shaping policies
- DSCP marking in firewall shaping policies

DSCP matching in firewall policies

Traffic is allowed or blocked according to the Differentiated Services Code Point (DSCP) values in the incoming packets.

The following CLI variables are available in the `config firewall policy` command:

`tos-mask <mask_value>`	Non-zero bit positions are used for comparison. Zero bit positions are ignored (default = 0x00). This variable replaces the `dscp-match` variable.
`tos <tos_value>`	Type of Service (ToC) value that is used for comparison (default = 0x00). This variable is only available when `tos-mask` is not zero. This variable replaces the `dscp-value` variable.
`tos-negate {enable | disable}`	Enable/disable negated ToS match (default = disable). This variable is only available when `tos-mask` is not zero. This variable replaces the `dscp-negate` variable.

DSCP matching in firewall shaping policies

Shaping is applied to the session or not according to the DSCP values in the incoming packets. The same logic and commands as in firewall policies are used.

DSCP marking in firewall shaping policies

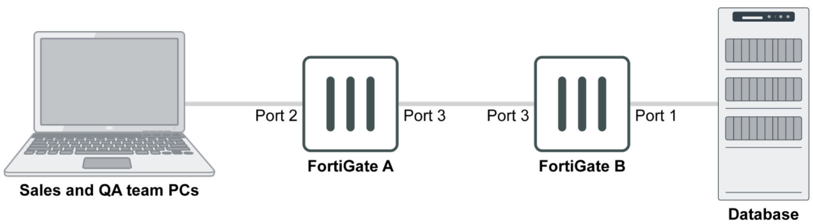
Traffic is allowed or blocked according to the DSCP values in the incoming packets. DSCP marking in firewall shaping policies uses the same logic and commands as in firewall policy and traffic-shaper.

When DSCP marking on `firewall shaper traffic-shaper`, `firewall shaping-policy`, and `firewall policy` all apply to the same session, `shaping-policy` overrides `policy`, and `shaper traffic-shaper` overrides both `shaping-policy` and `policy`.

The following CLI variables in `config firewall policy` are used to mark the packets:

`diffserv-forward {enable | disable}`	Enable/disable changing a packet's DiffServ values to the value specified in `diffservcode-forward` (default = disable).
`diffservcode-forward <dscp_value>`	The value that packet's DiffServ is set to (default = 000000). This variable is only available when `diffserv-forward` is enabled.
`diffserv-reverse {enable | disable}`	Enable/disable changing a packet's reverse (reply) DiffServ values to the value specified in `diffservcode-rev` (default = disable).
`diffservcode-rev <dscp_value>`	The value that packet's reverse (reply) DiffServ is set to (default = 000000). This variable is only available when `diffserv-rev` is enabled.

Examples



Example 1

FortiGate A marks traffic from the sales and QA teams with different DSCP values. FortiGate B does DSCP matching, allowing only the sales team to access the database.

1. Configure FortiGate A:

```
config firewall policy
    edit 1
        set srcintf "port2"
        set dstintf "port3"
        set srcaddr "QA"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set diffserv-forward enable
        set diffservcode-forward 110000
        set nat enable
    next
    edit 5
        set srcintf "port2"
        set dstintf "port3"
        set srcaddr "Sales"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set diffserv-forward enable
        set diffservcode-forward 111011
        set nat enable
```

```
    next
end
```

2. Configure FortiGate B:

```
config firewall policy
    edit 2
        set srcintf "port3"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "Database"
        set action accept
        set schedule "always"
        set service "ALL"
        set tos-mask 0xf0
        set tos 0xe0
        set fsso disable
        set nat enable
    next
end
```

Example 2

FortiGate A marks traffic from the sales and QA teams with different DSCP values. FortiGate B uses a firewall shaping policy to do the DSCP matching, limiting the connection speed of the sales team to the database to 10MB/s.

1. Configure FortiGate A:

```
config firewall policy
    edit 1
        set srcintf "port2"
        set dstintf "port3"
        set srcaddr "QA"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set diffserv-forward enable
        set diffservcode-forward 110000
        set nat enable
    next
    edit 5
        set srcintf "port2"
        set dstintf "port3"
        set srcaddr "Sales"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set diffserv-forward enable
        set diffservcode-forward 111011
        set nat enable
    next
end
```

2. Configure FortiGate B:

```
config firewall policy
    edit 2
        set srcintf "port3"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set nat enable
    next
end
config firewall shaper traffic-shaper
    edit "10MB/s"
        set guaranteed-bandwidth 60000
        set maximum-bandwidth 80000
    next
end
config firewall shaping-policy
    edit 1
        set service "ALL"
        set dstintf "port1"
        set tos-mask 0xf0
        set tos 0xe0
        set traffic-shaper "10MB/s"
        set srcaddr "all"
        set dstaddr "all"
    next
end
```

Example 3

FortiGate A has a traffic shaping policy to mark traffic from the QA team with a DSCP value of 100000, while reverse traffic is marked with 000011.

1. Configure FortiGate A:

```
config firewall shaping-policy
    edit 1
        set name "QA Team 50MB"
        set service "ALL"
        set dstintf "port3"
        set traffic-shaper "50MB/s"
        set traffic-shaper-reverse "50MB/s"
        set diffserv-forward enable
        set diffserv-reverse enable
        set srcaddr "QA"
        set dstaddr "all"
        set diffservcode-forward 100000
        set diffservcode-rev 000011
    next
end
```

Examples

This section includes the following traffic shaping configuration examples:

- Interface-based traffic shaping profile on page 953
- Interface-based traffic shaping with NP acceleration on page 961
- QoS assignment and rate limiting for FortiSwitch quarantined VLANs on page 962
- Ingress traffic shaping profile on page 963

Interface-based traffic shaping profile

A traffic shaping policy can be used for interface-based traffic shaping by organizing traffic into 30 class IDs. The shaping profile defines the percentage of the interface bandwidth that is allocated to each class. Each traffic class ID is shaped to the assigned speed according to the outgoing bandwidth limit configured to the interface.

Traffic classification

A shaping policy classifies traffic and organizes it into different class IDs, based on matching criteria. For traffic matching a criteria, you can choose to put it into 30 different shaping classes, identified by class ID 2 to 31.

You must select an outgoing interface for the traffic. The shaping policy is only applied when the traffic goes to one of the selected outgoing interfaces.

Criterion	Description
Source	• Address: match the source address of the traffic to the selected address or address group. • User: use the user credentials of the traffic to match the selected user or user group. At least one address, address group, or internet service must also be selected. • Internet service: match the traffic to the selected internet service. Internet services cannot be used if addresses or address or groups are used.
Destination	• Address: match the destination address of the traffic to the selected address or address group. • Internet service: match the traffic to the selected internet service. Internet services cannot be used if addresses or address or groups are used.
Schedule	Match the current date and time to the selected schedule. You can select a one-time schedule, recurring schedule, or schedule group. This setting is optional.
Service	Match the service of the traffic to the selected service or service group.
Application	Match the application of the traffic to the selected application, application category, or application group. Application control must be enabled in the related firewall policy to know the application of the traffic. See Application control on page 1190 for more information.
URL category	Match the URL of the traffic to the selected URL category. Web filter must be enabled in the related firewall policy to know the URL of the traffic. See Web filter on page 1122 for more information.

When multiple items are selected in one criterion, it is considered a match when traffic matches any one of them.

Traffic prioritization

Shaping profiles define how different shaping classes of traffic are prioritized. For each class, you can define three prioritization strategies: guaranteed bandwidth, maximum bandwidth, and priority.

For each shaping profile, a default shaping class must be defined. Traffic is prioritized based on the default shaping group in the following two circumstances:

- All traffic to the outgoing interface that does not match to any shaping policy
- Traffic with a shaping group that is not defined in a shaping profile

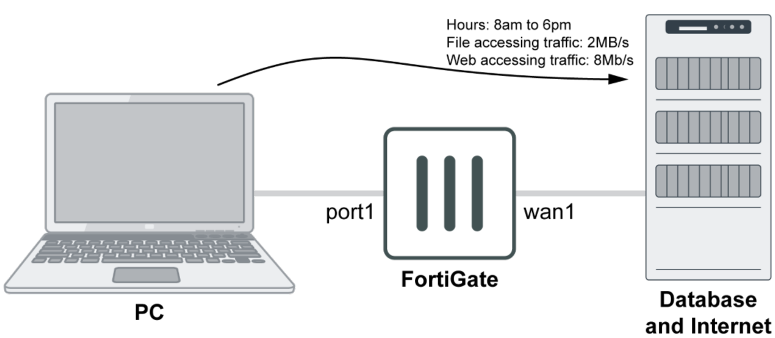
Prioritization strategy	Description
Guaranteed bandwidth	The percentage of the link speed that is reserved for the shaping group. The total guaranteed bandwidth for all shaping groups cannot exceed 100%.
Maximum bandwidth	The maximum percentage of the link speed that the shaping group can use.
Priority	The shaping class priority: top, critical, high, medium, or low. When groups are competing for bandwidth on the interface, the group with the higher priority wins.

Applying a shaping profile to an interface

Traffic shaping is accomplished by configuring the outgoing bandwidth and outgoing shaping profile on an interface. The shaping profile uses the outgoing bandwidth of the interface as the maximum link speed, and it only works when the outgoing bandwidth is configured.

This example shows how to apply interface-based traffic shaping to web and file accessing traffic according to a schedule:

- The link speed of the wan1 interface is 10 Mb/s.
- File access can use up to 2 Mb/s from 8:00 AM to 6:00 PM.
- Web access can use 8 Mb/s from 8:00 AM to 6:00 PM.



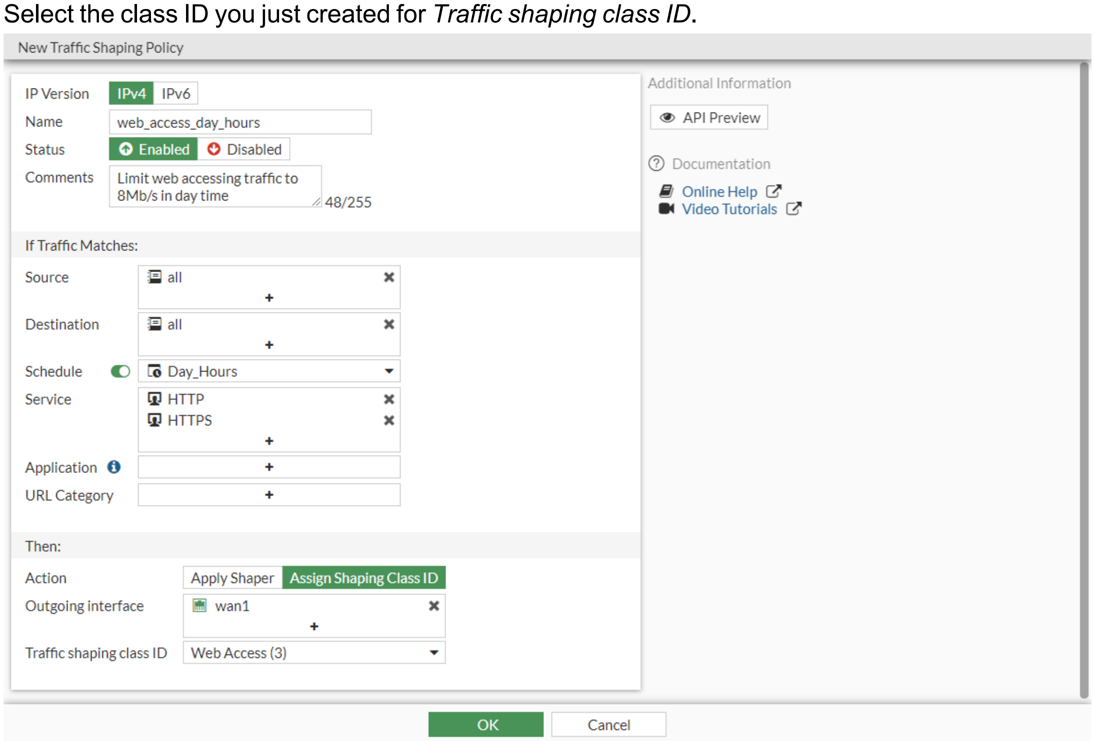
Putting the traffic into shaping classes

To create a recurring schedule in the GUI:

1. Go to *Policy & Objects > Schedules*.
2. Click *Create New > Schedule*.
3. Configure a recurring schedule called *Day_Hours* for everyday from 8:00 AM to 6:00 PM.
4. Click *OK*.

To create a traffic shaping policy and class ID for the web accessing traffic in the GUI:

1. Go to *Policy & Objects > Traffic Shaping*, select the *Traffic Shaping Policies* tab, and click *Create New*.
2. Enter a name for the policy, such as *web_access_day_hours*.
3. Enable *Schedule* and select the schedule you just created.
4. Set *Service* to web accessing services, such as *HTTP* and *HTTPS*.
5. Set *Action* to *Assign Shaping Class ID*, and *Outgoing interface* to *wan1*.
6. Click the *Traffic shaping class ID* drop down then click *Create*.
7. Enter an integer value for the *ID* (*3*) and a description for the *Name*, such as *Web Access*.
8. Click *OK*.
9. Select the class ID you just created for *Traffic shaping class ID*.

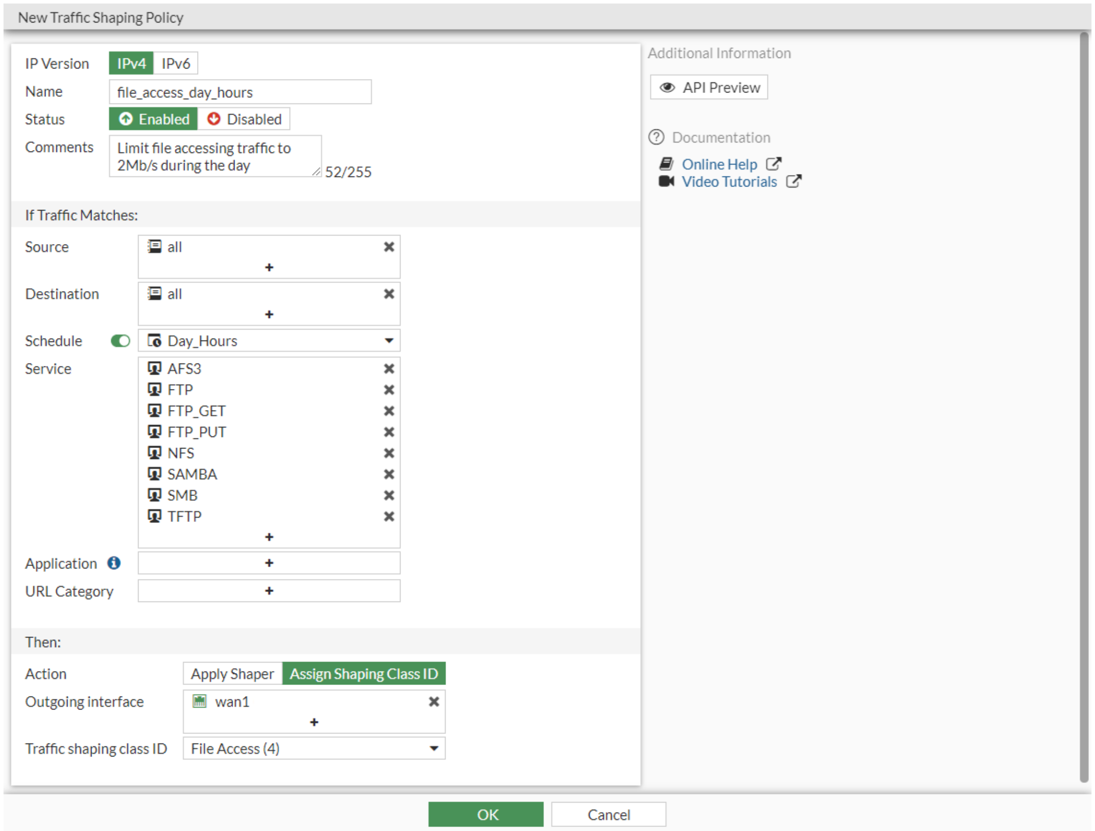


10. Configure the remaining settings as required.
11. Click *OK*.

To create a traffic shaping policy and class ID for the file accessing traffic in the GUI:

1. Go to *Policy & Objects > Traffic Shaping*, select the *Traffic Shaping Policies* tab, and click *Create New*.
2. Enter a name for the policy, such as *file_access_day_hours*.

3. Enable *Schedule* and select the schedule you just created.
4. Set *Service* to file accessing services, such as *ASF3*, *FTP* and *SMB*.
5. Set *Action* to *Assign Shaping Class ID*, and *Outgoing interface* to *wan1*.
6. Click the *Traffic shaping class ID* drop down then click *Create*.
7. Enter an integer value for the *ID* (*4*) and a description for the *Name*, such as *File Access*.
8. Click *OK*.
9. Select the class ID you just created for *Traffic shaping class ID*.



10. Configure the remaining settings as required.
11. Click *OK*.

To put the traffic into shaping classes in the CLI:

1. Create a recurring schedule:

```
config firewall schedule recurring
    edit "Day_Hours"
        set start 08:00
        set end 18:00
        set day sunday monday tuesday wednesday thursday friday saturday
    next
end
```

2. Create the traffic class IDs:

```
config firewall traffic-class
    edit 3
        set class-name "Web Access"
```

```
    next
    edit 4
        set class-name "File Access"
    next
end
```

3. Create the web and file accessing traffic shaping policies:

```
config firewall shaping-policy
    edit 2
        set name "web_access_day_hours"
        set comment "Limit web accessing traffic to 8Mb/s in day time"
        set service "HTTP" "HTTPS"
        set schedule "Day_Hours"
        set dstintf "wan1"
        set class-id 3
        set srcaddr "all"
        set dstaddr "all"
    next
    edit 3
        set name "file_access_day_hours"
        set comment "Limit file accessing traffic to 2Mb/s during the day"
        set service "AFS3" "FTP" "FTP_GET" "FTP_PUT" "NFS" "SAMBA" "SMB" "TFTP"
        set schedule "Day_Hours"
        set dstintf "wan1"
        set class-id 4
        set srcaddr "all"
        set dstaddr "all"
    next
end
```

Allocating bandwidth to the shaping classes

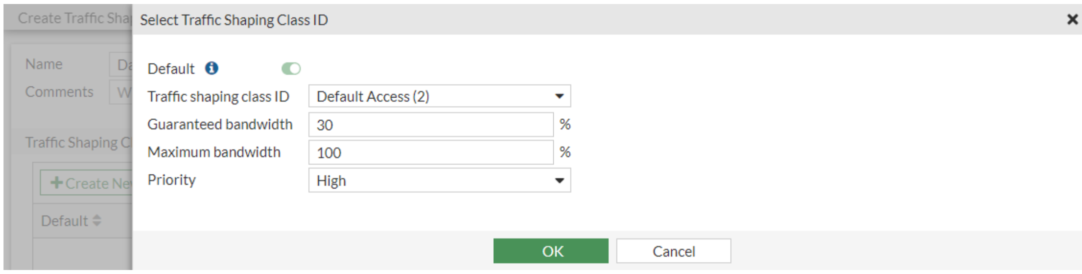
A traffic shaping profile defines the guaranteed and maximum bandwidths each class receives. In this example, file access can use up to 2 Mb/s and web access can use 8 Mb/s from 8:00 AM to 6:00 PM.

To create a traffic shaping profile using the GUI:

1. Go to *Policy & Objects* > *Traffic Shaping*, select the *Traffic Shaping Profiles* tab, and click *Create New*.
2. Enter a name for the profile, such as *Day_Hours_Profile*.
3. Configure a default traffic shaping class:
   This class has a high priority, meaning that when the other classes have reached their guaranteed bandwidths, this default class will use the rest of the available bandwidth.
   a. In the *Traffic Shaping Classes* table click *Create New*.
   b. Click the *Traffic shaping class ID* drop down then click *Create*.
   c. Enter a name for the class, such as *Default Access*.
   d. Click *OK*.
   e. Select the class ID you just created for *Traffic shaping class ID*.

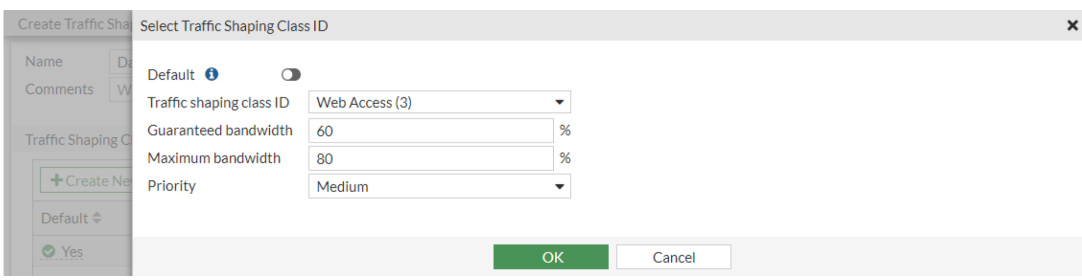
f. Configure the following settings, then click *OK*:

Guaranteed bandwidth	30
Maximum bandwidth	100
Priority	High



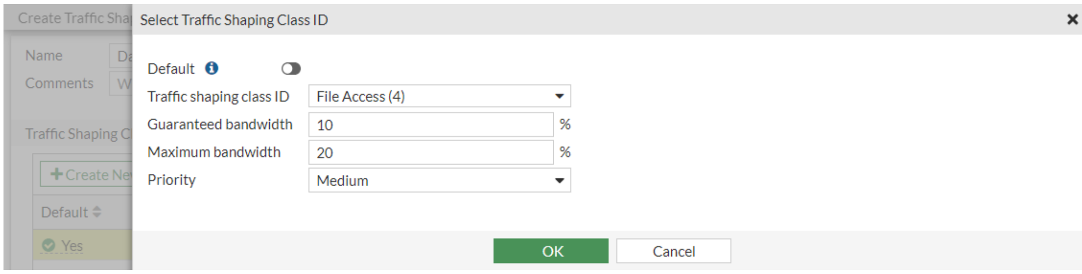
4. Configure a web accessing traffic shaping class:
When other types of traffic are competing for bandwidth, this class is guaranteed to 6 Mb/s, or 60% of the bandwidth.
   a. In the *Traffic Shaping Classes* table click *Create New*.
   b. Configure the following settings, then click *OK*:

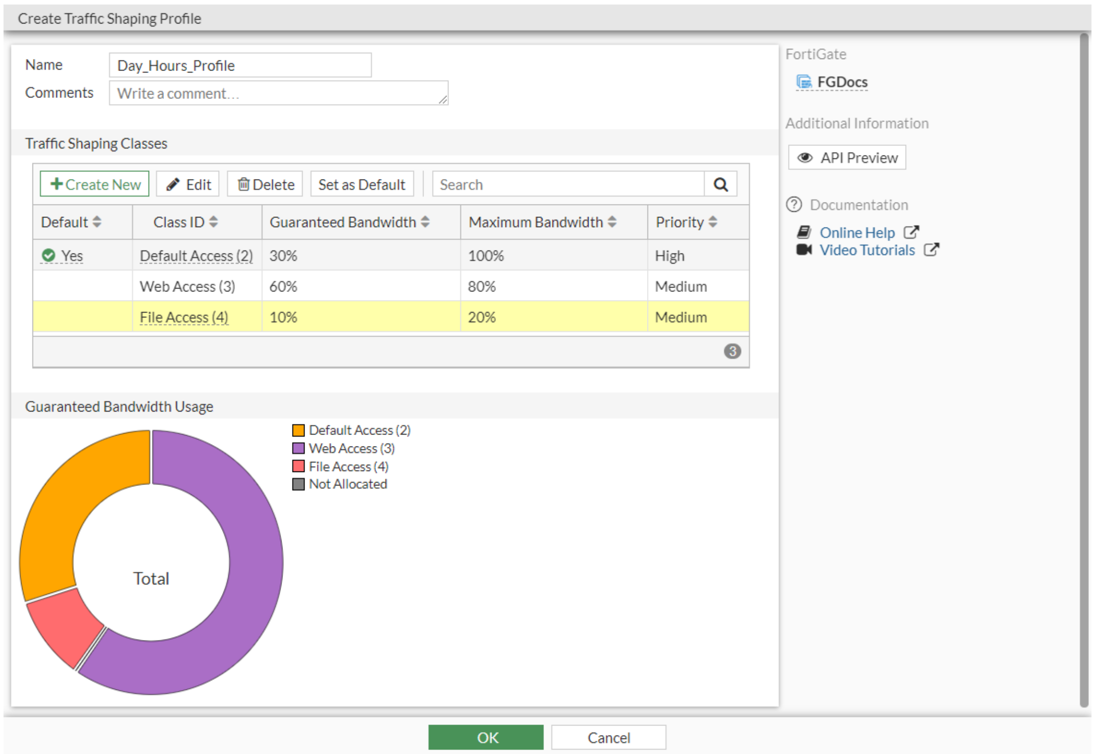
Traffic shaping class ID	Web Access
Guaranteed bandwidth	60
Maximum bandwidth	80
Priority	Medium



5. Configure a file accessing traffic shaping class:
When other types of traffic are competing for bandwidth, this group is guaranteed to 1 Mb/s, or 10% of the bandwidth.
   a. In the *Traffic Shaping Classes* table click *Create New*.
   b. Configure the following settings, then click *OK*:

Traffic shaping class ID	File Access
Guaranteed bandwidth	10
Maximum bandwidth	20
Priority	Medium





6. Click *OK*.

To create a traffic shaping profile using the CLI:

```
config firewall shaping-profile
    edit "Day_Hours_Profile"
        set default-class-id 2
        config shaping-entries
            edit 1
                set class-id 2
                set guaranteed-bandwidth-percentage 30
                set maximum-bandwidth-percentage 100
            next
            edit 2
                set class-id 3
                set priority medium
                set guaranteed-bandwidth-percentage 60
                set maximum-bandwidth-percentage 80
            next
            edit 3
                set class-id 4
```

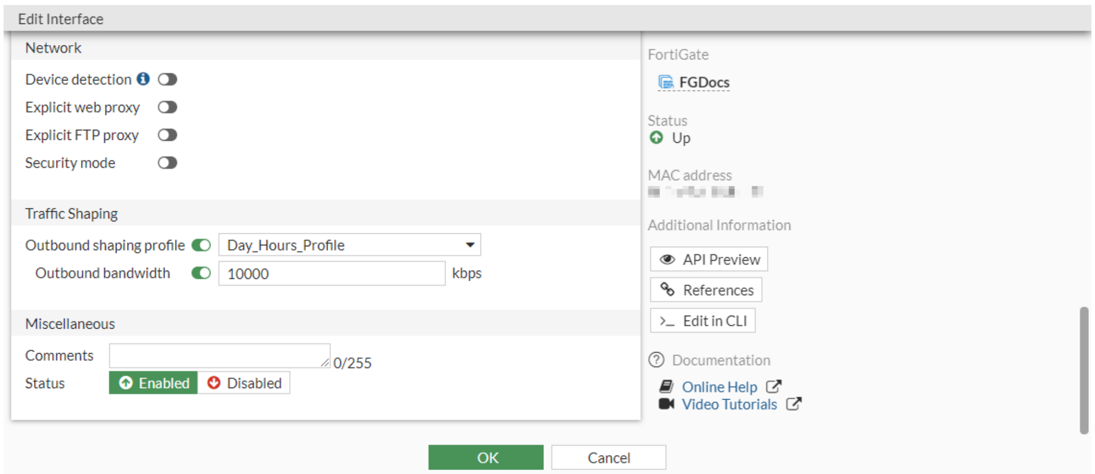
```
                set priority medium
                set guaranteed-bandwidth-percentage 10
                set maximum-bandwidth-percentage 20
            next
        end
    next
end
```

Defining the available bandwidth on an interface

In this example, the link speed of the wan1 interface is 10 Mb/s.

To set the bandwidth of the wan1 interface in the GUI:

1. Go to *Network > Interfaces*.
2. Edit the wan1 interface.
3. Under Traffic Shaping, enable *Outbound shaping profile* and select the profile that you just created, *Day_Hours_Profile*.
4. Enable *Outbound Bandwidth* and set it to *10000* Kbps.



5. Click *OK*.

To set the bandwidth of the wan1 interface in the CLI:

```
config system interface
    edit "wan1"
        set egress-shaping-profile "Day_Hours_Profile"
        set outbandwidth 10000
    next
end
```

Diagnose commands

To check that the specific traffic is put into the correct shaping group or class ID:

```
# diagnose firewall iprope list 100015
```

To check the speed limit for each class ID on an interface:

```
# diagnose netlink interface list wan1
```

Interface-based traffic shaping with NP acceleration

Interface-based traffic shaping with NP acceleration is supported on some devices.

An administrator configures the WAN interface's maximum outbound bandwidth and, based on that, creates a traffic shaping profile with a percentage based shaper. This allows for proper QoS and traffic shaping. VLAN interfaces are not supported.

This feature is supported on FortiGate 600E, 500E, and 300E models.

To configure interface-based traffic shaping:

1. Enable NPU offloading when doing interface-based traffic shaping according to the egress-shaping-profile:

```
config system npu
    set intf-shaping-offload enable
end
```

2. Configure shaping profiles:

```
config firewall shaping-profile
    edit "sdwan"
        set default-class-id 4
        config shaping-entries
            edit 1
                set class-id 4
                set guaranteed-bandwidth-percentage 3
                set maximum-bandwidth-percentage 5
            next
            edit 2
                set class-id 3
                set priority medium
                set guaranteed-bandwidth-percentage 50
                set maximum-bandwidth-percentage 100
            next
            edit 3
                set class-id 2
                set priority low
                set guaranteed-bandwidth-percentage 1
                set maximum-bandwidth-percentage 5
            next
        end
    next
end
```

The class number is limited to 16.

3. Configure a traffic shaper and shaping policy:

```
config firewall shaper traffic-shaper
    edit "Transactional"
        set priority medium
    next
end
```

```
config firewall shaping-policy
    edit 1
        set service "ALL"
        set dstintf "any"
        set traffic-shaper "Transactional"
        set class-id 3
        set srcaddr "all"
        set dstaddr "all"
    next
end
```

4. Apply the egress shaping profile on the interface:

```
config system interface
    edit "port2"
        set vdom "root"
        set ip 10.1.100.23 255.255.255.0
        set allowaccess ping
        set type physical
        set outbandwidth 500
        set egress-shaping-profile "sdwan"
        set snmp-index 4
    next
end
```

5. Configure a firewall policy:

```
config firewall policy
    edit 3
        set srcintf "port2"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set logtraffic all
        set nat enable
    next
end
```

QoS assignment and rate limiting for FortiSwitch quarantined VLANs

When devices are quarantined, they are isolated from the rest of the network. However, they can still impact the network if not controlled beyond isolation. A quarantined host, which offers heavy traffic, could congest the network and create a DOS-style reduction in service to authorized hosts.

Within the quarantined VLAN, two restrictions are available within the network:

- Traffic policing (also known as rate limiting)
- QoS (Quality of Service) assignment (also known as priority assignment)

Each quarantined host's traffic can be subject to rate limiting and priority adjustment. This reduces the impact that any quarantined host can have on authorized traffic on the network.

To configure QoS assignment and rate limiting for quarantined VLANs:

1. Configure a traffic policy, or use the default "quarantine" policy:

```
config switch-controller traffic-policy
    edit "quarantine"
        set description "Rate control for quarantined traffic"
        set guaranteed-bandwidth 163840
        set guaranteed-burst 8192
        set maximum-burst 163840
        set cos-queue 0
    next
end
```

2. Configure an interface:

```
config system interface
    edit "qtn.aggr1"
        set vdom "root"
        set ip 10.254.254.254 255.255.255.0
        set description "Quarantine VLAN"
        set security-mode captive-portal
        set replacemsg-override-group "auth-intf-qtn.aggr1"
        set device-identification enable
        set snmp-index 30
        set switch-controller-access-vlan enable
        set switch-controller-traffic-policy "quarantine"
        set color 6
        set interface "aggr1"
        set vlanid 4093
    next
end
```

By default, `switch-controller-traffic-policy` is empty. You need to apply the necessary traffic policy (not only limited to "quarantine").

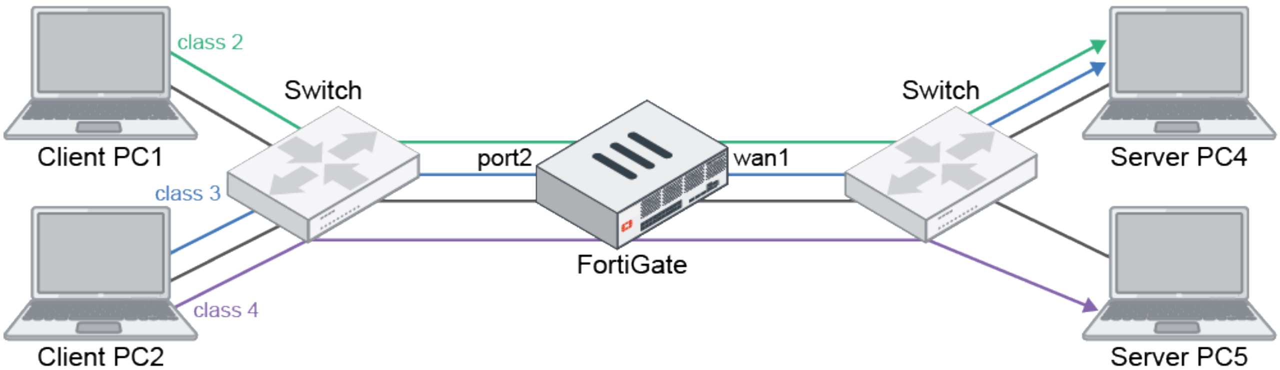
Ingress traffic shaping profile

A traffic shaping profile can be applied to an interface for traffic in the ingress direction. Similar to an egress traffic shaping profile, the guaranteed bandwidth and priority of the profile will be respected when an interface receives inbound traffic. When congestion occurs, any remaining bandwidth will be allotted to classes based on priority.

Ingress traffic shaping does not support NPU offloading.

Example

In this example, the port2 interface has a total inbound bandwidth of 100 Mbps. Traffic from certain clients to certain servers are assigned different classes.



IPv6 traffic from any client PCs to server PCs is assigned class 5.

For each class, the priority, guaranteed bandwidth, and maximum bandwidth are as follows:

Class	Priority	Guaranteed bandwidth	Maximum bandwidth
2	Low	10%	60%
3	High	20%	100%
4	High	30%	100%
5	Medium	10%	50%

Bandwidth will first be allotted to each class according to its guaranteed bandwidth. Then remaining available bandwidth will be allotted to class 3 and 4 first based on their priority. The allocation will be proportional to their guaranteed bandwidth ratio.

To configure ingress traffic shaping:

1. Configure the client and server addresses:

```
config firewall address
    edit "pc1"
        set subnet 10.1.100.11 255.255.255.255
    next
    edit "pc2"
        set subnet 10.1.100.22 255.255.255.255
    next
    edit "pc4"
        set subnet 172.16.200.44 255.255.255.255
    next
    edit "pc5"
        set subnet 172.16.200.55 255.255.255.255
    next
end
```

2. Configure the class IDs:

```
config firewall traffic-class
    edit 2
        set class-name "class2"
    next
    edit 3
        set class-name "class3"
    next
    edit 4
        set class-name "class4"
    next
    edit 4
        set class-name "class5"
    next
end
```

3. Configure traffic shaping policies to assign classes to each group of traffic.
   a. Configure a policy to assign traffic from PC1 to PC4 in class 2:

```
config firewall shaping-policy
    edit 1
        set name "shaping policy 1"
        set service "ALL"
        set dstintf "wan1"
        set class-id 2
        set srcaddr "pc1"
        set dstaddr "pc4"
    next
end
```

   b. Configure a policy to assign traffic from PC2 to PC4 in class 3:

```
config firewall shaping-policy
    edit 2
        set name "shaping policy 2"
        set service "ALL"
        set dstintf "wan1"
        set class-id 3
        set srcaddr "pc2"
        set dstaddr "pc4"
    next
end
```

   c. Configure a policy to assign traffic from PC2 to PC5 in class 4:

```
config firewall shaping-policy
    edit 3
        set name "shaping policy 3"
        set service "ALL"
        set dstintf "wan1"
        set class-id 4
        set srcaddr "pc2"
        set dstaddr "pc5"
    next
end
```

d. Configure a policy to assign all IPv6 traffic to class 5:

```
config firewall shaping-policy
    edit 4
        set name "shaping policy 4"
        set ip-version 6
        set service "ALL"
        set dstintf "wan1"
        set class-id 5
        set srcaddr6 "all"
        set dstaddr6 "all"
    next
end
```

4. Configure a shaping profile to set the priority, and the guaranteed and maximum bandwidth percentages for each class:

```
config firewall shaping-profile
    edit "ingShapeProfile"
        set default-class-id 2
        config shaping-entries
            edit 2
                set class-id 2
                set priority low
                set guaranteed-bandwidth-percentage 10
                set maximum-bandwidth-percentage 60
            next
            edit 3
                set class-id 3
                set guaranteed-bandwidth-percentage 20
                set maximum-bandwidth-percentage 100
            next
            edit 4
                set class-id 4
                set guaranteed-bandwidth-percentage 30
                set maximum-bandwidth-percentage 100
            next
            edit 5
                set class-id 5
                set priority medium
                set guaranteed-bandwidth-percentage 10
                set maximum-bandwidth-percentage 50
            next
        end
    next
end
```

5. Configure the inbandwidth and apply the ingress shaping profile on port2:

```
config system interface
    edit "port2"
        set ip 10.1.100.1 255.255.255.0
        set inbandwidth 100000
        set ingress-shaping-profile "ingShapeProfile"
        config ipv6
            set ip6-address 2000:10:1:100::1/64
        end
```

```
    next
end
```

Inbandwidth must be configured for traffic shaping to take effect.

6. Configure a firewall policy to allow traffic to go through. Since traffic shaping is for inbound traffic on port2, the policy is defined from port2 to wan1:

```
config firewall policy
    edit 2
        set srcintf "port2"
        set dstintf "wan1"
        set action accept
        set srcaddr "all"
        set dstaddr "all"
        set srcaddr6 "all"
        set dstaddr6 "all"
        set schedule "always"
        set service "ALL"
        set logtraffic all
        set auto-asic-offload disable
        set nat enable
    next
end
```

NPU must be disabled by configuring `set auto-asic-offload disable`.

Verifying that the traffic is being shaped

In each of the following cases, the server PCs (PC4 and PC5) are configured as iPerf servers. The client PCs (PC1 and PC2) are configured as iPerf clients. The client sends traffic to the server from the client to server direction, triggering inbound traffic shaping on the port2 interface. The inbound bandwidth on port2 is 100 Mbps.

Case 1: single stream, PC1 to PC4

Traffic is sent from PC1 to PC4. There is no other traffic. Traffic is marked with class ID 2 and allocated the maximum bandwidth 60 Mbps (60%).

```
# diagnose netlink interface list port2
if=port2 family=00 type=1 index=20 mtu=1500 link=0 master=0
ref=25 state=start present fw_flags=3800 flags=up broadcast run multicast
Qdisc=mq hw_addr=70:4c:a5:7d:d4:95 broadcast_addr=ff:ff:ff:ff:ff:ff
ingress traffic control:
        bandwidth=100000(kbps) lock_hit=50 default_class=2 n_active_class=4
        class-id=2      allocated-bandwidth=60000(kbps)         guaranteed-bandwidth=10000
(kbps)
                        max-bandwidth=60000(kbps)       current-bandwidth=60002(kbps)
                        priority=low    forwarded_bytes=58157K
                        dropped_packets=94K     dropped_bytes=125385K
        class-id=5      allocated-bandwidth=1000(kbps)  guaranteed-bandwidth=10000(kbps)
                        max-bandwidth=50000(kbps)       current-bandwidth=0(kbps)
                        priority=medium         forwarded_bytes=0
                        dropped_packets=0       dropped_bytes=0
        class-id=3      allocated-bandwidth=15000(kbps)         guaranteed-bandwidth=20000
(kbps)
                        max-bandwidth=100000(kbps)      current-bandwidth=0(kbps)
                        priority=high   forwarded_bytes=0
```

```
                        dropped_packets=0       dropped_bytes=0
        class-id=4      allocated-bandwidth=24000(kbps)         guaranteed-bandwidth=30000
(kbps)
                        max-bandwidth=100000(kbps)      current-bandwidth=0(kbps)
                        priority=high   forwarded_bytes=0
                        dropped_packets=0       dropped_bytes=0
stat: rxp=173465879 txp=2430534 rxb=194665548609 txb=2767375732 rxe=0 txe=0 rxd=0 txd=0 mc=0
collision=0 @ time=1628814469
re: rxl=0 rxo=0 rxc=0 rxf=0 rxfi=0 rxm=0
te: txa=0 txc=0 txfi=0 txh=0 txw=0
misc rxc=0 txc=0
input_type=0 state=3 arp_entry=0 refcnt=25
```

Case 2: dual stream, PC1 to PC4, PC2 to PC4

Traffic is sent from both PC1 and PC2 to PC4. PC1 to PC4 traffic is marked with class ID 2 and low priority, and PC2 to PC4 traffic is marked with class ID 3 and high priority. Both class 2 and 3 will be allocated their guaranteed bandwidth first, using up 10% and 20% respectively. The remaining available bandwidth is used by class 3 since it has a higher priority. Class 2 uses around 10 Mbps, and class 3 uses around 90 Mbps.

```
# diagnose netlink interface list port2
if=port2 family=00 type=1 index=20 mtu=1500 link=0 master=0
ref=36 state=start present fw_flags=3800 flags=up broadcast run multicast
Qdisc=mq hw_addr=70:4c:a5:7d:d4:95 broadcast_addr=ff:ff:ff:ff:ff:ff
ingress traffic control:
        bandwidth=100000(kbps) lock_hit=181 default_class=2 n_active_class=4
        class-id=2      allocated-bandwidth=10000(kbps)         guaranteed-bandwidth=10000
(kbps)
                        max-bandwidth=60000(kbps)       current-bandwidth=10001(kbps)
                        priority=low    forwarded_bytes=1799482K
                        dropped_packets=5998K   dropped_bytes=7965553K
        class-id=5      allocated-bandwidth=1000(kbps)  guaranteed-bandwidth=10000(kbps)
                        max-bandwidth=50000(kbps)       current-bandwidth=0(kbps)
                        priority=medium         forwarded_bytes=0
                        dropped_packets=0       dropped_bytes=0
        class-id=3      allocated-bandwidth=88000(kbps)         guaranteed-bandwidth=20000
(kbps)
                        max-bandwidth=100000(kbps)      current-bandwidth=88000(kbps)
                        priority=high   forwarded_bytes=345039K
                        dropped_packets=324K    dropped_bytes=430862K
        class-id=4      allocated-bandwidth=1000(kbps)  guaranteed-bandwidth=30000(kbps)
                        max-bandwidth=100000(kbps)      current-bandwidth=0(kbps)
                        priority=high   forwarded_bytes=0
                        dropped_packets=0       dropped_bytes=0
stat: rxp=181269891 txp=2433428 rxb=205136511596 txb=2771214402 rxe=0 txe=0 rxd=0 txd=0 mc=0
collision=0 @ time=1628815849
re: rxl=0 rxo=0 rxc=0 rxf=0 rxfi=0 rxm=0
te: txa=0 txc=0 txfi=0 txh=0 txw=0
misc rxc=0 txc=0
input_type=0 state=3 arp_entry=0 refcnt=36
```

Case 3: multiple streams

Multiple streams of traffic are sent at the same time:

- PC1 to PC4 traffic is assigned class 2 with low priority, and a guaranteed bandwidth of 10 Mbps.
- PC2 to PC4 traffic is assigned class 3 with high priority, and a guaranteed bandwidth of 20 Mbps.
- PC2 to PC5 traffic is assigned class 4 with high priority, and a guaranteed bandwidth of 30 Mbps.

All classes will be allocated their guaranteed bandwidth first, using up 10 Mbps, 20 Mbps, and 30 Mbps respectively. The remaining available bandwidth (40 Mbps) is shared by class 3 and class 4 based on their guaranteed bandwidth ratio of 20:30.

- Class 3's share of the remaining 40 Mbps traffic = 40 × 20/(20 + 30) =16 Mpbs
- Class 4's share of the remaining 40 Mbps traffic = 40 × 30/(20 + 30) =24 Mpbs

Each class is allocated roughly the following bandwidth:

- Class 2: 10 Mbps
- Class 3: 20 Mbps + 16 Mbps = 36 Mbps
- Class 4: 30 Mbps + 24 Mbps = 54 Mbps

```
# diagnose netlink interface list port2
if=port2 family=00 type=1 index=20 mtu=1500 link=0 master=0
ref=27 state=start present fw_flags=3800 flags=up broadcast run multicast
Qdisc=mq hw_addr=70:4c:a5:7d:d4:95 broadcast_addr=ff:ff:ff:ff:ff:ff
ingress traffic control:
        bandwidth=100000(kbps) lock_hit=148731 default_class=2 n_active_class=4
        class-id=2      allocated-bandwidth=10000(kbps)         guaranteed-bandwidth=10000
(kbps)
                        max-bandwidth=60000(kbps)       current-bandwidth=10004(kbps)
                        priority=low    forwarded_bytes=2267956K
                        dropped_packets=10389K  dropped_bytes=13796469K
        class-id=5      allocated-bandwidth=1000(kbps)  guaranteed-bandwidth=10000(kbps)
                        max-bandwidth=50000(kbps)       current-bandwidth=0(kbps)
                        priority=medium         forwarded_bytes=0
                        dropped_packets=0       dropped_bytes=0
        class-id=3      allocated-bandwidth=35000(kbps)         guaranteed-bandwidth=20000
(kbps)
                        max-bandwidth=100000(kbps)      current-bandwidth=35729(kbps)
                        priority=high   forwarded_bytes=2119502K
                        dropped_packets=6020K   dropped_bytes=7994926K
        class-id=4      allocated-bandwidth=54000(kbps)         guaranteed-bandwidth=30000
(kbps)
                        max-bandwidth=100000(kbps)      current-bandwidth=53907(kbps)
                        priority=high   forwarded_bytes=902415K
                        dropped_packets=4141K   dropped_bytes=5499248K
stat: rxp=197827723 txp=2433885 rxb=227356779526 txb=2771602657 rxe=0 txe=0 rxd=0 txd=0 mc=0
collision=0 @ time=1628816440
re: rxl=0 rxo=0 rxc=0 rxf=0 rxfi=0 rxm=0
te: txa=0 txc=0 txfi=0 txh=0 txw=0
misc rxc=0 txc=0
input_type=0 state=3 arp_entry=0 refcnt=27
```

Zero Trust Network Access

This section includes information about ZTNA related new features:

Zero Trust Network Access introduction

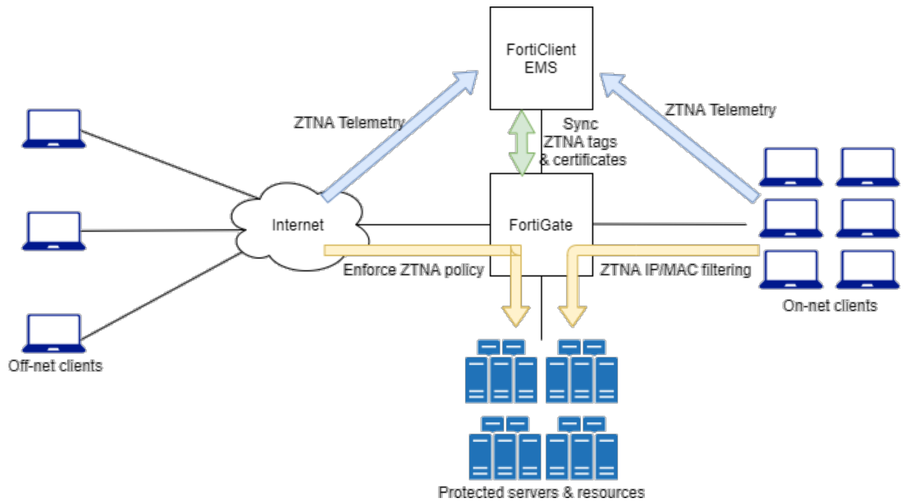
Zero Trust Network Access (ZTNA) is an access control method that uses client device identification, authentication, and Zero Trust tags to provide role-based application access. It gives administrators the flexibility to manage network access for On-net local users and Off-net remote users. Access to applications is granted only after device verification, authenticating the user's identity, authorizing the user, and then performing context based posture checks using Zero Trust tags.

Traditionally, a user and a device have different sets of rules for on-net access and off-net VPN access to company resources. With a distributed workforce and access that spans company networks, data centers, and cloud, managing the rules can become complex. User experience is also affected when multiple VPNs are needed to get to various resources. ZTNA can improve this experience.

ZTNA access proxy and IP/MAC based access control

- ZTNA access proxy allows users to securely access resources through an SSL encrypted access proxy. This simplifies remote access by eliminating the use of VPNs.
- IP/MAC based access control combines IP/MAC with uses ZTNA tags for identification and security posture check to implement role-based zero trust access.

ZTNA telemetry, tags, and policy enforcement

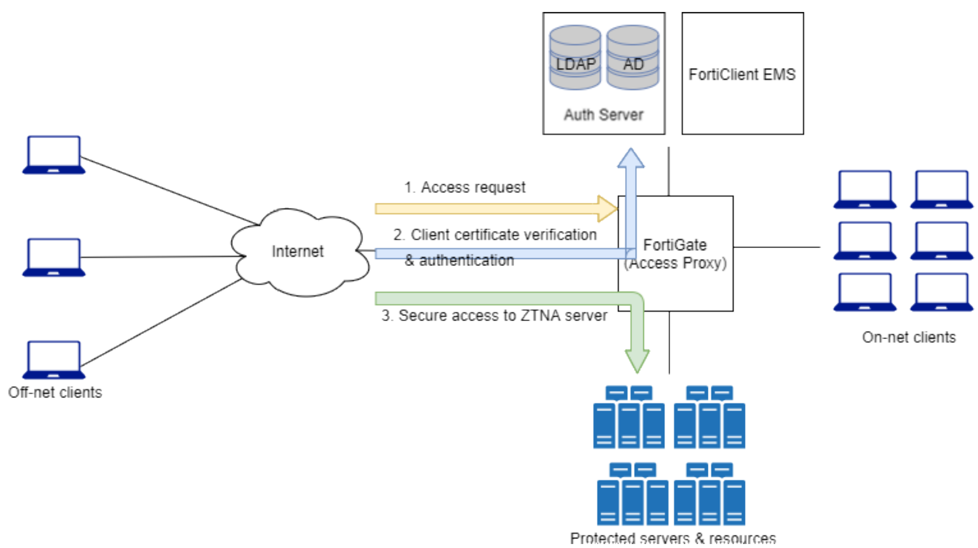


When On-net and Off-net FortiClient endpoints register to FortiClient EMS, device information, log on user information, and security posture are all shared over ZTNA telemetry with the EMS server. Clients also make a certificate signing request to obtain a client certificate from the EMS that is acting as the ZTNA Certificate Authority (CA).

Based on the client information, EMS applies matching Zero Trust tagging rules to tag the clients. These tags, and the client certificate information, are synchronized with the FortiGate in real-time. This allows the FortiGate to verify the client's identity using the client certificate, and grant access based on the ZTNA tags applied in the ZTNA rule.

For more information, see Establish device identity and trust context with FortiClient EMS on page 981.

Access proxy



The FortiGate access proxy can proxy HTTP, SSH, RDP, SMB, FTP, and other TCP traffic over secure connections with the client. This enables seamless access from the client to the protected servers, without needing to form IPsec or SSL VPN tunnels.

HTTPS access proxy

The FortiGate HTTPS access proxy works as a reverse proxy for the HTTP server. When a client connects to a webpage hosted by the protected server, the address resolves to the FortiGate's access proxy VIP. The FortiGate proxies the connection and takes steps to authenticate the user. It prompts the user for their certificate on the browser, and verifies this against the ZTNA endpoint record that is synchronized from the EMS. If an authentication scheme, such as SAML authentication, is configured, the client is redirected to a captive portal for sign-on. If this passes, traffic is allowed based on the ZTNA rules, and the FortiGate returns the webpage to the client.

For example configurations, see ZTNA HTTPS access proxy example on page 988, ZTNA HTTPS access proxy with basic authentication example on page 995, and ZTNA proxy access with SAML authentication example on page 1013.

TCP forwarding access proxy (TFAP)

The TCP forwarding access proxy works as a special type of HTTPS reverse proxy. Instead of proxying traffic to a web server, TCP traffic is tunneled between the client and the access proxy over HTTPS, and forwarded to the protected resource. The FortiClient endpoint configures the ZTNA connection by pointing to the proxy gateway, and then specifying the destination host that it wants to reach. An HTTPS connection is made to the FortiGate's access proxy VIP, where the client certificate is verified and access is granted based on the ZTNA rules. TCP traffic is forwarded from the FortiGate to the protected resource, and an end to end connection is established. To reduce overhead, you can disable access proxy encryption on the client, as some TCP protocols, like RDP, are already secure. The TCP forwarding access proxy supports UTM scanning and deep inspection for HTTP, HTTPS, SMTP, SMTPS, IMAP, IMAPS, POP3, POP3S, SMB, and CIFS.

For an example configuration, see ZTNA TCP forwarding access proxy example on page 1002.

SSH access proxy

The SSH access proxy provides some benefits to proxying SSH connections over TFAP, including allowing SSH deep inspection, performing optional SSH host-key validation, and allowing one time user authentication to authenticate the ZTNA SSH access proxy connection and SSH server connection.

For an example configuration, see ZTNA SSH access proxy example on page 1029.

Basic ZTNA configuration components

The basic components that are require to configure ZTNA access proxy on the FortiGate are:

1. FortiClient EMS fabric connector and ZTNA tags.
2. FortiClient EMS running version 7.0.0 or later or FortiClient EMS Cloud.
3. FortiClient running 7.0.0 or later.
4. ZTNA server
5. ZTNA rule
6. (Optional) User authentication
7. (Optional) HA configurations

For configuration details, see Basic ZTNA configuration on page 973.

Basic ZTNA configuration

To deploy a ZTNA access proxy, configure the following components on the FortiGate:

1. Configure a FortiClient EMS connector on page 973
2. Configure a ZTNA server on page 975
3. Configure a ZTNA rule on page 978
4. Optional authentication on page 979
5. Optional HA configurations on page 980

To configure ZTNA in the GUI, go to *System > Feature Visibility* and enable *Zero Trust Network Access*.

Configure a FortiClient EMS connector

To add an on-premise FortiClient EMS server in the GUI:

1. Go to *Security Fabric > Fabric Connectors*.
2. Click *Create New* and click *FortiClient EMS*.
3. Enter a name for the connector and the IP address or FQDN of the EMS.
4. Click *OK*.
5. A window appears to verify the EMS server certificate. Click *Accept*.
   See FortiClient EMS for more information.

To add an on-premise FortiClient EMS server in the CLI:

```
config endpoint-control fctems
    edit <name>
        set server <server IP or domain>
    next
end
```

To add FortiClient EMS Cloud in the GUI:

1. Go to *Security Fabric > Fabric Connectors*.
2. Double-click *FortiClient EMS* to open it for editing.
3. Set *Type* to *FortiClient EMS Cloud*.
4. Enter a name for the connector.
5. Click *OK*. A window appears to verify the EMS server certificate.
6. Click *Accept*.
   See FortiClient EMS for more information.

To add FortiClient EMS Cloud in the CLI:

```
config endpoint-control fctems
    edit <name>
```

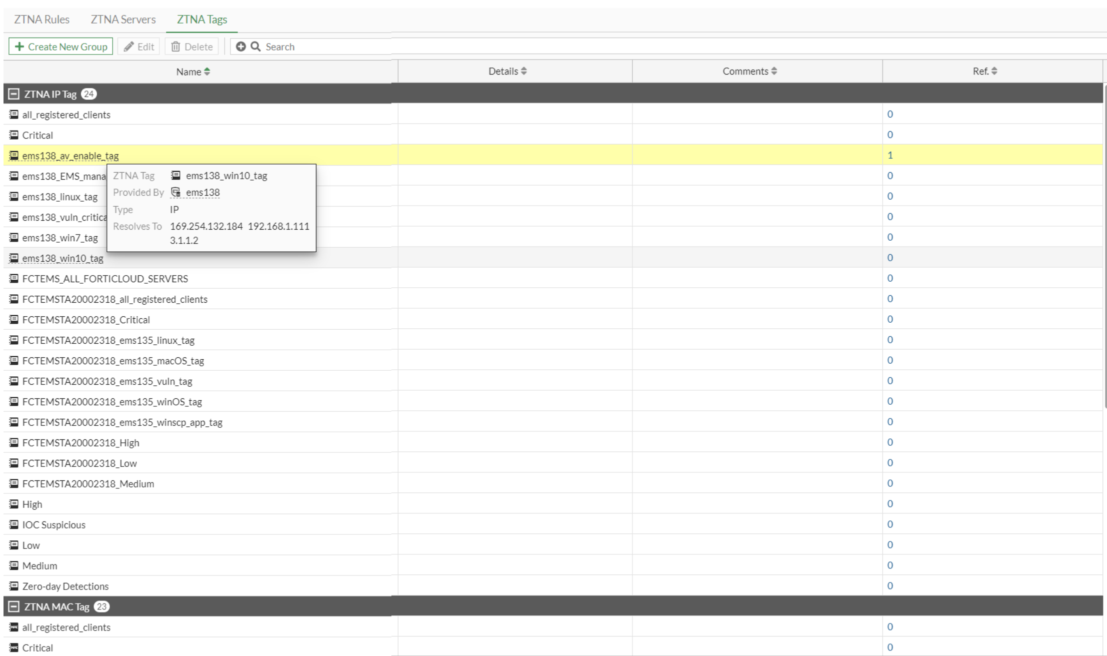
```
        set fortinetone-cloud-authentication enable
        set certificate <string>
    next
end
```

ZTNA tags

After the FortiGate connects to the FortiClient EMS, it automatically synchronizes ZTNA tags. ZTNA tags are generated from tagging rules configured on the FortiClient EMS. These tagging rules are based on various posture checks that can be applied on the endpoints. See Endpoint Posture Check Reference.

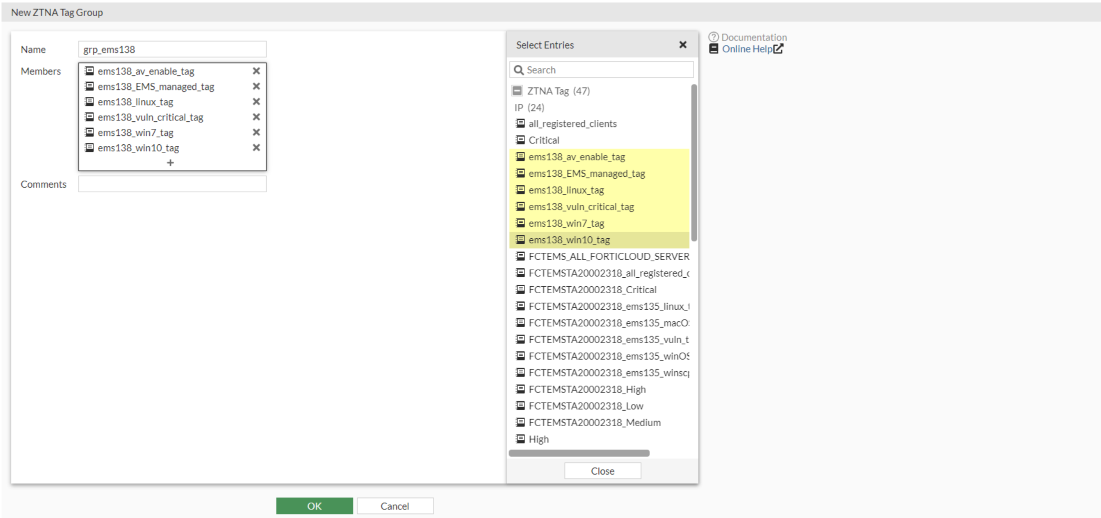
To view the synchronized ZTNA tags in the GUI:

1. Go to *Policy & Objects > ZTNA* and select the *ZTNA Tags* tab.
2. Hover the cursor over a tag name to view more information about the tag, such as its resolved addresses.



To create a ZTNA tag group in the GUI:

1. Go to *Policy & Objects > ZTNA* and select the *ZTNA Tags* tab.
2. Click *Create New Group*.
3. Enter a name for the group and select the group members.



4. Click *OK*.

To view the synchronized ZTNA tags in the CLI:

```
# diagnose firewall dynamic address
# diagnose firewall dynamic list
```

To create a ZTNA tag group in the CLI:

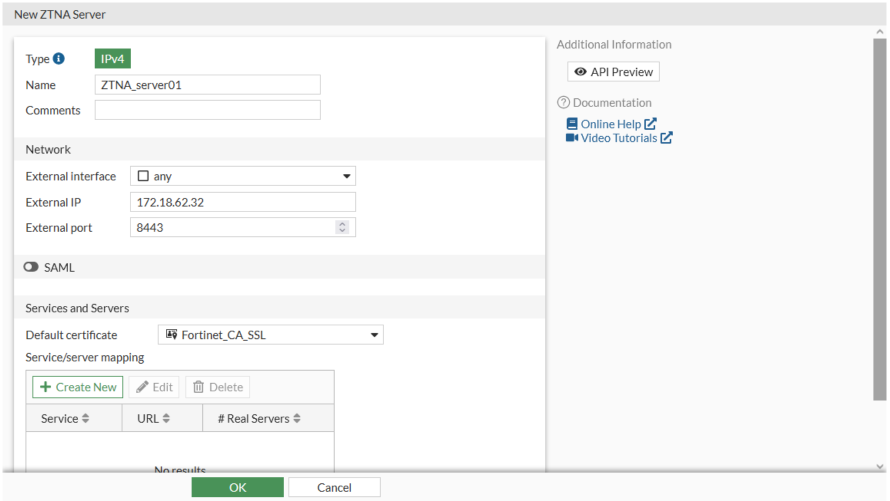
```
config firewall addrgrp
    edit <group name>
        set category ztna-ems-tag
        set member <members>
    next
end
```

Configure a ZTNA server

To configure a ZTNA server, define the access proxy VIP and the real servers that clients will connect to. The access proxy VIP is the FortiGate ZTNA gateway that clients make HTTPS connections to. The service/server mappings define the virtual host matching rules and the real server mappings of the HTTPS requests.

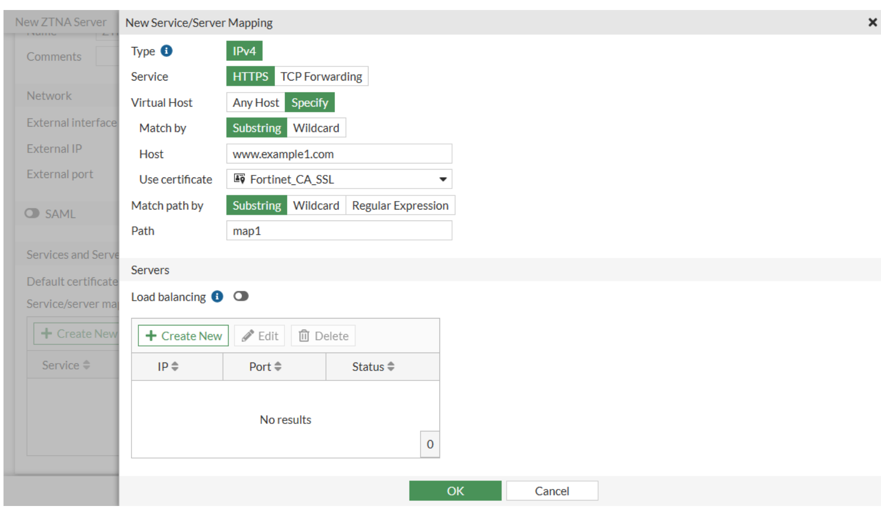
To create a ZTNA server for HTTPS access proxy in the GUI:

1. Go to *Policy & Objects* > *ZTNA* and select the *ZTNA Servers* tab.
2. Click *Create New*.
3. Enter a name for the server.
4. Select an external interface, enter the external IP address, and select the external port that the clients will connect to.
5. Select the *Default certificate*. Clients will be presented with this certificate when they connect to the access proxy VIP.



6. Add server mapping:
   a. In the *Service/server mapping* table, click *Create New*.
   b. Set *Service* to *HTTPS*.
   c. Set *Virtual Host* to *Any Host* or *Specify*.
      - *Any Host*: Any request that resolves to the access proxy VIP will be mapped to your real servers. For example, if both www.example1.com and www.example2.com resolve to the VIP, then both requests are mapped to your real servers.
      - *Specify*: Enter the name or IP address of the host that the request must match. For example, if www.example1.com is entered as the host, then only requests to www.example1.com will match.
   d. Configure the path as needed.

      The path can be matched by substring, wildcard, or regular expression. For example, if the virtual host is specified as www.example1.com, and the path substring is map1, then www.example1/map1 will be matched.



   e. If multiple servers will be configured, enable *Load balancing* and select an algorithm.
   f. Add a server:
      i. In the *Servers* table, click *Create New*.
      ii. Enter the server IP address and port number.
      iii. Set the server status.

iv. Click *OK*.
v. Add more servers as needed.
g. Click *OK*.
h. Add more server mappings as needed.
7. Click *OK*.

To create a ZTNA server and access proxy VIP in the CLI:

1. Configure an access proxy VIP:

```
config firewall vip
    edit <name>
        set type access-proxy
        set extip <external IP>
        set extintf <external interface>
        set server-type { https | ssh }
        set extport <external port>
        set ssl-certificate <certificate>
    next
end
```

2. If the virtual host is specified, configure the virtual host:

```
config firewall access-proxy-virtual-host
    edit <auto generated when configured from GUI>
        set ssl-certificate <certificate>
        set host <host name or IP>
        set host-type { sub-string | wildcard }
    next
end
```

3. Configure the server and path mapping:

```
config firewall access-proxy
    edit <name>
        set vip <vip name>
        set client-cert { enable | disable }
        set empty-cert-action { accept | block }
        set log-blocked-traffic {enable | disable}
        config api-gateway
            edit 1
                set url-map <mapped path>
                set service { http | https | tcp-forwarding | samlsp }
                set virtual-host <name of virtual-host if specified>
                set url-map-type { sub-string | wildcard | regex }
                config realservers
                    edit 1
                        set addr-type ip
                        set ip <ip of real server>
                        set port <port>
                        set status { active | standby | disable }
                        set health-check { enable | disable }
                    next
                end
                set ldb-method static
                set persistence none
```

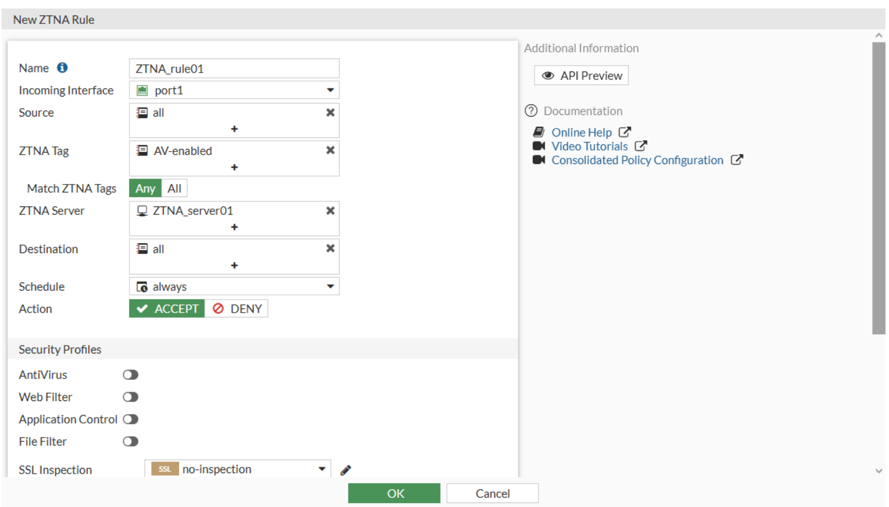
```
                    set ssl-dh-bits 2048
                    set ssl-algorithm high
                    set ssl-min-version tls-1.1
                    set ssl-max-version tls-1.3
                next
            end
        next
    end
```

Configure a ZTNA rule

A ZTNA rule is a proxy policy used to enforce access control. ZTNA tags or tag groups can be defined to enforce zero trust role based access. Security profiles can be configured to protect this traffic.

To configure a ZTNA rule in the GUI:

1. Go to *Policy & Objects* > *ZTNA* and select the *ZTNA Rules* tab.
2. Click *Create New*.
3. Enter a name for the rule.
4. Select an *Incoming Interface* and *Source*.
5. Add the ZTNA tags or tag groups that are allowed access. If multiple tags are included, select the *Match ZTNA Tags* method, *Any* or *All*.
6. Select the *ZTNA Server*.
7. Select the *Destination*.



8. Configure the remaining options as needed.
9. Click *OK*.

To configure a ZTNA rule in the CLI:

```
config firewall proxy-policy
    edit 1
        set name <ZTNA rule name>
        set proxy access-proxy
        set access-proxy <access proxy>
```

```
        set srcintf <interface>
        set srcaddr "all"
        set transparent {enable | disable}
        set dstaddr "all"
        set ztna-ems-tag <ZTNA tag(s)>
        set ztna-tags-match-logic {or | and}
        set action accept
        set schedule "always"
        set logtraffic all
        set poolname <ip_pool>
        set utm-status enable
        set ssl-ssh-profile <inspection profile>
    next
end
```

The `transparent` and `poolname` settings cannot be enabled at the same time. Use one setting at a time when configuring ZTNA rules.

Optional authentication

To configure authentication to the access proxy, you must configure an authentication scheme and authentication rule in the GUI or CLI. They are used to authenticate proxy-based policies, similar to configuring authentication for explicit and transparent proxy.

The authentication scheme defines the method of authentication that is applied. For ZTNA, basic HTTP and SAML methods are supported. Each method has additional settings to define the data source to check against. For example, with basic HTTP authentication, a user database can reference an LDAP server, RADIUS server, local database, or other supported authentication servers that the user is authenticated against.

The authentication rule defines the proxy sources and destinations that require authentication, and which authentication scheme to apply. For ZTNA, active authentication method is supported. The active authentication method references a scheme where users are actively prompted for authentication, like with basic authentication.

After the authentication rule triggers the method to authenticate the user, a successful authentication returns the groups that the user belongs to. In the ZTNA rule and proxy policy you can define a user or user group as the allowed source. Only users that match that user or group are allowed through the proxy policy.

To configure a basic authentication scheme:

```
config authentication scheme
    edit <name>
        set method basic
        set user-database <auth server>
    next
end
```

To configure an authentication rule:

```
config authentication rule
    edit <name>
        set status enable
        set protocol http
```

```
        set srcintf <interface>
        set srcaddr <address>
        set dstaddr <address>
        set ip-based enable
        set active-auth-method <active auth scheme>
    next
end
```

To apply a user group to a ZTNA rule in the GUI:

1. Go to *Policy & Objects* > *ZTNA* and select the *ZTNA Rules* tab.
2. Edit an existing rule, or click *Create New* to create a new rule.
3. Click in the *Source* field, select the *User* tab, and select the users and user groups that will be allowed access.
4. Configure the remaining settings as required.
5. Click *OK*.

To apply a user group to a ZTNA rule in the CLI:

```
config firewall proxy-policy
    edit <policy ID>
        set name <ZTNA rule name>
        set proxy access-proxy
        set access-proxy <access proxy>
        set srcintf <interface>
        set srcaddr "all"
        set dstaddr "all"
        set ztna-ems-tag <ZTNA tags>
        set ztna-tags-match-logic {or | and}
        set action accept
        set schedule "always"
        set logtraffic all
        set groups <user group>
        set utm-status enable
        set ssl-ssh-profile <inspection profile>
    next
end
```

The authentication rule and scheme defines the method used to authenticate users. With basic HTTP authentication, a sign in prompt is shown after the client certificate prompt. After the authentication passes, the returned groups that the user is a member of are checked against the user groups that are defined in the ZTNA rule. If a group matches, then the user is allowed access after passing a posture check.

For basic setup information, see ZTNA HTTPS access proxy with basic authentication example on page 995.

For advanced setup information, see ZTNA proxy access with SAML authentication example on page 1013 and ZTNA access proxy with SAML and MFA using FortiAuthenticator example on page 1036.

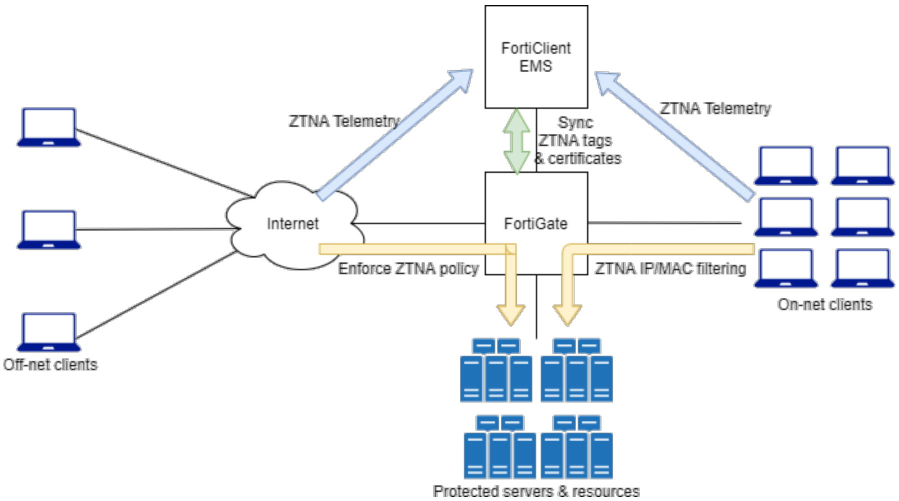
Optional HA configurations

User information and TLS sessions are synchronized between HA members for ZTNA proxy sessions. When a failover occurs, the new primary unit will continue allowing sessions from the logged in users without asking for the client certificate and re-authentication again.

There are no special configurations for HA. Refer to HA active-passive cluster setup on page 2035 and HA active-active cluster setup on page 2037 to configure your HA cluster.

Establish device identity and trust context with FortiClient EMS

How device identity is established through client certificates, and how device trust context is established between FortiClient, FortiClient EMS, and the FortiGate, are integral to ZTNA.



Device roles

FortiClient

FortiClient endpoints provide the following information to FortiClient EMS when they register to the EMS:

- Device information (network details, operating system, model, and others)
- Logged on user information
- Security posture (On-net/Off-net, antivirus software, vulnerability status, and others)

It also requests and obtains a client device certificate from the EMS ZTNA Certificate Authority (CA) when it registers to FortiClient EMS. The client uses this certificate to identify itself to the FortiGate.

FortiClient EMS

FortiClient EMS issues and signs the client certificate with the FortiClient UID, certificate serial number, and EMS serial number. The certificate is then synchronized to the FortiGate. EMS also shares its EMS ZTNA CA certificate with the FortiGate, so that the FortiGate can use it to authenticate the clients.

FortiClient EMS uses zero trust tagging rules to tag endpoints based on the information that it has on each endpoint. The tags are also shared with the FortiGate. See Endpoint Posture Check Reference for a list of the endpoint posture checks that EMS can perform.

Each ZTNA tag creates two firewall addresses in all VDOMs on a FortiGate. One firewall address is the IP address, and the other firewall address is the MAC address. Because each FortiGate model has a global limit and a per-VDOM limit for the maximum number of supported firewall addresses, the FortiGate model determines the maximum number of ZTNA tags allowable by that unit, which is the maximum number of firewall address divided by two. For each FortiGate model's limit, see the Maximum Values table.

FortiGate

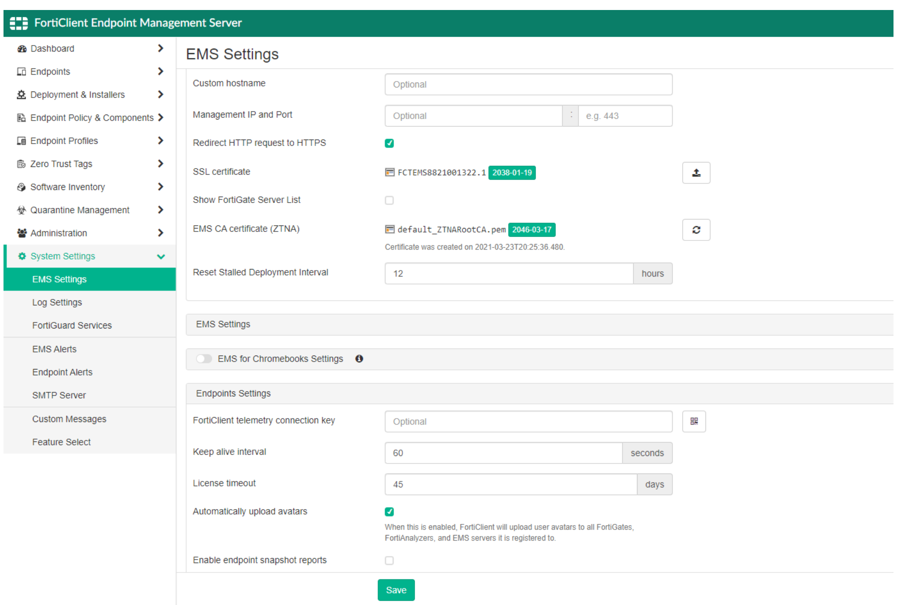
The FortiGate maintains a continuous connection to the EMS server to synchronize endpoint device information, including primarily:

- FortiClient UID
- Client certificate SN
- EMS SN
- Device credentials (user/domain)
- Network details (IP and MAC address and routing to the FortiGate)

When a device's information changes, such as when a client moves from on-net to off-net, or their security posture changes, EMS is updated with the new device information and then updates the FortiGate. The FortiGate's WAD daemon can use this information when processing ZTNA traffic. If an endpoint's security posture change causes it to no longer match the ZTNA rule criteria on an existing session, then the session is terminated.

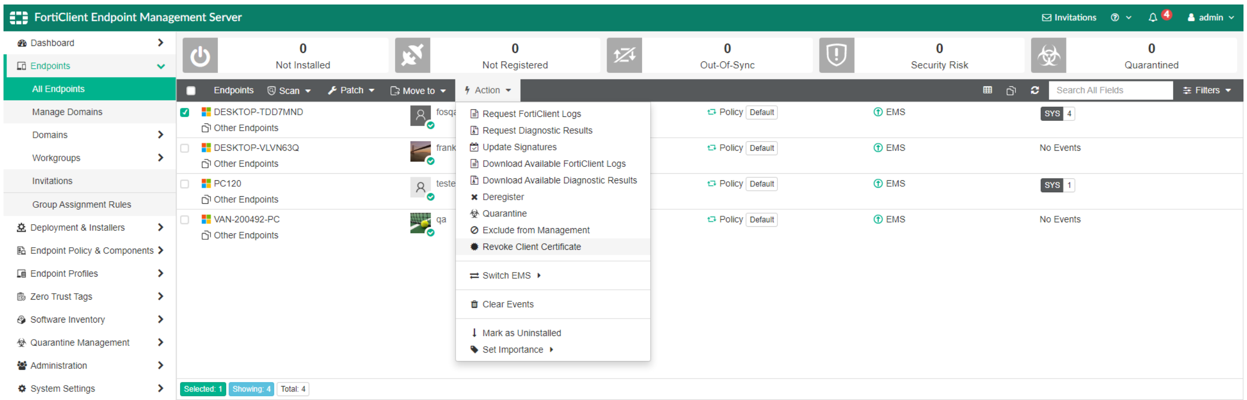
Certificate management on FortiClient EMS

FortiClient EMS has a *default_ZTNARootCA* certificate generated by default that the ZTNA CA uses to sign CSRs from the FortiClient endpoints. Clicking the refresh button revokes and updates the root CA, forcing updates to the FortiGate and FortiClient endpoints by generating new certificates for each client.



Do not confuse the EMS CA certificate (ZTNA) with the SSL certificate. The latter is the server certificate that is used by EMS for HTTPS access and fabric connectivity to the EMS server.

EMS can also manage individual client certificates. To revoke the current client certificate that is used by the endpoint: go to *Endpoint > All Endpoints*, select the client, and click *Action > Revoke Client Certificate*.



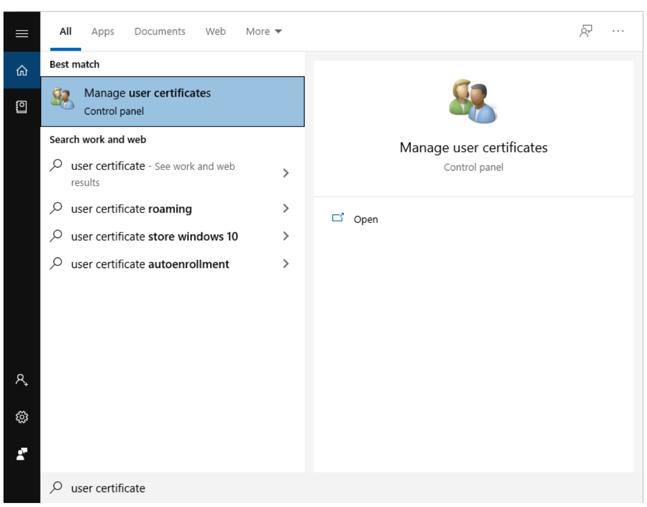
Locating and viewing the client certificate on an endpoint

In Windows, FortiClient automatically installs certificates into the certificate store. The certificate information in the store, such as certificate UID and SN, should match the information on EMS and the FortiGate.

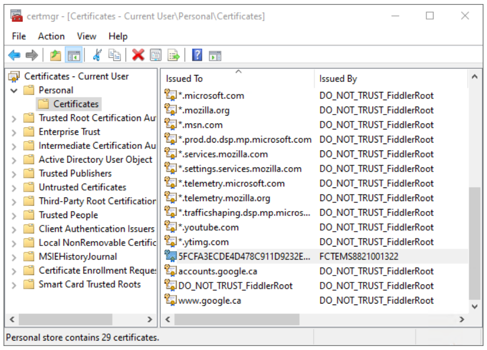
To locate certificates on other operating systems, consult the vendor documentation.

To locate the client certificate and EMS ZTNA CA certificate on a Windows PC:

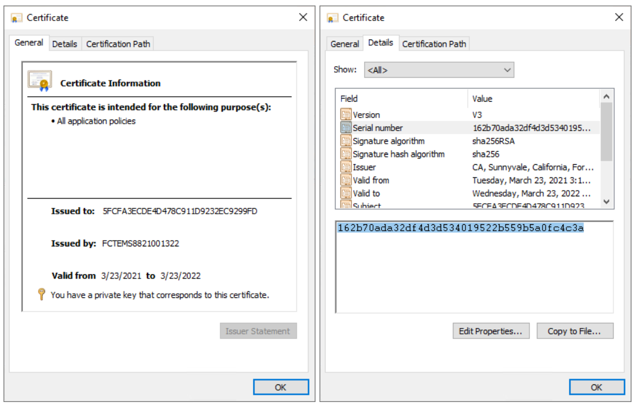
1. In the Windows search box, enter *user certificate* and click *Manage user certificates* from the results.



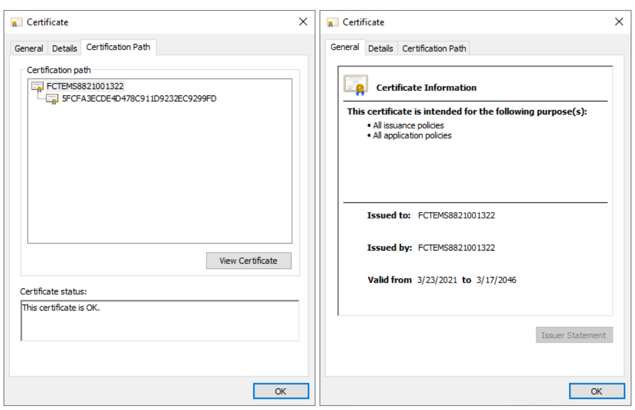
2. In the certificate manager, go to *Certificates - Current User > Personal > Certificates* and find the certificate that is issued by the FortiClient EMS.



3. Right-click on it and select Properties.
4. The *General* tab shows the client certificate UID and the issue and expiry dates. The *Details* tab show the certificate SN.



5. Go to the *Certificate Path* tab to see the full certificate chain.
6. Select the root CA and click *View Certificate* to view the details about the EMS ZTNA CA certificate.



Verifying that the client information is synchronized to the FortiGate

The following diagnose commands help to verify the presence of matching endpoint record, and information such as the client UID, client certificate SN, and EMS certificate SN on the FortiGate. If any of the information is missing or incomplete, client certificate authentication might fail because the corresponding endpoint entry is not found. More in-depth diagnosis would be needed to determine the reason for the missing records.

Command	Description
`# diagnose endpoint record list <ip>`	Show the endpoint record list. Optionally, filter by the endpoint IP address.
`# diagnose endpoint lls-comm send ztna find-uid <uid>`	Query endpoints by client UID.
`# diagnose endpoint lls-comm send ztna find-ip-vdom <ip> <vdom>`	Query endpoints by the client IP-VDOM pair.
`# diagnose wad dev query-by uid <uid>`	Query from WAD diagnose command by UID.
`# diagnose wad dev query-by ipv4 <ip>`	Query from WAD diagnose command by IP address.
`# diagnose test application fcnacd 7` `# diagnose test application fcnacd 8`	Check the FortiClient NAC daemon ZTNA and route cache.

To check the endpoint record list for IP address 10.6.30.214:

```
# diagnose endpoint record list 10.6.30.214
Record #1:
                IP Address = 10.6.30.214
                MAC Address = 00:0c:29:ba:1e:61
                MAC list = 00:0c:29:ba:1e:61;00:0c:29:ba:1e:6b;
                VDOM = root (0)
                EMS serial number: FCTEMS8821001322
                Client cert SN: 17FF6595600A1AF53B87627AB4EBEDD032593E64
                Quarantined: no
                Online status: online
                Registration status: registered
                On-net status: on-net
                Gateway Interface: port2
                FortiClient version: 7.0.0
                AVDB version: 84.778
                FortiClient app signature version: 18.43
                FortiClient vulnerability scan engine version: 2.30
                FortiClient UID: 5FCFA3ECDE4D478C911D9232EC9299FD
                …
                Number of Routes: (1)
                        Gateway Route #0:
                                - IP:10.1.100.214, MAC: 00:0c:29:ba:1e:6b, Indirect: no
                                - Interface:port2, VFID:0, SN: FG5H1E5819902474
online records: 1; offline records: 0; quarantined records: 0
```

SSL certificate based authentication

A client certificate is obtained when an endpoint registers to EMS. FortiClient automatically submits a CSR request and the FortiClient EMS signs and returns the client certificate. This certificate is stored in the operating system's certificate store for subsequent connections. The endpoint information is synchronized between the FortiGate and FortiClient EMS.

When an endpoint disconnects or is unregistered from EMS, its certificate is removed from the certificate store and revoked on EMS. The endpoint obtains a certificate again when it reconnected the EMS.

By default, client certificate authentication is enabled on the access proxy, so when the HTTPS request is received the FortiGate's WAD process challenges the client to identify itself with its certificate. The FortiGate makes a decision based on the following possibilities:

1. If the client responds with the correct certificate that the client UID and certificate SN can be extracted from:
   - If the client UID and certificate SN match the record on the FortiGate, the client is allowed to continue with the ZTNA proxy rule processing.
   - If the client UID and certificate SN do not match the record on the FortiGate, the client is blocked from further ZTNA proxy rule processing.
2. If the client cancels and responds with an empty client certificate:
   - If `empty-cert-action` is set to `accept`, the client is allowed to continue with ZTNA proxy rule processing.
   - If `empty-cert-action` is set to `block`, the client is blocked from further ZTNA proxy rule processing.

To configure the client certificate actions:

```
config firewall access-proxy
    edit <name>
        set client-cert {enable | disable}
        set empty-cert-action {accept | block}
    next
end
```

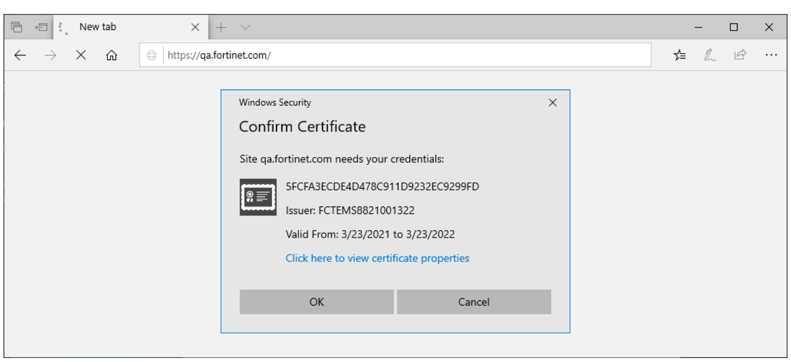
Example

In this example, a client connects to *qa.fortinet.com* and is prompted for a client certificate.

- `client-cert` is set to `enable`, and `empty-cert-action` is set to `block`.
- The ZTNA server is configured, and a ZTNA rule is set to allow this client.
- The domain resolves to the FortiGate access proxy VIP.

Scenario 1:

When prompted for the client certificate, the client clicks *OK* and provides a valid certificate that is verified by the FortiGate.

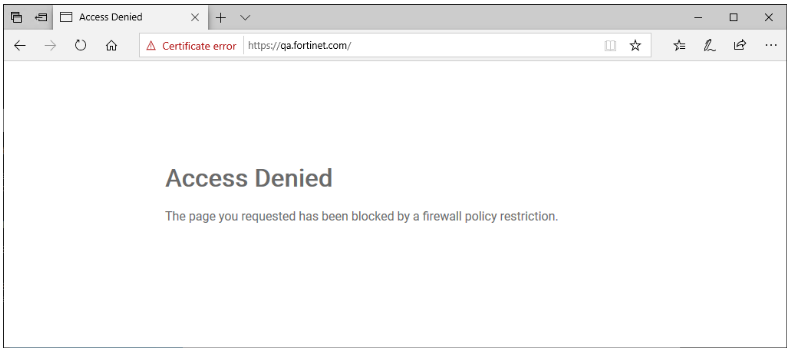


Result:

The client passes SSL certificate authentication and is allowed to access the website.

Scenario 2:

When prompted for the client certificate, the client clicks *Cancel*, resulting in an empty certificate response to the access proxy.



Result:

Because the certificate response is empty and `empty-cert-action` is set to `block`, the WAD daemon blocks the connection.

Currently, the Microsoft Edge, Google Chrome, and Safari browsers are supported by ZTNA.

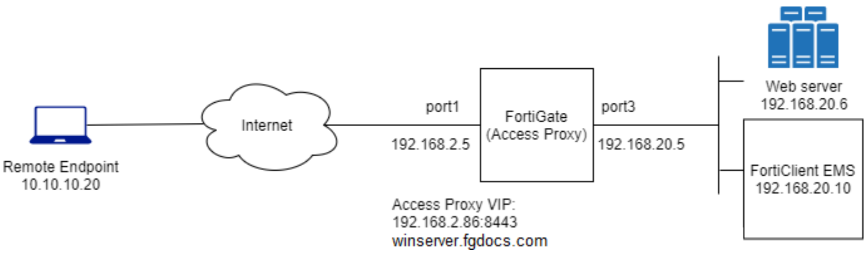
ZTNA configuration examples

This section includes the following ZTNA configuration examples:

- ZTNA HTTPS access proxy example on page 988
- ZTNA HTTPS access proxy with basic authentication example on page 995
- ZTNA TCP forwarding access proxy example on page 1002
- ZTNA TCP forwarding access proxy without encryption example on page 1008
- ZTNA proxy access with SAML authentication example on page 1013
- ZTNA IP MAC based access control example on page 1017
- ZTNA IPv6 examples on page 1023
- ZTNA SSH access proxy example on page 1029
- ZTNA access proxy with SAML and MFA using FortiAuthenticator example on page 1036
- ZTNA access proxy with SSL VPN web portal example on page 1048
- Posture check verification for active ZTNA proxy session examples on page 1053
- ZTNA TCP forwarding access proxy with FQDN example on page 1059
- ZTNA session-based form authentication on page 1062

ZTNA HTTPS access proxy example

In this example, an HTTPS access proxy is configured to demonstrate its function as a reverse proxy on behalf of the web server it is protecting. It verifies user identity, device identity, and trust context, before granting access to the protected source.



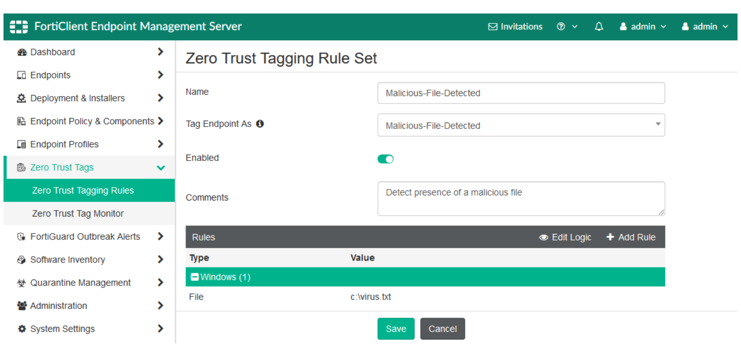
This example shows access control that allows or denies traffic based on ZTNA tags. Traffic is allowed when the FortiClient endpoint is tagged as *Low* risk, and denied when the endpoint is tagged with *Malicious-File-Detected*.

This example assumes that the FortiGate EMS fabric connector is already successfully connected.

To configure ZTNA in the GUI, go to *System > Feature Visibility* and enable *Zero Trust Network Access*.

To configure a Zero Trust tagging rule on the FortiClient EMS:

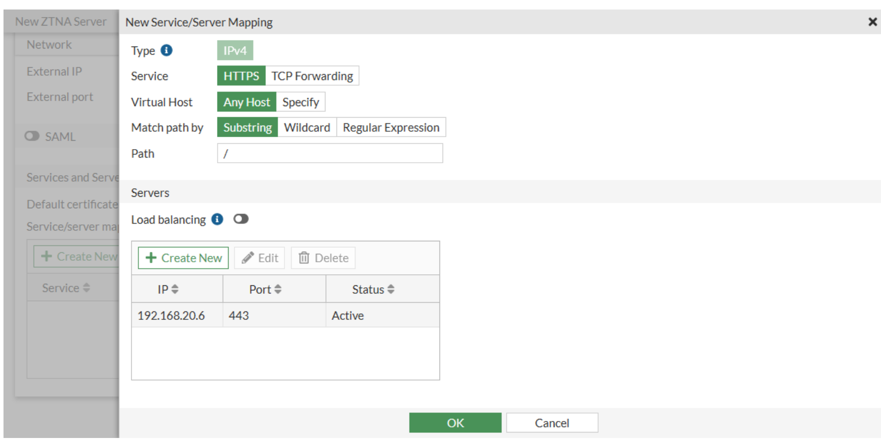
1. Log in to the FortiClient EMS.
2. Go to *Zero Trust Tags > Zero Trust Tagging Rules*, and click *Add*.
3. In the *Name* field, enter *Malicious-File-Detected*.
4. In the *Tag Endpoint As* dropdown list, select *Malicious-File-Detected*.
   EMS uses this tag to dynamically group together endpoints that satisfy the rule, as well as any other rules that are configured to use this tag.
5. Click *Add Rule* then configure the rule:
   a. For *OS*, select *Windows*.
   b. From the *Rule Type* dropdown list, select *File* and click the + button.
   c. Enter a file name, such as *C:\virus.txt*.
   d. Click *Save*.



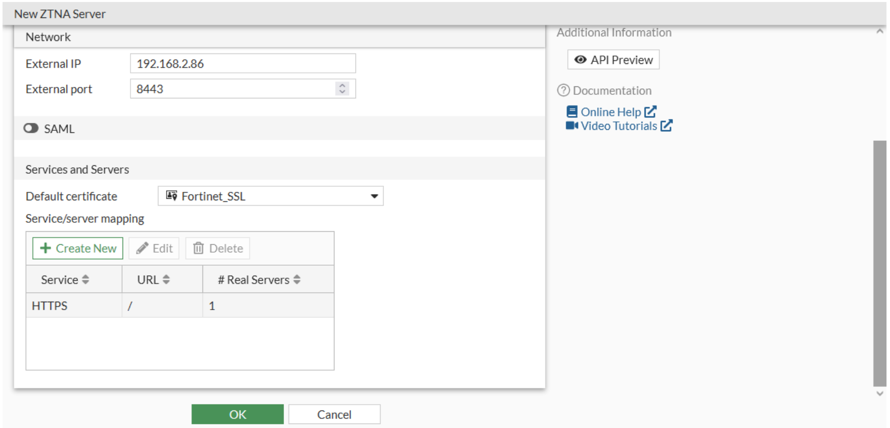
6. Click *Save*.

To configure a ZTNA server for HTTPS access proxy in the GUI:

1. Go to *Policy & Objects > ZTNA* and select the *ZTNA Servers* tab.
2. Click *Create New*.
3. Set *Name* to *WIN2K16-P1*.
4. Configure the network settings:
   a. Set *External interface* to *port1*.
   b. Set *External IP* to *192.168.2.86*.
   c. Set *External port* to *8443*.
5. Select the *Default certificate*. Clients will be presented with this certificate when they connect to the access proxy VIP.
6. Add server mapping:
   a. In the *Service/server mapping* table, click *Create New*.
   b. Set *Service* to *HTTPS*.
   c. Set *Virtual Host* to *Any Host*.
   d. Configure the path as needed. For example, to map to *winserver.fgdocs.com/fortigate*, enter */fortigate*.
   e. Add a server:
      i. In the *Servers* table, click *Create New*.
      ii. Set *IP* to *192.168.20.6*.
      iii. Set *Port* to *443*.
      iv. Click *OK*.



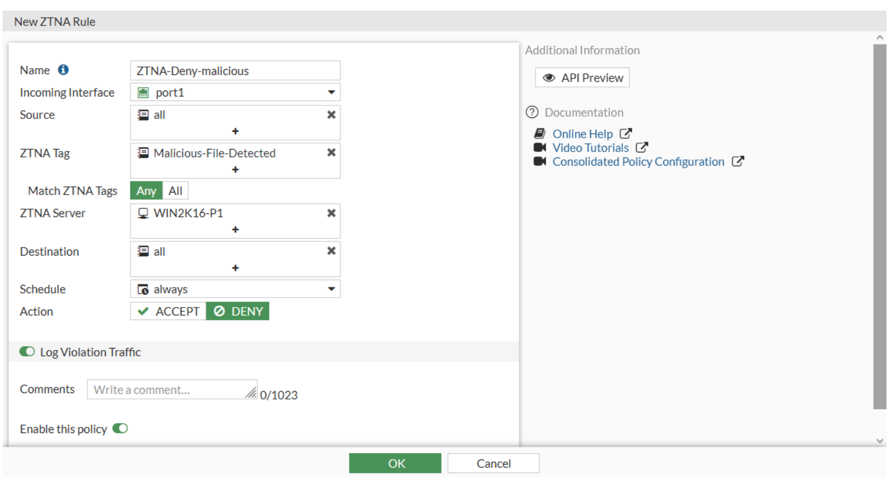
f. Click *OK*.



7. Click *OK*.

To configure ZTNA rules to allow and deny traffic based on ZTNA tags in the GUI:

1. Go to *Policy & Objects > ZTNA* and select the *ZTNA Rules* tab.
2. Create a rule to deny traffic:
   a. Click *Create New*.
   b. Set *Name* to *ZTNA-Deny-malicious*.
   c. Set *Incoming Interface* to *port1*.
   d. Set *Source* to *all*.
   e. Add the ZTNA tag *Malicious-File-Detected*.
   This tag is dynamically retrieved from EMS when you first created the Zero Trust Tagging Rule.
   f. Select the ZTNA server *WIN2K16-P1*.
   g. Set *Action* to *DENY*.
   h. Enable *Log Violation Traffic*.



   i. Click *OK*.
3. Create a rule to allow traffic:
   a. Click *Create New*.
   b. Set *Name* to *proxy-WIN2K16-P1*.

c. Set *Incoming Interface* to *port1*.
d. Set *Source* to *all*. This can also be set to specific IP addresses to only allow those addresses to connect to this HTTPS access proxy.
e. Add the ZTNA tag *Low*.
f. Select the ZTNA server *WIN2K16-P1*.
g. Configure the remaining options as needed.
h. Click *OK*.

4. On the ZTNA rules list, make sure that the deny rule (*ZTNA-Deny-malicious*) is above the allow rule (*proxy-WIN2K16-P1*).

To configure HTTPS access in the CLI:

1. Configure the access proxy VIP:

```
config firewall vip
    edit "WIN2K16-P1"
        set type access-proxy
        set extip 192.168.2.86
        set extintf "port1"
        set server-type https
        set extport 8443
        set ssl-certificate "Fortinet_SSL"
    next
end
```

2. Configure the server and path mapping:

```
config firewall access-proxy
    edit "WIN2K16-P1"
        set vip "WIN2K16-P1"
        set client-cert enable
        set log-blocked-traffic enable
        config api-gateway
            edit 1
                config realservers
                    edit 1
                        set ip 192.168.20.6
                    next
                end
            next
        end
    next
end
```

3. Configure ZTNA rules:

```
config firewall proxy-policy
    edit 3
        set name "ZTNA-Deny-malicious"
        set proxy access-proxy
        set access-proxy "WIN2K16-P1"
        set srcintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set ztna-ems-tag "FCTEMS0000109188_Malicious-File-Detected"
```

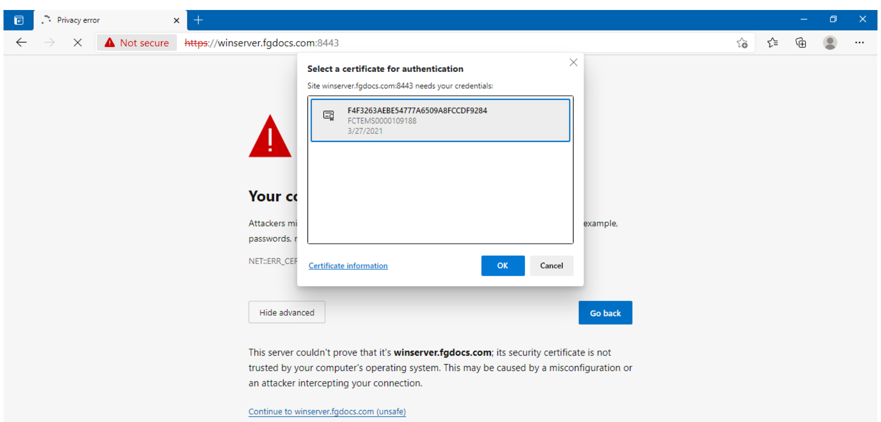
```
            set schedule "always"
            set logtraffic all
        next
        edit 2
            set name "proxy-WIN2K16-P1"
            set proxy access-proxy
            set access-proxy "WIN2K16-P1"
            set srcintf "port1"
            set srcaddr "all"
            set dstaddr "all"
            set ztna-ems-tag "FCTEMS0000109188_Low"
            set action accept
            set schedule "always"
            set logtraffic all
        next
    end
```

Testing the remote access to the HTTPS access proxy

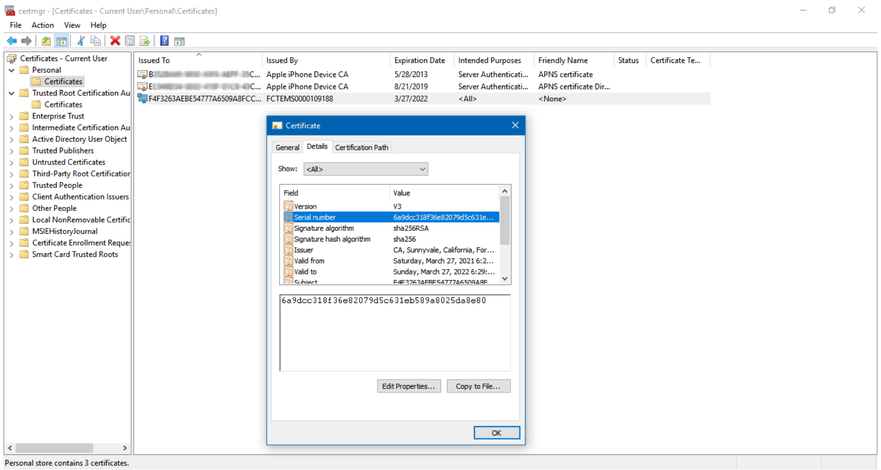
After FortiClient EMS and FortiGate are configured, the HTTPS access proxy remote connection can be tested.

Access allowed:

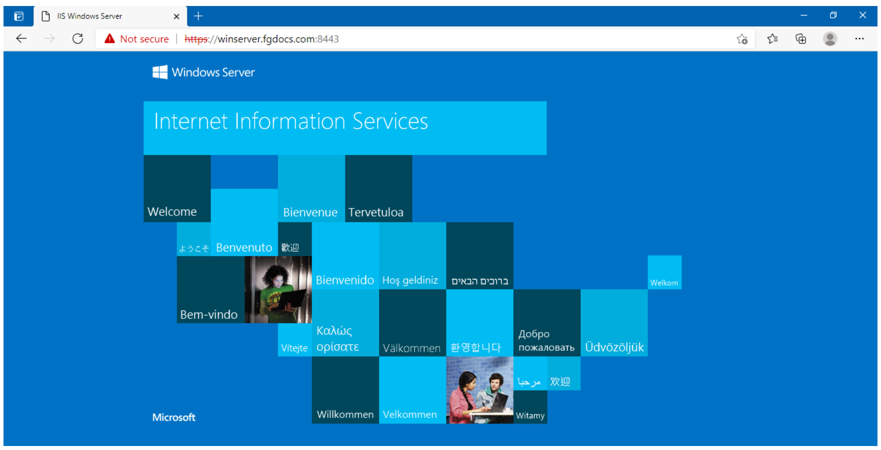
1. On the remote Windows PC, open FortiClient.
2. On the *Zero Trust Telemetry* tab, make sure that you are connected to the EMS server.
3. Open a browser and enter the address of the server and the access port. When entering the FQDN, make sure that the DNS can resolve the address to the IP address of the FortiGate. In this example, winserver.fgdocs.com resolves to 192.168.2.86.
4. The browser prompts for the client certificate to use. Select the EMS signed certificate, then click *OK*.



The certificate is in the *User Configuration* store, under *Personal* > *Certificates*. The details show the SN of the certificate, which matches the record on the FortiClient EMS and the FortiGate.

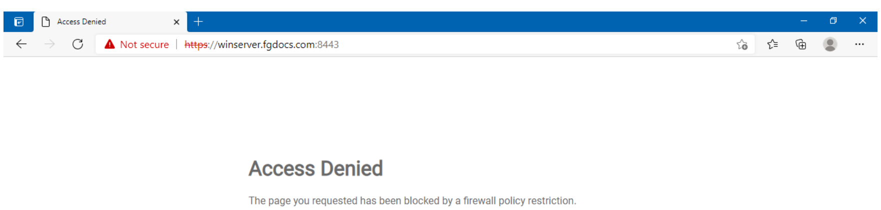


5. The client is verified by the FortiGate to authenticate your identity.
6. The FortiGate matches your security posture by verifying your ZTNA tag and matching the corresponding ZTNA rule, and you are allowed access to the web server.



Access denied:

1. On the remote Windows PC, trigger the Zero Trust Tagging Rule by creating the file in C:\virus.txt.
2. Open a browser and enter the address http://winserver.fgdocs.com:8443.
3. The client is verified by the FortiGate to authenticate your identity.
4. FortiGate checks your security posture. Because EMS has tagged the PC with the *Malicious-File-Detected* tag, it matches the *ZTNA-Deny-malicious* rule.
5. You are denied access to the web server.



Logs and debugs

Access allowed:

```
# diagnose endpoint record list
Record #1:
                IP Address = 10.10.10.20
                MAC Address = 9c:b7:0d:2d:5c:d1
                MAC list = 24:b6:fd:fa:54:c1;06:15:cd:45:f1:2e;9c:b7:0d:2d:5c:d1;
                VDOM =  (-1)
                EMS serial number: FCTEMS0000109188
                Client cert SN: 6A9DCC318F36E82079D5C631EB589A8025DA8E80
                Public IP address: 192.157.105.35
                Quarantined: no
                Online status: online
                Registration status: registered
                On-net status: on-net
                Gateway Interface:
                FortiClient version: 7.0.0
                AVDB version: 0.0
                FortiClient app signature version: 0.0
                FortiClient vulnerability scan engine version: 2.30
                FortiClient UID: F4F3263AEBE54777A6509A8FCCDF9284
                Host Name: Fortinet-KeithL
                OS Type: WIN64
….
                Number of Routes: (0)
online records: 1; offline records: 0; quarantined records: 0
```

```
# diagnose test application fcnacd 7
ZTNA Cache:
-uid F4F3263AEBE54777A6509A8FCCDF9284: { "tags": [ "all_registered_clients", "Low" ], "user_
name": "keithli", "client_cert_sn": "6A9DCC318F36E82079D5C631EB589A8025DA8E80", "ems_sn":
"FCTEMS0000109188" }
```

```
# diagnose endpoint lls-comm send ztna find-uid F4F3263AEBE54777A6509A8FCCDF9284
UID: F4F3263AEBE54777A6509A8FCCDF9284
        status code:ok
        Domain:
        User: keithli
        Cert SN:6A9DCC318F36E82079D5C631EB589A8025DA8E80
        EMS SN: FCTEMS0000109188
        Routes(0):
        Tags(2):
         - tag[0]: name=all_registered_clients
         - tag[1]: name=Low
```

```
# execute log display
1: date=2021-03-28 time=00:46:39 eventtime=1616917599923614599 tz="-0700" logid="0000000010"
type="traffic" subtype="forward" level="notice" vd="root" srcip=10.10.10.20 srcport=60185
srcintf="port1" srcintfrole="wan" dstcountry="Reserved" srccountry="Reserved"
dstip=192.168.20.6 dstport=443 dstintf="root" dstintfrole="undefined" sessionid=29515
srcuuid="2d8e1736-8ec6-51eb-885c-009bdf9c31d7" dstuuid="5445be2e-5d7b-51ea-e2c3-
ae6b7855c52f" service="HTTPS" wanoptapptype="web-proxy" proto=6 action="accept" policyid=2
policytype="proxy-policy" poluuid="5aba29de-8ec6-51eb-698f-25b59d5bf852" duration=6
wanin=104573 rcvdbyte=104573 wanout=2274 lanin=3370 sentbyte=3370 lanout=104445
```

```
srchwvendor="Fortinet" devtype="Network" srcfamily="Firewall" osname="Windows"
srchwversion="FortiWiFi-30E" appcat="unscanned"
```

Access denied:

```
# diagnose test application fcnacd 7
ZTNA Cache:
-uid F4F3263AEBE54777A6509A8FCCDF9284: { "user_name": "keithli", "client_cert_sn":
"6A9DCC318F36E82079D5C631EB589A8025DA8E80", "ems_sn": "FCTEMS0000109188", "tags": [
"Malicious-File-Detected", "all_registered_clients", "Low" ] }
```

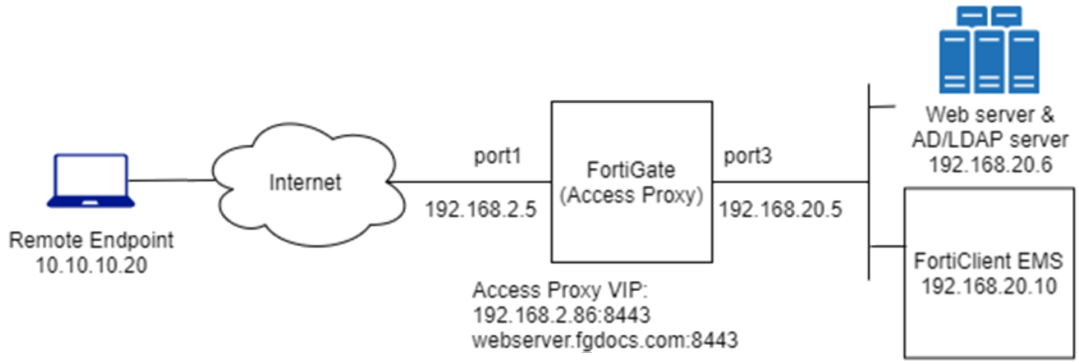
```
# diagnose endpoint lls-comm send ztna find-uid F4F3263AEBE54777A6509A8FCCDF9284
UID: F4F3263AEBE54777A6509A8FCCDF9284
        status code:ok
        Domain:
        User: keithli
        Cert SN:6A9DCC318F36E82079D5C631EB589A8025DA8E80
        EMS SN: FCTEMS0000109188
        Routes(0):
        Tags(3):
         - tag[0]: name=Malicious-File-Detected
         - tag[1]: name=all_registered_clients
         - tag[2]: name=Low
```

```
# execute log display
1: date=2021-03-28 time=01:21:55 eventtime=1616919715444980633 tz="-0700" logid="0000000013"
type="traffic" subtype="forward" level="notice" vd="root" srcip=10.10.10.20 srcport=60784
srcintf="port1" srcintfrole="wan" dstip=192.168.20.6 dstport=443 dstintf="root"
dstintfrole="undefined" srcuuid="2d8e1736-8ec6-51eb-885c-009bdf9c31d7" dstuuid="5445be2e-
5d7b-51ea-e2c3-ae6b7855c52f" srccountry="Reserved" dstcountry="Reserved" sessionid=33933
proto=6 action="deny" policyid=3 policytype="proxy-policy" poluuid="762ca074-8f9e-51eb-7614-
03a8801c6477" service="HTTPS" trandisp="noop" url="https://winserver.fgdocs.com/"
agent="Chrome/89.0.4389.90" duration=0 sentbyte=0 rcvdbyte=0 sentpkt=0 rcvdpkt=0
appcat="unscanned" crscore=30 craction=131072 crlevel="high" msg="Traffic denied because of
explicit proxy policy"
```

ZTNA HTTPS access proxy with basic authentication example

This example expands on the previous example (ZTNA HTTPS access proxy example on page 988), adding LDAP authentication to the ZTNA rule. Users are allowed based on passing the client certificate authentication check, user authentication, and security posture check.

Users that are in the AD security group *ALLOWED-VPN* are allowed access to the access proxy. Users that are not part of this security group are not allowed access.



This example assumes that the FortiGate EMS fabric connector is already successfully connected.

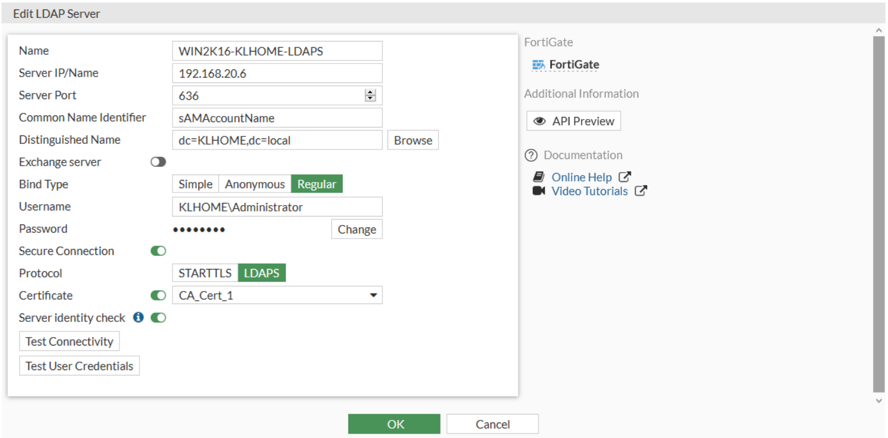
LDAP/Active Directory Users and Groups:

- Domain: KLHOME.local
- Users (Groups):
  - radCurtis (Domain Users, ALLOWED-VPN)
  - radKeith (Domain Users)

To configure a secure connection to the LDAP server in the GUI:

1. Go to *User & Authentication > LDAP Servers* and click *Create New*.
2. Configure the following settings:

Name	WIN2K16-KLHOME-LDAPS
Server IP/Name	192.168.20.6
Server Port	636
Common Name Identifier	sAMAccountName
Distinguished Name	dc=KLHOME,dc=local
Exchange server	Disabled
Bind Type	Regular Enter the *Username* and *Password* for LDAP binding and lookup.
Secure Connection	Enabled • Set *Protocol* to *LDAPS* • Enable *Certificate* and select the CA certificate to validate the server certificate.
Server identity check	Optionally, enable to verify the domain name or IP address against the server certificate.



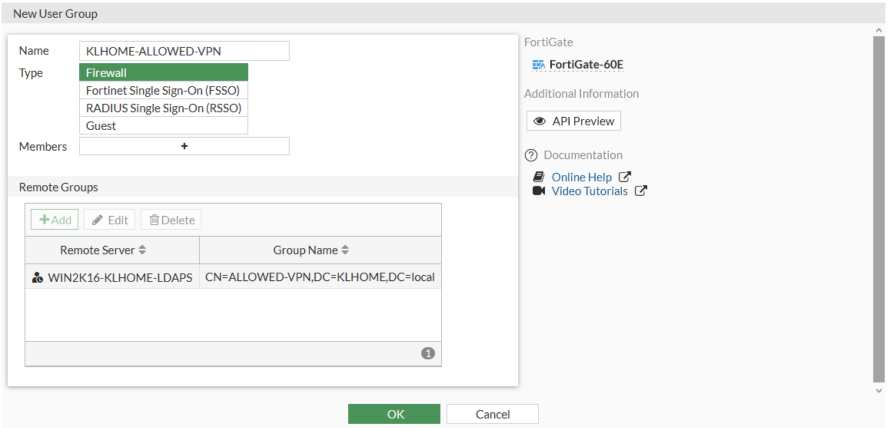
3. Click *Test Connectivity* to verify the connection to the server.
4. Click *OK*.

To configure a secure connection to the LDAP server in the CLI:

```
config user ldap
    edit "WIN2K16-KLHOME-LDAPS"
        set server "192.168.20.6"
        set cnid "sAMAccountName"
        set dn "dc=KLHOME,dc=local"
        set type regular
        set username "KLHOME\\Administrator"
        set password <password>
        set secure ldaps
        set ca-cert "CA_Cert_1"
        set port 636
    next
end
```

To configure a remote user group from the LDAP server in the GUI:

1. Go to *User & Authentication > User Groups* and click *Create New*.
2. Set the name to *KLHOME-ALLOWED-VPN*.
3. Set *Type* to *Firewall*.
4. In the *Remote Groups* table click *Add*:
   a. Set *Remote Server* to *WIN2K16-KLHOME-LDAPS*.
   b. Locate the *ALLOWED-VPN* group, right-click on it, and click *Add Selected*.
   c. Click *OK*.



5. Click *OK*.

To configure a remote user group from the LDAP server in the CLI:

```
config user group
    edit "KLHOME-ALLOWED-VPN"
        set member "WIN2K16-KLHOME-LDAPS"
        config match
            edit 1
                set server-name "WIN2K16-KLHOME-LDAPS"
                set group-name "CN=ALLOWED-VPN,DC=KLHOME,DC=local"
            next
        end
    next
end
```

Authentication scheme and rules

After the LDAP server and user group have been configured, an authentication scheme and rule must be configured.

To configure authentication schemes and rules in the GUI, go to *System > Feature Visibility* and enable *Explicit Proxy*.

Authentication scheme

The authentication scheme defines the method of authentication that is applied. In this example, basic HTTP authentication is used so that users are prompted for a username and password the first time that they connect to a website through the HTTPS access proxy.

To configure an authentication scheme in the GUI:

1. Go to *Policy & Objects > Authentication Rules* and click *Create New > Authentication Scheme*.
2. Set the name to *ZTNA-Auth-scheme*.
3. Set *Method* to *Basic*.
4. Set *User database* to *Other* and select *WIN2K16-KLHOME-LDAPS* as the LDAP server.
5. Click *OK*.

To configure an authentication scheme in the CLI:

```
config authentication scheme
    edit "ZTNA-Auth-scheme"
        set method basic
        set user-database "WIN2K16-KLHOME-LDAPS"
    next
end
```

Authentication rule

The authentication rule defines the proxy sources and destination that require authentication, and what authentication scheme is applied. In this example, active authentication through the basic HTTP prompt is used and applied to all sources.

To configure an authentication rule in the GUI:

1. Go to *Policy & Objects > Authentication Rules* and click *Create New > Authentication Rule*.
2. Set the name to *ZTNA-Auth-rule*.
3. Set *Source Address* to *all*.
4. Set *Protocol* to *HTTP*.
5. Enable *Authentication Scheme* and select *ZTNA-Auth-scheme*.
6. Click *OK*.

To configure an authentication rule in the CLI:

```
config authentication rule
    edit "ZTNA-Auth-rule"
        set srcaddr "all"
        set active-auth-method "ZTNA-Auth-scheme"
    next
end
```

Applying the user group to a ZTNA rule

A user or user group must be applied to the ZTNA rule that you need to control user access to. The authenticated user from the authentication scheme and rule must match the user or user group in the ZTNA rule.

In this example, the user group is applied to the two ZTNA rules that were configured in ZTNA HTTPS access proxy example on page 988.

To apply a user group to the ZTNA rules in the GUI:

1. Go to *Policy & Objects > ZTNA* and select the *ZTNA Rules* tab.
2. Edit the *ZTNA-Deny-malicious* rule.
3. Click in the *Source* field, select the *User* tab, select the *KLHOME-ALLOWED-VPN* group, then click *Close*.
4. Click *OK*.
5. Edit the *proxy-WIN2K16-P1* rule.
6. Click in the *Source* field, select the *User* tab, select the *KLHOME-ALLOWED-VPN* group, then click *Close*.
7. Click *OK*.

To apply a user group to the ZTNA rules in the CLI:

```
config firewall proxy-policy
    edit 3
        set name "ZTNA-Deny-malicious"
        set proxy access-proxy
        set access-proxy "WIN2K16-P1"
        set srcintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set ztna-ems-tag "FCTEMS0000109188_Malicious-File-Detected"
        set schedule "always"
        set logtraffic all
        set groups "KLHOME-ALLOWED-VPN"
    next
    edit 2
        set name "proxy-WIN2K16-P1"
        set proxy access-proxy
        set access-proxy "WIN2K16-P1"
        set srcintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set ztna-ems-tag "FCTEMS0000109188_Low"
        set action accept
        set schedule "always"
        set logtraffic all
        set groups "KLHOME-ALLOWED-VPN"
    next
end
```

Testing remote access to the HTTPS access proxy with user authentication

Scenario 1: access allowed - user radCurtis

1. On a remote Windows PC, open the FortiClient app, select the *Zero Trust Telemetry* tab, and confirm that you are connected to the EMS server.
2. In a browser, enter the address of the server and the access port.

   If entering an FQDN, make sure that DNS can resolve the address to the IP address of the FortiGate. In this example, *winserver.fgdocs.com* resolves to 192.168.2.86.
3. When the browser asks for the client certificate to use, select the EMS signed certificate, then click *OK*.

   The client certificate is verified by the FortiGate to authenticate your identity.
4. When prompted, enter the username *radCurtis* and the password, and click *Sign in*.

   As *radCurtis* is a member of the *ALLOWED-VPN* group in Active Directory, it will match the *KLHOME-ALLOWED-VPN* user group. After the user authentication passes, the FortiGate performs a posture check on the ZTNA group. When that passes, you are allowed access to the website.

Verifying the results

```
# diagnose firewall auth list

10.10.10.20, radCurtis
        type: fw, id: 0, duration: 13, idled: 13
```

```
        expire: 587, allow-idle: 600
        packets: in 0 out 0, bytes: in 0 out 0
        group_id: 8 16777220
        group_name: KLHOME-ALLOWED-VPN grp_16777220
```

```
# diagnose test application fcnacd 7
ZTNA Cache:
-uid F4F3263AEBE54777A6509A8FCCDF9284: { "tags": [ "all_registered_clients", "Low" ], "user_
name": "keith", "client_cert_sn": "6C7433E8E2CEDEB49B6C3C3C03677A3521EA4486", "ems_sn":
"FCTEMS0000109188" }
```

The `user_name` is the windows log in username learned by FortiClient. It might not match the username used in firewall user authentication.

```
# execute log display

1: date=2021-04-13 time=00:11:56 eventtime=1618297916023667886 tz="-0700" logid="0000000010"
type="traffic" subtype="forward" level="notice" vd="root" srcip=10.10.10.20 srcport=51513
srcintf="port1" srcintfrole="wan" dstcountry="Reserved" srccountry="Reserved"
dstip=192.168.20.6 dstport=443 dstintf="root" dstintfrole="undefined" sessionid=2319197
srcuuid="2d8e1736-8ec6-51eb-885c-009bdf9c31d7" dstuuid="5445be2e-5d7b-51ea-e2c3-
ae6b7855c52f" service="HTTPS" wanoptapptype="web-proxy" proto=6 action="accept" policyid=2
policytype="proxy-policy" poluuid="5aba29de-8ec6-51eb-698f-25b59d5bf852" duration=10
user="radCurtis" group="KLHOME-ALLOWED-VPN" authserver="WIN2K16-KLHOME-LDAPS" wanin=104573
rcvdbyte=104573 wanout=2364 lanin=3538 sentbyte=3538 lanout=104445 appcat="unscanned"
```

Scenario 2: access denied – user radKeith

1. If scenario 1 has just been tested, log in to the FortiGate and deauthenticate the user:
   a. Go to *Dashboard* > *Users & Devices* and expand the *Firewall Users* widget.
   b. Right-click on the user *radCurtis* and select deauthenticate.
2. On a remote Windows PC, open the FortiClient app, select the *Zero Trust Telemetry* tab, and confirm that you are connected to the EMS server.
3. In a browser, enter the address *winserver.fgdocs.com*.
4. When the browser asks for the client certificate to use, select the EMS signed certificate, then click *OK*. This option might not appear if you have already selected the certificate when testing scenario 1.
   The client certificate is verified by the FortiGate to authenticate your identity.
5. When prompted, enter the username *radKeith* and the password, and click *Sign in*.
   As *radKeith* is not a member of the *ALLOWED-VPN* group in Active Directory, it will not match the *KLHOME-ALLOWED-VPN* user group. Because no other policies are matched, this user is implicitly denied

Verifying the results

Go to *Dashboard* > *Users & Devices*, expand the *Firewall Users* widget, and confirm that user *radKeith* is listed, but no applicable user group is returned.

```
# execute log display

1: date=2021-04-13 time=12:29:21 eventtime=1618342161821542277 tz="-0700" logid="0000000013"
type="traffic" subtype="forward" level="notice" vd="root" srcip=10.10.10.20 srcport=52571
```

```
srcintf="port1" srcintfrole="wan" dstip=192.168.20.6 dstport=443 dstintf="root"
dstintfrole="undefined" srcuuid="5445be2e-5d7b-51ea-e2c3-ae6b7855c52f" srccountry="Reserved"
dstcountry="Reserved" sessionid=2394329 proto=6 action="deny" policyid=0 policytype="proxy-
policy" user="radKeith" authserver="WIN2K16-KLHOME-LDAPS" service="HTTPS" trandisp="noop"
url="https://winserver.fgdocs.com/" agent="Chrome/89.0.4389.114" duration=0 sentbyte=0
rcvdbyte=0 sentpkt=0 rcvdpkt=0 appcat="unscanned" crscore=30 craction=131072 crlevel="high"
msg="Traffic denied because of explicit proxy policy"
```

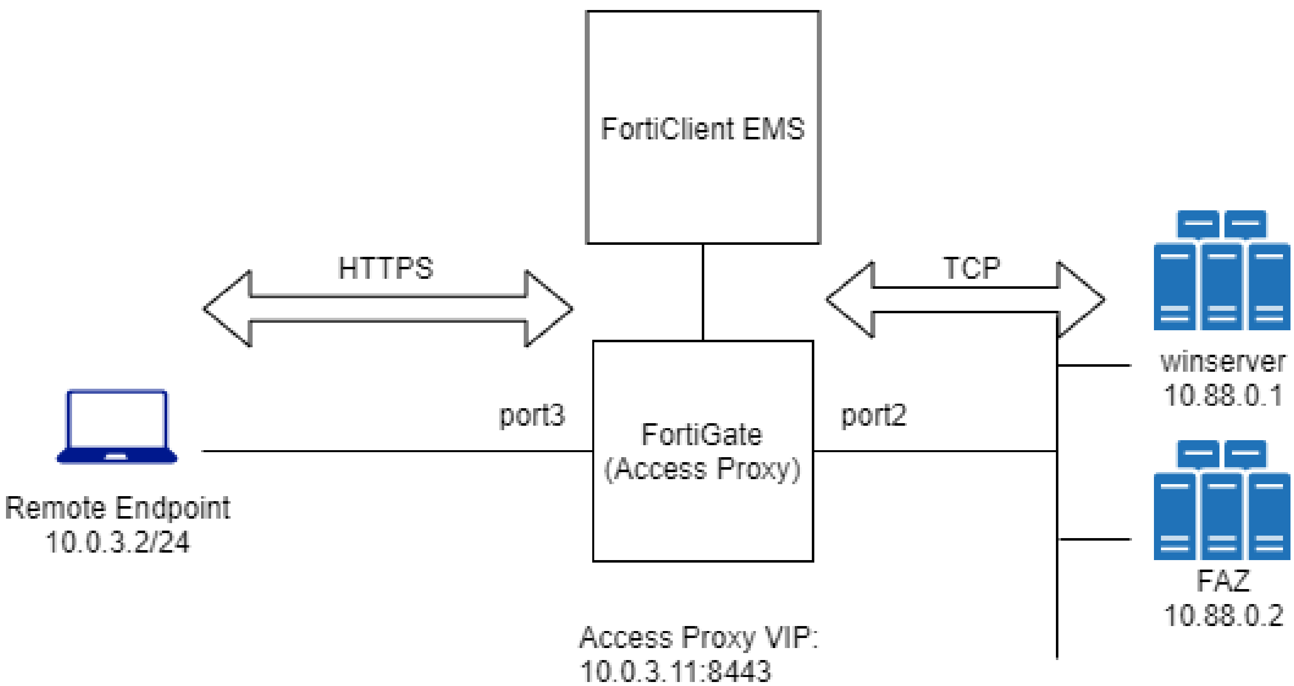
ZTNA TCP forwarding access proxy example

In this example, a TCP forwarding access proxy (TFAP) is configured to demonstrate an HTTPS reverse proxy that forwards TCP traffic to the designated resource. The access proxy tunnels TCP traffic between the client and the FortiGate over HTTPS, and forwards the TCP traffic to the protected resource. It verifies user identity, device identity, and trust context, before granting access to the protected source.

By default, encryption is disabled on FortiClient ZTNA rules, as this reduces overhead for end to end protocols that are already secure. For insecure end to end protocols, enable encryption.

RDP (Remote Desktop Protocol) and SMB (Server Message Block) protocol access are configured to one server, and SSH access to the other server.

FortiClient (Windows) must be running 7.0.3 or later to detect SMB.



This example assumes that the FortiGate EMS fabric connector is already successfully connected.

To configure the ZTNA server for TCP access proxy in the GUI:

1. Go to *Policy & Objects > ZTNA* and select the *ZTNA Servers* tab.
2. Click *Create New*.
3. Set *Name* to *ZTNA-tcp-server*.
4. Configure the network settings:
   a. Set *External interface* to *port3*.
   b. Set *External IP* to *10.0.3.11*.
   c. Set *External port* to *8443*.
5. Select the *Default certificate*. Clients will be presented with this certificate when they connect to the access proxy VIP.
6. Add server mapping:
   a. In the *Service/server mapping* table, click *Create New*.
   b. Set *Service* to *TCP Forwarding*.
   c. Add a server:
      i. In the *Servers* table, click *Create New*.
      ii. Create a new address for the FortiAnalyzer server at 10.88.0.2 and use it as the address.
      iii. Set *Port* to *22*.
      iv. Click *OK*.
   d. Add another server:
      i. In the *Servers* table, click *Create New*.
      ii. Create a new address for the winserver at 10.88.0.1 and use it as the address.
      iii. Set *Port* to *445, 3389* to correspond to SMB and RDP.
      iv. Click *OK*.
   e. Click *OK*.
7. Click *OK*.

To configure the ZTNA rule to allow traffic to the TCP access proxy in the GUI:

1. Go to *Policy & Objects > ZTNA*, select the *ZTNA Rules* tab, and click *Create New*.
2. Set *Name* to *ZTNA_remote*.
3. Set *Incoming Interface* to *port3*.
4. Set *Source* to *all*.
5. Select the ZTNA server *ZTNA-tcp-server*.
6. Configure the remaining options as needed.
7. Click *OK*.

To configure the access proxy VIP in the CLI:

```
config firewall vip
    edit "ZTNA-tcp-server"
        set type access-proxy
        set extip 10.0.3.11
        set extintf "port3"
        set server-type https
        set extport 8443
```

```
        set ssl-certificate "Fortinet_SSL"
    next
end
```

To configure the server addresses in the CLI:

```
config firewall address
    edit "FAZ"
        set subnet 10.88.0.2 255.255.255.255
    next
    edit "winserver"
        set subnet 10.88.0.1 255.255.255.255
    next
end
```

To configure access proxy server mappings in the CLI:

```
config firewall access-proxy
    edit "ZTNA-tcp-server"
        set vip "ZTNA-tcp-server"
        set client-cert enable
        config api-gateway
            edit 1
                set service tcp-forwarding
                config realservers
                    edit 1
                        set address "FAZ"
                        set mappedport 22
                    next
                    edit 2
                        set address "winserver"
                        set mappedport 445 3389
                    next
                end
            next
        end
    next
end
```

The mapped port (`mappedport`) restricts the mapping to the specified port or port range. If `mappedport` is not specified, then any port will be matched.

To configure a ZTNA rule (proxy policy) in the CLI:

```
config firewall proxy-policy
    edit 0
        set name "ZTNA_remote"
        set proxy access-proxy
        set access-proxy "ZTNA-tcp-server"
        set srcintf "port3"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set utm-status enable
        set ssl-ssh-profile "certificate-inspection"
```

```
        set logtraffic all
    next
end
```

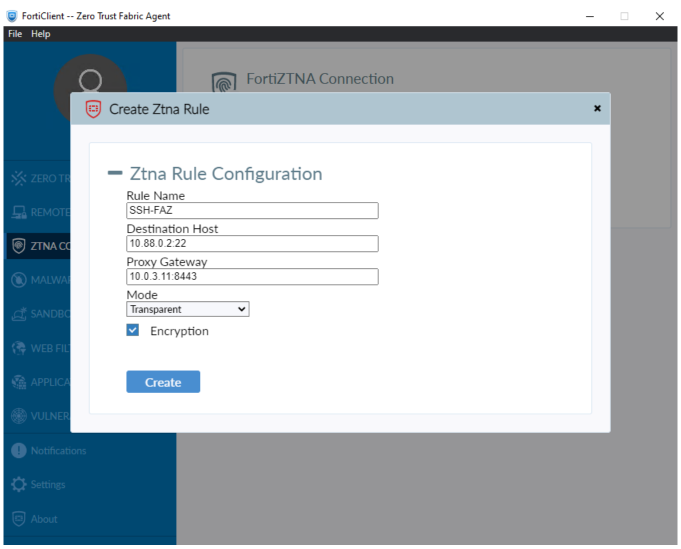
Test the connection to the access proxy

Before connecting, users must have a ZTNA connection rule in FortiClient.

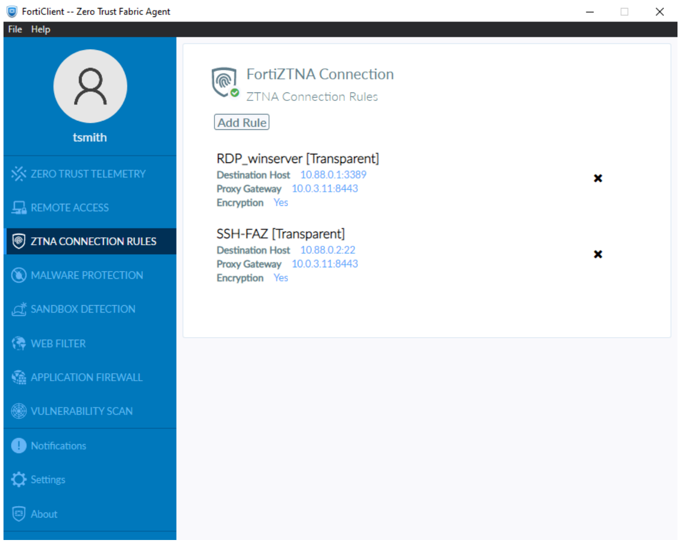
ZTNA TCP forwarding rules can be provisioned from the EMS server. See Provisioning ZTNA TCP forwarding rules via EMS for details.

To create a ZTNA rule in FortiClient:

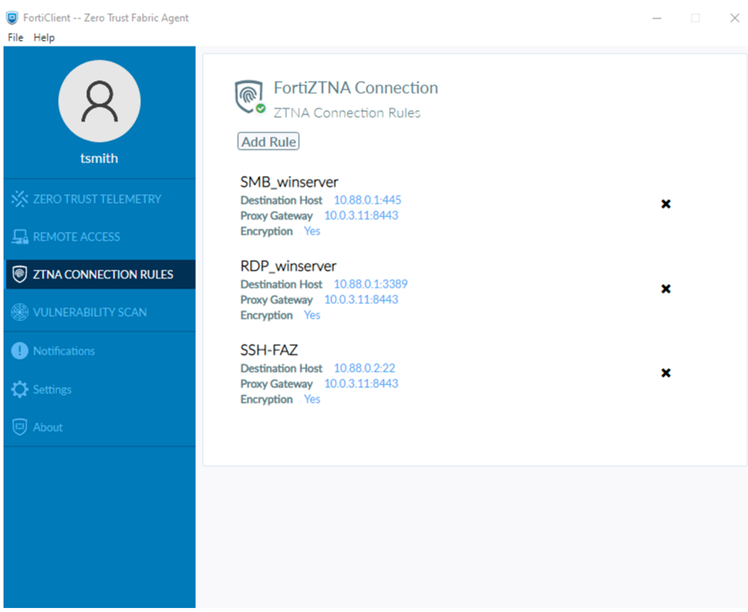
1. On the *ZTNA Connection Rules* tab, click *Add Rule*.
2. Set *Rule Name* to *SSH-FAZ*.
3. Set *Destination Host* to *10.88.0.2:22*. This is the real IP address and port of the server.
4. Set *Proxy Gateway* to *10.0.3.11:8443*. This is the access proxy address and port that are configured on the FortiGate.
5. Enable *Encryption*. This option determines whether or not the Client to FortiGate access proxy connection is encrypted in HTTPS.



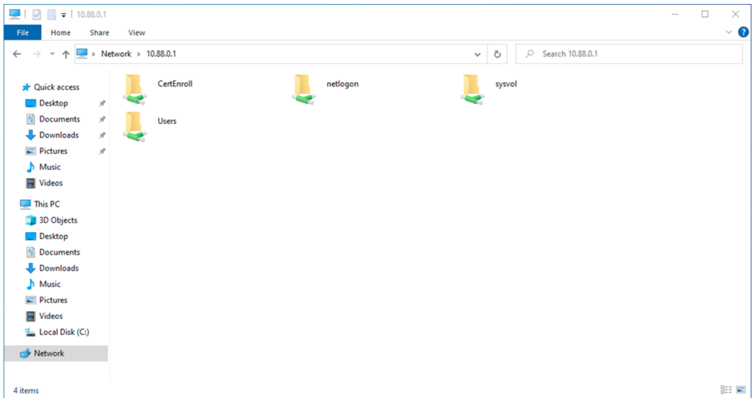
6. Click *Create*.
7. Create a second rule with the following settings:
   - *Rule Name*: *RDP_winserver*
   - *Destination Host*: *10.88.0.1:3389*
   - *Proxy Gateway*: *10.0.3.11:8443*
   - *Encryption*: Enabled



8. Create a third rule with the following settings:
   - *Rule Name*: *SMB_winserver*
   - *Destination Host*: *10.88.0.1:445*
   - *Proxy Gateway*: *10.0.3.11:8443*
   - *Encryption*: Enabled



After creating the ZTNA connection rules, you can SSH, RDP, and SMB directly to the server IP address and port.



Logs

```
# exec log filter category 0
# exec log filter field subtype ztna
# exec log display
```

SMB:

```
1: date=2022-05-19 time=23:32:15 eventtime=1653028335085873990 tz="-0700" logid="0005000024"
type="traffic" subtype="ztna" level="notice" vd="root" srcip=10.0.3.2 srcport=49823
srcintf="port3" srcintfrole="wan" dstcountry="Reserved" srccountry="Reserved"
dstip=10.88.0.1 dstport=445 dstintf="root" dstintfrole="undefined" sessionid=27520
service="SMB" proto=6 action="accept" policyid=2 policytype="proxy-policy"
poluuid="1c65cda2-d7d4-51ec-271f-deb3263318ad" policyname="ZTNA_remote" duration=11
gatewayid=1 vip="ZTNA-tcp-server" accessproxy="ZTNA-tcp-server"
clientdeviceid="9A016B5A6E914B42AD4168C066EB04CA" clientdevicetags="MAC_FCTEMS8822001975_
Low/FCTEMS8822001975_all_registered_clients/MAC_FCTEMS8822001975_all_registered_clients"
wanin=6018 rcvdbyte=6018 wanout=4541 lanin=6840 sentbyte=6840 lanout=10041
fctuid="9A016B5A6E914B42AD4168C066EB04CA" appcat="unscanned"
```

SSH:

```
2: date=2022-05-19 time=17:59:25 eventtime=1653008365264312800 tz="-0700" logid="0005000024"
type="traffic" subtype="ztna" level="notice" vd="root" srcip=10.0.3.2 srcport=52233
srcintf="port3" srcintfrole="wan" dstcountry="Reserved" srccountry="Reserved"
dstip=10.88.0.2 dstport=22 dstintf="root" dstintfrole="undefined" sessionid=18022
service="SSH" proto=6 action="accept" policyid=2 policytype="proxy-policy"
poluuid="1c65cda2-d7d4-51ec-271f-deb3263318ad" policyname="ZTNA_remote" duration=5
gatewayid=1 vip="ZTNA-tcp-server" accessproxy="ZTNA-tcp-server"
clientdeviceid="9A016B5A6E914B42AD4168C066EB04CA" clientdevicetags="MAC_FCTEMS8822001975_
Low/FCTEMS8822001975_all_registered_clients/MAC_FCTEMS8822001975_all_registered_clients"
wanin=3693 rcvdbyte=3693 wanout=2817 lanin=4915 sentbyte=4915 lanout=7586
fctuid="9A016B5A6E914B42AD4168C066EB04CA" appcat="unscanned"
```

RDP:

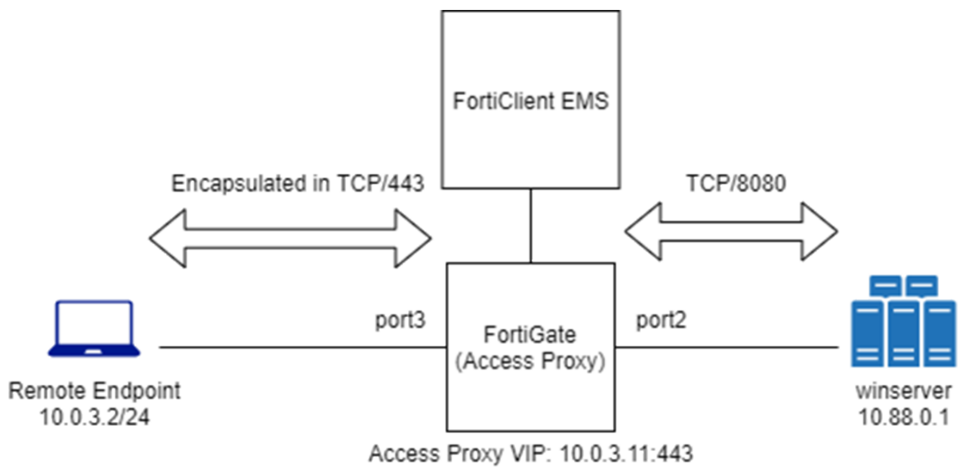
```
4: date=2022-05-19 time=17:56:23 eventtime=1653008183694431712 tz="-0700" logid="0005000024"
type="traffic" subtype="ztna" level="notice" vd="root" srcip=10.0.3.2 srcport=52213
srcintf="port3" srcintfrole="wan" dstcountry="Reserved" srccountry="Reserved"
dstip=10.88.0.1 dstport=3389 dstintf="root" dstintfrole="undefined" sessionid=17598
```

```
service="RDP" proto=6 action="accept" policyid=2 policytype="proxy-policy"
poluuid="1c65cda2-d7d4-51ec-271f-deb3263318ad" policyname="ZTNA_remote" duration=41
gatewayid=1 vip="ZTNA-tcp-server" accessproxy="ZTNA-tcp-server"
clientdeviceid="9A016B5A6E914B42AD4168C066EB04CA" clientdevicetags="MAC_FCTEMS8822001975_
Low/FCTEMS8822001975_all_registered_clients/MAC_FCTEMS8822001975_all_registered_clients"
wanin=178884 rcvdbyte=178884 wanout=33650 lanin=38920 sentbyte=38920 lanout=188121
fctuid="9A016B5A6E914B42AD4168C066EB04CA" appcat="unscanned"
```

ZTNA TCP forwarding access proxy without encryption example

TCP forwarding access proxy supports communication between the client and the access proxy without SSL/TLS encryption. The connection still begins with a TLS handshake. The client uses the HTTP 101 response to switch protocols and remove the HTTPS stack. Further end to end communication between the client and server are encapsulated in the specified TCP port, but not encrypted by the access proxy. This improves performance by reducing the overhead of encrypting an already secured underlying protocol, such as RDP, SSH, or FTPS. Users should still enable the encryption option for end to end protocols that are insecure.

In this example, the encryption option to access the web server on HTTP/8080 is disabled to show that traffic for an insecure connection protocol can be viewed in plain text in a protocol analyzer (such as Wireshark). In a real life application, the encryption option should be used for an insecure protocol.



To configure the ZTNA server for TCP access proxy in the GUI:

1. Go to *Policy & Objects > ZTNA* and select the *ZTNA Servers* tab.
2. Click *Create New*.
3. Set *Name* to *ZTNA-tcp-server*.
4. Configure the network settings:
    a. Set *External interface* to *port3*.
    b. Set *External IP* to *10.0.3.11*.
    c. Set *External port* to *8443*.
5. Select the *Default certificate*. Clients will be presented with this certificate when they connect to the access proxy VIP.
6. Add server mapping:
    a. In the *Service/server mapping* table, click *Create New*.
    b. Set *Service* to *TCP Forwarding*.

   c. Add a server:
      i. In the *Servers* table, click *Create New*.
      ii. Create a new address for the winserver at 10.88.0.1 and use it as the address.
      iii. Click *OK*.
   d. Click *OK*.
7. Click *OK*.

To configure the ZTNA rule to allow traffic to the TCP access proxy in the GUI:

1. Go to *Policy & Objects* > *ZTNA* and select the *ZTNA Rules* tab.
2. Click *Create New*.
3. Set *Name* to *ZTNA-TCP*.
4. Set *Incoming Interface* to *port3*.
5. Set *Source* to *all*.
6. Select the ZTNA server *ZTNA-tcp-server*.
7. Configure the remaining options as needed.
8. Click *OK*.

To configure the access proxy VIP in the CLI:

```
config firewall vip
    edit "ZTNA-tcp-server"
        set type access-proxy
        set extip 10.0.3.11
        set extintf "port3"
        set server-type https
        set extport 443
        set ssl-certificate "Fortinet_SSL"
    next
end
```

To configure the server addresses in the CLI:

```
config firewall address
    edit "winserver"
        set subnet 10.88.0.1 255.255.255.255
    next
end
```

To configure access proxy server mappings in the CLI:

```
config firewall access-proxy
    edit "ZTNA-tcp-server"
        set vip "ZTNA-tcp-server"
        set client-cert enable
        config api-gateway
            edit 1
                set service tcp-forwarding
                config realservers
                    edit 2
                        set address "winserver"
```

```
                    next
                end
            next
        end
    next
end
```

The mapped port (`mappedport`) is not specified so that it will map any ports that are defined in FortiClient's ZTNA connection rule.

To configure a ZTNA rule (proxy policy in the CLI):

```
config firewall proxy-policy
    edit 0
        set name "ZTNA-TCP"
        set proxy access-proxy
        set access-proxy "ZTNA-tcp-server"
        set srcintf "port3"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set logtraffic all
    next
end
```

Test the connection to the access proxy

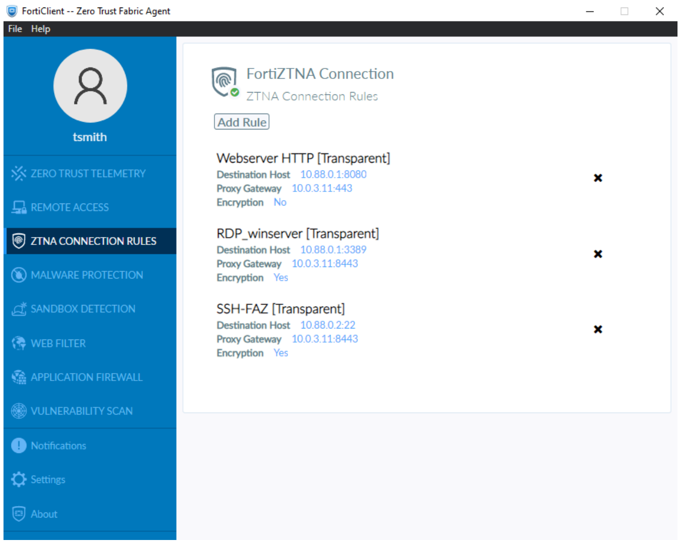
Before connecting, users must have a ZTNA connection rule in FortiClient.

ZTNA TCP forwarding rules can be provisioned from the EMS server. See Provisioning ZTNA TCP forwarding rules via EMS for details.

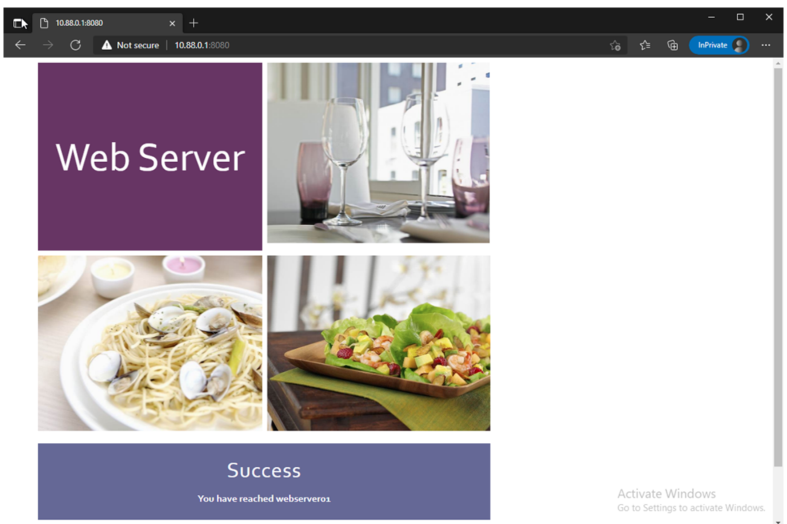
To create a ZTNA rule in FortiClient:

1. Go to the *ZTNA Connection Rules* tab and click *Add Rule*.
2. Set *Rule Name* to *Webserver HTTP*.
3. Set *Destination Host* to *10.88.0.1:8080*. This is the real IP address and port of the server.
4. Set *Proxy Gateway* to *10.0.3.11:443*. This is the access proxy address and port that are configured on the FortiGate.
5. Disable *Encryption*. This option determines whether or not the Client to FortiGate access proxy connection is encrypted in HTTPS.

6. Click *Create*.



After creating the ZTNA connection rule, open a browser and access the web page at *http://10.88.0.1:8080*.



Logs and debugs

1. The forward traffic log will show a log similar to this:

```
27: date=2021-07-13 time=13:05:00 eventtime=1626206700290129558 tz="-0700"
logid="0000000024" type="traffic" subtype="forward" level="notice" vd="root"
srcip=10.0.3.2 srcport=61409 srcintf="port3" srcintfrole="wan" dstcountry="Reserved"
srccountry="Reserved" dstip=10.88.0.1 dstport=8080 dstintf="root"
dstintfrole="undefined" sessionid=46959 service="tcp/8080" proto=6 action="accept"
policyid=3 policytype="proxy-policy" poluuid="fe0e1ae8-bdf9-51eb-b86f-c5e2adb934b3"
policyname="ZTNA-TCP" duration=114 wanin=38471 rcvdbyte=38471 wanout=775 lanin=2450
sentbyte=2450 lanout=40643 appcat="unscanned"
```

2. Use the following WAD debugs to can capture the details about the connection as seen by the FortiGate WAD daemon. Notice that the HTTP request has tls=0, indicating that the proxy connection between the client and access proxy is not encrypted.

```
# diagnose wad debug enable category all
```

```
# diagnose wad debug enable level verbose
```

```
# diagnose debug enable
```

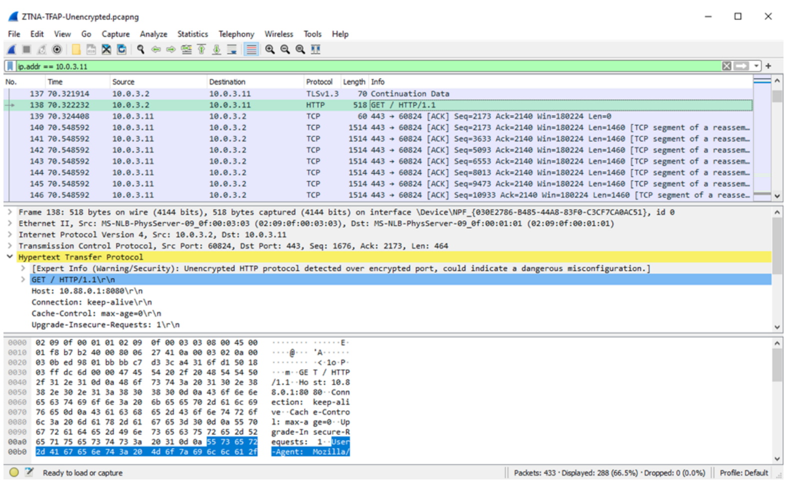
```
[I][p:224][s:46086][r:16777237] wad_dump_http_request              :2542
hreq=0x7f20bdaf5950 Received request from client: 10.0.3.2:62067

GET /tcp?address=10.88.0.1&port=8080&tls=0 HTTP/1.1
Host: 10.0.3.11:443
User-Agent: Forticlient
Accept: */*
Cookie:
Authorization: Basic
…
```

After reviewing the details, disable or reset the debugs:

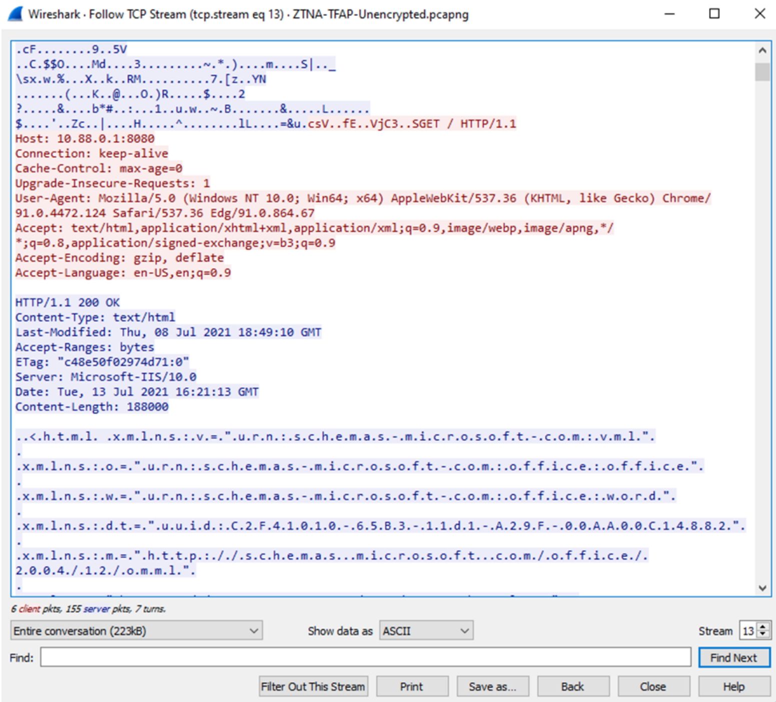
```
# diagnose debug reset
```

3. On the client PC, perform a packet capture to review the traffic flow between the client (10.0.3.2) and the access proxy (10.0.3.11) in detail. While the traffic is encapsulated in port 443, the underlying HTTP/8080 requests and traffic are decoded as clear text.

   Packet capture of traffic between 10.0.3.2:60824<->10.0.3.11:443:

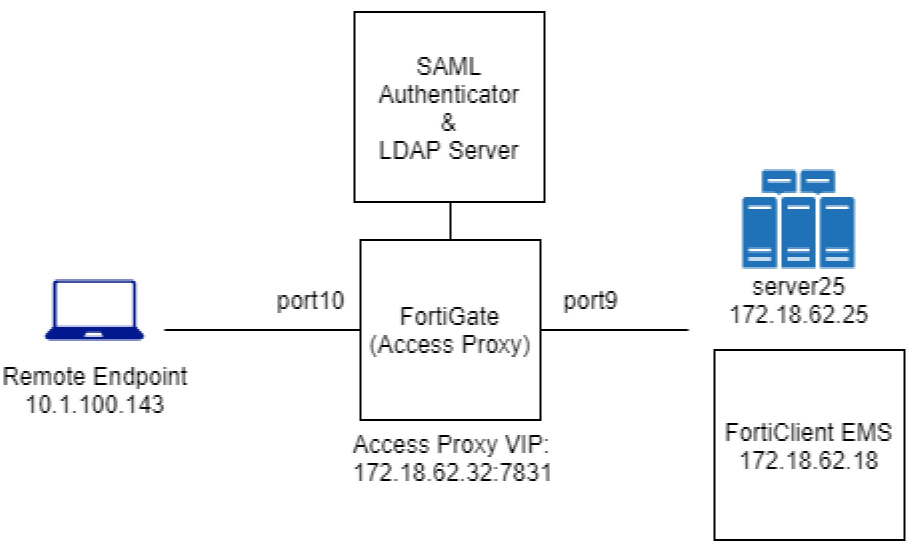


   Traffic stream:



ZTNA proxy access with SAML authentication example

In this example, an HTTPS access proxy is configured, and SAML authentication is applied to authenticate the client. The FortiGate acts as the SAML SP and a SAML authenticator serves as the IdP. In addition to verifying the user and device identity with the client certificate, the user is also authorized based on user credentials to establish a trust context before granting access to the protected resource.



This example assumes that the FortiGate EMS fabric connector is already successfully connected.

To configure the access proxy VIP:

```
config firewall vip
    edit "ZTNA_server01"
        set type access-proxy
        set extip 172.18.62.32
        set extintf "any"
        set server-type https
        set extport 7831
        set ssl-certificate "Fortinet_CA_SSL"
    next
end
```

To configure access proxy server mappings:

```
config firewall access-proxy
    edit "ZTNA_server01"
        set vip "ZTNA_server01"
        set client-cert enable
        config api-gateway
            edit 1
                set service https
                config realservers
                    edit 1
                        set ip 172.18.62.25
                        set port 443
                    next
                end
            next
        end
    next
end
```

To configure a SAML server:

```
config user saml
    edit "saml_ztna"
        set cert "Fortinet_CA_SSL"
        set entity-id "https://fgt9.myqalab.local:7831/samlap"
        set single-sign-on-url "https://fgt9.myqalab.local:7831/XX/YY/ZZ/saml/login/"
        set single-logout-url "https://fgt9.myqalab.local:7831/XX/YY/ZZ/saml/logout/"
        set idp-entity-id "http://MYQALAB.LOCAL/adfs/services/trust"
        set idp-single-sign-on-url "https://myqalab.local/adfs/ls"
        set idp-single-logout-url "https://myqalab.local/adfs/ls"
        set idp-cert "REMOTE_Cert_4"
        set digest-method sha256
        set adfs-claim enable
        set user-claim-type upn
        set group-claim-type group-sid
    next
end
```

To map the SAML server into an access proxy configuration:

```
config firewall access-proxy
    edit "ZTNA_server01"
        config api-gateway
             edit 3
                set service samlsp
                set saml-server "saml_ztna"
            next
        end
    next
end
```

To configure an LDAP server and an LDAP server group to verify user groups:

```
config user ldap
    edit "ldap-10.1.100.198"
        set server "10.1.100.198"
        set cnid "cn"
        set dn "dc=myqalab,dc=local"
        set type regular
        set username "cn=fosqa1,cn=users,dc=myqalab,dc=local"
        set password **********
        set group-search-base "dc=myqalab,dc=local"
    next
end
```

```
config user group
    edit "ldap-group-saml"
        set member "ldap-10.1.100.198"
    next
end
```

To configure the authentication rule and scheme to match the new SAML server:

```
config authentication rule
    edit "saml_ztna"
        set srcintf "port10"
        set srcaddr "all"
        set ip-based disable
        set active-auth-method "saml_ztna"
        set web-auth-cookie enable
    next
end
```

```
config authentication scheme
    edit "saml_ztna"
        set method saml
        set saml-server "saml_ztna"
        set saml-timeout 30
        set user-database "ldap-10.1.100.198"
    next
end
```

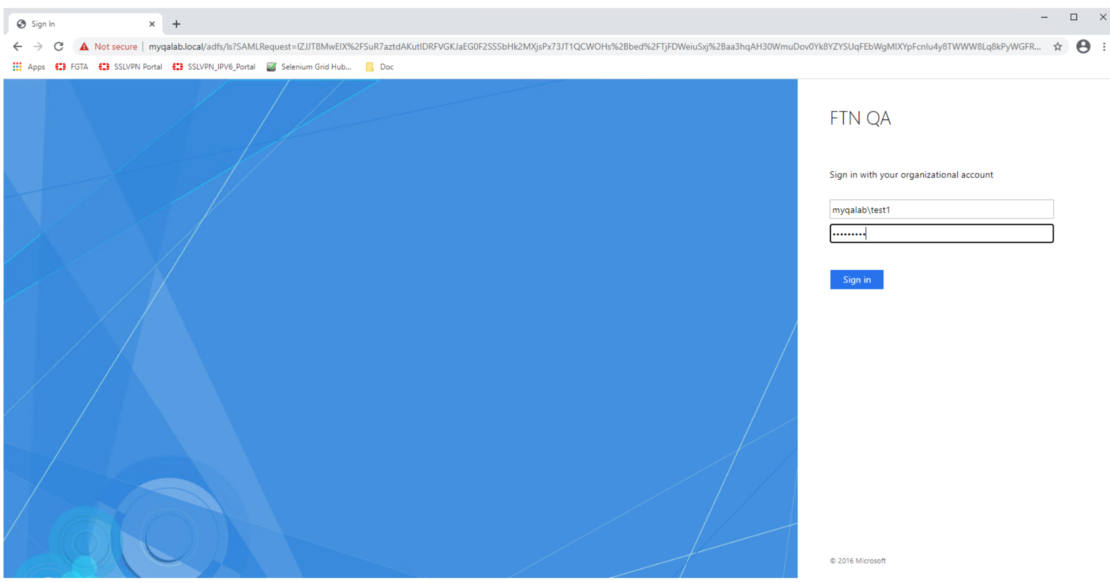
To enable user group authentication in an access-proxy type firewall proxy-policy:

```
config firewall proxy-policy
    edit 6
        set name "ZTNA_remote"
        set proxy access-proxy
        set access-proxy "ZTNA_server01"
        set srcintf "any"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set groups "ldap-group-saml"
        set utm-status enable
        set ssl-ssh-profile "certificate-inspection"
    next
end
```

Testing the connection

To test the connection:

1. On a client PC, try to access the webpage through the HTTPS access proxy. For example, go to http://172.18.62.32:7831 in a browser.
2. The client PC is prompted for a client certificate. After the certificate is validated, you are redirected to a SAML log in portal.



3. Enter your user credentials. The SAML server authenticates and sends a SAML assertion response message to the FortiGate.
4. The FortiGate queries the LDAP server for the user group, and then verifies the user group against the groups or groups defined in the proxy policy.
5. The user is proxied to the webpage on the real web server.

Logs and debugs

Use the following command to check the user information after the user has been authenticated:

```
# diagnose wad user list
ID: 7, VDOM: vdom1, IPv4: 10.1.100.143
  user name   : test1@MYQALAB.local
  worker      : 0
  duration    : 124
  auth_type   : Session
  auth_method : SAML
  pol_id      : 6
  g_id        : 13
  user_based  : 0
  expire      : no
  LAN:
    bytes_in=25953 bytes_out=14158
  WAN:
    bytes_in=8828 bytes_out=6830
```

Event log:

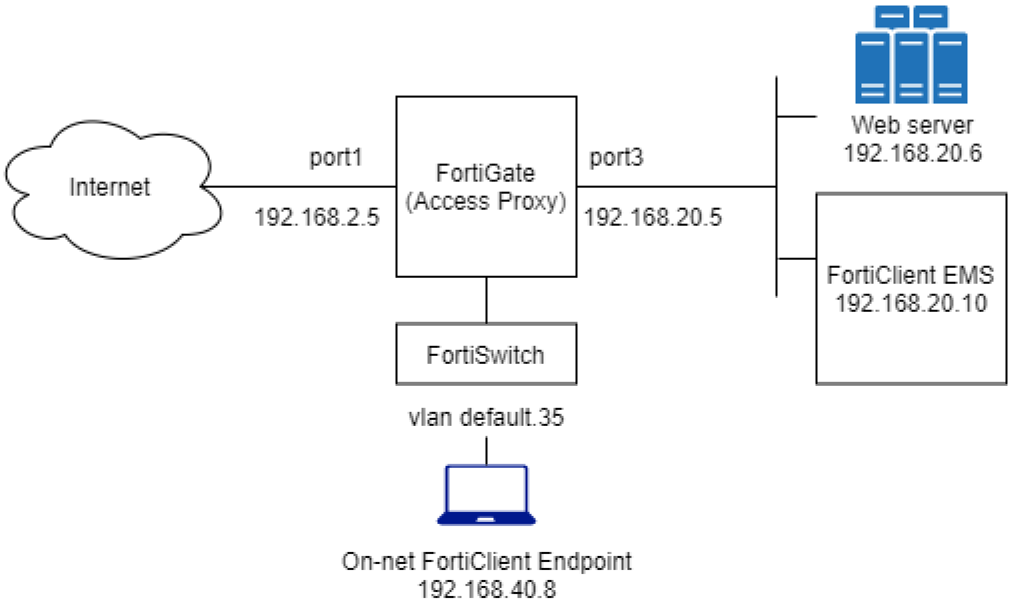
```
1: date=2021-03-24 time=19:02:21 eventtime=1616637742066893182 tz="-0700" logid="0102043025"
type="event" subtype="user" level="notice" vd="vdom1" logdesc="Explicit proxy authentication
successful" srcip=10.1.100.143 dstip=172.18.62.32 authid="saml" user="test1@MYQALAB.local"
group="N/A" authproto="HTTP(10.1.100.143)" action="authentication" status="success"
reason="Authentication succeeded" msg="User test1@MYQALAB.local succeeded in authentication"
```

Traffic log:

```
1: date=2021-03-24 time=19:09:06 eventtime=1616638146541253587 tz="-0700" logid="0000000010"
type="traffic" subtype="forward" level="notice" vd="vdom1" srcip=10.1.100.143 srcport=58084
srcintf="port10" srcintfrole="undefined" dstcountry="Reserved" srccountry="Reserved"
dstip=172.18.62.25 dstport=443 dstintf="vdom1" dstintfrole="undefined" sessionid=8028
service="HTTPS" wanoptapptype="web-proxy" proto=6 action="accept" policyid=6
policytype="proxy-policy" poluuid="8dcfe762-8d0b-51eb-82bf-bfbee59b89f2" duration=8
user="test1@MYQALAB.local" group="ldap-group-saml" authserver="ldap-10.1.100.198"
wanin=10268 rcvdbyte=10268 wanout=6723 lanin=7873 sentbyte=7873 lanout=10555
appcat="unscanned"
```

ZTNA IP MAC based access control example

In this example, firewall policies are configured that use ZTNA tags to control access between on-net devices and an internal web server. This mode does not require the use of the access proxy, and only uses ZTNA tags for access control. Traffic is passed when the FortiClient endpoint is tagged as *Low* risk only. Traffic is denied when the FortiClient endpoint is tagged with *Malicious-File-Detected*.

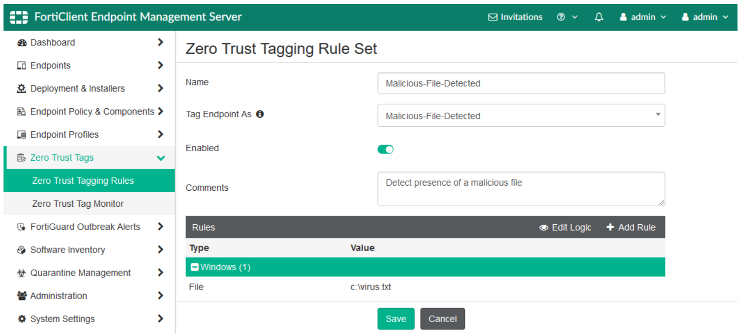


This example assumes that the FortiGate EMS fabric connector is already successfully connected.

To configure ZTNA in the GUI, go to *System > Feature Visibility* and enable *Zero Trust Network Access*.

To configure a Zero Trust tagging rule on the FortiClient EMS:

1. Log in to the FortiClient EMS.
2. Go to *Zero Trust Tags > Zero Trust Tagging Rules*, and click *Add*.
3. In the *Name* field, enter *Malicious-File-Detected*.
4. In the *Tag Endpoint As* dropdown list, select *Malicious-File-Detected*.
   EMS uses this tag to dynamically group together endpoints that satisfy the rule, as well as any other rules that are configured to use this tag.
5. Click *Add Rule* then configure the rule:
   a. For *OS*, select *Windows*.
   b. From the *Rule Type* dropdown list, select *File* and click the + button.
   c. Enter a file name, such as *C:\virus.txt*.
   d. Click *Save*.



6. Click *Save*.

To configure a firewall policy with IP/MAC based access control to deny traffic in the GUI:

1. Go to *Policy & Objects > Firewall Policy* and click *Create New*.
2. Set *Name* to *block-internal-malicious-access*.
3. Set *Incoming Interface* to *default.35*.
4. Set *Outgoing Interface* to *port3*.
5. Set *Source* to *all*.
6. Set *IP/MAC Based Access Control* to the *Malicious-File-Detected* tag.
7. Set *Destination* to *all*.
8. Set *Service* to *ALL*.
9. Set *Action* to *DENY*.
10. Enable *Log Violation Traffic*.
11. Configuring the remaining settings as needed.
12. Click *OK*.

To configure a firewall policy with IP/MAC based access control to allow access in the GUI:

1. Go to *Policy & Objects > Firewall Policy* and click *Create New*.
2. Set *Name* to *allow-internal-access*.
3. Set *Incoming Interface* to *default.35*.
4. Set *Outgoing Interface* to *port3*.
5. Set *Source* to *all*.
6. Set *IP/MAC Based Access Control* to the *Low* tag.
7. Set *Destination* to *all*.
8. Set *Service* to *ALL*.
9. Set *Action* to *ACCEPT*.
10. Enable *Log Allowed Traffic* and set it to *All Sessions*.
11. Configuring the remaining settings as needed.
12. Click *OK*.

To configure firewall policies with IP/MAC based access control to block and allow access in the CLI:

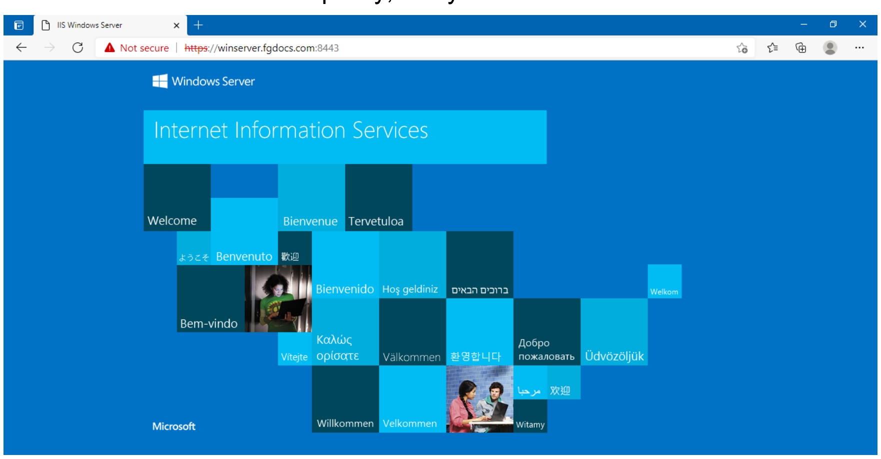
```
config firewall policy
    edit 29
```

```
        set name "block-internal-malicious-access"
        set srcintf "default.35"
        set dstintf "port3"
        set srcaddr "all"
        set dstaddr "all"
        set ztna-status enable
        set ztna-ems-tag "FCTEMS0000109188_Malicious-File-Detected"
        set schedule "always"
        set service "ALL"
        set logtraffic all
    next
    edit 30
        set name "allow-internal-access"
        set srcintf "default.35"
        set dstintf "port3"
        set srcaddr "all"
        set dstaddr "all"
        set ztna-status enable
        set ztna-ems-tag "FCTEMS0000109188_Low"
        set action accept
        set schedule "always"
        set service "ALL"
        set inspection-mode proxy
        set logtraffic all
        set nat enable
    next
end
```

Testing the access to the web server from the on-net client endpoint

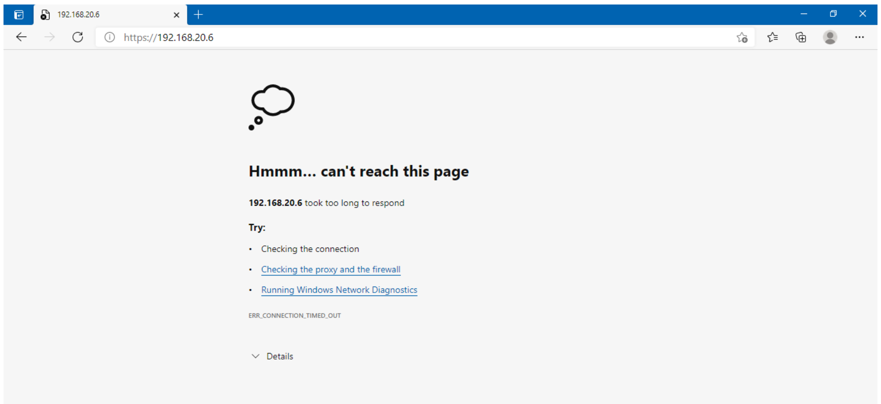
Access allowed:

1. On the remote Windows PC, open FortiClient.
2. On the *Zero Trust Telemetry* tab, make sure that you are connected to the EMS server.
3. Open a browser and enter the address of the server.
4. The FortiGate matches your security posture by verifying your ZTNA tag and matching the corresponding `allow-internal-access` firewall policy, and you are allowed access to the web server.



Access denied:

1. On the remote Windows PC, trigger the Zero Trust Tagging Rule by creating the file in C:\virus.txt.
2. Open a browser and enter the address of the server.
3. FortiGate checks your security posture. Because EMS has tagged the PC with the *Malicious-File-Detected* tag, it matches the *block-internal-malicious-access* firewall policy.
4. You are denied access to the web server.



Logs and debugs

Access allowed:

```
# diagnose endpoint record list
Record #1:
                IP Address = 192.168.40.8
                MAC Address = 24:b6:fd:fa:54:c1
                MAC list = 24:b6:fd:fa:54:c1;54:15:cd:3f:f8:30;9c:b7:0d:2d:5c:d1;
                VDOM = root (0)
                EMS serial number: FCTEMS0000109188
                Client cert SN: 563DA313367608678A3633E93C574F6F8BCB4A95
                Public IP address: 192.157.105.35
                Quarantined: no
                Online status: online
                Registration status: registered
                On-net status: on-net
                Gateway Interface: default.35
                FortiClient version: 7.0.0
                AVDB version: 0.0
                FortiClient app signature version: 0.0
                FortiClient vulnerability scan engine version: 2.30
                FortiClient UID: F4F3263AEBE54777A6509A8FCCDF9284
                ….
                Number of Routes: (1)
                        Gateway Route #0:
                                - IP:192.168.40.8, MAC: 24:b6:fd:fa:54:c1, Indirect: no
                                - Interface:default.35, VFID:0, SN: FGVM04TM21000144
online records: 1; offline records: 0; quarantined records: 0
```

```
# diagnose endpoint lls-comm send ztna find-ip-vdom 192.168.40.8 root
UID: F4F3263AEBE54777A6509A8FCCDF9284
        status code:ok
```

```
        Domain:
        User: keithli
        Cert SN:563DA313367608678A3633E93C574F6F8BCB4A95
        EMS SN: FCTEMS0000109188
        Routes(1):
         - route[0]: IP=192.168.40.8, VDom=root
        Tags(2):
         - tag[0]: name=all_registered_clients
         - tag[1]: name=Low
```

```
# diagnose firewall dynamic list
List all dynamic addresses:
FCTEMS0000109188_all_registered_clients: ID(51)
        ADDR(172.17.194.209)
        ADDR(192.168.40.8)
…

FCTEMS0000109188_Low: ID(78)
        ADDR(172.17.194.209)
        ADDR(192.168.40.8)
…

FCTEMS0000109188_Malicious-File-Detected: ID(190)
…
```

```
# diagnose test application fcnacd 7
ZTNA Cache:
-uid F4F3263AEBE54777A6509A8FCCDF9284: { "tags": [ "all_registered_clients", "Low" ], "user_
name": "keithli", "client_cert_sn": "563DA313367608678A3633E93C574F6F8BCB4A95", "gateway_
route_list": [ { "gateway_info": { "fgt_sn": "FGVM04TM21000144", "interface": "default.35",
"vdom": "root" }, "route_info": [ { "ip": "192.168.40.8", "mac": "24-b6-fd-fa-54-c1",
"route_type": "direct" } ] } ], "ems_sn": "FCTEMS0000109188" }
```

```
# execute log display
49 logs found.
10 logs returned.
3.5% of logs has been searched.
38: date=2021-03-28 time=23:07:38 eventtime=1616998058790134389 tz="-0700"
logid="0000000013" type="traffic" subtype="forward" level="notice" vd="root"
srcip=192.168.40.8 srcname="Fortinet-KeithL" srcport=51056 srcintf="default.35"
srcintfrole="undefined" dstip=192.168.20.6 dstport=443 dstintf="port3"
dstintfrole="undefined" srcuuid="5445be2e-5d7b-51ea-e2c3-ae6b7855c52f" dstuuid="5445be2e-
5d7b-51ea-e2c3-ae6b7855c52f" srccountry="Reserved" dstcountry="Reserved" sessionid=161585
proto=6 action="close" policyid=30 policytype="policy" poluuid="8f6ea492-9034-51eb-f197-
c00d803b7489" policyname="allow-internal-access" service="HTTPS" trandisp="snat"
transip=192.168.20.5 transport=51056 duration=2 sentbyte=3374 rcvdbyte=107732 sentpkt=50
rcvdpkt=80 fctuid="F4F3263AEBE54777A6509A8FCCDF9284" unauthuser="keithli"
unauthusersource="forticlient" appcat="unscanned" mastersrcmac="24:b6:fd:fa:54:c1"
srcmac="24:b6:fd:fa:54:c1" srcserver=0 dstosname="Windows" dstswversion="10"
masterdstmac="52:54:00:e3:4c:1a" dstmac="52:54:00:e3:4c:1a" dstserver=0
```

Access denied:

```
# diagnose endpoint lls-comm send ztna find-ip-vdom 192.168.40.8 root
UID: F4F3263AEBE54777A6509A8FCCDF9284
        status code:ok
        Domain:
```

```
        User: keithli
        Cert SN:563DA313367608678A3633E93C574F6F8BCB4A95
        EMS SN: FCTEMS0000109188
        Routes(1):
         - route[0]: IP=192.168.40.8, VDom=root
        Tags(3):
         - tag[0]: name=Malicious-File-Detected
         - tag[1]: name=all_registered_clients
         - tag[2]: name=Low
```

```
# diagnose firewall dynamic list
List all dynamic addresses:
FCTEMS0000109188_all_registered_clients: ID(51)
        ADDR(172.17.194.209)
        ADDR(192.168.40.8)
…
FCTEMS0000109188_Low: ID(78)
        ADDR(172.17.194.209)
        ADDR(192.168.40.8)
…
FCTEMS0000109188_Malicious-File-Detected: ID(190)
        ADDR(172.17.194.209)
        ADDR(192.168.40.8)
…
```

```
# diagnose test application fcnacd 7
ZTNA Cache:
-uid F4F3263AEBE54777A6509A8FCCDF9284: { "user_name": "keithli", "client_cert_sn":
"563DA313367608678A3633E93C574F6F8BCB4A95", "gateway_route_list": [ { "gateway_info": {
"fgt_sn": "FGVM04TM21000144", "interface": "default.35", "vdom": "root" }, "route_info": [ {
"ip": "192.168.40.8", "mac": "24-b6-fd-fa-54-c1", "route_type": "direct" } ] } ], "ems_sn":
"FCTEMS0000109188", "tags": [ "Malicious-File-Detected", "all_registered_clients", "Low" ] }
```

```
# execute log display
49 logs found.
10 logs returned.
3.5% of logs has been searched.

11: date=2021-03-28 time=23:14:41 eventtime=1616998481409744928 tz="-0700"
logid="0000000013" type="traffic" subtype="forward" level="notice" vd="root"
srcip=192.168.40.8 srcname="Fortinet-KeithL" srcport=51140 srcintf="default.35"
srcintfrole="undefined" dstip=192.168.20.6 dstport=443 dstintf="port3"
dstintfrole="undefined" srcuuid="5445be2e-5d7b-51ea-e2c3-ae6b7855c52f" dstuuid="5445be2e-
5d7b-51ea-e2c3-ae6b7855c52f" srccountry="Reserved" dstcountry="Reserved" sessionid=162808
proto=6 action="deny" policyid=29 policytype="policy" poluuid="2835666c-9034-51eb-135d-
2f56e5f0f7a2" policyname="block-internal-malicious-access" service="HTTPS" trandisp="noop"
duration=0 sentbyte=0 rcvdbyte=0 sentpkt=0 fctuid="F4F3263AEBE54777A6509A8FCCDF9284"
unauthuser="keithli" unauthusersource="forticlient" appcat="unscanned" crscore=30
craction=131072 crlevel="high" mastersrcmac="24:b6:fd:fa:54:c1" srcmac="24:b6:fd:fa:54:c1"
srcserver=0
```

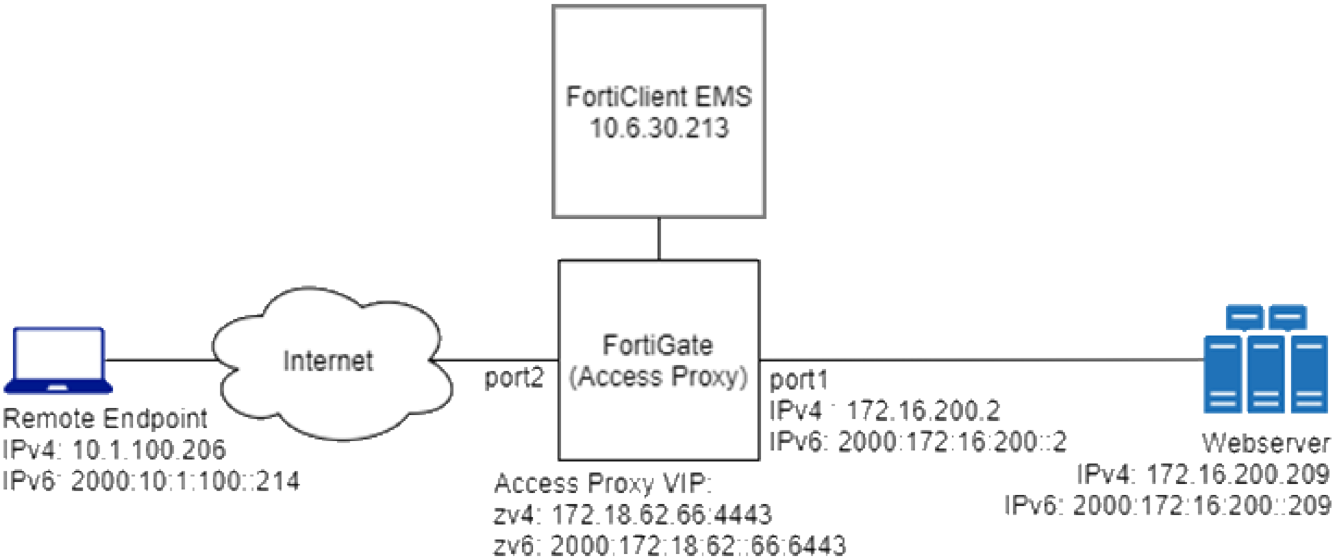
ZTNA IPv6 examples

IPv6 can be configured in ZTNA in several scenarios:

- IPv6 Client — IPv6 Access Proxy — IPv6 Server
- IPv6 Client — IPv6 Access Proxy — IPv4 Server

- IPv4 Client — IPv4 Access Proxy — IPv6 Server

These examples show the basic configuration for each scenario. It is assumed that the EMS fabric connector is already successfully connected.



Example 1: IPv6 Client — IPv6 Access Proxy — IPv6 Server

To configure the FortiGate:

1. Configure the IPv6 access proxy VIP:

```
config firewall vip6
    edit "zv6"
        set type access-proxy
        set extip 2000:172:18:62::66
        set server-type https
        set extport 6443
        set ssl-certificate "cert"
    next
end
```

2. Configure a virtual host:

```
config firewall access-proxy-virtual-host
    edit "vhost_ipv6"
        set ssl-certificate "cert"
        set host "qa6.test.com"
    next
end
```

   The client uses this address to connect to the access proxy.

3. Configure an IPv6 access proxy and IPv6 api-gateway, apply the VIP6 and virtual host to it, and assign an IPv6 address to the realserver:

```
config firewall access-proxy6
    edit "zs6"
        set vip "zv6"
```

```
                config api-gateway6
                    edit 1
                        set virtual-host "vhost_ipv6"
                        config realservers
                            edit 1
                                set ip 2000:172:16:200::209
                            next
                        end
                    next
                end
            next
        end
```

4. Apply the IPv6 access proxy to a proxy policy:

```
config firewall proxy-policy
    edit 1
        set name "ztna_rule"
        set proxy access-proxy
        set access-proxy6 "zs6"
        set srcintf "port2"
        set action accept
        set schedule "always"
        set logtraffic all
        set srcaddr6 "all"
        set dstaddr6 "all"
        set utm-status enable
        set ssl-ssh-profile "custom-deep-inspection"
        set webfilter-profile "monitor-all"
    next
end
```

5. Apply the IPv6 VIP to a firewall policy:

```
config firewall policy
    edit 4
        set name "ZTNA"
        set srcintf "port2"
        set dstintf "any"
        set action accept
        set srcaddr6 "all"
        set dstaddr6 "zv6"
        set schedule "always"
        set service "ALL"
        set inspection-mode proxy
        set logtraffic all
        set nat enable
    next
end
```

To test the configuration:

1. On an IPv6 client, ensure that the address qa6.test.com resolves to the IPv6 VIP address of 2000:172:18:62::66.
2. In a browser, connect to https://qa6.test.com:6443.
3. After device certificate verification, the browser will open up the webpage on the IPv6 real server.
4. In the Forward Traffic Log, the following log is available:

```
3: date=2021-06-25 time=13:38:18 eventtime=1624653498459580215 tz="-0700"
logid="0000000024" type="traffic" subtype="forward" level="notice" vd="root"
srcip=2000:10:1:100::214 srcport=55957 srcintf="port2" srcintfrole="undefined"
dstcountry="Reserved" srccountry="Reserved" dstip=2000:172:16:200::209 dstport=443
dstintf="root" dstintfrole="undefined" sessionid=92406 service="HTTPS" proto=6
action="accept" policyid=1 policytype="proxy-policy" poluuid="7afdac8c-d5db-51eb-dfc6-
67bb86e4bdcf" policyname="ztna_rule" duration=5 wanin=2031 rcvdbyte=2031 wanout=1332
lanin=1247 sentbyte=1247 lanout=950 appcat="unscanned" utmaction="allow" countweb=1
utmref=65445-0
```

Example 2: IPv6 Client — IPv6 Access Proxy — IPv4 Server

To configure the FortiGate:

1. Configure the IPv6 access proxy VIP:

```
config firewall vip6
    edit "zv6"
        set type access-proxy
        set extip 2000:172:18:62::66
        set server-type https
        set extport 6443
        set ssl-certificate "cert"
    next
end
```

2. Configure a virtual host:

```
config firewall access-proxy-virtual-host
    edit "vhost_ipv6"
        set ssl-certificate "cert"
        set host "qa6.test.com"
    next
end
```

The client uses this address to connect to the access proxy.

3. Configure an IPv6 access proxy and IPv6 api-gateway, apply the VIP6 and virtual host to it, and assign an IPv4 address to the realserver:

```
config firewall access-proxy6
    edit "zs6"
        set vip "zv6"
        config api-gateway6
            edit 1
                set virtual-host "vhost_ipv6"
                config realservers
                    edit 1
                        set ip 172.16.200.209
                    next
                end
            next
        end
    next
end
```

4. Apply the IPv6 access proxy to a proxy policy:

```
config firewall proxy-policy
    edit 1
        set name "ztna_rule"
        set proxy access-proxy
        set access-proxy6 "zs6"
        set srcintf "port2"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set logtraffic all
        set srcaddr6 "all"
        set dstaddr6 "all"
        set utm-status enable
        set ssl-ssh-profile "custom-deep-inspection"
        set webfilter-profile "monitor-all"
    next
end
```

5. Apply the IPv6 VIP to a firewall policy:

```
config firewall policy
    edit 4
        set name "ZTNA"
        set srcintf "port2"
        set dstintf "any"
        set action accept
        set srcaddr6 "all"
        set dstaddr6 "zv6"
        set schedule "always"
        set service "ALL"
        set inspection-mode proxy
        set logtraffic all
        set nat enable
    next
end
```

To test the configuration:

1. On an IPv6 client, ensure that the address qa6.test.com resolves to the IPv6 VIP address of 2000:172:18:62::66.
2. In a browser, connect to https://qa6.test.com:6443.
3. After device certificate verification, the browser will open up the webpage on the IPv4 real server.
4. In the Forward Traffic Log, the following log is available:

```
2: date=2021-06-25 time=13:46:54 eventtime=1624654014129553521 tz="-0700"
logid="0000000024" type="traffic" subtype="forward" level="notice" vd="root"
srcip=2000:10:1:100::214 srcport=60530 srcintf="port2" srcintfrole="undefined"
dstcountry="Reserved" srccountry="Reserved" dstip=172.16.200.209 dstport=443
dstintf="root" dstintfrole="undefined" sessionid=219 service="HTTPS" proto=6
action="accept" policyid=1 policytype="proxy-policy" poluuid="7afdac8c-d5db-51eb-dfc6-
67bb86e4bdcf" policyname="ztna_rule" duration=5 wanin=2028 rcvdbyte=2028 wanout=1321
lanin=1236 sentbyte=1236 lanout=947 appcat="unscanned" utmaction="allow" countweb=1
utmref=65443-14
```

Example 3: IPv4 Client — IPv4 Access Proxy — IPv6 Server

To configure the FortiGate:

1. Configure the IPv4 access proxy VIP:

```
config firewall vip
    edit "zv4"
        set type access-proxy
        set extip 172.18.62.66
        set extintf "any"
        set server-type https
        set extport 4443
        set ssl-certificate "cert"
    next
end
```

2. Configure a virtual host:

```
config firewall access-proxy-virtual-host
    edit "vhost_ipv4"
        set ssl-certificate "cert"
        set host "qa.test.com"
    next
end
```

   The client uses this address to connect to the access proxy.

3. Configure an IPv4 access proxy and IPv6 api-gateway, apply the VIP and virtual host to it, and assign an IPv6 address to the realserver:

```
config firewall access-proxy
    edit "zs4"
        set vip "zv4"
        config api-gateway6
            edit 1
                set virtual-host "vhost_ipv4"
                config realservers
                    edit 1
                        set ip 2000:172:16:200::209
                    next
                end
            next
        end
    next
end
```

4. Apply the IPv4 access proxy to a proxy policy:

```
config firewall proxy-policy
    edit 1
        set name "ztna_rule"
        set proxy access-proxy
        set access-proxy "zs4"
        set srcintf "port2"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
```

```
        set logtraffic all
        set srcaddr6 "all"
        set dstaddr6 "all"
        set utm-status enable
        set ssl-ssh-profile "custom-deep-inspection"
        set webfilter-profile "monitor-all"
    next
end
```

5. Apply the IPv4 VIP to a firewall policy:

```
config firewall policy
    edit 4
        set name "ZTNA"
        set srcintf "port2"
        set dstintf "any"
        set action accept
        set srcaddr "all"
        set dstaddr "zv4"
        set schedule "always"
        set service "ALL"
        set inspection-mode proxy
        set logtraffic all
        set nat enable
    next
end
```

To test the configuration:

1. On an IPv4 client, ensure that the address qa6.test.com resolves to the IPv4 VIP address of 172.18.62.66.
2. In a browser, connect to https://qa6.test.com:6443.
3. After device certificate verification, the browser will open up the webpage on the IPv6 real server.
4. In the Forward Traffic Log, the following log is available:

```
1: date=2021-06-25 time=13:52:30 eventtime=1624654350689576485 tz="-0700"
logid="0000000024" type="traffic" subtype="forward" level="notice" vd="root"
srcip=10.1.100.206 srcport=53492 srcintf="port2" srcintfrole="undefined"
dstcountry="Reserved" srccountry="Reserved" dstip=2000:172:16:200::209 dstport=443
dstintf="root" dstintfrole="undefined" sessionid=726 service="HTTPS" proto=6
action="accept" policyid=1 policytype="proxy-policy" poluuid="7afdac8c-d5db-51eb-dfc6-
67bb86e4bdcf" policyname="ztna_rule" duration=0 wanin=1901 rcvdbyte=1901 wanout=736
lanin=569 sentbyte=569 lanout=3040 appcat="unscanned" utmaction="allow" countweb=1
utmref=65443-28
```

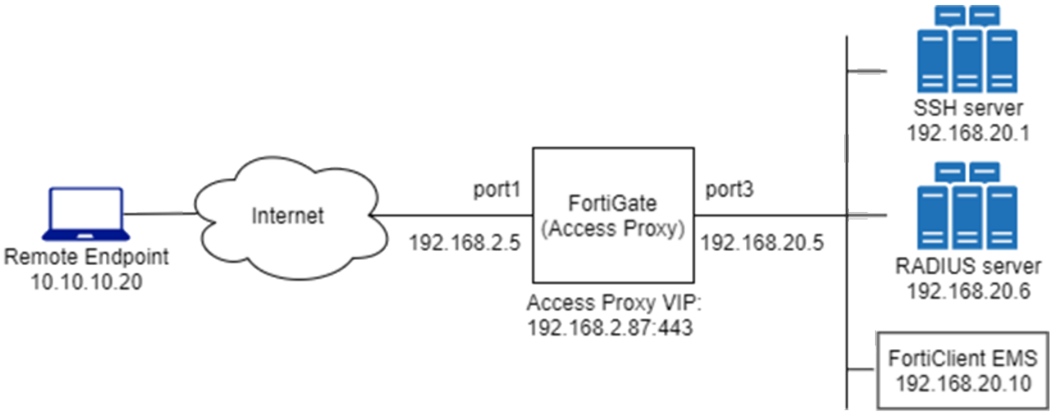
ZTNA SSH access proxy example

ZTNA can be configured with SSH access proxy to provide a seamless SSH connection to the server.

Advantages of using an SSH access proxy instead of a TCP forwarding access proxy include:

- Establishing device trust context with user identity and device identity checks.
- Applying SSH deep inspection to the traffic through the SSH related profile.
- Performing optional SSH host-key validation of the server.

- Using one-time user authentication to authenticate the ZTNA SSH access proxy connection and the SSH server connection.

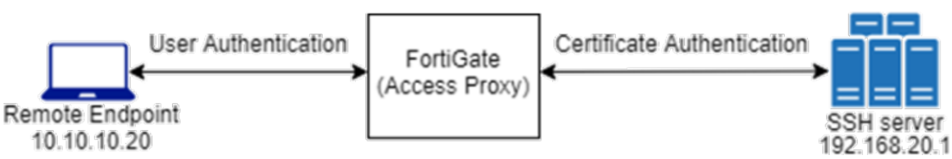


Perform SSH host-key validation of the server

To act as a reverse proxy for the SSH server, the FortiGate must perform SSH host-key validation to verify the identity of the SSH server. The FortiGate does this by storing the public key of the SSH server in its SSH host-key configurations. When a connection is made to the SSH server, if the public key matches one that is used by the server, then the connection is established. If there is no match, then the connection fails.

One-time user authentication

SSH access proxy allows user authentication to occur between the client and the access proxy, while using the same user credentials to authenticate with the SSH server. The following illustrates how this works:



1. The remote endpoint registers to FortiClient EMS and receives the client certificate.
2. The remote endpoint tries to connect to the SSH access proxy. It must use the same username that is later used for access proxy authentication.
3. The FortiGate challenges the endpoint with device identity validation.
4. The remote endpoint provides the EMS issued certificate for device identification.
5. The FortiGate challenges the endpoint with user authentication. For example, this could be done with basic or SAML authentication.
6. The users enters their credentials on the remote endpoint.
7. The FortiGate authenticates the user and collects the username.
8. Using the FortiGate's CA or the customer's CA certificate, the FortiGate signs an SSH certificate and embeds the username in its principal.
9. The FortiGate attempts to connect to the SSH server using the certificate authentication.
10. The SSH server verifies the authenticity of the certificate, and matches the username principal against its authorized_keys file.
11. If the username matches a record in the file, then the SSH connection is established. If no match is found, then the SSH connection fails.

Example

In this example, an SSH connection is established using SSH access proxy with host-key validation and one-time authentication.

- The SSH server is a Linux based server that uses sshd to provide remote access
- For SSH host-key validation, the public key of the SSH server has been imported into the FortiGate.
- For one-time authentication using certificate authentication:
  - The SSH server must allow certificate authentication.
  - The SSH server must have the proper entry in its authorized_keys file that contains the user principal and the FortiGate CA's public key.
  - The entry is present in the user directory corresponding to the user that is trying to log in.

To pre-configure the Linux SSH server:

1. Retrieve the public key used for host-key validation:
   a. Locate the public key files in the SSH server:
   ```
   $ ls -la /etc/ssh/*.pub
   -rw-r--r-- 1 root root 186 Mar 29  2020 /etc/ssh/ssh_host_ecdsa_key.pub
   -rw-r--r-- 1 root root 106 Mar 29  2020 /etc/ssh/ssh_host_ed25519_key.pub
   -rw-r--r-- 1 root root 406 Mar 29  2020 /etc/ssh/ssh_host_rsa_key.pub2
   ```
   b. Choose the publish key file based on the hash type (in this case, ECDSA), and show it's content:
   ```
   $ cat /etc/ssh/ssh_host_ecdsa_key.pub
   ecdsa-sha2-nistp256 AAAAE2*********IpEik=
   ```
   This key will be used when configuring the FortiGate.
2. Retrieve the FortiGate CA's public key from the FortiGate:
   ```
   # show full firewall ssh local-ca Fortinet_SSH_CA
   config firewall ssh local-ca
       edit "Fortinet_SSH_CA"
           set password ENC <hidden password>
           set private-key "-----BEGIN OPENSSH PRIVATE KEY-----
   <hidden private key>
   -----END OPENSSH PRIVATE KEY-----"
           set public-key "ssh-rsa AAAAB3**********JLXlxj3"
           set source built-in
       next
   end
   ```
3. On the Linux server, enable the SSH service to use the authorized_keys file:
   a. Locate and edit the */etc/ssh/sshd_config* file.
   b. Ensure that the *AuthorizedKeysFile* line is uncommented, for example:
   ```
   AuthorizedKeysFile      .ssh/authorized_keys .ssh/authorized_keys2
   ```
4. Allow remote SSH log in with certificate authentication and principal name:
   a. Log in to the SSH server using the account that will be granted remote SSH access (in this example: *radCurtis*):
   b. Locate the account's authorized_keys file in the *~/.ssh* directory:
   ```
   $ ls -la ~/.ssh
   total 12
   ```

```
drwxrwxr-x 2 radCurtis radCurtis 4096 Aug 10 19:14 .
drwxr-xr-x 5 radCurtis radCurtis 4096 Aug 10 19:13 ..
-rw-rw-r-- 1 radCurtis radCurtis  419 Aug 10 19:14 authorized_keys
```

c. If the directory and file do not exist, create the directory:

```
$ mkdir ~/.ssh
```

d. Create an entry containing the following keywords and add them to the authorized_keys file:

```
echo 'cert-authority,principals="radCurtis" ssh-rsa AAAAB3**********JLXlxj3' >>
authorized_keys
```

Where:

- `cert-authority` - indicates that this entry is used in certificate authentication by validating the certificate using the public key provided in this entry.
- `principals="radCurtis"` - indicates the user that must match with the username embedded in the SSH certificate.
- `ssh-rsa AAAAB3**********JLXlxj3` - indicates the FortiGate CA's public key that is used to validate the SSH certificate.

5. Restart the sshd service:

```
$ sudo systemctl stop sshd
$ sudo systemctl start sshd
```

The SSH server can now accept SSH connection from *radCurtis@<server IP>*, where the SSH certificate used by the FortiGate to log in contains radCurtis embedded as a principal.

When a user connects from a SSH client using *<username>@<server IP>*, sshd will locate the authorized_keys file in the directory */home/<username>/.ssh/authorized_keys*. If the authorized_keys is not in that directory, authentication will fail on the SSH server side.

If you suspect that authentication is failing on the SSH server, use the following commands to manually start sshd in debug mode to troubleshoot:

```
$ sudo systemctl stop sshd
```

```
$ /usr/sbin/sshd -ddd -p 22
```

To configure the FortiGate :

1. Configure a new VIP to allow access to the SSH access proxy over 192.168.2.87:443:

```
config firewall vip
    edit "ZTNA_SSH"
        set type access-proxy
        set extip 192.168.2.87
        set extintf "any"
        set server-type https
        set extport 443
        set ssl-certificate "Fortinet_CA_SSL"
    next
end
```

2. Configure the address object for the SSH server:

```
config firewall address
    edit "SSH_server"
        set subnet 192.168.20.1 255.255.255.255
    next
end
```

3. Configure the host-key that will be used to authenticate the SSH server. The public-key was retrieved when pre-configure the Linux SSH server (step 1b).

```
config firewall ssh host-key
    edit "ed25519"
        set type ECDSA
        set usage access-proxy
        set public-key "AAAAE2**********IpEik="
    next
end
```

4. Configure the access proxy SSH client certificate:

   A CA certificate is assigned to sign the SSH certificate that will be used in the SSH authentication. The SSH certificate will have the username embedded in the certificate principal.

```
config firewall access-proxy-ssh-client-cert
    edit "ssh-access-proxy"
        set source-address enable
        set auth-ca "Fortinet_SSH_CA"
    next
end
```

5. Configure the access-proxy server setting:

```
config firewall access-proxy
    edit "ZTNA_SSH"
        set vip "ZTNA_SSH"
        set client-cert enable
        config api-gateway
            edit 1
                set url-map "tcp"
                set service tcp-forwarding
                config realservers
                    edit 1
                        set address "SSH_server"
                        set type ssh
                        set ssh-client-cert "ssh-access-proxy"
                        set ssh-host-key-validation enable
                        set ssh-host-key "ed25519"
                    next
                end
            next
        end
    next
end
```

6. Configure the RADIUS setting, user setting, and user group to apply user authentication to the access proxy connection using RADIUS:

```
config user radius
    edit "Win2k16-Radius"
        set server "192.168.20.6"
```

```
        set secret ENC <secret>
    next
end
config user local
    edit "radCurtis"
        set type radius
        set radius-server "Win2k16-Radius"
    next
end
config user group
        edit "radius_group"
        set member "radCurtis" "Win2k16-Radius"
    next
end
```

7. Create the authentication scheme and rule to perform the authentication:

```
config authentication scheme
    edit "basic_auth"
        set method basic
        set user-database "Win2k16-Radius"
    next
end
config authentication rule
    edit "ztna-basic"
        set srcaddr "all"
        set ip-based disable
        set active-auth-method "basic_auth"
        set web-auth-cookie enable
    next
end
```

8. Configure the ZTNA rule to allow traffic to the SSH server, and apply user authentication, posture check, and a security profile where necessary:

```
config firewall proxy-policy
    edit 5
        set name "SSH-proxy"
        set proxy access-proxy
        set access-proxy "ZTNA_SSH"
        set srcintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set ztna-ems-tag "FCTEMS8821001056_ems138_av_tag"
        set action accept
        set schedule "always"
        set groups "radius_group"
        set utm-status enable
        set ssl-ssh-profile "custom-deep-inspection"
    next
end
```

To check the results:

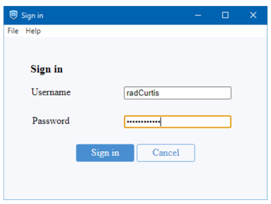
1. On the remote client, open FortiClient, go to the *Zero Trust Telemetry* tab, and make sure that it is connected to the EMS server.
2. Go to the *ZTNA Connection Rules* tab and click *Add Rule*.

3. Configure the rule, then click *Create*:

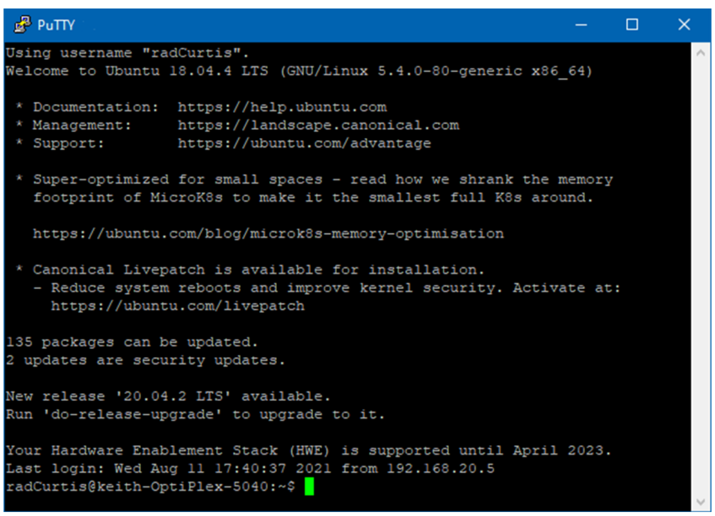
Rule Name	SSH-Linux
Destination Host	192.168.20.1:22
Proxy Gateway	192.168.2.87:443
Mode	Transparent
Encryption	Disabled (recommended)

   When Encryption is disabled, the connection between the client and FortiGate access proxy is not encapsulated in HTTPS after the client and FortiGate connection is established. This allows for less overhead, because SSH is already a secure connection. This option is available in FortiClient 7.0.1 and later releases.

4. Open an SSH client, such as PuTTy, and make an SSH connection to *radCurtis@192.168.20.1* on port *22*.
5. After device authentication is performed and passes in the background, FortiClient prompts the user to sign in. Enter the username, *radCurtis*, and password, then click *Sign in*.



   After successful user authentication, the SSH connection is established without an additional log in.



6. On the FortiGate, check the logged in user:
   a. Go to *Dashboard > Users & Devices* and expand the *Firewall Users* widget.
   b. Check the WAD proxy user list:

```
# diagnose wad user list
ID: 2, VDOM: root, IPv4: 10.10.10.25
  user name   : radCurtis
  worker      : 0
  duration    : 614
  auth_type   : Session
  auth_method : Basic
  pol_id      : 5
```

```
    g_id        : 12
    user_based  : 0
    expire      : 53
    LAN:
      bytes_in=3403 bytes_out=5699
    WAN:
      bytes_in=3681 bytes_out=3132
```

7. The successful connection is logged in the forward traffic logs after the SSH connection has disconnected:

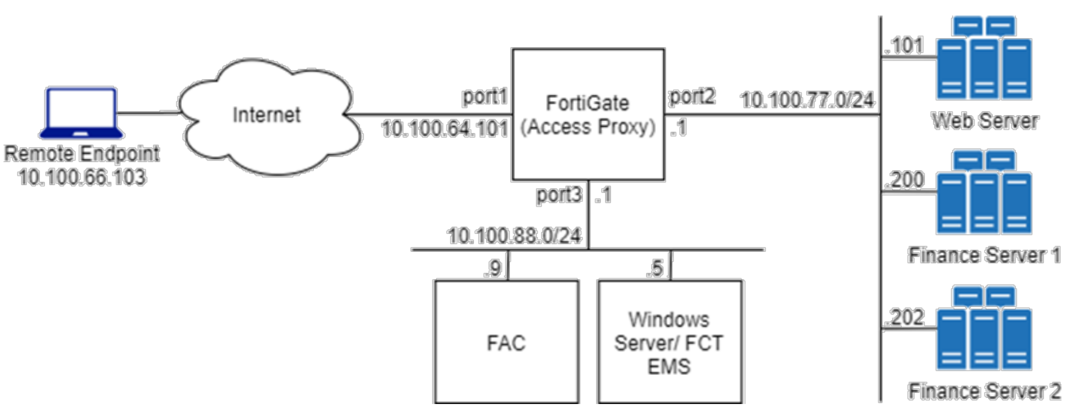
```
# execute log display
25 logs found.
10 logs returned.

1: date=2021-08-11 time=17:59:56 eventtime=1628729996110159120 tz="-0700"
logid="0000000024" type="traffic" subtype="forward" level="notice" vd="root"
srcip=10.10.10.25 srcport=50627 srcintf="port1" srcintfrole="wan" dstcountry="Reserved"
srccountry="Reserved" dstip=192.168.20.1 dstport=22 dstintf="root"
dstintfrole="undefined" sessionid=1926338 srcuuid="5445be2e-5d7b-51ea-e2c3-ae6b7855c52f"
service="SSH" proto=6 action="accept" policyid=5 policytype="proxy-policy"
poluuid="16fb5550-e976-51eb-e76c-d45e96dfa5dc" policyname="SSH-proxy" duration=67
user="radCurtis" group="radius_group" authserver="Win2k16-Radius" wanin=3681
rcvdbyte=3681 wanout=3132 lanin=3403 sentbyte=3403 lanout=5699 appcat="unscanned"
```

ZTNA access proxy with SAML and MFA using FortiAuthenticator example

ZTNA access proxy supports device verification using device certificates that are issued by EMS. To authenticate users, administrators can use either basic or SAML authentication. An advantage of SAML authentication is that multi-factor authentication (MFA) can be provided by the SAML Identity Provider (IdP).

In these examples, a FortiAuthenticator is used as the IdP, and MFA is applied to user authentication for remote users accessing the web, RDP, and SSH resources over the ZTNA access proxy. It is assumed that the FortiGate EMS fabric connector has already been successfully connected.

- Configuring the FortiAuthenticator on page 1037
- Configuring the FortiGate SAML settings on page 1039
- Example 1 - Applying SAML and MFA to ZTNA HTTPS access proxy on page 1042
- Example 2 - Applying SAML and MFA to a ZTNA TCP forwarding access proxy for RDP connections on page 1044
- Example 3 - Applying SAML and MFA to a ZTNA SSH access proxy on page 1046



DNS resolutions:

- ztna.fortidemo.fortinet.com:20443 -> 10.100.64.201:20443
- entcore.fortidemo.fortinet.com:20443 -> 10.100.64.201:20443
- fac.fortidemo.fortinet.com - > 10.100.64.103

The FortiAuthenticator (FAC) integrates with Active Directory (AD) on the Windows Domain Controller, which is also acting as the EMS server. Users are synchronized from the AD to the FAC, and remote users are configured with token-based authentication. SAML authentication is configured on the FortiGate, pointing to the FAC as the SAML IdP. The SAML server is applied to the ZTNA access proxy authentication scheme and rule, to provide the foundation for applying user authentication on individual ZTNA rules.

Configuring the FortiAuthenticator

First configure the FortiAuthenticator to synchronize users from AD using LDAP, apply MFA to individual remote users, and be the IdP.

To create a remote authentication server pointing to the Windows AD:

1. Go to *Authentication > Remote Auth. Servers > LDAP* and click *Create New*.
2. Configure the following:

Name	AD
Primary server name / IP	10.100.88.5
Port	389 (or another port if using LDAPS)
Based distinguished name	DC=FORTI-ARBUTUS,DC=LOCAL
Bind type	Regular
Username	<user account used for LDAP bind>
Password	<password of user>
User object class	person (default)
Username attribute	sAMAccountName (default)
Group object class	group (default)
Obtain group membership from	Group attribute
Group membership attribute	memberOf (default)
Secure connection	Enable if using LDAPS or STARTTLS

3. Click *OK*.
4. In the *Remote LDAP Users* section click *Go*.
5. Select the users to import then click *OK*.
6. Click *OK*.
   For more details, see LDAP in the FortiAuthenticator Administration Guide.

To configure a remote LDAP user to use MFA:

1. Go to *Authentication > User Management > Remote Users*, and edit a user.
2. Enable *Token-based authentication* then select the method of token code delivery.
   For this example, select *FortiToken > Mobile*, select the Token from the drop-down list, and set the *Activation delivery method* to email.
3. In the *User Information* section, add the email address that will be used for the FortiToken activation.
4. Click *OK*.
   An activation email is sent to the user that they can use to install the token to their FortiToken Mobile app.
   For more details, see Remote users in the FortiAuthenticator Administration Guide.

To configure SAML IdP:

1. Go to *Authentication > SAML IdP > General* and enable *Enable SAML Identity Provider portal*.
2. The *Server address* is the device FQDN or IP address (configured in the System Information widget at *System > Dashboard > Status*). In this example, it is *fac.fortidemo.fortinet.com*.
3. Set *Username input format* to *username@realm*.
4. Click *Add a realm* in the *Realms* table:
   a. Set *Realm* to the just created LDAP realm (*AD*).
   b. Optionally, enable *Filter* and select the required users groups. In this example, *Customer Support* and *Marketing* are configured.
5. Set *Default IdP certificate* to the certificate that will be used in the HTTPS connection to the IdP portal.
6. Click *OK*.
7. Go to *Authentication > SAML IdP > Service Providers*, and click *Create New* to create a service provider (SP) for the FortiGate SP.
8. Configure the following, which must match what will be configured on the FortiGate:

SP name	Enterprise Core
IdP prefix	ztna
Server certificate	Same certificate as the default IdP certificate used in SAML IdP > General
SP entity ID	https://entcore.fortidemo.fortinet.com:20443/ztna/saml/metadata/
SP ACS (login) URL	https://entcore.fortidemo.fortinet.com:20443/ztna/saml/login/
SP SLS (logout) URL	https://entcore.fortidemo.fortinet.com:20443/ztna/saml/logout/
Participate in single logout	Enable

   Where the *SP entity ID*, *SP ACS (login) URL*, and *SP SLS (logout) URL* break down as follows:
   - *entcore.fortidemo.fortinet.com* - The FQDN that resolves to the FortiGate SP.
   - *20443* - The port that is used to map to the FortiGate's SAML SP service.
   - */ztna/saml* - The custom, user defined fields.
   - */metadata*, */login*, and */logout* - The standard convention used to identify the SP entity, log in portal, and log out portal.
9. Click *OK*.
10. Edit the just created SP object and, under *SAML Attribute*, click *Create New*.

11. Set *SAML attribute* to the username and set *User attribute* to *Username*, then click *OK*.
12. Click *OK*.

For more details, see Configuring SAML settings in the SAML Interoperability Guide.

Configuring the FortiGate SAML settings

On the FortiGate, a SAML user is used to define the SAML SP and IdP settings. This user is then applied to the ZTNA proxy using an authentication scheme, rule, and settings. A ZTNA server is then created to allow access to the SAML SP server so that end users can reach the FortiGate SP's captive portal. The SAML user must then be added to a ZTNA rule to trigger authentication when accessing the ZTNA access proxy.

To create a new SAML user/server:

1. Create the SAML user object:

```
config user saml
    edit "su-ztna"
        set cert "FortiDemo"
        set entity-id "https://entcore.fortidemo.fortinet.com:20443/ztna/saml/metadata/"
        set single-sign-on-url
"https://entcore.fortidemo.fortinet.com:20443/ztna/saml/login/"
        set single-logout-url
"https://entcore.fortidemo.fortinet.com:20443/ztna/saml/logout/"
        set idp-entity-id "http://fac.fortidemo.fortinet.com/saml-idp/ztna/metadata/"
        set idp-single-sign-on-url "https://fac.fortidemo.fortinet.com/saml-
idp/ztna/login/"
        set idp-single-logout-url "https://fac.fortidemo.fortinet.com/saml-
idp/ztna/logout/"
        set idp-cert "REMOTE_Cert_1"
        set user-name "username"
        set digest-method sha1
    next
end
```

Where:

- The `FortiDemo` certificate is a local certificate that is used to sign SAML messages that are exchanged between the client and the FortiGate SP. In this example, it is used to sign *entcore.fortidemo.fortinet.com*.
- The `REMOTE_Cert_1` certificate is a remote certificate that is used to identify the IdP. In this example, *fac.fortidemo.fortinet.com*.
- The URLs used in the SAML user settings are the same as the ones defined on the FortiAuthenticator.

2. Add the SAML user object to a new user group:

```
config user group
    edit "ztna-users"
        set member "su-ztna"
    next
end
```

To apply the SAML server to proxy authentication:

1. Apply the SAML server to an authentication scheme:

```
config authentication scheme
    edit "saml-scheme"
        set method saml
        set saml-server "su-ztna"
    next
end
```

2. Apply the authentication scheme to an authentication rule:

```
config authentication rule
    edit "saml-rule"
        set srcintf "any"
        set srcaddr "all"
        set ip-based disable
        set active-auth-method "saml-scheme"
        set web-auth-cookie enable
    next
end
```

3. Configure the active authentication scheme, and a captive portal to serve the log in page for the SAML requests:

```
config firewall address
    edit "entcore.fortidemo.fortinet.com"
        set type fqdn
        set fqdn "entcore.fortidemo.fortinet.com"
    next
end
```

```
config authentication setting
    set active-auth-scheme "saml-scheme"
    set captive-portal "entcore.fortidemo.fortinet.com"
end
```

To configure a ZTNA access proxy to allow SAML authentication requests to the SP:

1. Configure the ZTNA server:
   a. Go to *Policy & Objects > ZTNA*, select the *ZTNA Servers* tab, and click *Create New*.
   b. Configure the following:

Name	ZTNA-access
External interface	Any
External IP	10.100.64.201
External port	20443
SAML	Enabled
SAML SSO Server	su-ztna
Default certificate	FortiDemo

   c. Click *OK*.
2. Define the ZTNA rule to allow access to the ZTNA server:

a. Go to *Policy & Objects* > *ZTNA*, select the *ZTNA Rules* tab, and click *Create New*.
b. Configure the following:

Name	ZTNA-Rule
Source (Address)	all
Source (User)	ztna-users
ZTNA Server	ZTNA-access
Action	Accept

c. Click *OK*.

To configure a VIP and a firewall policy to forward IdP authentication traffic to the FortiAuthenticator:

Remote clients connect to the FortiAuthenticator IdP behind the FortiGate using a VIP. In this example, users connect to the FQDN fac.fortidemo.fortinet.com that resolves to the VIP's external IP address.

1. Configure the VIP to forward traffic to the FortiAuthenticator:
   a. Go to *Policy & Objects* > *Virtual IPs* and click *Create New* > *Virtual IP*.
   b. Configure the following:

Name	FortiAuthenticator
Interface	Any
External IP address	10.100.64.103
Map to > IPv4 address/range	10.100.88.9
Port Forwarding	Enabled
Protocol	TCP
External service port	443
Map to IPv4 port	443

   c. Click *OK*.
2. Configure a firewall policy to allow VIP:
   a. Go to *Policy & Objects* > *Firewall Policy* and click *Create New*.

b. Configure the following:

Name	WAN to FAC
ZTNA	Disabled
Incoming Interface	Any
Outgoing Interface	Any
Source	All
ZTNA Server	FortiAuthenticator
Schedule	always
Service	ALL
Action	Accept
NAT	disabled

c. Click *OK*.

Example 1 - Applying SAML and MFA to ZTNA HTTPS access proxy

In this HTTPS access proxy example, two real servers are implemented with round robin load balancing performed between them. The HTTPS access proxy is configured on the same ZTNA server as was configured in the authentication step. The same ZTNA rule and firewall policy also apply.

To configure the ZTNA server for HTTPS access proxy with load balancing:

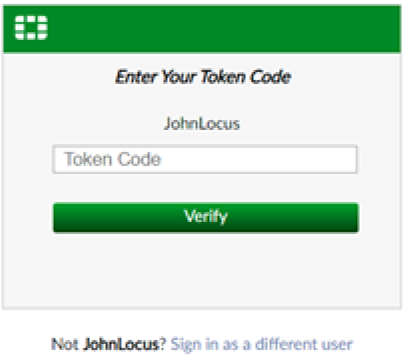
1. Go to *Policy & Objects* > *ZTNA* and select the *ZTNA Servers* tab.
2. Edit the ZTNA-access server.
3. In the *Service/server mapping* table, click *Create New*:
    a. Set *Service* to *HTTPS*.
    b. Set *Virtual Host* to *Any Host*.
    c. In the *Servers* table click *Create New*:
        i. Set *IP* to 10.100.77.200
        ii. Set *Port* to *443*.
        iii. Set *Status* to *Active*.
        iv. Click *OK*.
    d. Create a second server with *IP 10.100.77.202*.
    e. Click *OK*.
4. Enable *Load balancing*, and select the *Round Robin* algorithm.
5. Click *OK*.

Testing and verification:

From the remote endpoint, user John Locus attempts to connect to the Finance server over ZTNA:

1. On the remote Windows computer, open FortiClient and register to the EMS server.
2. Open a browser and attempt to connect to the web server at *https://ztna.fortidemo.fortinet.com:20443*.

3. Device authentication prompts the user for their device certificate. Select the certificate issued by EMS and click *OK*.
4. FortiGate receives the SAML request and redirects the user to the IdP login screen. Enter the username and password for John Locus and click *Login*.
5. A second prompt opens asking for the *Token Code*. Enter the code then click *Verify*.



6. The FortiAuthenticator IdP verifies the login, then sends the SAML assertion back to the user.
7. The browser redirects the assertion to the FortiGate SP, which decides if the user is allowed access.
8. On a successful log in, FortiGate redirects the user to the web page that they are trying to access.

Logs and debugs:

On the FortiGate, a successful connection can be seen in *Log & Report > Forward Traffic log*, or by using the CLI:

```
# execute log filter category 0
# execute log filter field srcip 10.100.66.103
# execute log display
...
1: date=2021-08-25 time=23:34:15 eventtime=1629959656098675227 tz="-0700" logid="0000000024"
type="traffic" subtype="forward" level="notice" vd="root" srcip=10.100.66.103 srcport=51341
srcintf="port1" srcintfrole="wan" dstcountry="Reserved" srccountry="Reserved"
dstip=10.100.77.202 dstport=443 dstintf="root" dstintfrole="undefined" sessionid=3396047
srcuuid="d8dd134a-0517-51ec-2ff0-3032b84564e7" service="HTTPS" proto=6 action="accept"
policyid=1 policytype="proxy-policy" poluuid="256bb090-0518-51ec-f431-5dcc0baa725b"
policyname="ZTNA-Rule" duration=16 user="johnlocus" group="ztna-users" wanin=2837
rcvdbyte=2837 wanout=1495 lanin=2581 sentbyte=2581 lanout=5505 appcat="unscanned"

2: date=2021-08-25 time=23:34:04 eventtime=1629959645171823879 tz="-0700" logid="0000000024"
type="traffic" subtype="forward" level="notice" vd="root" srcip=10.100.66.103 srcport=62691
srcintf="port1" srcintfrole="wan" dstcountry="Reserved" srccountry="Reserved"
dstip=10.100.77.200 dstport=443 dstintf="root" dstintfrole="undefined" sessionid=3396036
srcuuid="d8dd134a-0517-51ec-2ff0-3032b84564e7" service="HTTPS" proto=6 action="accept"
policyid=1 policytype="proxy-policy" poluuid="256bb090-0518-51ec-f431-5dcc0baa725b"
policyname="ZTNA-Rule" duration=5 user="johnlocus" group="ztna-users" wanin=2837
rcvdbyte=2837 wanout=1546 lanin=1576 sentbyte=1576 lanout=1033 appcat="unscanned"
...
4: date=2021-08-25 time=23:29:21 eventtime=1629959362319670887 tz="-0700" logid="0000000024"
type="traffic" subtype="forward" level="notice" vd="root" srcip=10.100.66.103 srcport=62107
srcintf="port1" srcintfrole="wan" dstcountry="Reserved" srccountry="Reserved"
dstip=10.100.64.201 dstport=20443 dstintf="root" dstintfrole="undefined" sessionid=3388350
srcuuid="d8dd134a-0517-51ec-2ff0-3032b84564e7" service="tcp/20443" proto=6 action="accept"
policyid=41 policytype="policy" poluuid="256d6bd8-0518-51ec-7b6a-18df00b82da3"
```

```
policyname="ZTNA Policy" duration=1 user="johnlocus" authserver="su-ztna" wanin=0 rcvdbyte=0
wanout=0 lanin=14246 sentbyte=14246 lanout=993 appcat="unscanned"
```

Log number four shows that the session was first allowed through the ZTNA firewall policy. Log numbers one and two show the traffic allowed through the ZTNA proxy-policy over two successive sessions. Note that they have different destination IP addresses (`dstip`), indicating that ZTNA was performing server load balancing.

Use the following command to show if the FortiGate's WAD process has an active record of the SAML user login:

```
# diagnose wad user list

ID: 6, VDOM: root, IPv4: 10.100.66.103
  user name   : johnlocus
  worker      : 0
  duration    : 611
  auth_type   : Session
  auth_method : SAML
  pol_id      : 1
  g_id        : 3
  user_based  : 0
  expire      : 283
  LAN:
    bytes_in=38016 bytes_out=16166
  WAN:
    bytes_in=12422 bytes_out=7217
```

Example 2 - Applying SAML and MFA to a ZTNA TCP forwarding access proxy for RDP connections

In this TCP forwarding access proxy example, RDP connections are allowed to be forwarded to the Windows/EMS server. Traffic to TCP/3389 is allowed through the ZTNA proxy.

To configure the ZTNA server for TCP forwarding on TCP/3389:

1. Create a firewall address for the Windows/EMS server:
   a. Go to *Policy & Objects > Addresses* and click *Create New > Address*.
   b. Configure the following:

Name	EMS-Server
Type	Subnet
IP/Netmask	10.100.88.5/32
Interface	any

   c. Click *OK*.
2. Go to *Policy & Objects > ZTNA* and select the *ZTNA Servers* tab.
3. Edit the ZTNA-access server.
4. In the *Service/server mapping* table, click *Create New*:
   a. Set *Service* to *TCP Forwarding*.
   b. In the *Servers* table click *Create New*:

   i. Set *Address* to *EMS-Server*.
   ii. Set *Ports* to *3389*.
   iii. Click *OK*.
  c. Click *OK*.
5. Click *OK*.

Testing and verification:

On the remote endpoint, manually configure ZTNA connection rules to forward RDP traffic to the ZTNA access proxy. The rules can also be pushed from the EMS server; for details see Provisioning ZTNA TCP forwarding rules via EMS.

Configure the ZTNA connection rule:

1. On the remote Windows computer, open FortiClient.
2. Register to the EMS server.
3. On the *ZTNA Connection Rules* tab, click *Add Rule* to add a TCP forwarding rule.
4. Configure the following:

Rule Name	RDP-server
Destination Host	10.100.88.5:3389
Proxy Gateway	ztna.fortidemo.fortinet.com:20443
Mode	Transparent
Encryption	Disabled *Encryption* can be enabled or disabled. When it is disabled, the client to access proxy connection is not encrypted in HTTPS. Because RDP is encrypted by default, disabling *Encryption* does not reduce security.

5. Click *Create*.

Connect over RDP:

1. On the remote PC, open a new RDP connection.
2. Enter the IP address 10.100.588.5. By default, RDP session use port 3389.
   When the connection to the ZTNA access proxy is established, FortiGate will redirect the SAML login request to the FortiAuthenticator IdP.
   A FortiClient prompt will open with the FortiAuthenticator login screen.
3. Enter the username and password then click *Login*.
4. A second prompt opens asking for the *Token Code*. Enter the code from your FortiToken app, then click *Verify*.
5. FortiAuthenticator verifies the token code, determines if the login is successful, then sends the SAML assertion back to the client.
6. The client redirects the response back to the FortiGate SP.
7. If the log in was successful, the user can now log on to the RDP session.

Logs and debugs:

On the FortiGate, a successful connection can be seen in *Log & Report > Forward Traffic log*, or by using the CLI:

```
# execute log filter category 0
# execute log filter field srcip 10.100.66.103
# execute log display
...
3: date=2021-08-25 time=23:05:52 eventtime=1629957952722272222 tz="-0700" logid="0000000024"
type="traffic" subtype="forward" level="notice" vd="root" srcip=10.100.66.103 srcport=59980
srcintf="port1" srcintfrole="wan" dstcountry="Reserved" srccountry="Reserved"
dstip=10.100.88.5 dstport=3389 dstintf="root" dstintfrole="undefined" sessionid=3349083
srcuuid="d8dd134a-0517-51ec-2ff0-3032b84564e7" service="RDP" proto=6 action="accept"
policyid=1 policytype="proxy-policy" poluuid="256bb090-0518-51ec-f431-5dcc0baa725b"
policyname="ZTNA-Rule" duration=8 user="johnlocus" group="ztna-users" wanin=0 rcvdbyte=0
wanout=0 lanin=1444 sentbyte=1444 lanout=665 appcat="unscanned"

4: date=2021-08-25 time=23:05:49 eventtime=1629957949064583213 tz="-0700" logid="0000000024"
type="traffic" subtype="forward" level="notice" vd="root" srcip=10.100.66.103 srcport=59982
srcintf="port1" srcintfrole="wan" dstcountry="Reserved" srccountry="Reserved"
dstip=10.100.64.201 dstport=20443 dstintf="root" dstintfrole="undefined" sessionid=3349102
srcuuid="d8dd134a-0517-51ec-2ff0-3032b84564e7" service="tcp/20443" proto=6 action="accept"
policyid=41 policytype="policy" poluuid="256d6bd8-0518-51ec-7b6a-18df00b82da3"
policyname="ZTNA Policy" duration=5 user="johnlocus" authserver="su-ztna" wanin=0 rcvdbyte=0
wanout=0 lanin=14237 sentbyte=14237 lanout=1032 appcat="unscanned"
```

Use the following command to show if the FortiGate's WAD process has an active record of the SAML user login:

```
# diagnose wad user list

ID: 6, VDOM: root, IPv4: 10.100.66.103
  user name   : johnlocus
  worker      : 0
  duration    : 611
  auth_type   : Session
  auth_method : SAML
  pol_id      : 1
  g_id        : 3
  user_based  : 0
  expire      : no
  LAN:
    bytes_in=51471 bytes_out=233862
  WAN:
    bytes_in=211247 bytes_out=24410
```

Example 3 - Applying SAML and MFA to a ZTNA SSH access proxy

In this SSH access proxy example, SSH connections can be forwarded to the web server.

To configure the ZTNA server for SSH access proxy:

1. Create a firewall address for the web server:
   a. Go to *Policy & Objects > Addresses* and click *Create New > Address*.
   b. Configure the following:

Name	Web-Server1
Type	Subnet

IP/Netmask	10.100.77.101/32
Interface	any

c. Click *OK*.

2. Go to *Policy & Objects > ZTNA* and select the *ZTNA Servers* tab.
3. Edit the ZTNA-access server.
4. In the *Service/server mapping* table, edit the *TCP Forwarding* entry:
   a. In the *Servers* table click *Create New*:
      i. Set *Address* to *Web-Server1*.
      ii. Set *Ports* to *22*.
      iii. Optionally, enable *Additional SSH Options* to configure other SSH options as needed.
      iv. Click *OK*.
   b. Click *OK*.
5. Click *OK*.

Testing and verification:

On the remote endpoint, manually configure ZTNA connection rules to forward SSH traffic to the ZTNA access proxy. The rules can also be pushed from the EMS server; for details see Provisioning ZTNA TCP forwarding rules via EMS.

Configure the ZTNA connection rule:

1. On the remote Windows computer, open FortiClient.
2. Register to the EMS server.
3. On the *ZTNA Connection Rules* tab, click *Add Rule* to add a TCP forwarding rule.
4. Configure the following:

Rule Name	SSH-webserver
Destination Host	10.100.77.101:22
Proxy Gateway	ztna.fortidemo.fortinet.com:20443
Mode	Transparent
Encryption	Disabled

5. Click *Create*.

Connect over SSH:

1. On the remote PC, open a new SSH connection.
2. Enter the host root@10.100.77.101 on port 22.
   When the connection to the ZTNA access proxy is established, FortiGate will redirect the SAML login request to the FortiAuthenticator IdP.
   A FortiClient prompt will open with the FortiAuthenticator login screen.
3. Enter the username and password then click *Login*.
4. A second prompt opens asking for the *Token Code*. Enter the code from your FortiToken app, then click *Verify*.
5. FortiAuthenticator verifies the token code, determines if the login is successful, then sends the SAML assertion back to the client.

6. The client redirects the response back to the FortiGate SP.
7. If the log in was successful, the user can now log on to the SSH session.

Logs and debugs:

On the FortiGate, a successful connection can be seen in *Log & Report > Forward Traffic log*, or by using the CLI:

```
# execute log filter category 0
# execute log filter field srcip 10.100.66.103
# execute log display
...
2: date=2021-08-25 time=23:20:03 eventtime=1629958804111962686 tz="-0700" logid="0000000024"
type="traffic" subtype="forward" level="notice" vd="root" srcip=10.100.66.103 srcport=60043
srcintf="port1" srcintfrole="wan" dstcountry="Reserved" srccountry="Reserved"
dstip=10.100.77.101 dstport=22 dstintf="root" dstintfrole="undefined" sessionid=3372964
srcuuid="d8dd134a-0517-51ec-2ff0-3032b84564e7" service="SSH" proto=6 action="accept"
policyid=1 policytype="proxy-policy" poluuid="256bb090-0518-51ec-f431-5dcc0baa725b"
policyname="ZTNA-Rule" duration=1 user="johnlocus" group="ztna-users" wanin=39 rcvdbyte=39
wanout=0 lanin=1444 sentbyte=1444 lanout=726 appcat="unscanned"

3: date=2021-08-25 time=23:20:03 eventtime=1629958803100280597 tz="-0700" logid="0000000024"
type="traffic" subtype="forward" level="notice" vd="root" srcip=10.100.66.103 srcport=60042
srcintf="port1" srcintfrole="wan" dstcountry="Reserved" srccountry="Reserved"
dstip=10.100.64.201 dstport=20443 dstintf="root" dstintfrole="undefined" sessionid=3372934
srcuuid="d8dd134a-0517-51ec-2ff0-3032b84564e7" service="tcp/20443" proto=6 action="accept"
policyid=41 policytype="policy" poluuid="256d6bd8-0518-51ec-7b6a-18df00b82da3"
policyname="ZTNA Policy" duration=0 user="johnlocus" authserver="su-ztna" wanin=0 rcvdbyte=0
wanout=0 lanin=14237 sentbyte=14237 lanout=1032 appcat="unscanned"
```

Use the following command to show if the FortiGate's WAD process has an active record of the SAML user login:

```
# diagnose wad user list

ID: 6, VDOM: root, IPv4: 10.100.66.103
  user name    : johnlocus
  worker       : 0
  duration     : 192
  auth_type    : Session
  auth_method  : SAML
  pol_id       : 1
  g_id         : 3
  user_based   : 0
  expire       : 475
  LAN:
    bytes_in=19647 bytes_out=10063
  WAN:
    bytes_in=2886 bytes_out=1860
```

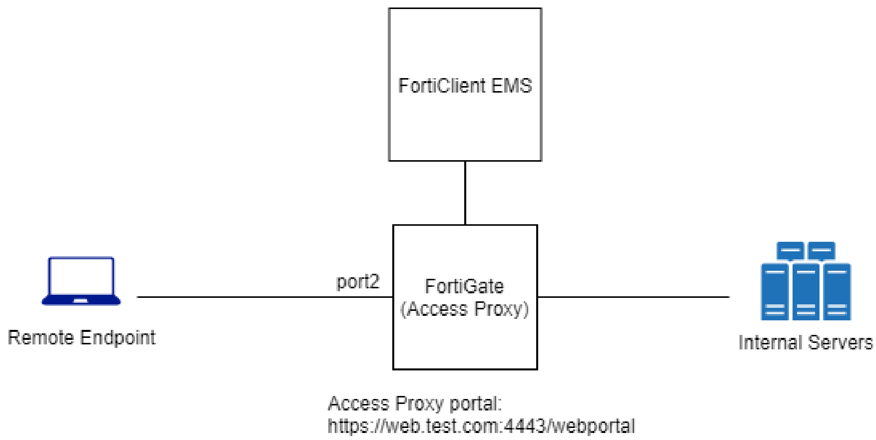
ZTNA access proxy with SSL VPN web portal example

SSL VPN web portals can be defined in ZTNA access proxy settings. The ZTNA access proxy handles the access control processes (client certificate authentication, posture check, user authentication and authorization), and establishes the HTTPS connection between the end user and the access proxy. Then, it forwards the user to the web portal where they can use predefined bookmarks to access TCP based services like HTTPS, RDP, VNC, FTP, SFTP, SSH, Telnet, and SMB. Existing SSL VPN portal configurations can be used.

The web portal service can only be configured in the CLI.

Example

In this example, a remote client connects to the ZTNA access proxy and completes the client certificate check. If successful, the remaining access control procedures are automatically completed, and the user is forwarded to the web portal. The web portal is configured with predefined bookmarks that connect to internal servers and external websites. The user can access any resource that is defined in the bookmarks to create an end-to-end connection.



To configure the SSL VPN web portal:

1. Go to *VPN > SSL-VPN Portals* and click *Create New*.
2. Enter the name, *test_ssl*.
3. Disable *Tunnel Mode*.
4. Enable *Web Mode*.
5. Create the bookmarks:
   a. Under *Predefined Bookmarks*, click *Create New*.
   b. Enter the name of the service.
   c. Select the service *Type*.
   d. Enter the *URL* to access the service.
   e. Click *OK*.
   f. Repeat these steps to create other bookmarks.
6. Click *OK*.

To configure the ZTNA access proxy:

1. Configure a VIP for the ZTNA access proxy. The `ssl-certificate` can be replaced with a server certificate:

```
config firewall vip
    edit "ztna_webportal"
        set type access-proxy
        set extip 172.18.62.68
        set extintf "any"
```

```
        set server-type https
        set extport 4443
        set ssl-certificate "*.test.com"
    next
end
```

2. Configure the virtual host to be used to connect to the ZTNA access proxy. The host should resolve to the VIP's address:

```
config firewall access-proxy-virtual-host
    edit "webportal"
        set ssl-certificate "*.test.com"
        set host "web.test.com"
    next
end
```

3. Configure the ZTNA access proxy to be in web portal mode:

```
config firewall access-proxy
    edit "ztna_webportal"
        set vip "ztna_webportal"
        set client-cert enable
        config api-gateway
            edit 1
                set url-map "/webportal"
                set service web-portal
                set virtual-host "webportal"
                set ssl-vpn-web-portal "test_ssl"
            next
        end
    next
end
```

4. Apply the access proxy to a proxy policy (specify the ZTNA tags as needed):

```
config firewall proxy-policy
    edit 1
        set name "ztna_rule"
        set proxy access-proxy
        set access-proxy "ztna_webportal"
        set srcintf "any"
        set srcaddr "all"
        set dstaddr "all"
        set ztna-ems-tag "FCTEMS8821000000_High"
        set action accept
        set schedule "always"
        set logtraffic all
        set srcaddr6 "all"
        set dstaddr6 "all"
        set utm-status enable
        set profile-type group
        set profile-group "profile group1"
        set logtraffic-start enable
    next
end
```

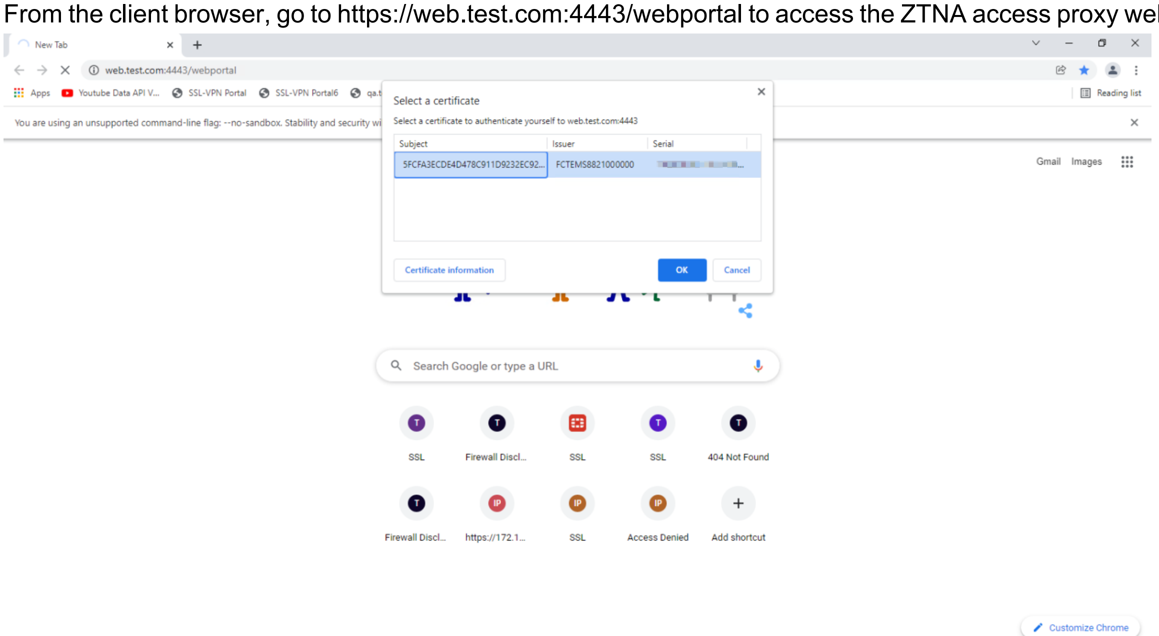
The SSL VPN bookmarks are learned by the WAD daemon and are ready to use.

5. Verify the bookmarks:

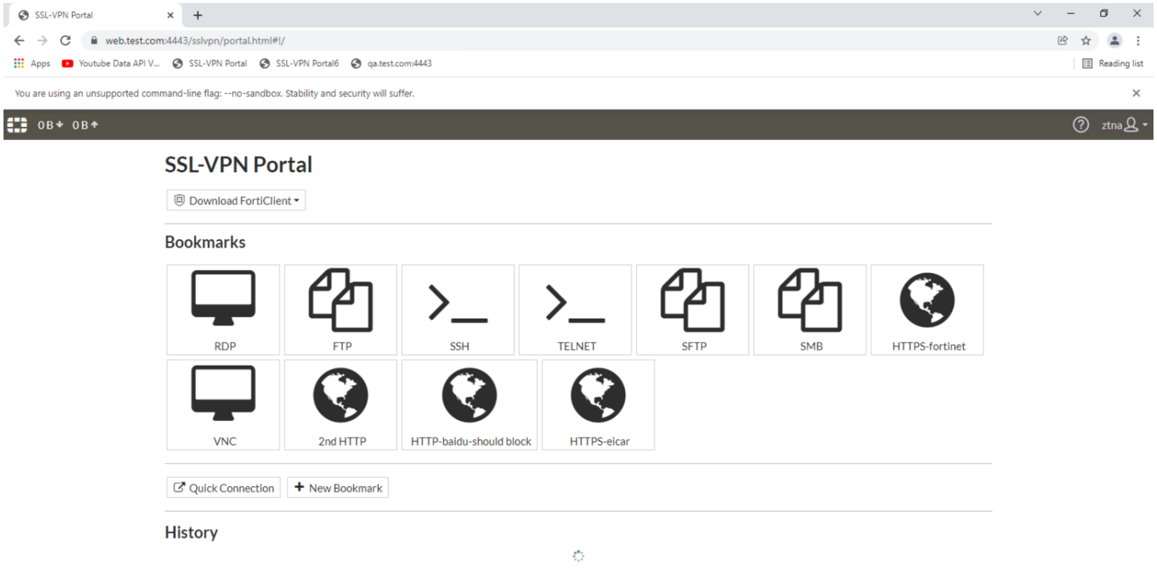
```
# diagnose test app wad 351
[bookmark: (portal/group/name=test_ssl/gui-bookmarks/2nd HTTP)]:
    type  :1
    url   :http://httpbin.org
    host  :
    folder:
    domain:
    port  :0
[bookmark: (portal/group/name=test_ssl/gui-bookmarks/FTP)]:
    type  :4
    url   :
    host  :
    folder:172.16.200.215
    domain:
    port  :0
[bookmark: (portal/group/name=test_ssl/gui-bookmarks/HTTPS-fortinet)]:
    type  :1
    url   :https://www.fortinet.com
    host  :
    folder:
    domain:
    port  :0
[bookmark: (portal/group/name=test_ssl/gui-bookmarks/RDP)]:
    type  :9
    url   :
    host  :172.18.62.213
    folder:
    domain:
    port  :3389
…
```

To test the connection:

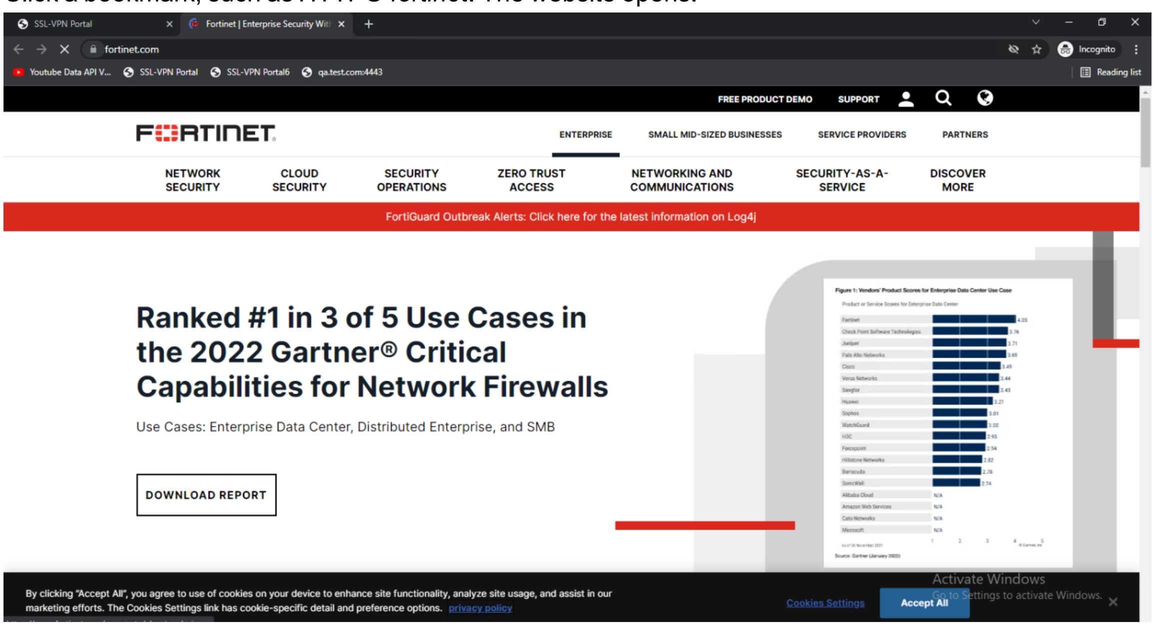
1. From the client browser, go to https://web.test.com:4443/webportal to access the ZTNA access proxy web portal.



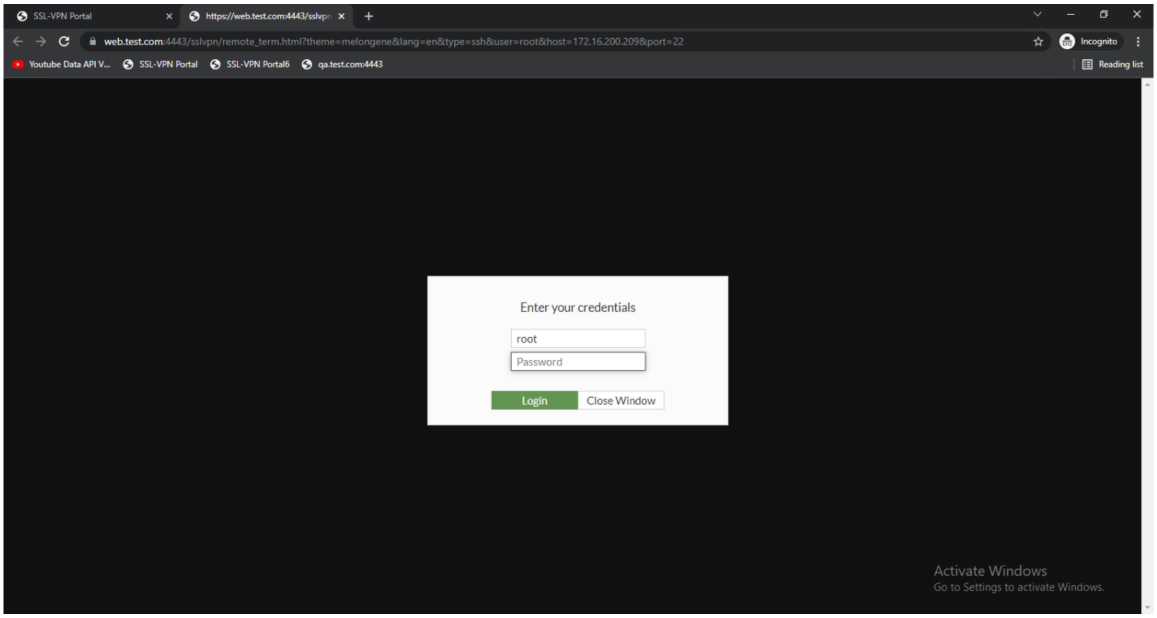
2. Once the client passes the certificate check, posture check, and access is granted, the user is redirected to the web portal. The list of predefined bookmarks appears.



3. Click a bookmark, such as *HTTPS-fortinet*. The website opens.



4. From the web portal, click another bookmark, such as *SSH*. The page opens with the credential login screen to access the server.

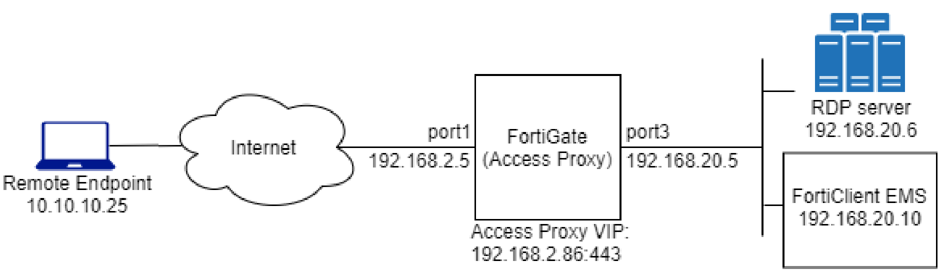


Posture check verification for active ZTNA proxy session examples

Endpoint posture changes trigger active ZTNA proxy sessions to be re-verified and terminated if the endpoint is no longer compliant with the ZTNA policy.

The FortiGate monitors changes to the endpoint tags that are updated by EMS with the fcnacd process. When a change is detected, the endpoint's active ZTNA sessions must match the ZTNA policy again before data can pass.

Changes to the ZTNA policy, such as changing the ZTNA tag matching logic, will also trigger re-verification of the client device against the policy.



The remote endpoint accesses the RDP server through the TCP forwarding access proxy. The proxy is managed by the FortiClient EMS server, which has a ZTNA tagging rule that assigns the *AV-enabled* tag to endpoints that have Windows antivirus enabled, and the *Low* risk host tag to endpoints that are low risk.

These examples assume that the FortiGate EMS fabric connector has already connected successfully, and a ZTNA server named WIN2K16-P1-RDP that forwards traffic to the RDP server has been configured.

Example 1 - The ZTNA tag status changes on the endpoint

In this example, a ZTNA rule is configured to allow access for endpoints that have the *AV-enabled* tag. After an RDP sessions is established, Windows antivirus is disabled on the remote endpoint. The FortiGate re-verifies the session and the active RDP session is removed from the FortiGate session table, causing the RDP session to be disconnected.

To configure the ZTNA rule in the GUI:

1. Go to *Policy & Objects > ZTNA*, select the *ZTNA Rules* tab, and click *Create New*.
2. Set *Name* to *TCP-forward-WIN2K16*.
3. Set *Incoming Interface* to *port1*.
4. Set *Source* to *all*.
5. In *ZTNA Tag* add *AV-enabled*
6. In *ZTNA Server* add *WIN2K16-P1-RDP*.
7. Set *Destination* to *all*.
8. Set *Action* to *ACCEPT*.
9. Configure the remaining options as needed.
10. Click *OK*.

To configure the ZTNA rule in the CLI:

```
config firewall proxy-policy
    edit 4
        set name "TCP-forward-WIN2K16"
        set proxy access-proxy
        set access-proxy "WIN2K16-P1-RDP"
        set srcintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set ztna-ems-tag "FCTEMS0000109188_AV-enabled"
        set action accept
        set schedule "always"
        set logtraffic all
    next
end
```

To test the example:

1. On the remote endpoint, open FortiClient.
2. On the *Zero Trust Telemetry* tab, make sure that you are connected to the EMS server.
3. Add a ZTNA rule:
   a. On the *ZTNA Connection Rules* tab, click *Add Rule*.
   b. Configure the ZTNA rule:

Rule Name	RDP-WIN2K16
Destination Host	192.168.20.6:3389
Proxy Gateway	192.168.2.86:443
Encryption	Disabled

   c. Click *Create*.
4. Ensure that the endpoint has Windows antivirus enabled.
5. Open an RDP session to connect to the RDP server at 192.168.20.6.
6. After a successful connection, on the FortiGate:

a. The endpoint is detected and marked with the *AV-enabled* tag:

```
# diagnose test application fcnacd 7

ZTNA Cache V2:
Entry #1:

 - UID: F4F3263AEBE54777A6509A8FCCDF9284
 - Domain:
 - User: keithli
 - Owner:
 - Certificate SN: 1626C2C10E6AD97D71FA9E2D9C314C1F5C03D68B
 - EMS SN: FCTEMS0000109188
 - online: true
 - Tags (3):
  -- Tag (#0): AV-enabled
  -- Tag (#1): all_registered_clients
  -- Tag (#2): Low
lls_idx_mask = 0x00000001,
```

b. A session is created:

```
# diagnose sys session filter dst 192.168.2.86
# diagnose sys session filter src 10.10.10.25
# diagnose sys session list

session info: proto=6 proto_state=01 duration=191 expire=3599 timeout=3600
flags=00000000 socktype=0 sockport=1012 av_idx=0 use=3
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/0
state=log local may_dirty f24
statistic(bytes/packets/allow_err): org=58031/376/1 reply=66864/351/1 tuples=2
tx speed(Bps/kbps): 303/2 rx speed(Bps/kbps): 349/2
orgin->sink: org pre->in, reply out->post dev=3->7/7->3 gwy=192.168.2.86/0.0.0.0
hook=pre dir=org act=noop 10.10.10.25:60668->192.168.2.86:443(0.0.0.0:0)
hook=post dir=reply act=noop 192.168.2.86:443->10.10.10.25:60668(0.0.0.0:0)
pos/(before,after) 0/(0,0), 0/(0,0)
src_mac=08:5b:0e:ea:7f:d4
misc=7 policy_id=4 pol_uuid_idx=14853 auth_info=0 chk_client_info=0 vd=0
serial=00000c0b tos=00/00 app_list=0 app=0 url_cat=0
sdwan_mbr_seq=0 sdwan_service_id=0
rpdb_link_id=00000000 rpdb_svc_id=0 ngfwid=n/a
npu_state=00000000
total session 1
```

c. The forward traffic log indicates that traffic is allowed:

```
# execute log filter category 0
# execute log filter field dstip 192.168.20.6
# execute log display
...
11: date=2021-10-18 time=11:22:16 eventtime=1634581336644493852 tz="-0700"
logid="0000000024" type="traffic" subtype="forward" level="notice" vd="root"
srcip=10.10.10.25 srcport=60660 srcintf="port1" srcintfrole="wan"
dstcountry="Reserved" srccountry="Reserved" dstip=192.168.20.6 dstport=3389
dstintf="root" dstintfrole="undefined" sessionid=2550 srcuuid="5445be2e-5d7b-51ea-
```

```
e2c3-ae6b7855c52f" service="RDP" proto=6 action="accept" policyid=4
policytype="proxy-policy" poluuid="ce8f82d0-8fb3-51eb-0a17-5e6a6a51ff27"
policyname="TCP-forward-WIN2K16" duration=0 wanin=1578 rcvdbyte=1578 wanout=1107
lanin=2788 sentbyte=2788 lanout=3750 srchwvendor="Fortinet" devtype="Network"
srcfamily="Firewall" osname="FortiOS" srchwversion="FortiWiFi-30E" appcat="unscanned"
```

7. On the remote endpoint, disable Windows antivirus.
   FortiClient EMS detects a change in ,and removes the *AV-enabled* tag on the FortiClient endpoint.
8. Due to the change in posture, the RDP session is disconnected:
   a. The endpoint is no longer marked with the *AV-enabled* tag:

```
# diagnose test application fcnacd 7

ZTNA Cache V2:
Entry #1:

 - UID: F4F3263AEBE54777A6509A8FCCDF9284
 - Domain:
 - User: keithli
 - Owner:
 - Certificate SN: 1626C2C10E6AD97D71FA9E2D9C314C1F5C03D68B
 - EMS SN: FCTEMS0000109188
 - online: true
 - Tags (2):
  -- Tag (#0): all_registered_clients
  -- Tag (#1): Low
lls_idx_mask = 0x00000001,
```

   b. The previous session is removed:

```
# diagnose sys session filter dst 192.168.2.86
# diagnose sys session filter src 10.10.10.25
# diagnose sys session list
total session 0
```

   c. The forward traffic log indicates that traffic is denied:

```
# execute log display
7: date=2021-10-18 time=11:31:45 eventtime=1634581905530844852 tz="-0700"
logid="0000000013" type="traffic" subtype="forward" level="notice" vd="root"
srcip=10.10.10.25 srcport=60668 srcintf="port1" srcintfrole="wan" dstip=192.168.20.6
dstport=3389 dstintf="root" dstintfrole="undefined" srcuuid="5445be2e-5d7b-51ea-e2c3-
ae6b7855c52f" dstuuid="5445be2e-5d7b-51ea-e2c3-ae6b7855c52f" srccountry="Reserved"
dstcountry="Reserved" sessionid=3083 proto=6 action="deny" policyid=4
policytype="proxy-policy" poluuid="ce8f82d0-8fb3-51eb-0a17-5e6a6a51ff27"
policyname="TCP-forward-WIN2K16" service="RDP" trandisp="noop" duration=0 sentbyte=0
rcvdbyte=0 sentpkt=0 rcvdpkt=0 appcat="unscanned" utmaction="block" countztna=1
msg="Denied: failed to match a proxy-policy" utmref=65349-5754
```

   d. The ZTNA log indicates that traffic is denied:

```
# execute log filter category 21
# execute log display
6: date=2021-10-18 time=11:31:45 eventtime=1634581905530840484 tz="-0700"
logid="2101060510" type="utm" subtype="ztna" eventtype="ztna-policy-match"
level="warning" vd="root" msg="Connection is blocked due to unable to match a proxy-
policy" policyid=4 sessionid=3083 srcip=10.10.10.25 dstip=192.168.20.6 srcport=60668
```

```
dstport=3389 srcintf="port1" srcintfrole="wan" dstintf="root" dstintfrole="undefined"
proto=6 action="blocked" service="HTTPS" gatewayid=1 vip="WIN2K16-P1-RDP"
accessproxy="WIN2K16-P1-RDP" clientdeviceid="F4F3263AEBE54777A6509A8FCCDF9284"
clientdevicetags="MAC_FCTEMS0000109188_Low/FCTEMS0000109188_all_registered_
clients/MAC_FCTEMS0000109188_all_registered_clients/FCTEMS0000109188_Low"
```

Example 2 - The ZTNA rule tag checking logic changes

In this example, a ZTNA rule is configured to allow access to endpoints that have at least one of the *AV-enabled* or *Low* ZTNA tags. A remote user who has Windows antivirus disabled, but is low risk, successfully establishes an RDP session over the ZTNA access proxy. An administrator changes the ZTNA rule's tag matching logic from *Any* to *All*, causing the RDP session to be disconnected.

To configure the ZTNA rule in the GUI:

1. Go to *Policy & Objects > ZTNA*, select the *ZTNA Rules* tab.
2. Edit the *TCP-forward-WIN2K16* rule.
3. In *ZTNA Tag*, add *Low*.
4. Ensure that *Match ZTNA Tags* is set to *Any*.
5. Click *OK*.

To configure the ZTNA rule in the CLI:

```
config firewall proxy-policy
    edit 4
        set name "TCP-forward-WIN2K16"
        set proxy access-proxy
        set access-proxy "WIN2K16-P1-RDP"
        set srcintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set ztna-ems-tag "FCTEMS0000109188_AV-enabled" "FCTEMS0000109188_Low"
        set ztna-tags-match-logic or
        set action accept
        set schedule "always"
        set logtraffic all
    next
end
```

To test the example:

1. On the remote Windows PC, disable antivirus protection.
2. Open an RDP session to connect to the RDP server at 192.168.20.6.
3. After a successful connection, on the FortiGate:
   a. The endpoint is detected and marked with the *Low* tag, but not the *AV-enabled* tag:

   ```
   # diagnose test application fcnacd 7

   ZTNA Cache V2:
   Entry #1:

    - UID: F4F3263AEBE54777A6509A8FCCDF9284
   ```

```
 - Domain:
 - User: keithli
 - Owner:
 - Certificate SN: 1626C2C10E6AD97D71FA9E2D9C314C1F5C03D68B
 - EMS SN: FCTEMS0000109188
 - online: true
 - Tags (2):
  -- Tag (#0): all_registered_clients
  -- Tag (#1): Low
lls_idx_mask = 0x00000001,
```

b. A session is created:

```
# diagnose sys session filter dst 192.168.2.86
# diagnose sys session filter src 10.10.10.25
# diagnose sys session list

session info: proto=6 proto_state=01 duration=29 expire=3598 timeout=3600
flags=00000000 socktype=0 sockport=1012 av_idx=0 use=3
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/0
state=log local may_dirty f24
statistic(bytes/packets/allow_err): org=54763/299/1 reply=90223/313/1 tuples=2
tx speed(Bps/kbps): 1860/14 rx speed(Bps/kbps): 3064/24
orgin->sink: org pre->in, reply out->post dev=3->7/7->3 gwy=192.168.2.86/0.0.0.0
hook=pre dir=org act=noop 10.10.10.25:55147->192.168.2.86:443(0.0.0.0:0)
hook=post dir=reply act=noop 192.168.2.86:443->10.10.10.25:55147(0.0.0.0:0)
pos/(before,after) 0/(0,0), 0/(0,0)
src_mac=08:5b:0e:ea:7f:d4
misc=7 policy_id=4 pol_uuid_idx=14853 auth_info=0 chk_client_info=0 vd=0
serial=00003255 tos=00/00 app_list=0 app=0 url_cat=0
sdwan_mbr_seq=0 sdwan_service_id=0
rpdb_link_id=00000000 rpdb_svc_id=0 ngfwid=n/a
```

c. The forward traffic log indicates that traffic is allowed:

```
# execute log filter category 0
# execute log display
...
1: date=2021-10-18 time=12:46:01 eventtime=1634586361077487880 tz="-0700"
logid="0000000024" type="traffic" subtype="forward" level="notice" vd="root"
srcip=10.10.10.25 srcport=55140 srcintf="port1" srcintfrole="wan"
dstcountry="Reserved" srccountry="Reserved" dstip=192.168.20.6 dstport=3389
dstintf="root" dstintfrole="undefined" sessionid=12542 srcuuid="5445be2e-5d7b-51ea-
e2c3-ae6b7855c52f" service="RDP" proto=6 action="accept" policyid=4
policytype="proxy-policy" poluuid="ce8f82d0-8fb3-51eb-0a17-5e6a6a51ff27"
policyname="TCP-forward-WIN2K16" duration=138 wanin=140349 rcvdbyte=140349
wanout=47118 lanin=48799 sentbyte=48799 lanout=142521 appcat="unscanned"
```

4. On the FortiGate, edit the ZTNA rule *TCP-forward-WIN2K16*:
   - In the GUI, set *Match ZTNA Tags* to *All*.
   - In the CLI, set `ztna-tags-match-logic` to `and`.
5. Due to the ZTNA rule update, the FortiGate re-verifies the session, and the RDP session is disconnected:

a. The previous session is removed:

```
# diagnose sys session filter dst 192.168.2.86
# diagnose sys session filter src 10.10.10.25
# diagnose sys session list
total session 0
```

b. The ZTNA log indicates that traffic is denied:

```
# execute log filter category 21
# execute log display
1: date=2021-10-18 time=12:53:57 eventtime=1634586837921889075 tz="-0700"
logid="2101060510" type="utm" subtype="ztna" eventtype="ztna-policy-match"
level="warning" vd="root" msg="Connection is blocked due to unable to match a proxy-
policy" policyid=0 sessionid=13865 srcip=10.10.10.25 dstip=192.168.2.86 srcport=55162
dstport=443 srcintf="port1" srcintfrole="wan" dstintf="root" dstintfrole="undefined"
proto=6 action="blocked" service="HTTPS" gatewayid=1 vip="WIN2K16-P1-RDP"
accessproxy="WIN2K16-P1-RDP" clientdeviceid="F4F3263AEBE54777A6509A8FCCDF9284"
clientdevicetags="MAC_FCTEMS0000109188_Low/FCTEMS0000109188_all_registered_
clients/MAC_FCTEMS0000109188_all_registered_clients/FCTEMS0000109188_Low"
```

ZTNA TCP forwarding access proxy with FQDN example

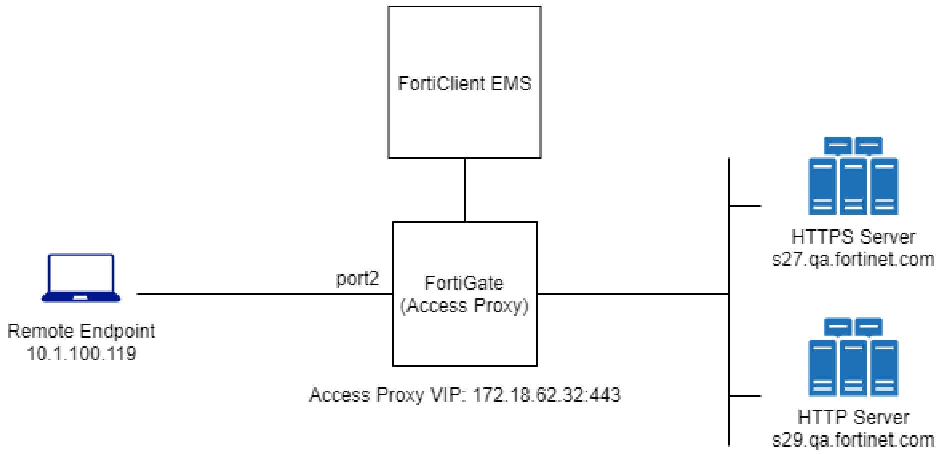
When defining ZTNA connection rules on FortiClient for TCP forwarding, it is sometimes desirable to configure the destination host address as an FQDN address instead of an IP address. Since the real servers are often servers in the corporate network, this layer of obfuscation prevents internal IPs from easily leaking to the public, and also makes the destination more easily recognizable by the end users.

One obstacle to overcome is getting remote hosts to resolve an internal FQDN that is typically only resolvable by an internal DNS in the corporate network. This can be solved with the following:

1. When an FQDN address is added as a destination host in a ZTNA connection rule, FortiClient creates a virtual IP for this FQDN address and adds this to the computer's host file (Windows). The same is true when a ZTNA connection rule entry is pushed from EMS.
2. The virtual IP mapped to the FQDN address is not the real address of the server. It allows applications to resolve the FQDN address to this virtual IP. FortiClient listens to any traffic destined for it and forwards the traffic using the TCP forwarding URL with FQDN to the ZTNA access proxy.
3. The FortiGate access proxy will resolve the FQDN using the internal DNS on the corporate network, matching the traffic to the ZTNA real server configuration with the same domain and address.
4. If a valid ZTNA real server entry is found, traffic is forwarded to the real server.

Example

In this example, two servers in the internal network are added to the FortiGate access proxy for TCP forwarding. The remote client configures two ZTNA connection rules, with the destination host field pointing to the FQDN addresses of the internal servers. These FQDN addresses are configured in the FortiGate's DNS database so they can be resolved by the FortiGate. It is recommended to use an internal DNS server for production environments.



This example assumes that the FortiGate EMS Fabric connector is already successfully connected.

This features requires a minimum FortiClient and FortiClient EMS version of 7.0.3.

To configure the TCP forwarding access proxy:

1. Go to *Policy & Objects > ZTNA* and select the *ZTNA Servers* tab.
2. Click *Create New*.
3. Set *Name* to *ZTNA_S1*.
4. Configure the network settings:
   a. Set *External interface* to *any*.
   b. Set *External IP* to *172.18.62.32*.
   c. Set *External port* to *443*.
5. Select the *Default certificate*. Clients will be presented with this certificate when they connect to the access proxy VIP.
6. Add server mapping:
   a. In the *Service/server mapping* table, click *Create New*.
   b. For *Service*, select *TCP Forwarding*.
   c. Add a server:
      i. In the *Servers* table, click *Create New*.
      ii. Create a new FQDN address for the HTTPS server at s27.qa.fortinet.com, then click *OK*.
      iii. Apply the new address object as the address for the new server.
      iv. Click *OK*.
   d. Add another server using the same steps for s29.qa.fortinet.com.
7. Click *OK*. Now that the ZTNA server is complete, the domain settings must be configured in the CLI to map domains to the real servers.

To map domains to the real servers:

```
config firewall access-proxy
    edit "ZTNA_S1"
        set vip "ZTNA_S1"
        set client-cert enable
        config api-gateway
```

```
            edit 2
                set url-map "/tcp"
                set service tcp-forwarding
                config realservers
                    edit 4
                        set address "s27.qa.fortinet.com"
                        set domain "qa.fortinet.com"
                    next
                    edit 5
                        set address "s29.qa.fortinet.com"
                        set domain "qa.fortinet.com"
                    next
                end
            next
        end
    next
end
```

To configure the ZTNA rule:

1. Go to *Policy & Objects > ZTNA* and select the *ZTNA Rules* tab.
2. Click *Create New*.
3. Set *Name* to *ZTNA_TCP*.
4. Set *Incoming Interface* to *port2*.
5. Set *Source* to *all*.
6. Select the ZTNA server *ZTNA_S1*.
7. Configure the remaining options as needed.
8. Click *OK*.

To configure the DNS entries for each server:

1. Enable the DNS database visibility:
   a. Go to *System > Feature Visibility*.
   b. Enable *DNS Database*.
   c. Click *Apply*.
2. Go to *Network > DNS Servers*. Under *DNS Database*, click *Create New*.
3. Set *DNS Zone* to *ZTNA*.
4. Set *Domain Name* to *qa.fortinet.com*.
5. Add the DNS entries:
   a. Under *DNS Entries*, click *Create New*.
   b. Set *Hostname* to *s27*.
   c. Set *IP Address* to the HTTPS server address.
   d. Click *OK*.
   e. Add another DNS entry using the same steps for the s29.qa.fortinet.com HTTP server.
6. Click *OK*.

Testing the connection to the access proxy

Before connecting, users must have a ZTNA connection rule in FortiClient.

ZTNA TCP forwarding rules can be provisioned from the EMS server. See Provisioning ZTNA TCP forwarding rules via EMS for more details.

To create the ZTNA rules in FortiClient and connect:

1. From the *ZTNA Connection Rules* tab, click *Add Rule*.
2. Create a rule for the HTTPS server:
   a. Set *Rule Name* to *server27*.
   b. Set *Destination Host* to *s27.qa.fortinet.com:443*.
   c. Set *Proxy Gateway* to *172.18.62.32:443*.
   d. Disable *Encryption*.
   e. Click *Create*.
3. Create a rule for the HTTP server:
   a. Set *Rule Name* to *server29*.
   b. Set *Destination Host* to *s29.qa.fortinet.com:80*.
   c. Set *Proxy Gateway* to *172.18.62.32:443*.
   d. Disable *Encryption*.
   e. Click *Create*.
4. Upon creating the ZTNA rules, two new entries are added to the Windows PC's host file in folder C:\Windows\System32\drivers\etc. View the file, and observe the new entries for the virtual IP and FQDN pairing for each ZTNA connection rule.

```
# ----- FORTICLIENT ZTNA VIP START -----
10.235.0.1 s27.qa.fortinet.com
10.235.0.2 s29.qa.fortinet.com
# ----- FORTICLIENT ZTNA VIP END -----
```

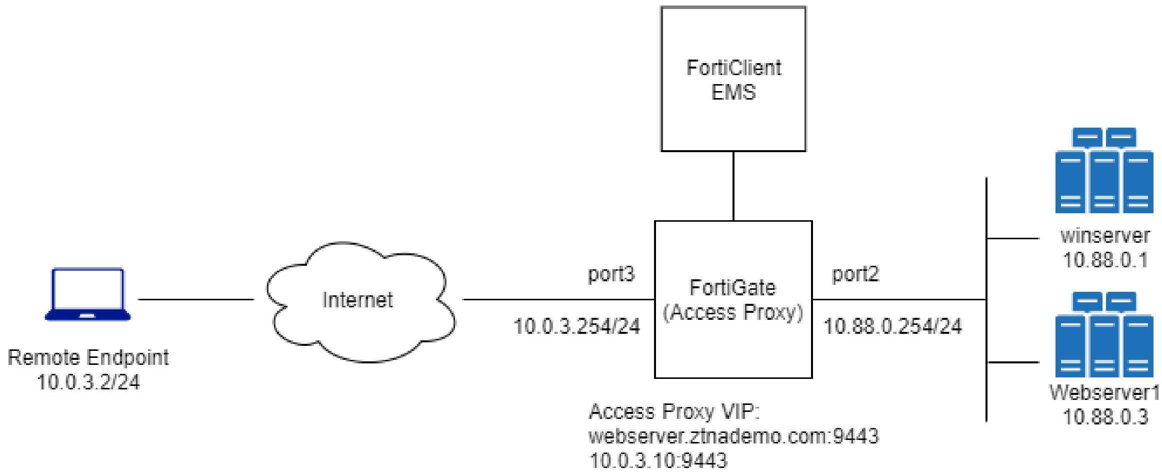
5. The Windows PC now resolves the FQDNs to the virtual IPs, and FortiClient will listen to the traffic to these IPs and forward them to the TCP access proxy.
6. Have the remote user connect to the HTTPS and HTTP servers on a browser. After device verification, the user is able to successfully connect to the remote servers.

ZTNA session-based form authentication

Session-based form authentication for ZTNA allows users to log in through an authentication portal with support for multi-factor authentication (MFA). This added advantage over the basic type authentication method allows FortiToken MFA to be applied directly to FortiGate users. FortiToken MFA can be applied to local users or remote users. Session-based form authentication can also be applied to explicit and transparent web proxies.

Example

In this example, the FortiGate is configured with a ZTNA HTTPS access proxy to protect access to the web server. It uses session-based form authentication with cookies and `auth-portal` enabled. It connects to the internal Windows Active Directory using LDAPS for user authentication, and assigns FortiToken MFA to individual users.



This example assumes that the FortiGate EMS Fabric connector is already successfully connected.

To configure the LDAP server:

1. Go to *User & Authentication > LDAP Servers* and click *Create New*.
2. Configure the following settings:

Name	*LDAP-fortiad*
Server IP/Name	*10.88.0.1*
Server Port	*389*
Common Name Identifier	*sAMAccountName*
Distinguished Name	*dc=fortiad,dc=info*
Exchange server	Disable this setting.
Bind Type	*Regular* Enter the *Username* and *Password* for LDAP binding and lookup.
Secure Connection	Enable and set the *Protocol* to *LDAPS*.
Certificate	Enable and select the CA certificate to validate the server certificate.
Server identity check	Optionally, enable to verify the domain name or IP address against the server certificate.

3. Click *Test Connectivity* to verify the connection to the server.
4. Click *OK*.

To configure a user with FortiToken MFA:

1. Go to *User & Authentication > User Definition* and click *Create New*.
2. Set *User Type* to *Remote LDAP User* and click *Next*.
3. Set *LDAP Server* to *LDAP-fortiad* and click *Next*.
4. For *Remote Users*, right-click on a user from the list under the corresponding OU and click *Add Selected*. In this example, the user *tsmith* under the *Marketing* OU is selected.
5. Click *Submit*.

6. Double-click the new user, *tsmith*, to edit the settings.
7. Enable *Two-factor Authentication*. Select either *FortiToken Cloud* or *FortiToken*. In this example, *FortiToken* is selected with a mobile FortiToken available on this FortiGate.
8. Enter an *Email Address* for the user to get a token activation notification.
9. Click *OK*.

To configure a user group:

1. Go to *User & Authentication* > *User Groups* and click *Create New*.
2. Enter the name of the group, *FortiAD-MFA-group*.
3. Set *Type* to *Firewall*.
4. Click the +in the *Members* field and add the user, *tsmith*.
5. Click *OK*.

To configure the authentication scheme:

1. Go to *Policy & Objects* > *Authentication Rules* and click *Create New* > *Authentication Scheme*.
2. Enter the name, *ZTNA-Auth-scheme*.
3. Set *Method* to *Form-based*.
4. Set *User database* to *Other* and select the *LDAP-fortiad* LDAP server.
5. Enable *Two-factor authentication*.
6. Click *OK*.

To configure the authentication rule:

```
config authentication rule
    edit "ztna_form_rule"
        set srcaddr "all"
        set ip-based disable
        set active-auth-method "ZTNA-Auth-scheme"
        set web-auth-cookie enable
    next
end
```

By disabling `ip-based`, the rule is session-based, so web authentication cookies must be enabled.

To configure the ZTNA basic server settings in the GUI:

Configuring the ZTNA server requires some settings that can only be configured in the CLI. The basic settings are configured in the GUI first, then the advanced CLI-only configurations are added after.

1. Go to *Policy & Objects* > *ZTNA* and select the *ZTNA Servers* tab.
2. Click *Create New*.
3. Enter the server name, *ZTNA_S1*.
4. Configure the network settings:

   a. Set *External interface* to *port3*.
   b. Set *External IP* to *10.0.3.10*.
   c. Set *External port* to *9443*.
5. Select the *Default certificate*. Clients will be presented with this certificate when they connect to the access proxy VIP. In this example, the custom certificate, *ztna-wildcard* is selected.
6. Add server mapping:
   a. In the *Service/server mapping* table, click *Create New*.
   b. Set *Service* to *HTTPS*.
   c. Set *Virtual Host* to *Any Host*.
   d. Configure the path as needed.
   e. Add a server:
      i. In the *Servers* table, click *Create New*.
      ii. Set *IP* to *10.88.0.3*.
      iii. Set *Port* to *9443*.
      iv. Click *OK* to complete the server settings.
   f. Click *OK* to complete the HTTPS service mapping.
7. Click *OK*.

To configure the advanced authentication settings in the CLI:

The following steps are required to create a virtual host and to enable the authentication portal.

1. Create an access proxy virtual host that points to the ZTNA access proxy. The FQDN of the host must be able to resolve to the external address 10.0.3.10. The client will be redirected to this page for form authentication:

```
config firewall access-proxy-virtual-host
    edit "auth-portal-vhost"
        set ssl-certificate "ztna-wildcard"
        set host "authportal.ztnademo.com"
    next
end
```

2. Enable `auth-portal` on the access proxy and point it to the virtual host:

```
config firewall access-proxy
    edit "ZTNA_S1"
        set auth-portal enable
        set auth-virtual-host "auth-portal-vhost"
    next
                              end
```

When `auth-virtual-host` is configured in the access proxy, it acts as a single sign-on (SSO) point. This means users will be authenticated once when accessing any domains or services in ZTNA_S1.

When `auth-virtual-host` is not configured, users will be re-authenticated for each domain or service in ZTNA_S1.

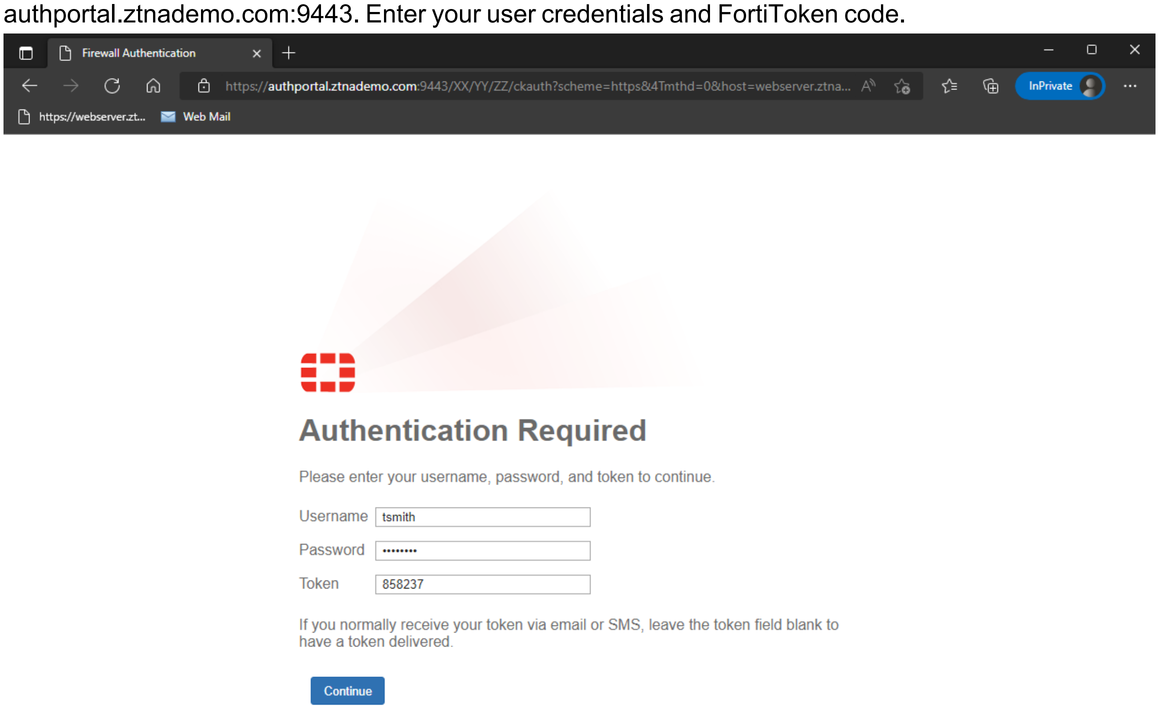
To apply the authentication to the ZTNA rule:

1. Go to *Policy & Objects > ZTNA* and select the *ZTNA Rules* tab.
2. Click *Create New*.
3. Enter the name, *ZTNA_R1*.
4. Set *Incoming Interface* to *port3*.
5. Set *Source* to *all*. This can also be set to specific IP addresses to only allow those addresses to connect to this HTTPS access proxy.
6. Click the + in the *Source* and from the *User* tab, select the *FortiAD-MFA-group* user group.
7. Click the + in the *ZTNA Tag* field and select the *Low* tag.
8. Set *ZTNA Server* to *ZTNA_S1*.
9. Set *Destination* to *Webserver1*, which is an address object for 10.88.0.3/32.
10. Configure the remaining options as needed.
11. Click *OK*.

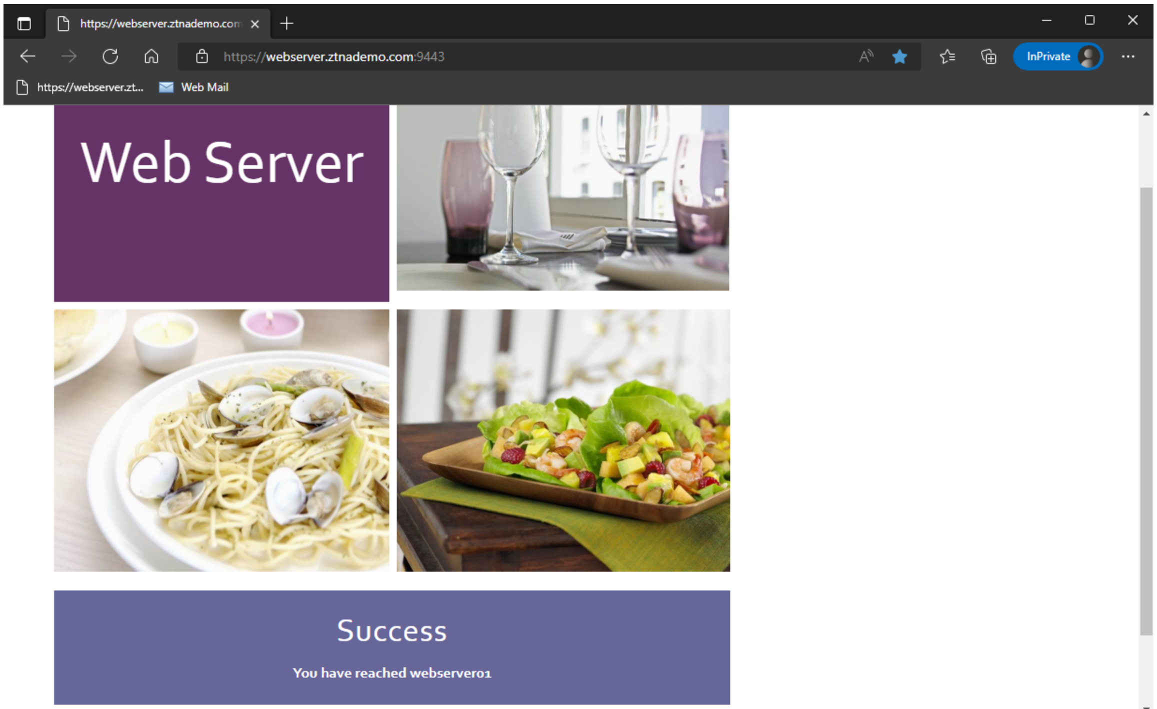
Testing the connection

To test the remote access to the HTTPS access proxy with user authentication:

1. On the remote Windows PC, open FortiClient.
2. From the *Zero Trust Telemetry* tab, make sure that you are connected to the EMS server.
3. Open a browser and enter the address or FQDN of the server and the access port. In this example, https://webserver.ztnademo.com:9443 resolves to https://10.0.3.10:9443.
4. The browser prompts for the client certificate to use. Select the EMS signed certificate, then click *OK*.
5. The client is verified by the FortiGate to authenticate your identity.
6. Form authentication redirects you to the captive portal defined by the `auth-virtual-host`, authportal.ztnademo.com:9443. Enter your user credentials and FortiToken code.



7. After the user authentication passes, the FortiGate performs a posture check on the endpoint. When the posture check passes, you are allowed access to the website.



To verify the logs:

1. Verify the logged in users in the WAD daemon:

```
# diagnose wad user list
ID: 2, VDOM: root, IPv4: 10.0.3.2
  user name    : tsmith
  worker       : 1
  duration     : 42
  auth_type    : Session
  auth_method  : Form
  pol_id       : 1
  g_id         : 4
  user_based   : 0
  expire       : no
  LAN:
    bytes_in=5117 bytes_out=302717
  WAN:
    bytes_in=304915 bytes_out=4407
```

2. Verify the endpoint information:

```
# diagnose endpoint record list
Record #1:
                IP Address = 10.0.3.2
                MAC Address = 02:09:0f:00:03:03
                MAC list = 02:09:0f:00:04:03;02:09:0f:00:03:03;
                VDOM =  (-1)
                EMS serial number: FCTEMS8822000000
                Client cert SN: 5BDEE2D7B7FCA460D9CEC67BBF4D1FA33E3D281A
```

```
                Public IP address: 67.249.72.215
                Quarantined: no
                Online status: online
                Registration status: registered
                On-net status: on-net
                Gateway Interface:
                FortiClient version: 7.0.2
                AVDB version: 1.0
                FortiClient app signature version: 13.364
                FortiClient vulnerability scan engine version: 2.31
                FortiClient UID: 9A016B5A6E914B42AD4168C066EB04CA
                Host Name: WIN10-01
                OS Type: WIN64
                OS Version: Microsoft Windows 10 Professional Edition, 64-bit  (build
19042) (version 2009)
                Host Description:
                Domain: fortiad.info
                Last Login User: tsmith
                …
                Number of Routes: (0)
online records: 1; offline records: 0; quarantined records: 0
```

3. Verify the detected tags on the endpoint:

```
# diagnose test app fcnacd 7
ZTNA Cache V2:
Entry #1:
 - UID: 9A016B5A6E914B42AD4168C066EB04CA
 - EMS SN: FCTEMS88220010000
 - Domain: fortiad.info
 - User: tsmith
 - Owner:
 - Certificate SN: 5BDEE2D7B7FCA460D9CEC67BBF4D1FA33E3D281A
 - online: true
 - Tags (2):
  -- Tag (#0): all_registered_clients
  -- Tag (#1): Low
lls_idx_mask = 0x00000001,
```

4. Verify the ZTNA logs.
   - In the GUI, go to *Log & Report > ZTNA Traffic*.
   - In the CLI:

```
# execute log filter category 0
# execute log filter field subtype ztna
# execute log display
17 logs found.
10 logs returned.

1: date=2022-05-19 time=13:04:41 eventtime=1652990680922903215 tz="-0700"
logid="0005000024" type="traffic" subtype="ztna" level="notice" vd="root"
srcip=10.0.3.2 srcport=63111 srcintf="port3" srcintfrole="wan" dstcountry="Reserved"
srccountry="Reserved" dstip=10.88.0.3 dstport=9443 dstintf="root"
dstintfrole="undefined" sessionid=8313 service="tcp/9443" proto=6 action="accept"
policyid=1 policytype="proxy-policy" poluuid="b513a216-d7a9-51ec-7965-6ba166e99004"
policyname="ZTNA_R1" duration=66 user="tsmith" group="FortiAD-MFA-group" gatewayid=1
vip="ZTNA_S1" accessproxy="ZTNA_S1" clientdeviceid="9A016B5A6E914B42AD4168C066EB04CA"
```

```
clientdevicetags="MAC_FCTEMS8822000000_Low/FCTEMS8822000000_all_registered_
clients/MAC_FCTEMS8822000000_all_registered_clients" wanin=303042 rcvdbyte=303042
wanout=3925 lanin=4430 sentbyte=4430 lanout=301660
fctuid="9A016B5A6E914B42AD4168C066EB04CA" appcat="unscanned"
```

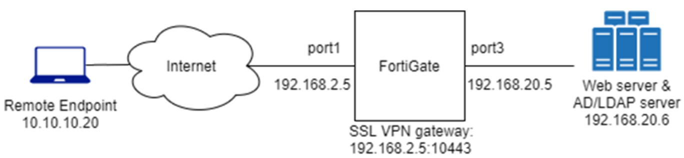
Migrating from SSL VPN to ZTNA

ZTNA can be used to replace VPN-based teleworking solutions to enhance the user experience and to increase security. A typical teleworking configuration may utilize SSL VPN tunnel or web portal mode with LDAP user authentication. Common objects defined for this setup can be reused when migrating to ZTNA, such as the remote LDAP server, user group, and address objects.

SSL VPN teleworking scenarios

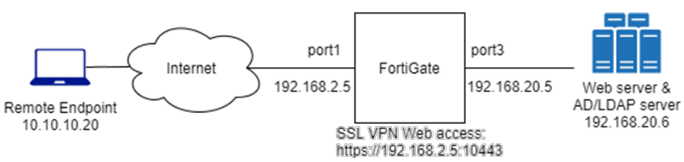
SSL VPN tunnel mode access with LDAP user authentication

Remote users that are in the *ALLOWED-VPN* active directory group have access to a specific web server when they connect through the SSL VPN tunnel. The FortiGate enables split tunneling to the web server so that only traffic to that destination is routed through the tunnel. The web server hosts internal websites that are only accessible by employees.



SSL VPN web mode access with LDAP user authentication

Remote users that are in the *ALLOWED-VPN* active directory group have access to a specific web server when they connect through the SSL VPN web portal. The web server hosts internal websites that are only accessible by employees. The pre-defined bookmark to the internal website is the only site that allows remote access.



Common configurations

This section includes configurations for common objects used in the SSL VPN configuration that can be reused in the ZTNA deployment:

- LDAP server
- User group
- Firewall address for protected server

LDAP server

To configure an LDAP server:

```
config user ldap
    edit "WIN2K16-KLHOME-LDAPS"
        set server "192.168.20.6"
        set server-identity-check disable
        set cnid "sAMAccountName"
        set dn "dc=KLHOME,dc=local"
        set type regular
        set username "KLHOME\\Administrator"
        set password **********
        set secure ldaps
        set ca-cert "CA_Cert_1"
        set port 636
    next
end
```

User group

To configure the user group:

```
config user group
    edit "KLHOME-ALLOWED-VPN"
        set member "WIN2K16-KLHOME-LDAPS"
        config match
            edit 1
                set server-name "WIN2K16-KLHOME-LDAPS"
                set group-name "CN=ALLOWED-VPN,DC=KLHOME,DC=local"
            next
        end
    next
end
```

Firewall address for protected server

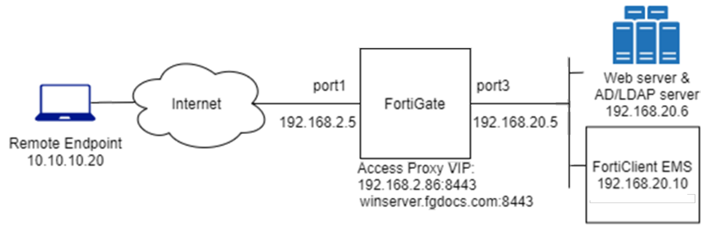
Firewall addresses can be reused in the server settings for TCP forwarding configurations.

To configure the firewall address:

```
config firewall address
    edit "winserver"
        set subnet 192.168.20.6 255.255.255.255
    next
end
```

Migrating to ZTNA

The preceding simple SSL VPN tunnel and web mode teleworking solutions can be migrated to ZTNA configurations, providing device authentication using client certificates and additional security posture checks.



Instead of connecting to the SSL VPN tunnel or web portal, the remote user connects to the HTTPS access proxy that forwards traffic to the web server after authentication and security posture checks are completed. This provides granular control over who can access the web resource using role-based access control. It also gives the user transparent access to the website using only their browser.

Migrating to ZTNA includes the following steps:

1. Connecting to FortiClient EMS
2. Configuring ZTNA tags on FortiClient EMS
3. Configuring a VIP to allow remote users access to FortiClient EMS
4. Configuring the ZTNA server
5. Configuring the authentication scheme and rule
6. Configuring the ZTNA rules

Connecting to FortiClient EMS

The first step to configure ZTNA is to connect to and authorize a FortiClient EMS using the EMS connector. There are different ways to connect to an on-premise FortiClient EMS server and a FortiClient EMS Cloud. Refer to the first step of Configure a FortiClient EMS connector on page 973 for instructions.

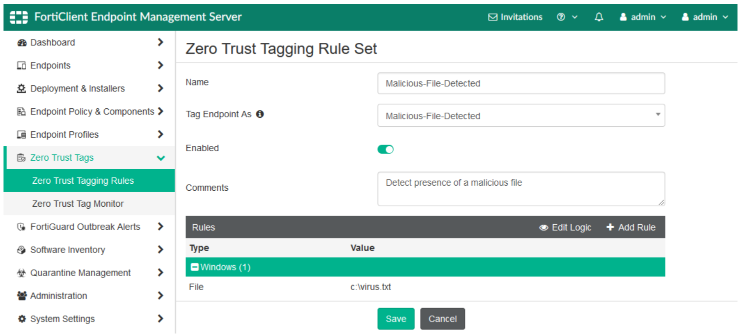
Configuring ZTNA tags on FortiClient EMS

ZTNA tags and tagging rules define security posture checks that connecting devices must pass before they are allowed to access protected resources and applications. In the following example, a Zero Trust tagging rule is configured to detect if a virus file exists on an endpoint.

To configure a Zero Trust tagging rule on the FortiClient EMS:

1. Log in to the FortiClient EMS.
2. Go to *Zero Trust Tags > Zero Trust Tagging Rules*, and click *Add*.
3. In the *Name* field, enter *Malicious-File-Detected*.
4. In the *Tag Endpoint As* dropdown list, select *Malicious-File-Detected*.

   EMS uses this tag to dynamically group together endpoints that satisfy the rule, as well as any other rules that are configured to use this tag.
5. Click *Add Rule* then configure the rule:
   a. For *OS*, select *Windows*.
   b. From the *Rule Type* dropdown list, select *File* and click the + button.
   c. Enter a file name, such as *C:\virus.txt*.
   d. Click *Save*.



6. Click *Save*.

Configuring a VIP to allow remote users access to FortiClient EMS

A ZTNA solution requires users to be registered and connected to the FortiClient EMS server. When an EMS server is behind the FortiGate, a VIP needs to be defined to allow remote users access to register to the FortiClient EMS. The only port required to be forwarded is TCP/8013. This VIP also needs to be applied in a firewall policy to allow this traffic.

To configure a VIP to allow traffic to the EMS server:

1. Go to *Policy & Objects* > *Virtual IPs* and click *Create New* > *Virtual IP*.
2. Set *Name* to *VIP-EMS*.
3. Configure the VIP settings:
   a. Set *Interface* to *port1*.
   b. Set *External IP address/range* to *192.168.2.5*.
   c. Set *Map to* to *192.168.20.10*.
   d. Enable *Port Forwarding*.
   e. Set *External service port* to *8013*.
   f. Set *Map to IPv4 port* to *8013*.
4. Click *OK*.

To configure the firewall policy:

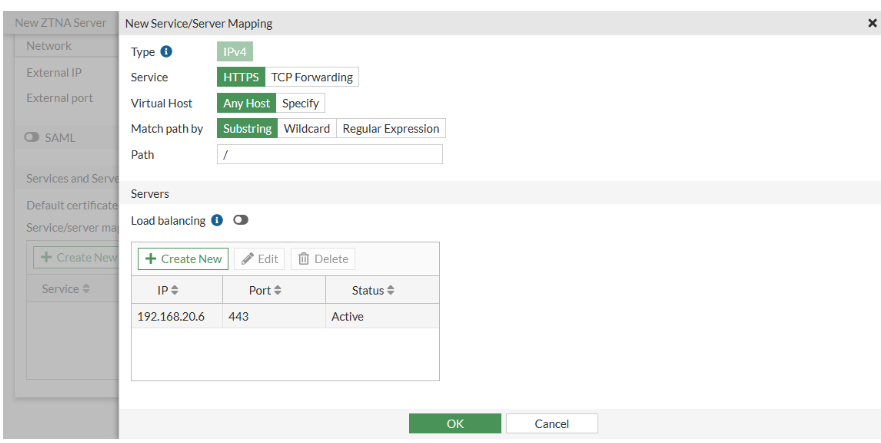
1. Go to *Policy & Objects* >*Firewall Policy* and click *Create New*.
2. Set *Name* to *ZTNA-VIP*.
3. Configure the policy settings:
   a. Set *Incoming Interface* to *port1*.
   b. Set *Outgoing Interface* to *port3*.
   c. Set *Source* to *all*.
   d. Set *Destination* to *VIP-EMS*.
   e. For *Service*, select an option that is for TCP/8013.
   f. Disable *NAT*.
   g. Configure the remaining options as needed.
4. Click *OK*.

Configuring the ZTNA server

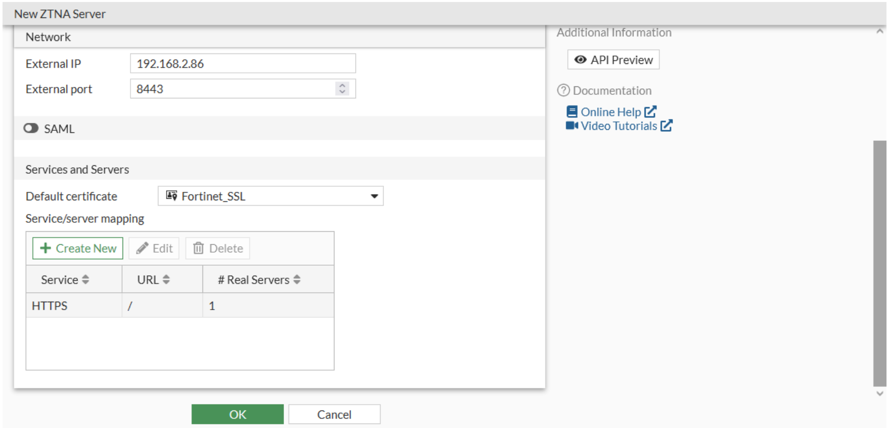
The ZTNA server defines the external IP and port used for the FortiGate access proxy. It also defines the protected resources that can be accessed through the HTTPS access proxy or TCP forwarding access proxy. The following configuration defines a HTTPS access proxy for accessing the web server on 192.168.20.6.

To configure a ZTNA server for HTTPS access proxy:

1. Go to *Policy & Objects > ZTNA* and select the *ZTNA Servers* tab.
2. Click *Create New*.
3. Set *Name* to *WIN2K16-P1*.
4. Configure the network settings:
   a. Set *External interface* to *port1*.
   b. Set *External IP* to *192.168.2.86*.
   c. Set *External port* to *8443*.
5. Select the *Default certificate*. Clients will be presented with this certificate when they connect to the access proxy VIP.
6. Add server mapping:
   a. In the *Service/server mapping* table, click *Create New*.
   b. Set *Service* to *HTTPS*.
   c. Set *Virtual Host* to *Any Host*.
   d. Configure the path as needed. For example, to map to *winserver.fgdocs.com/fortigate*, enter */fortigate*.
   e. Add a server:
      i. In the *Servers* table, click *Create New*.
      ii. Set *IP* to *192.168.20.6*.
      iii. Set *Port* to *443*.
      iv. Click *OK*.



f. Click *OK*.



7. Click *OK*.

Configuring the authentication scheme and rule

The authentication scheme defines the authentication method that is applied. In this example, basic HTTP authentication is used so that users are prompted for a username and password the first time that they connect to a website through the HTTPS access proxy. The LDAP server defined for the SSL VPN configurations can be reused here.

To configure an authentication scheme:

1. Go to *Policy & Objects > Authentication Rules* and click *Create New > Authentication Scheme*.
2. Set the name to *ZTNA-Auth-scheme*.
3. Set *Method* to *Basic*.
4. Set *User database* to *Other* and select *WIN2K16-KLHOME-LDAPS* as the LDAP server.
5. Click *OK*.

The authentication rule defines the proxy sources and destination that require authentication, and what authentication scheme is applied. In this example, active authentication through the basic HTTP prompt is used and applied to all sources.

To configure an authentication rule:

1. Go to *Policy & Objects > Authentication Rules* and click *Create New > Authentication Rule*.
2. Set the name to *ZTNA-Auth-rule*.
3. Set *Source Address* to *all*.
4. Set *Protocol* to *HTTP*.
5. Enable *Authentication Scheme* and select *ZTNA-Auth-scheme*.
6. Click *OK*.

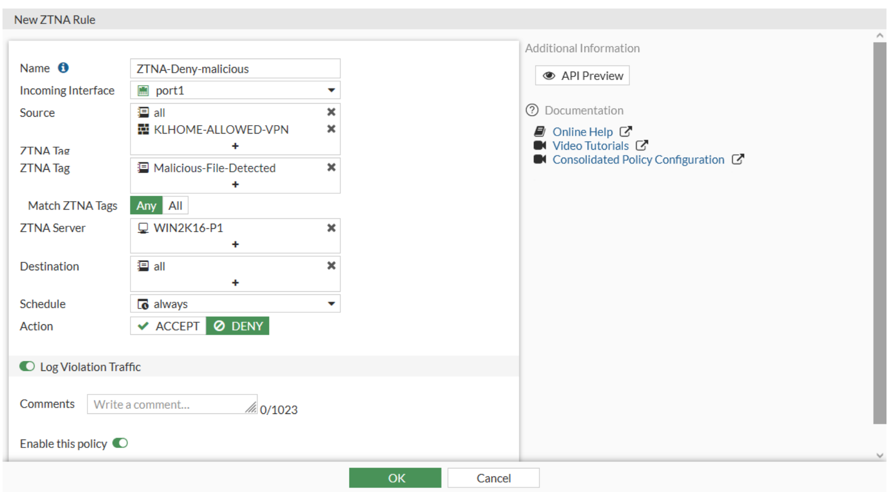
Configuring the ZTNA rules

A user or user group must be applied to the ZTNA rule used to control user access. The authenticated user from the authentication scheme and rule must match the user or user group in the ZTNA rule. The user group, *KLHOME-*

ALLOWED-VPN, defined in the SSL VPN configurations is reused in this example. The ZTNA tag, *Malicious-File-Detected*, is used to define a rule to deny access when the connecting device has the malicious file detected.

To configure ZTNA rules to allow and deny traffic based on ZTNA tags:

1. Go to *Policy & Objects* > *ZTNA* and select the *ZTNA Rules* tab.
2. Create a rule to deny traffic:
   a. Click *Create New*.
   b. Set *Name* to *ZTNA-Deny-malicious*.
   c. Set *Incoming Interface* to *port1*.
   d. Set *Source* to *all*, then click the + and from the *User* tab, select the *KLHOME-ALLOWED-VPN* group.
   e. Add the ZTNA tag *Malicious-File-Detected*.
      This tag is dynamically retrieved from EMS when the Zero Trust tagging rule is first created.
   f. Select the ZTNA server *WIN2K16-P1*.
   g. Set *Action* to *DENY*.
   h. Enable *Log Violation Traffic*.



   i. Click *OK*.
3. Create a rule to allow traffic:
   a. Click *Create New*.
   b. Set *Name* to *proxy-WIN2K16-P1*.
   c. Set *Incoming Interface* to *port1*.
   d. Set *Source* to *all*, then click the + and from the *User* tab, select the *KLHOME-ALLOWED-VPN* group. The *Source* can also be set to specific IP addresses to only allow those addresses to connect to this HTTPS access proxy.
   e. Add the ZTNA tag *Low*.
   f. Select the ZTNA server *WIN2K16-P1*.
   g. Set *Action* to *ACCEPT*.
   h. Configure the remaining options as needed.
   i. Click *OK*.
4. In the *ZTNA Rules* list, make sure that the deny rule (*ZTNA-Deny-malicious*) is above the allow rule (*proxy-WIN2K16-P1*).

Testing the connection

Once ZTNA is configured, connect to the FortiGate access proxy using an endpoint that is registered to EMS. The user should be prompted for their device certificate, username, and password the first time they connect. Once they have authenticated and they pass the security posture checks, they will be allowed to access the website.

See ZTNA HTTPS access proxy example on page 988 and ZTNA HTTPS access proxy with basic authentication example on page 995 for sample verifications and results.

Disabling the SSL VPN

Once testing is complete and the ZTNA servers and policies are configured, the users can be migrated to using ZTNA. Use the following checklist to verify if the remote users are ready to migrate:

1. The users have installed a supported FortiClient version and have installed the ZTNA module.
2. The endpoints can register to FortiClient EMS.
3. If using a TCP forwarding access proxy, ensure that ZTNA rules are either pushed from FortiClient EMS, or the users know how to configure them manually.

Next, SSL VPN access can be disabled in a phased approach by disabling SSL VPN firewall policies that allow access to resources that are accessible using ZTNA.

Once all applications and resources have been migrated, the SSL VPN can be disabled entirely by going to *VPN > SSL-VPN Settings*, and deselecting the *Enable SSL-VPN* toggle.

ZTNA scalability support for up to 50 thousand concurrent endpoints

ZTNA scalability supports up to 50 thousand concurrent endpoints. Communication between FortiOS and FortiClient EMS has efficient queries that request incremental updates. Retrieved device information can be written to the FortiClient NAC daemon cache.

FortiOS can receive tag information from the EMS common tags API. This feature requires FortiClient EMS 7.0.3 or later.

The APIs `api/v1/report/fct/uid_tags` and `api/v1/report/fct/tags` replace the API `api/v1/report/fct/host_tags`.

To use the common tags API capability:

1. Enable the common tags API when connecting the EMS:

```
config endpoint-control fctems
    edit "local.ems"
        set server "10.6.30.213"
        set capabilities fabric-auth silent-approval websocket websocket-malware push-
ca-certs common-tags-api
    next
end
```

2. The FortiGate uses the new APIs to obtain device information from the EMS:

```
[ec_ems_context_submit_work:414] Call submitted successfully.
    obj-id: 11, desc: REST API to get updates of tag endpoints., entry:
```

```
api/v1/report/fct/tags.
[ec_ems_context_submit_work:414] Call submitted successfully.
    obj-id: 12, desc: REST API to get updates of tags associated with FCT UID., entry:
api/v1/report/fct/uid_tags.
[ec_ez_worker_process:334] Processing call for obj-id: 11, entry:
"api/v1/report/fct/tags"
[dynamic_addr_ha_act:215] called (EMS SN N/A).
[dynamic_addr_ha_act:215] called (EMS SN N/A).
[ec_ez_worker_process:441] Call completed successfully.
    obj-id: 11, desc: "REST API to get updates of tag endpoints.", entry:
"api/v1/report/fct/tags".
[ec_ez_worker_process:334] Processing call for obj-id: 12, entry:
"api/v1/report/fct/uid_tags"
[ec_record_sync_tags_info_store:1419] Received 1 tags for
3D86DF70B85E16CBAD67908A897B4494 with sn FCTEMS8888888888
[ec_record_sync_tags_info_store:1419] Received 1 tags for
DA12930442F13F84D2441F03FCB6A10E with sn FCTEMS8888888888
[ec_record_sync_tags_info_store:1419] Received 1 tags for
25C59C275F257F4C5FBC7F6F5F56788E with sn FCTEMS8888888888
[ec_ez_worker_process:441] Call completed successfully.
    obj-id: 12, desc: "REST API to get updates of tags associated with FCT UID.", entry:
"api/v1/report/fct/uid_tags".
[ec_ems_context_submit_work:414] Call submitted successfully.
    obj-id: 7, desc: REST API to get updates about system info., entry:
api/v1/report/fct/sysinfo.
[ec_ems_context_submit_work:414] Call submitted successfully.
    obj-id: 11, desc: REST API to get updates of tag endpoints., entry:
api/v1/report/fct/tags.
[ec_ez_worker_process:334] Processing call for obj-id: 11, entry:
"api/v1/report/fct/tags"
[ec_ez_worker_process:441] Call completed successfully.
    obj-id: 11, desc: "REST API to get updates of tag endpoints.", entry:
"api/v1/report/fct/tags".
(......)
```

3. Confirm that the device information from the EMS is written to the FortiClient NAC daemon cache:

```
# diagnose endpoint record list
    ...
          Avatar source: OS
            Phone number:
            Number of Routes: (1)
                    Gateway Route #0:
                            - IP:10.1.91.6, MAC: 4f:8d:c2:73:dd:fe, Indirect: no
                            - Interface:port2, VFID:1, SN: FG5H1E5999999999
online records: 37174; offline records: 0; quarantined records: 0; out-of-sync records:
0
```

4. Use the tags that are pulled from the EMS in a firewall address:

```
config firewall address
    edit "FCTEMS8888888888_ZT_AD_MGMT"
        set type dynamic
        set sub-type ems-tag
        set obj-tag "ZT_AD_MGMT"
        set tag-type "zero_trust"
```

```
    next
end
```

5. Check the tags' resolved IP and MAC addresses:

```
# diagnose firewall fqdn getinfo-ip FCTEMS8888888888_ZT_AD_MGMT
getinfo FCTEMS8888888888_ZT_AD_MGMT id:114 generation:106 count:187 data_len:6160 flag 0
```

```
# diagnose firewall fqdn getinfo-mac MAC_FCTEMS8888888888_ZT_AD_MGMT
getinfo MAC_FCTEMS8888888888_ZT_AD_MGMT id:163 generation:105 count:371 data_len:2226
flag 0
```

```
# diagnose firewall dynamic address  FCTEMS8888888888_ZT_AD_MGMT
CMDB name: FCTEMS8888888888_ZT_AD_MGMT
TAG name: ZT_AD_MGMT
FCTEMS8888888888_ZT_AD_MGMT: ID(114)
        ADDR(10.1.10.4)
(......)
        ADDR(10.1.99.195)
Total IP dynamic range blocks: 190.
Total IP dynamic addresses: 281.
```

```
# diagnose firewall dynamic address MAC_FCTEMS8888888888_ZT_AD_MGMT
CMDB name: MAC_FCTEMS8888888888_ZT_AD_MGMT
TAG name: ZT_AD_MGMT
MAC_FCTEMS8888888888_ZT_AD_MGMT: ID(163)
        MAC(52:f1:9d:06:1c:db)
        MAC(4b:77:2b:db:82:15)
        MAC(df:6e:9e:d9:04:1e)
Total MAC dynamic addresses: 393.
```

ZTNA troubleshooting and debugging

The following debug commands can be used to troubleshoot ZTNA issues:

Command	Description
`# diagnose endpoint fctems test-connectivity <EMS>`	Verify FortiGate to FortiClient EMS connectivity.
`# execute fctems verify <EMS>`	Verify the FortiClient EMS's certificate.
`# diagnose test application fcnacd 2`	Dump the EMS connectivity information.
`# diagnose debug app fcnacd -1` `# diagnose debug enable`	Run real-time FortiClient NAC daemon debugs.
`# diagnose endpoint record list <ip>`	Show the endpoint record list. Optionally, filter by the endpoint IP address.
`# diagnose endpoint lls-comm send ztna find-uid <uid>`	Query endpoints by client UID.
`# diagnose endpoint lls-comm send ztna find-ip-vdom <ip> <vdom>`	Query endpoints by the client IP-VDOM pair.
`# diagnose wad dev query-by uid <uid>`	Query from WAD diagnose command by UID.

Command	Description
`# diagnose wad dev query-by ipv4 <ip>`	Query from WAD diagnose command by IP address.
`# diagnose firewall dynamic list`	List EMS ZTNA tags and all dynamic IP and MAC addresses.
`# diagnose test application fcnacd 7` `# diagnose test application fcnacd 8`	Check the FortiClient NAC daemon ZTNA and route cache.
`# diagnose wad worker policy list`	Display statistics associated with access proxy rules.

`# diagnose wad debug enable category all`
`# diagnose wad debug enable level verbose`
`# diagnose debug enable`
	Run real-time WAD debugs.

`# diagnose debug reset`	Reset debugs when completed

The WAD daemon handles proxy related processing. The FortiClient NAC daemon (fcnacd) handles FortiGate to EMS connectivity.

Troubleshooting usage and output

1. Verify the FortiGate to EMS connectivity and EMS certificate:

```
# diagnose endpoint fctems test-connectivity WIN10-EMS
Connection test was successful:
```

```
# execute fctems verify WIN10-EMS
Server certificate already verified.
```

```
# diagnose test application fcnacd 2
EMS context status:
FortiClient EMS number 1:
        name: WIN10-EMS confirmed: yes
        fetched-serial-number: FCTEMS0000109188
Websocket status: connected
```

2. If fcnacd does not report the proper status, run real-time fcnacd debugs:

```
# diagnose debug app fcnacd -1
# diagnose debug enable
```

3. Verify the following information about an endpoint:
   - Network information
   - Registration information
   - Client certificate information
   - Device information
   - Vulnerability status
   - Relative position with the FortiGate

```
# diagnose endpoint record list 10.6.30.214
Record #1:
                IP Address = 10.6.30.214
                MAC Address = 00:0c:29:ba:1e:61
                MAC list = 00:0c:29:ba:1e:61;00:0c:29:ba:1e:6b;
                VDOM = root (0)
                EMS serial number: FCTEMS8821001322
                Client cert SN: 17FF6595600A1AF53B87627AB4EBEDD032593E64
                Quarantined: no
                Online status: online
                Registration status: registered
                On-net status: on-net
                Gateway Interface: port2
                FortiClient version: 7.0.0
                AVDB version: 84.778
                FortiClient app signature version: 18.43
                FortiClient vulnerability scan engine version: 2.30
                FortiClient UID: 5FCFA3ECDE4D478C911D9232EC9299FD
                Host Name: ADPC
    …
                Number of Routes: (1)
                        Gateway Route #0:
                                - IP:10.1.100.214, MAC: 00:0c:29:ba:1e:6b, Indirect: no
                                - Interface:port2, VFID:0, SN: FG5H1E5819902474
online records: 1; offline records: 0; quarantined records: 0
```

4. Query the endpoint information, include ZTNA tags, by UID or IP address:

```
# diagnose endpoint lls-comm send ztna find-uid 5FCFA3ECDE4D478C911D9232EC9299FD
UID: 5FCFA3ECDE4D478C911D9232EC9299FD
        status code:ok
        Domain: qa.wangd.com
        User: user1
        Cert SN:17FF6595600A1AF53B87627AB4EBEDD032593E64
        EMS SN: FCTEMS8821001322
        Routes(1):
         - route[0]: IP=10.1.100.214, VDom=root
        Tags(3):
         - tag[0]: name=ZT_OS_WIN
         - tag[1]: name=all_registered_clients
         - tag[2]: name=Medium
```

```
# diagnose endpoint lls-comm send ztna find-ip-vdom 10.1.100.214 root
UID: 5FCFA3ECDE4D478C911D9232EC9299FD
        status code:ok
        Domain: qa.wangd.com
        User: user1
        Cert SN:17FF6595600A1AF53B87627AB4EBEDD032593E64
        EMS SN: FCTEMS8821001322
        Routes(1):
         - route[0]: IP=10.1.100.214, VDom=root
        Tags(3):
         - tag[0]: name=ZT_OS_WIN
         - tag[1]: name=all_registered_clients
         - tag[2]: name=Medium
```

5. Query endpoint information from WAD by UID or IP address:

```
# diagnose wad dev query-by uid 5FCFA3ECDE4D478C911D9232EC9299FD
Attr of type=0, length=32, value(ascii)=5FCFA3ECDE4D478C911D9232EC9299FD
Attr of type=4, length=30, value(ascii)=MAC_FCTEMS8821001322_ZT_OS_WIN
Attr of type=4, length=26, value(ascii)=FCTEMS8821001322_ZT_OS_WIN
Attr of type=4, length=43, value(ascii)=MAC_FCTEMS8821001322_all_registered_clients
Attr of type=4, length=39, value(ascii)=FCTEMS8821001322_all_registered_clients
Attr of type=4, length=27, value(ascii)=MAC_FCTEMS8821001322_Medium
Attr of type=4, length=23, value(ascii)=FCTEMS8821001322_Medium
Attr of type=5, length=18, value(ascii)=FOSQA@qa.wangd.com
Attr of type=6, length=40, value(ascii)=17FF6595600A1AF53B87627AB4EBEDD032593E64
```

```
# diagnose wad dev query-by ipv4 10.1.100.214
Attr of type=0, length=32, value(ascii)=5FCFA3ECDE4D478C911D9232EC9299FD
Attr of type=4, length=30, value(ascii)=MAC_FCTEMS8821001322_ZT_OS_WIN
Attr of type=4, length=26, value(ascii)=FCTEMS8821001322_ZT_OS_WIN
Attr of type=4, length=43, value(ascii)=MAC_FCTEMS8821001322_all_registered_clients
Attr of type=4, length=39, value(ascii)=FCTEMS8821001322_all_registered_clients
Attr of type=4, length=27, value(ascii)=MAC_FCTEMS8821001322_Medium
Attr of type=4, length=23, value(ascii)=FCTEMS8821001322_Medium
Attr of type=5, length=18, value(ascii)=FOSQA@qa.wangd.com
Attr of type=6, length=40, value(ascii)=17FF6595600A1AF53B87627AB4EBEDD032593E64
```

6. List all the dynamic ZTNA IP and MAC addresses learned from EMS:

```
# diagnose firewall dynamic list
List all dynamic addresses:
FCTEMS0000109188_all_registered_clients: ID(51)
        ADDR(172.17.194.209)
        ADDR(192.168.40.8)
…
FCTEMS0000109188_Low: ID(78)
        ADDR(172.17.194.209)
        ADDR(192.168.40.8)
…
FCTEMS0000109188_Malicious-File-Detected: ID(190)
        ADDR(172.17.194.209)
        ADDR(192.168.40.8)
…
```

7. Check the FortiClient NAC daemon ZTNA and route cache:

```
# diagnose test application fcnacd 7
ZTNA Cache:
-uid 5FCFA3ECDE4D478C911D9232EC9299FD: { "tags": [ "ZT_OS_WIN", "all_registered_
clients", "Medium" ], "domain": "qa.wangd.com", "user_name": "user1", "client_cert_sn":
"17FF6595600A1AF53B87627AB4EBEDD032593E64", "owner": "FOSQA@qa.wangd.com", "gateway_
route_list": [ { "gateway_info": { "fgt_sn": "FG5H1E5819902474", "interface": "port2",
"vdom": "root" }, "route_info": [ { "ip": "10.1.100.214", "mac": "00-0c-29-ba-1e-6b",
"route_type": "direct" } ] } ], "ems_sn": "FCTEMS8821001322" }
```

```
# diagnose test application fcnacd 8
IP-VfID Cache:
IP: 10.1.100.206, vfid: 0, uid: 3DED29B54386416E9888F2DCBD2B9D21
IP: 10.1.100.214, vfid: 0, uid: 5FCFA3ECDE4D478C911D9232EC9299FD
```

8. Troubleshoot WAD with real-time debugs to understand how the proxy handled a client request:

```
# diagnose wad debug enable category all
# diagnose wad debug enable level verbose
```

```
# diagnose debug enable

[0x7fbd7a46bb60] Received request from client: 10.10.10.20:56312
GET / HTTP/1.1 Host: 192.168.2.86:8443 Connection: keep-alive Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/89.0.4389.90 Safari/537.36 Edg/89.0.774.57
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,ap
plication/signed-exchange;v=b3;q=0.9 Sec-Fetch-Site: none Sec-Fetch-Mode: navigate Sec-
Fetch-User: ?1 Sec-Fetch-Dest: document Accept-Encoding: gzip, deflate, br Accept-
Language: en-US,en;q=0.9 [p:29957][s:458767][r:1] wad_http_marker_uri(1269): path=/
len=1
[p:29957][s:458767][r:1] wad_http_parse_host(1641): host_len=17
[p:29957][s:458767][r:1] wad_http_parse_host(1677): len=12
[p:29957][s:458767][r:1] wad_http_parse_host(1686): len=4
[p:29957][s:458767][r:1] wad_http_str_canonicalize(2180): path=/ len=1 changes=0
[p:29957][s:458767][r:1] wad_http_str_canonicalize(2189): path=/ len=1 changes=0
[p:29957][s:458767][r:1] wad_http_normalize_uri(2232): host_len=12 path_len=1 query_
len=0
[p:29957][s:458767][r:1] wad_vs_proxy_match_gwy(2244): 6:WIN2K16-P1: matching gwy with
vhost(_def_virtual_host_)
[p:29957][s:458767][r:1] wad_vs_proxy_match_vhost(2293): 6:WIN2K16-P1: matching vhost
by: 192.168.2.86
[p:29957][s:458767][r:1] wad_vs_matcher_map_find(477): Empty matcher!
[p:29957][s:458767][r:1] wad_vs_proxy_match_vhost(2296): 6:WIN2K16-P1: no host matched.
[p:29957][s:458767][r:1] wad_vs_proxy_match_gwy(2263): 6:WIN2K16-P1: matching gwy by (/)
with vhost(_def_virtual_host_).
[p:29957][s:458767][r:1] wad_pattern_matcher_search(1210): pattern-match succ:/
[p:29957][s:458767][r:1] wad_vs_proxy_match_gwy(2271): 6:WIN2K16-P1: Matched gwy(1) type
(https).
[p:29957][s:458767][r:1] wad_http_vs_check_dst_ovrd(776): 6:WIN2K16-P1:1: Found server:
192.168.20.6:443
[p:29957][s:458767][r:1] wad_http_req_exec_act(9296): dst_addr_type=3 wc_nontp=0 sec_
web=1 web_cache=0 req_bypass=0
[p:29957][s:458767][r:1] wad_http_req_check_policy(8117): starting policy matching(vs_
pol= 1):10.10.10.20:56312->192.168.20.6:443
[p:29957][s:458767][r:1] wad_fw_addr_match_ap(1524): matching ap:WIN2K16(7) with vip
addr:WIN2K16-P1(10)
[p:29957][s:458767][r:1] wad_fw_addr_match_ap(1524): matching ap:WIN2K16-P1(10) with vip
addr:WIN2K16-P1(10)
[p:29957][s:458767][r:1] wad_http_req_policy_set(6811): match pid=29957 policy-id=2 vd=0
in_if=3, out_if=7 10.10.10.20:56312 -> 192.168.20.6:443
[p:29957][s:458767][r:1] wad_cifs_profile_init(93): CIFS Profile 0x7fbd7a5bf200 [] of
type 0 created
[p:29957][s:458767][r:1] wad_http_req_proc_policy(6622): web_cache(http/https=0/0, fwd_
srv=<nil>.
[p:29957][s:458767][r:1] wad_auth_inc_user_count(1668): increased user count,
quota:128000, n_shared_user:2, vd_used: 2, vd_max: 0, vd_gurantee: 0
[p:29957][s:458767][r:1] __wad_fmem_open(563): fmem=0xaaee3e8, fmem_name='cmem 336
bucket', elm_sz=336, block_sz=73728, overhead=20, type=advanced
[p:29957][s:458767][r:1] __wad_hauth_user_node_hold(2107): wad_hauth_user_node_alloc
(1568): holding node 0x7fbd76d48060
mapping user_node:0x7fbd76d48060, user_ip:0x7fbd7a57b408(0), user:0x7fbd7a5cf420(0)
[p:29957][s:458767][r:1] __wad_hauth_user_node_hold(2107): wad_user_node_stats_hold
(483): holding node 0x7fbd76d48060
```

```
[p:29957][s:458767][r:1] __wad_hauth_user_node_hold(2107): wad_http_session_upd_user_
node (4813): holding node 0x7fbd76d48060
[p:29957][s:458767][r:1] wad_http_req_proc_policy(6698): policy result:vf_id=0:0 sec_
profile=0x7fbd7a5bef00 set_cookie=0
[p:29957][s:458767][r:1] wad_http_urlfilter_check(381): uri_norm=1 inval_host=0 inval_
url=0 scan-hdr/body=1/0 url local=0 block=0 user-cat=0 allow=0 ftgd=0 keyword=0 wisp=0
[p:29957][s:458767][r:1] wad_http_req_proc_waf(1309): req=0x7fbd7a46bb60 ssl.deep_scan=1
proto=10 exempt=0 waf=(nil) body_len=0 ua=Chrome/89.0.4389.90 skip_scan=0
[p:29957][s:458767][r:1] wad_http_req_proc_antiphish(5376): Processing antiphish request
[p:29957][s:458767][r:1] wad_http_req_proc_antiphish(5379): No profile
[p:29957][s:458767][r:1] wad_http_connect_server(4696): http session 0x7fbd7a532ac8
req=0x7fbd7a46bb60
[p:29957][s:458767][r:1] wad_http_srv_still_good(4575): srv((nil)) nontp(0) dst_type(3)
req: dst:192.168.20.6:443, proto:10)
hcs: dst:N/A:0, proto:1)
```

Always reset the debugs after using them:

```
# diagnose debug reset
```

Security Profiles

This section contains information about configuring FortiGate security features, including:

If you are unable to view a security profile feature, go to *System* > *Feature Visibility* to enable it.

Inspection modes

FortiOS supports flow-based and proxy-based inspection in firewall policies. You can select the inspection mode when configuring a policy.

Flow-based inspection takes a snapshot of content packets and uses pattern matching to identify security threats in the content.

Proxy-based inspection reconstructs content that passes through the FortiGate and inspects the content for security threats.

Certain security profiles allows users to display flow-based or proxy-based feature sets.

This following topics provide information about inspection modes for various security profile features:

- Flow mode inspection (default mode) on page 1085
- Proxy mode inspection on page 1085
- Inspection mode feature comparison on page 1086

Flow mode inspection (default mode)

When a firewall policy's inspection mode is set to flow, traffic flowing through the policy will not be buffered by the FortiGate. Unlike proxy mode, the content payload passing through the policy will be inspected on a packet by packet basis with the very last packet held by the FortiGate until the scan returns a verdict. If a violation is detected in the traffic, a reset packet is issued to the receiver, which terminates the connection and prevents the payload from being sent successfully.

Flow-based inspection identifies and blocks security threats in real time as they are identified. All applicable flow-based security modules are applied simultaneously in one single pass, using Direct Filter Approach (DFA) pattern matching to identify possible attacks or threats. Pattern matching is offloaded and accelerated by CP8 or CP9 processors.

Flow-based inspection typically requires lower processing resources than proxy-based inspection and does not change packets, unless a threat is found and packets are blocked. Flow-based inspection is selected by default on new firewall policies. It is the recommended inspection mode, unless proxy-specific features are required. For more information, see Inspection mode feature comparison on page 1086.

Proxy mode inspection

When a firewall policy’s inspection mode is set to proxy, traffic flowing through the policy will be buffered by the FortiGate for inspection. This means that the packets for a file, email message, or web page will be held by the FortiGate until the entire payload is inspected for violations (virus, spam, or malicious web links). After FortiOS finishes the inspection, the payload is either released to the destination (if the traffic is clean) or dropped and replaced with a replacement message (if the traffic contains violations).

To optimize inspection, the policy can be configured to block or ignore files or messages that exceed a certain size. To prevent the receiving end user from timing out, you can apply client comforting. This allows small portions of the payload to be sent while it is undergoing inspection.

In proxy-based antivirus scanning, certain techniques are used to streamline scanning with either in-process or stream-based scanning. For more information, see Proxy mode stream-based scanning on page 1.

Proxy mode provides some security profile capabilities that are not available to flow-based scanning:

- Video Filter
- Web Application Firewall (WAF)
- Content Disarm and Reconstruction (CDR)
- Web quota
- Sandbox Inline Scanning

For a complete list, see Inspection mode feature comparison on page 1086.

Some features are exclusively proxy-based:

- SSL Offloading
- Explicit Web Proxy
- ZTNA

Verify the capabilities that you need when deciding to use proxy-based or flow-based policy. Applying the same type of scan mode in all your policies also helps optimize your performance.

Inspection mode feature comparison

The following table shows which UTM profile can be configured on a flow mode or proxy mode inspection policy.

Some UTM profiles are hidden in the GUI and can only be configured using the CLI. To configure profiles in a firewall policy in CLI, enable the `utm-status` setting.

Some profiles might have feature differences between flow-based and proxy-based Inspection. From the GUI and CLI, you can set the *Feature set* option to be *Flow-based* or *Proxy-based* to display only the settings for that mode.

	Flow Mode Inspection Policy		Proxy Mode Inspection Policy		Feature set option
UTM Profile	GUI	CLI	GUI	CLI	
AntiVirus	Yes	Yes	Yes	Yes	GUI/CLI
Web Filter	Yes	Yes	Yes	Yes	GUI/CLI
DNS Filter	Yes	Yes	Yes	Yes	N/A
Application Control	Yes	Yes	Yes	Yes	N/A
Intrusion Prevention System	Yes	Yes	Yes	Yes	N/A
File Filter	Yes	Yes	Yes	Yes	GUI/CLI
Email Filter	Yes	Yes	Yes	Yes	GUI/CLI
Data Leak Prevention	No	Yes	No	Yes	CLI
VoIP	Yes	Yes	Yes	Yes	N/A
ICAP	No	No	Yes	Yes	N/A
Web Application Firewall	No	No	Yes	Yes	N/A
SSL/SSH Inspection	Yes	Yes	Yes	Yes	N/A

The following sections outline differences between flow-based and proxy-based inspection for a security profile.

Feature comparison between Antivirus inspection modes

The following table indicates which Antivirus features are supported by their designated scan modes.

Part1	Replacement Message	Content Disarm	Mobile Malware	Virus Outbreak	Sandbox Post-Transfer Scanning	Sandbox Inline Scanning	NAC Quarantine
Proxy	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Flow	Yes*	No	Yes	Yes	Yes	No	Yes

*IPS Engine caches the URL and a replacement message is presented after the second attempt.

Part 2	Archive Blocking	Emulator	Client Comforting	Infection Quarantine	Heuristics	Treat EXE as Virus
Proxy	Yes	Yes	Yes	Yes (1)	Yes	Yes (2)
Flow	Yes	Yes	No	Yes	Yes	Yes (2)

1. Only available on FortiGate models with HDD or when FortiAnalyzer or FortiGate Cloud is connected and enabled.
2. Only applies to inspection on IMAP, POP3, SMTP, and MAPI protocols.

Part 3	External Blocklist	EMS Threat Feed	AI/ML Based Detection	FortiNDR Inline Detection
Proxy	Yes	Yes	Yes	Yes
Flow	Yes	Yes	Yes	No

Feature comparison between Web Filter inspection modes

The following table indicates which Web Filter features are supported by their designated inspection modes.

	FortiGuard Category-Based Filter	Category Usage Quota	Override Blocked Categories	Search Engines	Static URL Filter	Rating Option	Proxy Option	Web Profile Override
Proxy	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Flow	Yes (1)	No	Yes (2)	No	Yes	Yes	Limited (3)	No

1. Local Category and Remote Category filters do not support the warning and authenticate actions.
2. Local Category and Remote Category filters cannot be overridden.
3. Only HTTP POST Action is supported.

Feature comparison between Email Filter inspection modes

The following tables indicate which Email Filters are supported by the specified inspection modes for local filtering and FortiGuard-assisted filtering.

Local Filtering	Banned Word Check	Block/Allow List	HELO/ EHLO DNS Check	Return Address DNS Check	DNSBL/ ORBL Check	MIME Header Check
Proxy	Yes	Yes	Yes	Yes	Yes	Yes
Flow	Yes	Yes	No	No	No	Yes

FortiGuard-Assisted Filtering	Phishing URL Check	Anti-Spam Block List Check	Submit Spam to FortiGuard	Spam Email Checksum Check	Spam URL Check
Proxy	Yes	Yes	Yes	Yes	Yes
Flow	No	No	No	No	No

Feature comparison between DLP inspection modes

The following table indicates which DLP filters are supported by their designated inspection modes.

	Credit Card Filter	SSN Filter	Regex Filter	File-Type Filter	File-Pattern Filter	Fingerprint Filter	Watermark Filter	Encrypted Filter	File-Size Filter
Proxy	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Flow	Yes	Yes	Yes	Yes	Yes	No	No	Yes	Yes*

*File-size filtering only works if file size is present in the protocol exchange.

Antivirus

FortiOS offers the unique ability to implement both flow-based and proxy-based antivirus concurrently, depending on the traffic type, users, and locations. Flow-based antivirus offers higher throughput performance.

FortiOS includes two preloaded antivirus profiles:

- *default*
- *wifi-default*

You can customize these profiles, or you can create your own to inspect certain protocols, remove viruses, analyze suspicious files with FortiSandbox, and apply botnet protection to network traffic. Once configured, you can add the antivirus profile to a firewall policy.

This functionality requires a subscription to FortiGuard Antivirus.

The following topics provide information about antivirus profiles:

- Proxy mode stream-based scanning on page 1090
- Databases on page 1093
- Content disarm and reconstruction on page 1094
- FortiGuard outbreak prevention on page 1096
- External malware block list on page 1098
- Malware threat feed from EMS on page 1101
- Checking flow antivirus statistics on page 1104

- CIFS support on page 1106
- Using FortiSandbox post-transfer scanning with antivirus on page 1111
- Using FortiSandbox inline scanning with antivirus on page 1113
- Using FortiNDR inline scanning with antivirus on page 1119

Protocol comparison between antivirus inspection modes

The following table indicates which protocols can be inspected by the designated antivirus scan modes.

	HTTP	FTP	IMAP	POP3	SMTP	NNTP	MAPI	CIFS	SSH
Proxy	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes*	Yes
Flow	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes	No

* Proxy mode antivirus inspection on CIFS protocol has the following limitations:

- Cannot detect infections within some archive files.
- Cannot detect oversized files.

Other antivirus differences between inspection modes

Starting from 6.4.0, the scan mode option is no longer available for flow-based AV.

This means that AV no longer exclusively uses the default or legacy scan modes when handling traffic on flow-based firewall policies. Instead, AV in flow-based policies uses a hybrid of the two scan modes. Flow AV may use a pre-filtering database for malware detection in some circumstances as opposed to the full AV signature database in others. The scan method is determined by the IPS engine algorithm that is based on the type of file being scanned. When handling oversized files in flow-based AV, the action can either be pass (default) or block. When theaction is pass, IPS appends to-be-scan data into the AV scan buffer. If the appended file size exceeds the oversize-limit that is defined in the protocol option profile, then the AV session is cleared and the file is bypassed from AV scanning.

In contrast, proxy mode maintains the scan mode option, which can be toggled between default or legacy mode. In default mode, the WAD daemon receives the file and then decides if it can do an in-process scan of the file in simple AV configuration scenarios. If the file is in an oversized archive that is supported by the stream-based decompressor, then it is sent to stream-based scan for best effort inspection. Stream-based scan decompresses and scans the entire archive without archiving the file. If the file is not supported by stream-based scan, then it is buffered and then sent to the scanunit daemon for inspection on content that is under the oversize limit.

In legacy mode, stream-based scanning is disabled, so oversized archive files and files that cannot be handled by WAD in-process scan are buffered and sent to the scanunit daemon for processing.

AI-based malware detection

The AV Engine AI malware detection model integrates into regular AV scanning to help detect potentially malicious Windows Portable Executables (PEs) in order to mitigate zero-day attacks. Previously, this type of detection was handled by heuristics that analyzed file behavior. With AV Engine AI, the module is trained by FortiGuard AV against many malware samples to identify file features that make up the malware. The AV Engine AI package can be downloaded by FortiOS via FortiGuard on devices with an active AV subscription. The `machine-learning-`

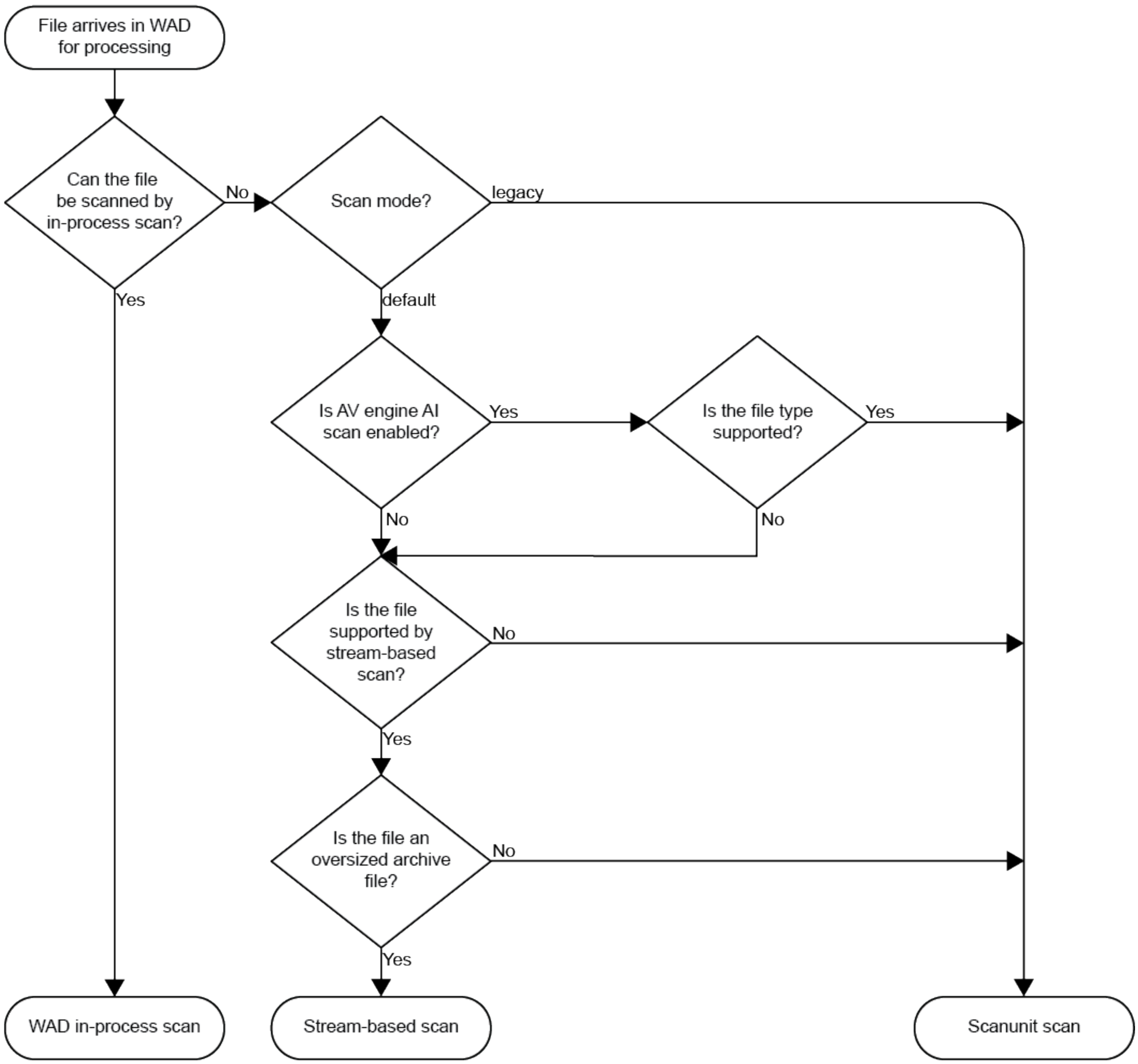
`detection` setting is enabled by default at a per-VDOM level. Files detected by the AV Engine AI are identified with the W32/AI.Pallas.Suspicious virus signature.

To configure machine learning-based malware detection:

```
config antivirus settings
    set machine-learning-detection {enable| monitor | disable}
end
```

Proxy mode stream-based scanning

In proxy mode, AV scanning is processed as follows:



Can the file be scanned by in-process scan?

- This is determined by the WAD daemon.
- In-process scan can be used for simple AV configurations to quickly scan a file without handing it off to another process.
- The following, more complex feature sets cannot be processed by in-process scan:
  - AV engine AI scan
  - DLP
  - Quarantine
  - FortiGuard outbreak prevention and external block list
  - Content disarm

Scan mode?

- To configure the scan mode:

```
config antivirus profile
    edit <name>
        set feature-set proxy
        set scan-mode {default | legacy}
    next
end
```

`default`	Enable stream-based scanning (default).
`legacy`	Disable stream-based scanning.

Is AV engine AI scan enabled?

- When enabled, supported files (such as EXE, PDF, and MS Office) are forwarded to the scanunit scan.
- AV engine AI scan is enabled by default. To disable it:

```
config antivirus settings
    set machine-learning-detection disable
end
```

Is the file supported by stream-based scan?

- Stream-based scan supports the following archive file types: ZIP, GZIP, BZIP2, TAR, and ISO (ISO 9660).
- In FortiOS 7.0, stream-based scan is supported in HTTP(S), FTP(S), and SCP/SFTP.
- In FortiOS 6.4 and 6.2, stream-based scan is only supported in HTTP(S).
- Stream-based scan does not support HTTP POST.
- Stream-based scan is not supported when the following features are enabled:
  - DLP
  - Quarantine
  - FortiGuard outbreak prevention and external block list
  - Content Disarm
- If a file is not supported, it is buffered and sent to scanunit for scanning.

Is the file an oversized archive file?

- An oversized archive file is a compressed file that is oversized according to the following setting:

```
config firewall profile-protocol-options
    edit <profile>
        config <protocol>
            set oversize-limit <size>
        end
    next
end
```

- If the file is not oversized, it is buffered and sent to scanunit for scanning.

Notes

Stream-based scans:

- Are performed with no oversize limits on a best effort basis.
- Can inspect the contents of large archive files without buffering the entire file.
- Decompress and scan the entire archive.

Legacy scan mode:

- Used to disable stream-based scanning for troubleshooting purposes.
- Limited by the oversize and uncompressed-oversize limits:

```
config firewall profile-protocol-options
    edit <profile>
        config <protocol>
            set oversize-limit <size>
            set uncompressed-oversize-limit <size>
        end
    next
end
```

TCP windows

Some file transfer applications can negotiate large TCP windows. For example, WinSCP can negotiate an initial TCP window size of about 2 GB.

The TCP window options can be used to prevent overly large initial TCP window sizes, helping avoid channel flow control issues. It allows stream-based scan's flow control to limit peers from sending data that exceeds a policy's configured oversize limit.

To configure TCP window size options:

```
config firewall profile-protocol-options
    edit <string>
        config {ftp | ssh}
            set stream-based-uncompressed-limit <integer>
            set tcp-window-type {system | static | dynamic}
            set tcp-window-size <integer>
            set tcp-window-minimum <integer>
            set tcp-window-maximum <integer>
```

```
        end
    next
end
```

`{ftp | ssh}`	• `ftp`: Configure FTP protocol options. • `ssh`: Configure SFTP and SCP protocol options.
`stream-based-uncompressed-limit <integer>`	The maximum stream-based uncompressed data size that will be scanned, in MB (default = 0 (unlimited)). Stream-based uncompression used only under certain conditions.).
`tcp-window-type {system | static | dynamic}`	The TCP window type to use for this protocol. • `system`: Use the system default TCP window size for this protocol (default). • `static`: Manually specify the TCP window size. • `dynamic`: Vary the TCP window size based on available memory within the limits configured in `tcp-window-minimum` and `tcp-window-maximum`.
`tcp-window-size <integer>`	The TCP static window size (65536 - 33554432, default = 262144). This option is only available when `tcp-window-type` is `static`.
`tcp-window-minimum <integer>`	The minimum TCP dynamic window size (65536 - 1048576, default = 131072). This option is only available when `tcp-window-type` is `dynamic`.
`tcp-window-maximum <integer>`	The maximum TCP dynamic window size (1048576 - 33554432, default = 8388608). This option is only available when `tcp-window-type` is `dynamic`.

Databases

The antivirus scanning engine uses a virus signatures database to record the unique attributes of each infection. The antivirus scan searches for these signatures and when one is discovered, the FortiGate determines if the file is infected and takes action.

All FortiGates have the normal antivirus signature database. Some models have additional databases that you can use. The database you use depends on your network and security needs, and on your FortiGate model.

The extended virus definitions database is the default setting and provides comprehensive antivirus protection. Low-end FortiGate models cannot support the extreme database. The FortiGate 300D is the lowest model that supports the extreme database. All VMs support the extreme database. The `use-extreme-db` setting is only available on models that support the extreme database.

Extended	This is the default setting. This database includes currently spreading viruses, as determined by the FortiGuard Global Security Research Team, plus recent viruses that are no longer active. These viruses may have been spreading within the last year but have since nearly or completely disappeared.
Extreme	This includes the extended database, plus a large collection of zoo viruses. These are viruses that have not spread in a long time and are largely dormant. Some zoo viruses might rely on operating systems and hardware that are no longer widely used.

To change the antivirus database:

```
config antivirus settings
    set use-extreme-db {enable | disable}
end
```

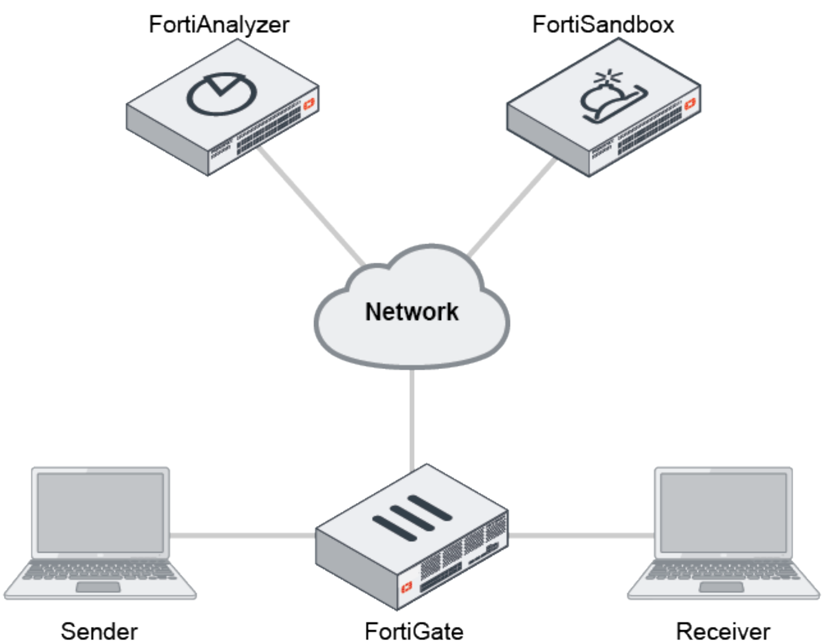
Content disarm and reconstruction

Content disarm and reconstruction (CDR) allows the FortiGate to sanitize Microsoft Office documents and PDF files (including those that are in ZIP archives) by removing active content, such as hyperlinks, embedded media, JavaScript, macros, and so on from the files (disarm) without affecting the integrity of its textual content (reconstruction). It allows network administrators to protect their users from malicious document files.

Files processed by CDR can be stored locally for quarantine on FortiAnalyzer, FortiSandbox, or FortiGate models with a hard disk. The original copies can also be obtained in the event of a false positive.

CDR is supported on HTTP, SMTP, POP3, and IMAP. Note that SMTP splice and client-comfort mode are not supported. CDR does not support flow-based inspection modes.

Sample topology

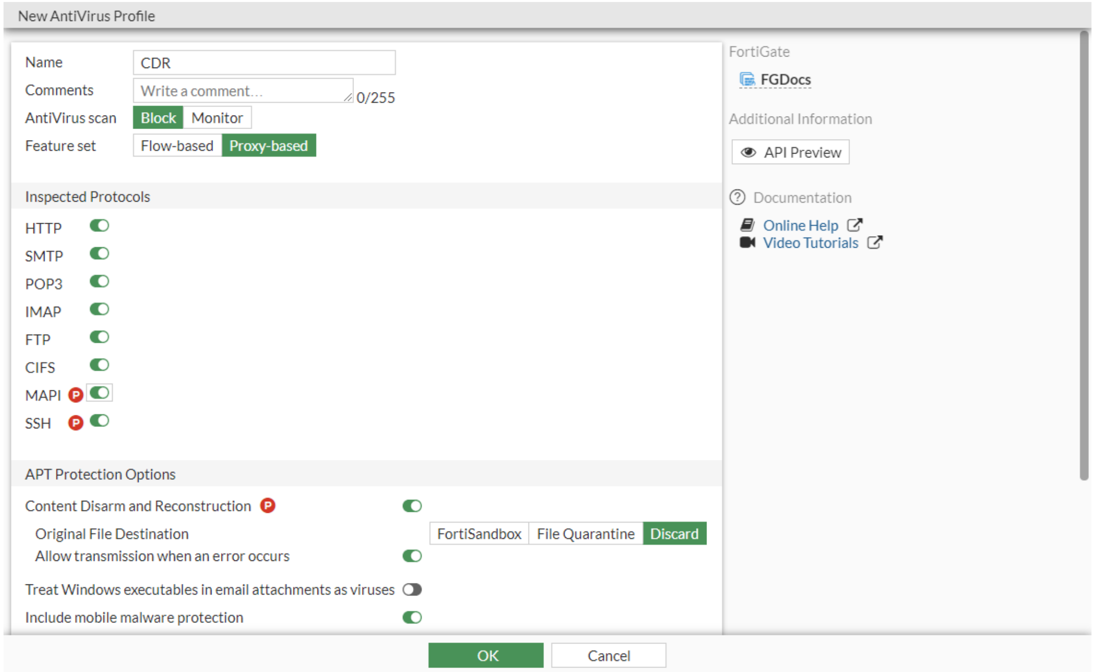


In this example, the a Microsoft Office document with an embedded hyperlink (that redirects to an external website) is sent to the receiver. When the user receives the file, the hyperlink in the document is deactivated.

To configure CDR:

1. Go to *Security Profiles > AntiVirus*.
2. Edit an antivirus profile, or create a new one.

3. Under *APT Protection Options*, enable *Content Disarm and Reconstruction*.



4. Select a quarantine location from the available options:

FortiSandbox	Saves the original document file to a connected FortiSandbox.
File Quarantine	Saves the original document file to disk (if possible) or a connected FortiAnalyzer based on the FortiGate log settings (`config log fortianalyzer setting`).
Discard	The default setting, which discards the original document file.

5. Click *OK*.

To edit the CDR detection parameters:

By default, stripping of all active Microsoft Office and PDF content types are enabled. In this example, stripping macros in Microsoft Office documents will be disabled.

```
config antivirus profile
    edit av
        config content-disarm
            set office-macro disable
            set detect-only {enable | disable}
            set cover-page {enable | disable}
        end
    next
end
```

Where:

detect-only	Only detect disarmable files, do not alter content. Disabled by default.
cover-page	Attach a cover page to the file's content when the file has been processed by CDR. Enabled by default.

FortiGuard outbreak prevention

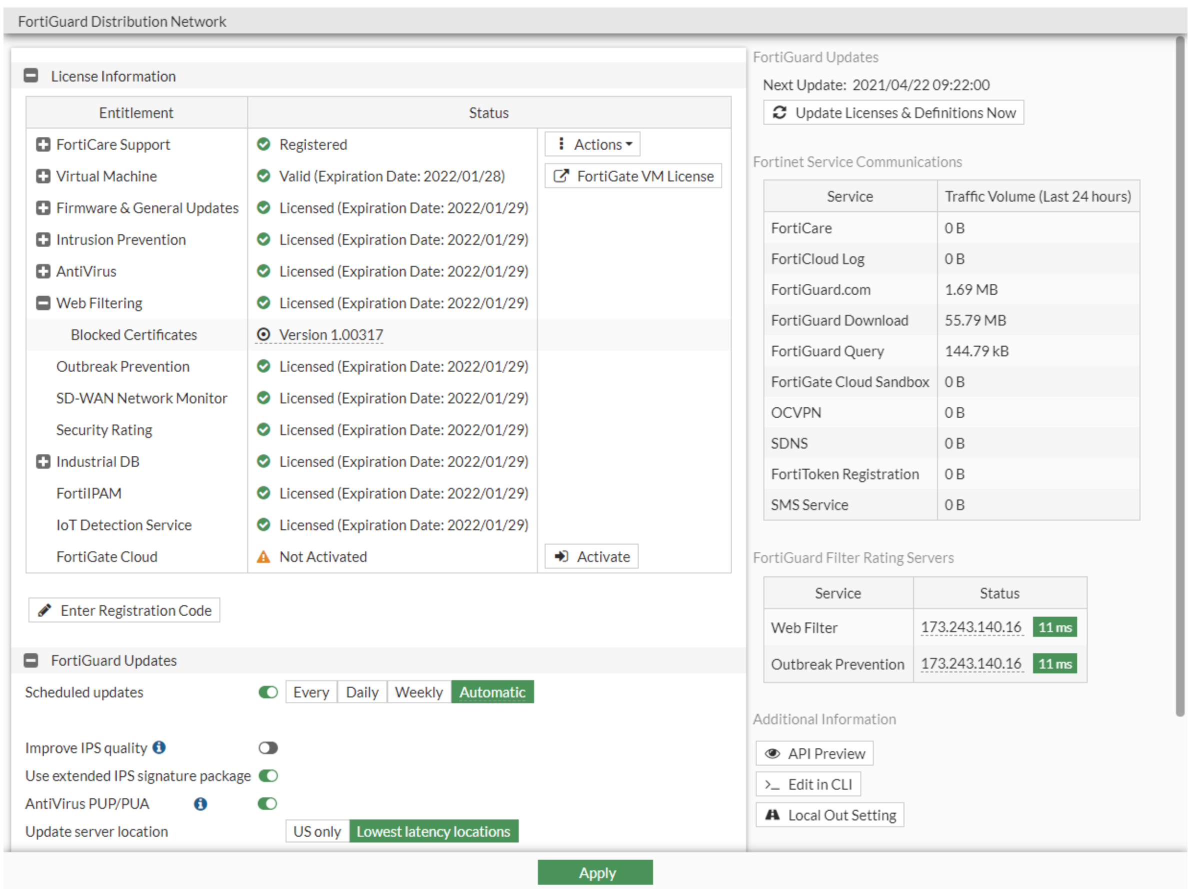
FortiGuard Virus Outbreak Protection Service (VOS) allows the FortiGate antivirus database to be subsidized with third-party malware hash signatures curated by FortiGuard. The hash signatures are obtained from FortiGuard's Global Threat Intelligence database. The antivirus database queries FortiGuard with the hash of a scanned file. If FortiGuard returns a match, the scanned file is deemed to be malicious. Enabling the AV engine scan is not required to use this feature.

FortiGuard VOS can be used in both proxy-based and flow-based policy inspections across all supported protocols.

The FortiGate must be registered with a valid FortiGuard outbreak prevention license.

To verify FortiGuard antivirus license information:

1. Go to *System > FortiGuard* and locate the *Outbreak Prevention* section in the table.



2. See the instructions in the video, How to Purchase or Renew FortiGuard Services, if required.

To enable FortiGuard outbreak prevention:

1. Go to *Security Profiles > AntiVirus*.
2. Edit an antivirus profile, or create a new one.

3. Under *Virus Outbreak Protection*, enable *Use FortiGuard outbreak prevention database*.
4. Click *OK*.

To verify FortiGuard antivirus license information:

```
# diagnose debug rating
Locale       : english

Service      : Web-filter
Status       : Enable
License      : Contract

Service      : Antispam
Status       : Disable

Service      : Virus Outbreak Prevention
Status       : Enable
License      : Contract

-=- Server List (Tue Feb 19 16:36:15 2019) -=-

IP                     Weight    RTT Flags  TZ    Packets  Curr Lost Total Lost
   Updated Time
192.168.100.185          -218      2 DI     -8        113          0          0 Tue Feb
19 16:35:55 2019
```

To enable all scanunit debug categories:

```
# diagnose sys scanunit debug all
Set meta-category: all(0xffffffff)
Enabled categories(0xffffffff): daemon job quarantine analytics outbreak-prevention dlp
antispam file-filter
```

```
# diagnose debug enable
# su 4739 open
su 4739 req vfid 1 id 1 ep 0 new request, size 313, policy id 1, policy type 0
su 4739 req vfid 1 id 1 ep 0 received; ack 1, data type: 0
su 4739 job 1 request info:
su 4739 job 1   client 10.1.100.11:39412 server 172.16.200.44:80
su 4739 job 1   object_name 'zhvo_test.com'
su 4739 file-typing NOT WANTED options 0x0 file_filter no
su 4739 enable databases 0b (core mmdb extended)
su 4739 job 1 begin http scan
su 4739 scan file 'zhvo_test.com' bytes 68
su 4739 job 1 outbreak-prevention scan, level 0, filename 'zhvo_test.com'
su 4739 scan result 0
su 4739 job 1 end http scan
su 4739 job 1 inc pending tasks (1)
su 4739 not wanted for analytics: analytics submission is disabled (m 0 r 0)
su 4739 job 1 suspend
su 4739 outbreak-prevention recv error
su 4739 ftgd avquery id 0 status 1
su 4739 job 1 outbreak-prevention infected entryid=0
su 4739 report AVQUERY infection priority 1
su 4739 insert infection AVQUERY SUCCEEDED loc (nil) off 0 sz 0 at index 0 total infections
1 error 0
```

```
su 4739 job 1 dec pending tasks 0
su 4739 job 1 send result
su 4739 job 1 close
su 4739 outbreak-prevention recv error
```

External malware block list

The external malware block list allows users to add their own malware signatures in the form of MD5, SHA1, and SHA256 hashes. The FortiGate's antivirus database retrieves an external malware hash list from a remote server and polls the hash list every *n* minutes for updates. Enabling the AV engine scan is not required to use this feature.

The external malware block list can be used in both proxy-based and flow-based policy inspections, but it is not supported in AV quick scan mode.

Note that using different types of hashes simultaneously may slow down the performance of malware scanning. It is recommended to use one type of hash.

To create the external block list:

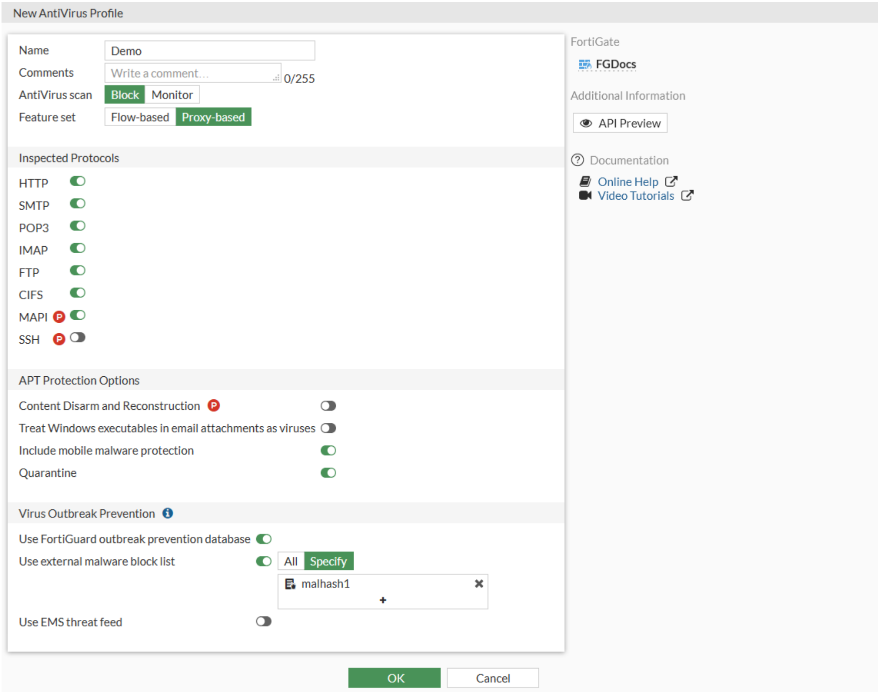
1. Create the malware hash list.
   The malware hash list follows a strict format in order for its contents to be valid. Malware hash signature entries must be separated into each line. A valid signature needs to follow this format:

   ```
   # MD5 Entry with hash description
   aa67243f746e5d76f68ec809355ec234  md5_sample1

   # SHA1 Entry with hash description
   a57983cb39e25ab80d7d3dc05695dd0ee0e49766  sha1_sample2

   # SHA256 Entry with hash description
   ae9bc0b4c5639d977d720e4271da06b50f7c60d1e2070e9c75cc59ab30e49379  sha256_sample1

   # Entry without hash description
   0289b0d967cb7b1fb1451339c7b9818a621903090e0020366ab415c549212521

   # Invalid entries
   7688499dc71b932feb126347289c0b8a_md5_sample2
   7614e98badca10b5e2d08f8664c519b7a906fbd5180ea5d04a82fce9796a4b87sha256_sample3
   ```
2. Configure the external malware block list source:
   a. Go to *Security Fabric > External Connectors* and click *Create New*.
   b. Click *Malware Hash*.
   c. Configure the settings as needed. The URI must point to the malware hash list on the remote server.
   d. Click *OK*.
3. To view entries inside the malware block list on the *External Connectors* page, hover over the malware hash card and click *View Entries*.

To configure antivirus to use an external block list in the GUI:

1. Go to *Security Profiles > AntiVirus* and edit the antivirus profile.
2. In the *Virus Outbreak Prevention* section, enable *Use external malware block list* and click *Specify*.
3. Click the + in the field and select a threat feed.

4. Optionally, enable *Quarantine*.



5. Configure the other settings as needed.
6. Click *OK*.

To configure antivirus to use an external block list in the CLI:

```
config antivirus profile
    edit "Demo"
        set feature-set proxy
        set mobile-malware-db enable
        config http
            set av-scan disable
            set outbreak-prevention block
            set external-blocklist block
            set quarantine enable
            set emulator enable
            set content-disarm disable
        end
        config ftp
            set av-scan disable
            set outbreak-prevention block
            set external-blocklist block
            set quarantine enable
            set emulator enable
        end
        config imap
            set av-scan monitor
            set outbreak-prevention block
            set external-blocklist block
            set quarantine enable
            set emulator enable
            set executables default
```

```
        set content-disarm disable
    end
    config pop3
        set av-scan monitor
        set outbreak-prevention block
        set external-blocklist block
        set quarantine enable
        set emulator enable
        set executables default
        set content-disarm disable
    end
    config smtp
        set av-scan monitor
        set outbreak-prevention block
        set external-blocklist block
        set quarantine enable
        set emulator enable
        set executables default
        set content-disarm disable
    end
    config mapi
        set av-scan monitor
        set outbreak-prevention block
        set external-blocklist block
        set quarantine enable
        set emulator enable
        set executables default
    end
    config nntp
        set av-scan disable
        set outbreak-prevention disable
        set external-blocklist disable
        set quarantine disable
        set emulator enable
    end
    config cifs
        set av-scan monitor
        set outbreak-prevention block
        set external-blocklist block
        set quarantine enable
        set emulator enable
    end
    config ssh
        set av-scan disable
        set outbreak-prevention disable
        set external-blocklist disable
        set quarantine disable
        set emulator enable
    end
    set outbreak-prevention-archive-scan enable
    set external-blocklist-enable-all disable
    set external-blocklist "malhash1"
    set av-virus-log enable
    set av-block-log enable
    set extended-log disable
    set scan-mode default
```

```
    next
end
```

The quarantine setting is configured in each protocol (`set quarantine`). The malware threat feed is also specified (`set external-blocklist-enable-all disable`) to the threat connector, malhash1 (`set external-blocklist "malhash1"`).

To verify the scanunit daemon updated itself with the external hashes:

```
# diagnose sys scanunit malware-list list
md5 'aa67243f746e5d76f68ec809355ec234' profile 'malhash1' description 'md5_sample1'
sha1 'a57983cb39e25ab80d7d3dc05695dd0ee0e49766' profile 'malhash1' description 'sha1_
sample2'
sha256 '0289b0d967cb7b1fb1451339c7b9818a621903090e0020366ab415c549212521' profile 'malhash1'
description ''
sha256 'ae9bc0b4c5639d977d720e4271da06b50f7c60d1e2070e9c75cc59ab30e49379' profile 'malhash1'
description 'sha256_sample1'
```

Malware threat feed from EMS

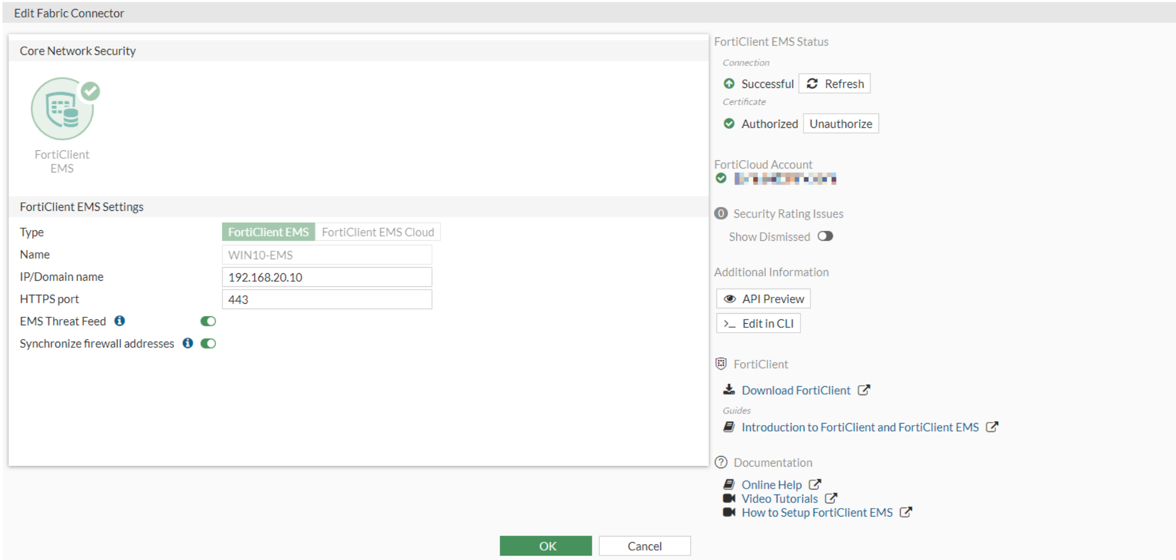
A FortiGate can pull malware threat feeds from FortiClient EMS, which in turn receives malware hashes detected by FortiClients. The malware hash can be used in an antivirus profile when AV scanning is enabled with block or monitor actions. This feature is supported in proxy and flow mode.

If an external malware blocklist and the FortiGuard outbreak prevention database are also enabled in the antivirus profile, the checking order is: AV local database, EMS threat feed, external malware blocklist, FortiGuard outbreak prevention database. If the EMS threat feed and external malware blocklist contain the same hash value, then the EMS infection will be reported if both of them are blocked.

To configure an EMS threat feed in an antivirus profile in the GUI:

1. Enable the EMS threat feed:
   a. Go to *Security Fabric > Fabric Connectors* and double-click the *FortiClient EMS* card.
   b. Enable *EMS Threat Feed*.

c. Configure the other settings if needed (see Configuring FortiClient EMS on page 2234 for more details).



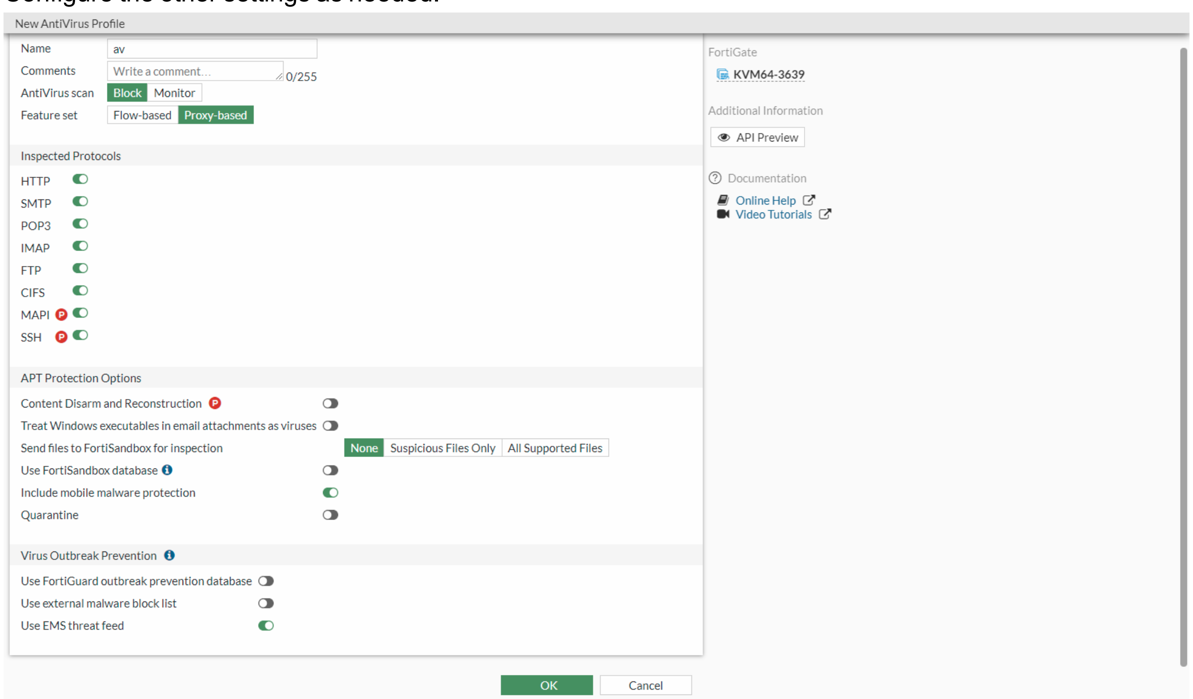
d. Click *OK*.

2. Create the antivirus profile:

a. Go to *Security Profiles > AntiVirus* and click *Create New*.

b. In the *Virus Outbreak Prevention* section, enable *Use EMS threat feed*.

c. Configure the other settings as needed.



d. Click *OK*.

To configure an EMS threat feed in an antivirus profile in the CLI:

1. Enable the EMS threat feed:

```
config endpoint-control fctems
    edit "WIN10-EMS"
        set fortinetone-cloud-authentication disable
        set server "192.168.20.10"
        set https-port 443
        set source-ip 0.0.0.0
        set pull-sysinfo enable
        set pull-vulnerabilities enable
        set pull-avatars enable
        set pull-tags enable
        set pull-malware-hash enable
        unset capabilities
        set call-timeout 30
        set websocket-override disable
    next
end
```

2. Create the antivirus profile:

```
config antivirus profile
    edit "av"
        config http
            set av-scan block
        end
        config ftp
            set av-scan block
        end
        config imap
            set av-scan block
        end
        config pop3
            set av-scan block
        end
        config smtp
            set av-scan block
        end
        config cifs
            set av-scan block
        end
        set external-blocklist-enable-all enable
        set ems-threat-feed enable
    next
end
```

Sample log

```
# execute log filter category utm-virus
# execute log display
```

```
1: date=2021-03-19 time=16:06:46 eventtime=1616195207055607417 tz="-0700" logid="0208008217"
type="utm" subtype="virus" eventtype="ems-threat-feed" level="notice" vd="vd1" policyid=1
msg="Detected by EMS threat feed." action="monitored" service="HTTPS" sessionid=1005
srcip=10.1.100.24 dstip=172.16.200.214 srcport=54674 dstport=443 srcintf="port2"
srcintfrole="undefined" dstintf="port1" dstintfrole="undefined" proto=6 direction="incoming"
```

```
filename="creditcardSSN.pdf" quarskip="Quarantine-disabled" virus="Email scan" dtype="File
Hash" filehash="22466078c2d52dfd5ebbbd6c4207ddec6ac61aa82f960dc54cfbc83b8eb42ed1"
filehashsrc="test" url="https://172.16.200.214/hash/creditcardSSN.pdf" profile="av"
agent="curl/7.68.0" analyticssubmit="false" crscore=10 craction=2 crlevel="medium"

2: date=2021-03-19 time=16:06:13 eventtime=1616195173832494609 tz="-0700" logid="0208008216"
type="utm" subtype="virus" eventtype="ems-threat-feed" level="warning" vd="vd1" policyid=1
msg="Blocked by EMS threat feed." action="blocked" service="HTTPS" sessionid=898
srcip=10.1.100.24 dstip=172.16.200.214 srcport=54672 dstport=443 srcintf="port2"
srcintfrole="undefined" dstintf="port1" dstintfrole="undefined" proto=6 direction="incoming"
filename="BouncingButton.pdf" quarskip="Quarantine-disabled" virus="Email scan" dtype="File
Hash" filehash="a601431acd5004c37bf8fd02fccfdacbb54b27c8648d1d41ad14fa3eaf8651d3"
filehashsrc="test" url="https://172.16.200.214/hash/BouncingButton.pdf" profile="av"
agent="curl/7.68.0" analyticssubmit="false" crscore=10 craction=2 crlevel="medium"
```

Checking flow antivirus statistics

Two CLI commands are used for the antivirus statistics:

- `diagnose ips av stats show`
- `diagnose ips av stats clear`

SNMP uses an API to get the antivirus statistics.

To check flow antivirus statistics:

1. Create an antivirus profile:

```
config antivirus profile
    edit "av-test"
        config http
            set av-scan monitor
        end
        config ftp
            set av-scan block
            set quarantine enable
        end
    next
end
```

2. Enable the profile in a firewall policy:

```
config firewall policy
    edit 1
        set name "policy1"
        set srcintf "port2"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set fsso disable
        set av-profile "av-test"
        set ssl-ssh-profile "custom-deep-inspection"
```

```
        set nat enable
    next
end
```

3. On the client PC, download the EICAR Standard Anti-Virus Test File via HTTP.
4. Check the antivirus statistics on the FortiGate. Since the action is set to monitor for HTTP, `HTTP virus detected` increases by 1:

```
# diagnose ips av stats show
AV stats:
HTTP virus detected: 1
HTTP virus blocked: 0
SMTP virus detected: 0
SMTP virus blocked: 0
POP3 virus detected: 0
POP3 virus blocked: 0
IMAP virus detected: 0
IMAP virus blocked: 0
NNTP virus detected: 0
NNTP virus blocked: 0
FTP virus detected: 0
FTP virus blocked: 0
SMB virus detected: 0
SMB virus blocked: 0
```

5. On the client PC, download the EICAR file via FTP.
6. Check the antivirus statistics on the FortiGate. Since quarantine is enabled for FTP, `FTP virus detected` and `FTP virus blocked` increase by 1:

```
# diagnose ips av stats show
AV stats:
HTTP virus detected: 1
HTTP virus blocked: 0
SMTP virus detected: 0
SMTP virus blocked: 0
POP3 virus detected: 0
POP3 virus blocked: 0
IMAP virus detected: 0
IMAP virus blocked: 0
NNTP virus detected: 0
NNTP virus blocked: 0
FTP virus detected: 1
FTP virus blocked: 1
SMB virus detected: 0
SMB virus blocked: 0
```

7. Check the antivirus statistics using an SNMP walk:

```
root:~# snmpwalk -c public -v 1 10.1.100.6 1.3.6.1.4.1.12356.101.8.2.1.1
iso.3.6.1.4.1.12356.101.8.2.1.1.1.1 = Counter32: 2  (fgAvVirusDetected)
iso.3.6.1.4.1.12356.101.8.2.1.1.2.1 = Counter32: 1  (fgAvVirusBlocked)
iso.3.6.1.4.1.12356.101.8.2.1.1.3.1 = Counter32: 1  (fgAvHTTPVirusDetected)
iso.3.6.1.4.1.12356.101.8.2.1.1.4.1 = Counter32: 0
iso.3.6.1.4.1.12356.101.8.2.1.1.5.1 = Counter32: 0
iso.3.6.1.4.1.12356.101.8.2.1.1.6.1 = Counter32: 0
iso.3.6.1.4.1.12356.101.8.2.1.1.7.1 = Counter32: 0
iso.3.6.1.4.1.12356.101.8.2.1.1.8.1 = Counter32: 0
```

```
iso.3.6.1.4.1.12356.101.8.2.1.1.9.1 = Counter32: 0
iso.3.6.1.4.1.12356.101.8.2.1.1.10.1 = Counter32: 0
iso.3.6.1.4.1.12356.101.8.2.1.1.11.1 = Counter32: 1  (fgAvFTPVirusDetected)
iso.3.6.1.4.1.12356.101.8.2.1.1.12.1 = Counter32: 1  (fgAvFTPVirusBlocked)
iso.3.6.1.4.1.12356.101.8.2.1.1.13.1 = Counter32: 0
iso.3.6.1.4.1.12356.101.8.2.1.1.14.1 = Counter32: 0
iso.3.6.1.4.1.12356.101.8.2.1.1.15.1 = Counter32: 0
iso.3.6.1.4.1.12356.101.8.2.1.1.16.1 = Counter32: 0
iso.3.6.1.4.1.12356.101.8.2.1.1.17.1 = Counter32: 0
iso.3.6.1.4.1.12356.101.8.2.1.1.18.1 = Counter32: 0
iso.3.6.1.4.1.12356.101.8.2.1.1.19.1 = Counter32: 0
iso.3.6.1.4.1.12356.101.8.2.1.1.20.1 = Counter32: 0
iso.3.6.1.4.1.12356.101.8.2.1.1.21.1 = Counter32: 0
iso.3.6.1.4.1.12356.101.8.2.1.1.22.1 = Counter32: 0
```

8. Optionally, reset the antivirus statistics to zero:

```
# diagnose ips av stats clear
```

CIFS support

Antivirus scanning on Common Internet File System (CIFS) traffic is supported in flow-based and proxy-based inspection. The file filter profile handles the configuration of file filtering on CIFS. The antivirus profile handles the antivirus configuration for CIFS scanning.

File filtering for CIFS is performed by inspecting the first 4 KB of the file to identify the file's magic number. If a match occurs, CIFS file filtering prevents the CIFS command that contains that file from running. The file filter functions differently for un-encrypted and encrypted CIFS traffic:

- For un-encrypted CIFS traffic, the standalone file filter works in flow and proxy mode.
- For encrypted CIFS traffic, the CIFS profile must be enabled in the firewall policy because the SMB server's credential settings are still be configured in CIFS profile. Using the standalone file filter only works in proxy mode.

For a CIFS profile to be available for assignment in a policy, the policy must use proxy inspection mode. See Proxy mode inspection on page 1085 for details. Note that in proxy inspection mode, special condition archive files (encrypted, corrupted, mailbomb, and so on) marked by the antivirus engine are blocked automatically.

Messages that are compressed with LZNT1, LZ77, and LZ77+Huffman algorithms can be scanned in proxy mode.

Configure file-type filtering and antivirus scanning on CIFS traffic

To configure file-type filtering and antivirus scanning on CIFS traffic:

1. Configure a CIFS domain controller on page 1106
2. Configure a CIFS profile on page 1107
3. Configure an antivirus profile on page 1109

Configure a CIFS domain controller

The domain controller must be configured when CIFS traffic is encrypted. The configuration tells the FortiGate the network location of the domain controller and the superuser credentials.

To configure the CIFS domain controller:

```
config user domain-controller
    edit "SERVER_NAME"
        set hostname "host"
        set domain-name "EXAMPLE.COM"
        set username "admin-super"
        set password *********
        set ip 172.16.201.40
    next
end
```

Configure a CIFS profile

To create a CIFS profile, configure the server credential type and create a file filter profile.

Set the CIFS server credential type

The CIFS server credential type can be `none`, `credential-replication`, or `credential-keytab`.

none

The CIFS profile assumes the CIFS traffic is unencrypted. This is the default value.

```
config firewall profile-protocol-options
    edit "cifs"
        config cifs
            set server-credential-type none
        end
    next
end
```

credential-replication

To decrypt CIFS traffic, FortiOS obtains the session key from the domain controller by logging in to the superuser account. The domain controller must be configured.

```
config firewall profile-protocol-options
    edit "cifs"
        config cifs
            set server-credential-type credential-replication
            set domain-controller "SERVER_NAME"
        end
    next
end
```

Variable	Description
domain-controller <string>	The previously configured domain to decrypt CIFS traffic for.

credential-keytab

To decrypt CIFS traffic, FortiOS uses a series of keytab values. This method is used when the SMB connection is authenticated by Kerberos. Keytab entries must be configured, and are stored in FortiOS in plaintext.

```
config firewall profile-protocol-options
    edit "cifs"
        config cifs
            set server-credential-type credential-keytab
            config server-keytab
                edit "keytab1"
                    set keytab
"BQIAAABFAAEAC0VYQU1QTEUuQ09NAAdleGFtcGxlAAAAAVUmAlwBABIAILdV5P6NXT8RrTvapcMJQxDYCjRQiD0Bzxh
wS9h0VgyM"
                next
            end
        end
    next
end
```

Variable	Description
keytab <keytab>	Base64 encoded keytab file containing the credentials of the server.

Configure CIFS file filtering

Multiple rules can be added to a file filter profile. See File filter on page 1225.

To configure a file filter for CIFS traffic:

```
config file-filter profile
    edit "cifs"
        set comment "block zip files on unencrypted cifs traffic"
        set feature-set flow
        set replacemsg-group ''
        set log enable
        config rules
            edit "rule1"
                set protocol cifs
                set action block
                set direction any
                set password-protected any
                set file-type zip
            next
        end
    next
end
```

Variable	Description
comment <string>	A brief comment describing the entry.
feature-set {flow | proxy}	Flow or proxy mode feature set (default = `flow`).
replacemsg-group <string>	Replacement message group.
log {enable | disable}	Enable/disable file filter logging (default = `enable`).
scan-archive-contents [enable | disable]	Enable/disable scanning of archive contents (default = `enable`).

Variable	Description
protocol {http ftp smtp imap pop3 mapi cifs ssh}	Filter based on the specified protocol(s).
action {log-only | block}	The action to take for matched files: • `log-only`: Allow the content and write a log message (default). • `block`: Block the content and write a log message.
direction {incoming | outgoing | any}	Match files transmitted in the session's originating (`incoming`) and/or reply (`outgoing`) direction (default = `any`).
password-protected [yes | any]	Match only password-protected files (`yes`) or any file (default = `any`).
file-type <file_type>	The file types to be matched. See Supported file types on page 1230 for details.

Configure an antivirus profile

The antivirus profile handles the antivirus configuration for CIFS scanning.

To configure an antivirus profile:

```
config antivirus profile
    edit "av"
        ...
        config cifs
            set av-scan {disable | block | monitor}
            set outbreak-prevention {disable | block | monitor}
            set external-blocklist {disable | block | monitor}
            set quarantine {enable | disable}
            set archive-block {encrypted corrupted partiallycorrupted multipart nested
mailbomb fileslimit timeout unhandled}
            set archive-log {encrypted corrupted partiallycorrupted multipart nested
mailbomb fileslimit timeout unhandled}
            set emulator {enable | disable}
        end
    next
end
```

Variable	Description
av-scan	Enable antivirus scan service: • `disable`: Disable (default). • `block`: Block the virus infected files. • `monitor`: Log the virus infected files.
outbreak-prevention {disable | block | monitor}	Enable the virus outbreak prevention service: • `disable`: Disable (default). • `block`: Block the matched files. • `monitor`: Log the matched files.
external-blocklist {disable | block | monitor}	Enable the external blocklist: • `disable`: Disable (default). • `block`: Block the matched files.

Variable	Description
	• `monitor`: Log the matched files.
quarantine {enable | disable}	Enable/disable quarantine for infected files (default = `disable`).
archive-block {encrypted corrupted partiallycorrupted multipart nested mailbomb fileslimit timeout unhandled}	Select the archive types to block: • `encrypted`: Block encrypted archives. • `corrupted`: Block corrupted archives. • `partiallycorrupted`: Block partially corrupted archives. • `multipart`: Block multipart archives. • `nested`: Block nested archives. • `mailbomb`: Block mail bomb archives. • `fileslimit`: Block exceeded archive files limit. • `timeout`: Block scan timeout. • `unhandled`: Block archives that FortiOS cannot open.
archive-log {encrypted corrupted partiallycorrupted multipart nested mailbomb fileslimit timeout unhandled}	Select the archive types to log: • `encrypted`: Log encrypted archives. • `corrupted`: Log corrupted archives. • `partiallycorrupted`: Log partially corrupted archives. • `multipart`: Log multipart archives. • `nested`: Log nested archives. • `mailbomb`: Log mail bomb archives. • `fileslimit`: Log exceeded archive files limit. • `timeout`: Log scan timeout. • `unhandled`: Log archives that FortiOS cannot open.
emulator {enable | disable}	Enable/disable the virus emulator (default = `enable`).

Log samples

File-type detection events generated by CIFS profiles are logged in the `utm-cifs` log category. Antivirus detection over the CIFS protocol generates logs in the `utm-virus` category. See the FortiOS Log Message Reference for more information.

Logs generated by CIFS profile file filter:

```
date=2019-03-28 time=10:39:19 logid="1800063001" type="utm" subtype="cifs" eventtype="cifs-filefilter" level="notice" vd="vdom1" eventtime=1553794757 msg="File was detected by file filter." direction="incoming" action="passthrough" service="CIFS" srcip=10.1.100.11 dstip=172.16.200.44 srcport=33372 dstport=445 srcintf="wan2" srcintfrole="wan" dstintf="wan1" dstintfrole="wan" policyid=1 proto=16 profile="cifs" filesize="1154" filename="virus\\test.png" filtername="2" filetype="png"
```

```
date=2019-03-28 time=10:39:12 logid="1800063001" type="utm" subtype="cifs" eventtype="cifs-filefilter" level="notice" vd="vdom1" eventtime=1553794751 msg="File was detected by file filter." direction="incoming" action="passthrough" service="CIFS" srcip=10.1.100.11 dstip=172.16.200.44 srcport=33370 dstport=445 srcintf="wan2" srcintfrole="wan" dstintf="wan1" dstintfrole="wan" policyid=1 proto=16 profile="cifs" filesize="81975" filename="virus\\screen.png" filtername="2" filetype="png"
```

```
date=2019-03-28 time=10:33:55 logid="1800063000" type="utm" subtype="cifs" eventtype="cifs-filefilter" level="warning" vd="vdom1" eventtime=1553794434 msg="File was blocked by file filter." direction="incoming" action="blocked" service="CIFS" srcip=10.1.100.11 dstip=172.16.200.44 srcport=33352 dstport=445 srcintf="wan2" srcintfrole="wan" dstintf="wan1" dstintfrole="wan" policyid=1 proto=16 profile="cifs" filesize="28432" filename="filetypes\\mpnotify.exe" filtername="3" filetype="exe"
```

```
date=2019-03-28 time=10:33:45 logid="1800063000" type="utm" subtype="cifs" eventtype="cifs-filefilter" level="warning" vd="vdom1" eventtime=1553794424 msg="File was blocked by file filter." direction="incoming" action="blocked" service="CIFS" srcip=10.1.100.11 dstip=172.16.200.44 srcport=33348 dstport=445 srcintf="wan2" srcintfrole="wan" dstintf="wan1" dstintfrole="wan" policyid=1 proto=16 profile="cifs" filesize="96528" filename="filetypes\\winmine.exe" filtername="3" filetype="exe"
```

Logs generated by AV profile for infections detected over CIFS:

```
date=2019-04-09 time=15:19:02 logid="0204008202" type="utm" subtype="virus" eventtype="outbreak-prevention" level="warning" vd="vdom1" eventtime=1554848342519005401 msg="Blocked by Virus Outbreak Prevention service." action="blocked" service="SMB" sessionid=177 srcip=10.1.100.11 dstip=172.16.200.44 srcport=37444 dstport=445 srcintf="wan2" srcintfrole="wan" dstintf="wan1" dstintfrole="wan" policyid=1 proto=6 direction="incoming" filename="outbreak\\zhvo_test.com" quarskip="File-was-not-quarantined." virus="503e99fe40ee120c45bc9a30835e7256fff3e46a" dtype="File Hash" filehash="503e99fe40ee120c45bc9a30835e7256fff3e46a" filehashsrc="fortiguard" profile="av" analyticssubmit="false" crscore=50 craction=2 crlevel="critical"
```

```
date=2019-04-09 time=15:18:59 logid="0211008192" type="utm" subtype="virus" eventtype="infected" level="warning" vd="vdom1" eventtime=1554848339909808987 msg="File is infected." action="blocked" service="SMB" sessionid=174 srcip=10.1.100.11 dstip=172.16.200.44 srcport=37442 dstport=445 srcintf="wan2" srcintfrole="wan" dstintf="wan1" dstintfrole="wan" policyid=1 proto=6 direction="incoming" filename="sample\\eicar.com" quarskip="File-was-not-quarantined." virus="EICAR_TEST_FILE" dtype="Virus" ref="http://www.fortinet.com/ve?vn=EICAR_TEST_FILE" virusid=2172 profile="av" analyticscksum="275a021bbfb6489e54d471899f7db9d1663fc695ec2fe2a2c4538aabf651fd0f" analyticssubmit="false" crscore=50 craction=2 crlevel="critical"
```

Using FortiSandbox post-transfer scanning with antivirus

Antivirus profiles can submit potential zero-day viruses to FortiSandbox for inspection. Based on FortiSandbox's analysis, the FortiGate can supplement its own antivirus database with FortiSandbox's threat intelligence to detect files determined as malicious or suspicious. This augments the FortiGate antivirus with zero-day detection.

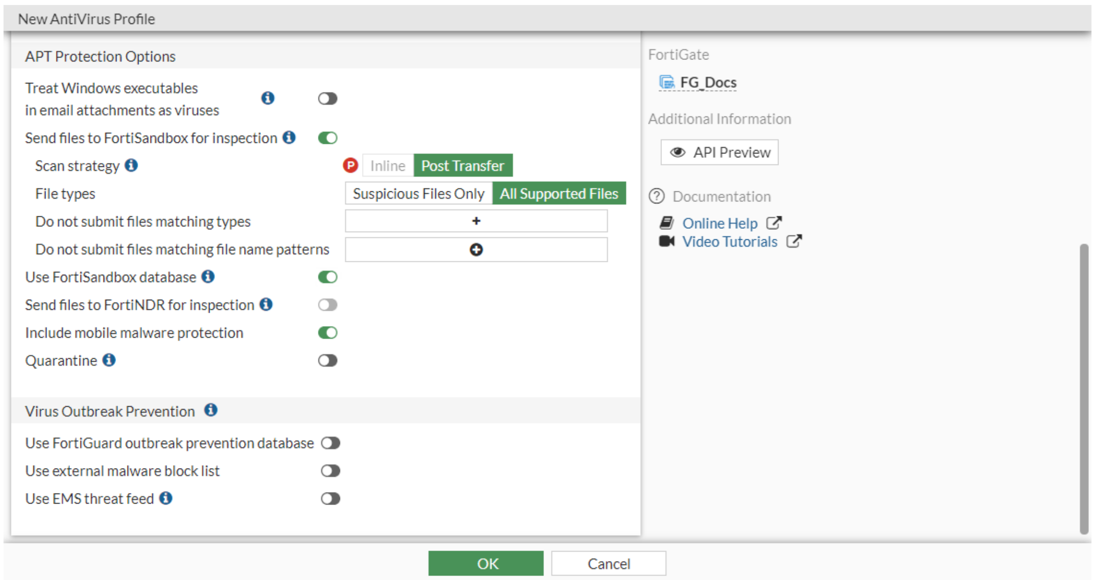
FortiSandbox can be used with antivirus in both proxy-based and flow-based inspection modes. The FortiGate first examines the file for any known viruses. When a match is found, the file is tagged as known malware. If no match is found, the files are forwarded to FortiSandbox using the following options:

- *All Supported Files*: all files matching the file types defined in the scan profile of the FortiSandbox are forwarded.
- *Suspicious Files Only*: files classified by the antivirus as having any possibility of active content are forwarded to FortiSandbox. When using FortiGate Cloud Sandbox, we recommend selecting this option due to its submission limits.
- *None*: files are not forwarded to FortiSandbox.

For more information, see Configuring Sandboxing on page 2228.

To enable FortiSandbox inspection in an antivirus profile:

1. Go to *Security Profiles > AntiVirus*.
2. Create, edit, or clone an antivirus profile.
3. In the *APT Protection Options* section, set *Send Files to FortiSandbox for Inspection* to either *Suspicious Files Only* or *All Supported Files*.
4. Optionally, for *Do not submit files matching types*, click the + to exclude certain file types from being sent to FortiSandbox.
5. Optionally, for *Do not submit files matching file name patterns*, click the + to enter a wildcard pattern to exclude files from being sent to FortiSandbox.



6. Enable *Use FortiSandbox Database*.
7. Click *OK*.

FortiGate diagnostics

To view the detection count:

```
# diagnose test application quarantined 7
Total: 0

Statistics:
        vfid: 0, detected: 2, clean: 1252, risk_low: 6, risk_med: 2, risk_high: 1, limit_
reached:0
```

To verify the address is configured correctly:

```
# diagnose test application quarantined 1
…
fortisandbox-fsb1 is enabled: analytics, realtime=yes, taskfull=no
addr=172.18.52.154/514, source-ip=0.0.0.0, keep-alive=no. ssl_opt=3, hmac_alg=0
…
```

To run the diagnostics for real-time debugging:

```
# diagnose debug application quarantined -1
# diagnose debug enable
```

To check the FortiGate Cloud server status:

```
# diagnose test application forticldd 3
…
    Active APTServer status:  up
```

To view FortiGate Cloud Sandbox submission statistics for advanced debugging:

```
# diagnose test application quarantined 2
```

FortiSandbox diagnostics

To run the OFTP debug for advanced debugging:

```
# diagnose-debug device <client serial number>
```

Using FortiSandbox inline scanning with antivirus

FortiSandbox inline scanning can be used in proxy inspection mode. When inline scanning is enabled, the client's file is held while it is sent to FortiSandbox for inspection. During this time, the FortiGate may apply client comforting (see Protocol options on page 915). For example, leaking a certain amount of bytes at a certain time interval to the client. Once a verdict is returned, the appropriate action (allow or block) is performed on the held file. If there is an error connecting to the FortiSandbox or a timeout on the FortiSandbox scanning the file within the default 50 seconds, the file can be passed, logged, or blocked based on FortiGate's configuration.

Inline scanning requires a FortiSandbox appliance running version 4.2 or later, and the FortiSandbox must be reachable by port 4443. This feature is not supported on FortiSandbox Cloud or FortiGate Cloud Sandbox. See Understanding Inline Block feature in the FortiSandbox Best Practices for more information.

FortiSandbox inline scanning is disabled by default. FortiSandbox inline scanning is best used in conjunction with AV engine scanning since there is a higher rate of detection by using both at the same time.

To enable FortiSandbox inline scanning:

```
config system fortisandbox
    set status enable
    set inline-scan {enable | disable}
    set server <fortisandbox_server_ip>
end
```

To configure the FortiSandbox scanning options in an antivirus profile:

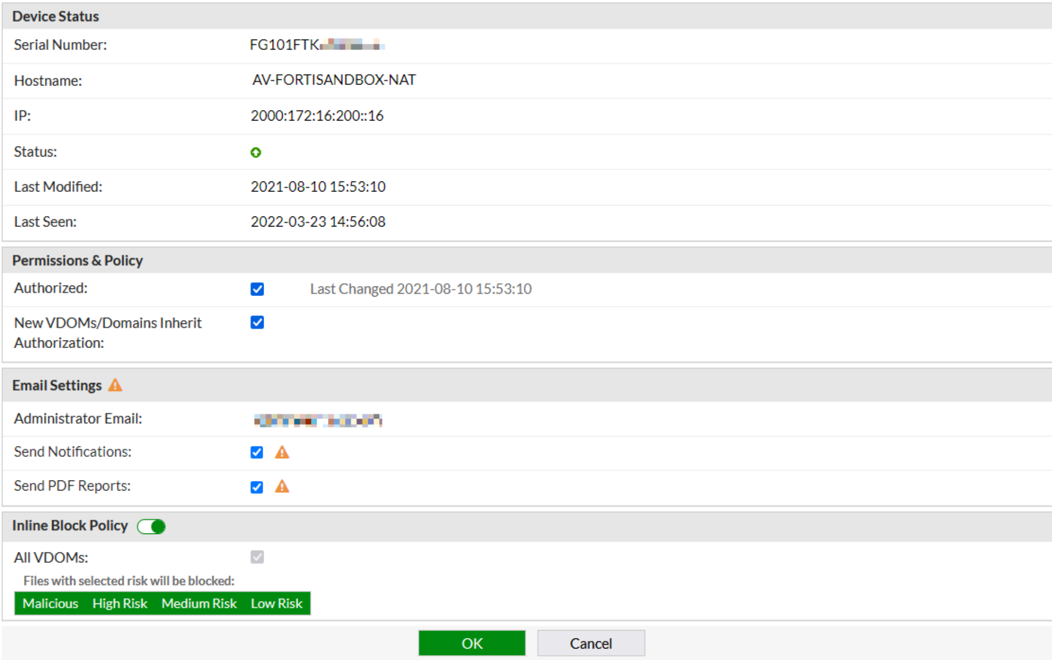
```
config antivirus profile
    edit <name>
```

```
        set fortisandbox-mode {inline | analytics-suspicious | analytics-everything}
        set fortisandbox-error-action {ignore | log-only | block}
        set fortisandbox-timeout-action {ignore | log-only | block}
        set fortisandbox-max-upload <integer>
        config {http | ftp | imap | pop3 | smtp | mapi | cifs | ssh}
            set av-scan {disable | block | monitor}
            set fortisandbox {disable | block | monitor}
        end
    next
end
```

`fortisandbox-mode {inline	analytics-suspicious	analytics-everything}`	Set the FortiSandbox scan mode: • `inline`: FortiSandbox inline scanning • `analytics-suspicious`: FortiSandbox post-transfer scanning; submit supported files if heuristics or other methods determine they are suspicious • `analytics-everything`: FortiSandbox post-transfer scanning; submit supported files and known infected files (default)
`fortisandbox-error-action {ignore	log-only	block}`	Set the action to take if FortiSandbox inline scanning encounters an error reaching the FortiSandbox: • `ignore`: take no action • `log-only`: log the FortiSandbox inline scan error, but allow the file (default) • `block`: block the file upon FortiSandbox inline scan error
`fortisandbox-timeout-action {ignore	log-only	block}`	Set the action to take if FortiSandbox inline scanning encounters a scan timeout: • `ignore`: take no action • `log-only`: log the FortiSandbox inline scan timeout, but allow the file (default) • `block`: block the file upon FortiSandbox inline scan timeout
`fortisandbox-max-upload <integer>`	Set the maximum size of files that can be uploaded to FortiSandbox (1 - 396, default = 10).
`av-scan {disable	block	monitor}`	Enable the antivirus scan service. Set to `block` or `monitor` to work with FortiSandbox (default = `disable`).
`fortisandbox {disable	block	monitor}`	Set the protocol level parameter for FortiSandbox file scanning: • `disable` (default), `block`, and `monitor` are available for inline scanning • `disable` (default) and `monitor` are available for post-transfer scanning

Basic configuration

This example assumes that *Inline Block Policy* is already enabled in FortiSandbox for the FortiGate with selected risk levels (see FortiGate devices in the FortiSandbox Administration Guide for more information). The inline block policy in this example blocks all risk levels: malicious, high risk, medium risk, and low risk.

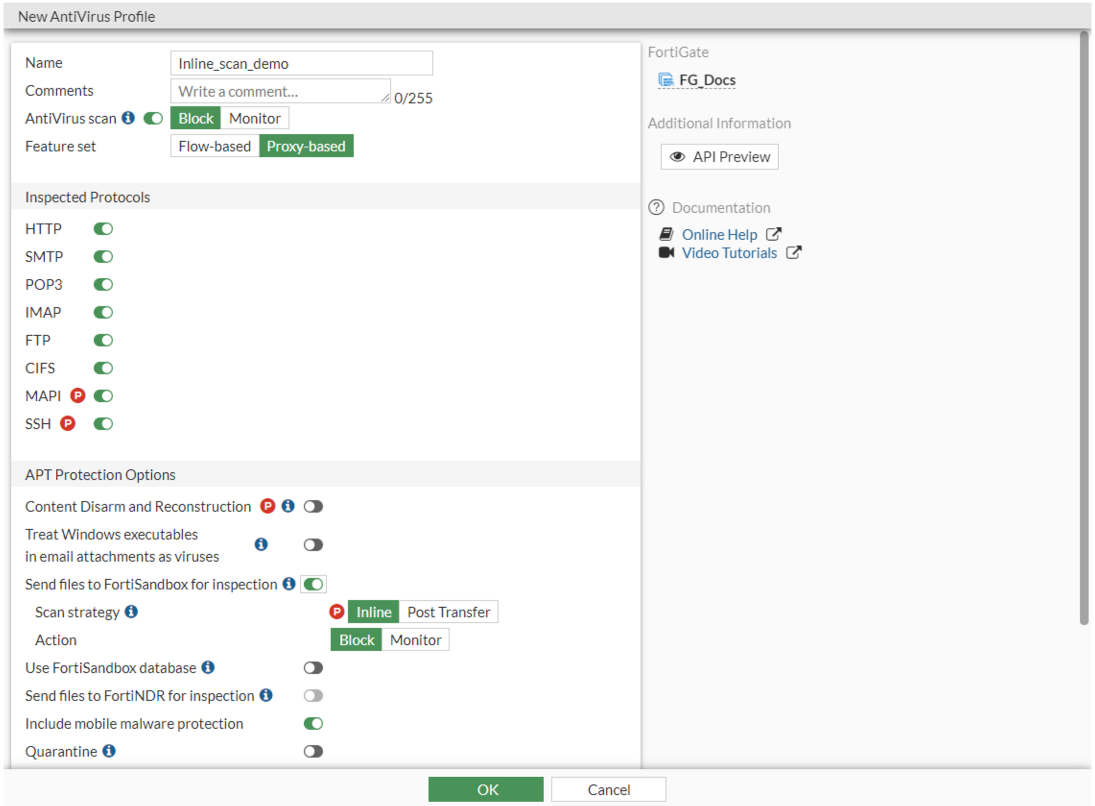


To configure FortiSandbox inline scanning in the GUI:

1. Enable FortiSandbox inline scanning globally:

```
config system fortisandbox
    set status enable
    set inline-scan enable
    set server "172.18.70.76"
end
```

2. Configure the antivirus profile:
   a. Go *to Security Profiles > AntiVirus* and click *Create New*.
   b. Set the *Feature set* to *Proxy-based*.
   c. Enable the protocols to inspect.
   d. Enable *AntiVirus scan* and set it to *Block*.
   e. Enable *Send files to FortiSandbox* for inspection and set the *Action* to *Block*. The *Scan strategy* appears as *Inline* because it was configured in the CLI.



f. Click *OK*.

To configure FortiSandbox inline scanning in the CLI:

1. Enable FortiSandbox inline scanning globally:

```
config system fortisandbox
    set status enable
    set inline-scan enable
    set server "172.18.70.76"
end
```

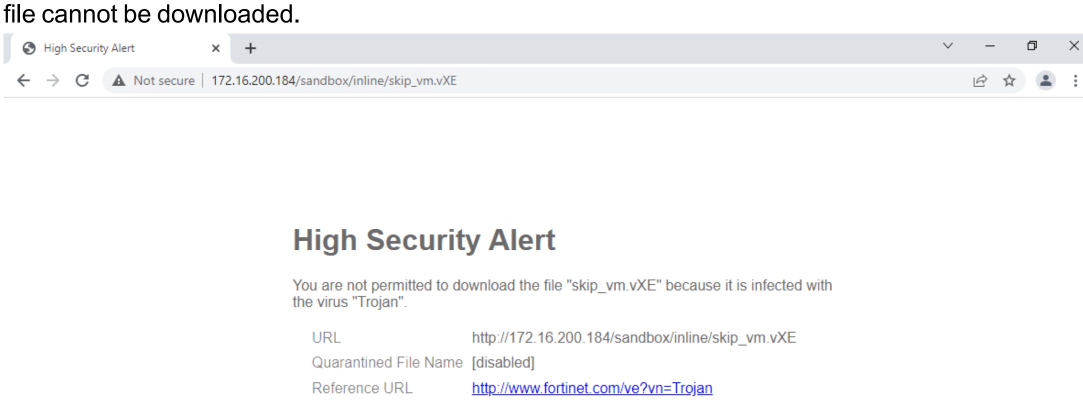
2. Configure the antivirus profile:

```
config antivirus profile
    edit "Inline_scan_demo"
        set feature-set proxy
        set fortisandbox-mode inline
        config http
            set av-scan block
            set fortisandbox block
        end
        config ftp
            set av-scan block
            set fortisandbox block
        end
        config imap
            set av-scan block
            set fortisandbox block
```

```
        end
        config pop3
            set av-scan block
            set fortisandbox block
        end
        config smtp
            set av-scan block
            set fortisandbox block
        end
        config mapi
            set av-scan block
            set fortisandbox block
        end
        config cifs
            set av-scan block
            set fortisandbox block
        end
        config ssh
            set av-scan block
            set fortisandbox block
        end
    next
end
```

To verify that infected files are blocked inline:

1. On a client, open a web browser and download an infected file.
2. The file is held while being scanned by FortiSandbox. Once FortiSandbox determines that file's risk level is not tolerated by the inline block policy, the FortiGate drops the connection and displays a replacement message that the file cannot be downloaded.



3. In FortiOS, view the antivirus log.
   - In the GUI, go to *Log & Report > Security Events* and click the *AntiVirus* card.
   - In the CLI:

```
# execute log filter category 2
# execute log display
1 logs found.
1 logs returned.

1: date=2022-03-23 time=16:19:37 eventtime=1648077577156255080 tz="-0700"
```

```
logid="0210008232" type="utm" subtype="virus" eventtype="fortisandbox"
level="warning" vd="vdom1" policyid=1 poluuid="9170ca3e-aade-51ec-772b-1d31f135fe26"
policytype="policy" msg="Blocked by FortiSandbox." action="blocked" service="HTTP"
sessionid=10545 srcip=10.1.100.181 dstip=172.16.200.184 srcport=37046 dstport=80
srccountry="Reserved" dstcountry="Reserved" srcintf="port1" srcintfrole="undefined"
dstintf="port9" dstintfrole="undefined" srcuuid="5b426c60-aade-51ec-f020-
b3d334ba18d3" dstuuid="5b426c60-aade-51ec-f020-b3d334ba18d3" proto=6
direction="incoming" filename="skip_vm.vXE" quarskip="File-was-not-quarantined"
virus="Trojan" viruscat="Unknown" dtype="fortisandbox"
ref="http://www.fortinet.com/ve?vn=Trojan" virusid=0
url="http://172.16.200.184/sandbox/inline/skip_vm.vXE" profile="Inline_scan_demo"
agent="curl/7.68.0" httpmethod="GET" analyticssubmit="false" fsaaction="deny"
fsaseverity="high-risk" fsaverdict="block" fsafileid=0 fsafiletype="exe" crscore=50
craction=2 crlevel="critical
```

Configuration with FortiSandbox scanning error and timeout actions

In this example, the HTTP protocol settings for `av-scan` and `fortisandbox` in the AV profile are both set to `block`. All files traversing HTTP in this configuration are scanned by the AV engine first, and then by FortiSandbox inline scanning for further file analysis. Based on the FortiSandbox results, FortiOS will take the appropriate action.

Files can be blocked if they contain a scan error or timeout. The scan timeout is configured in FortiSandbox and set to 50 seconds. If the file scan takes longer than 50 seconds, FortiSandbox returns a timeout to the FortiGate, and file is dropped with the current configuration. If a user tries to download the same file again, the cached result is provided by FortiSandbox to the FortiGate based on the previous file scan.

This example assumes FortiSandbox inline scanning has been configured globally. The FortiGate will block the file if there is an inline scanning error or timeout.

To configure the antivirus profile to block files if there is an inline scanning error or timeout:

```
config antivirus profile
    edit "av"
        set feature-set proxy
        set fortisandbox-mode inline
        config http
            set av-scan block
            set fortisandbox block
        end
        set fortisandbox-error-action block
        set fortisandbox-timeout-action block
    next
end
```

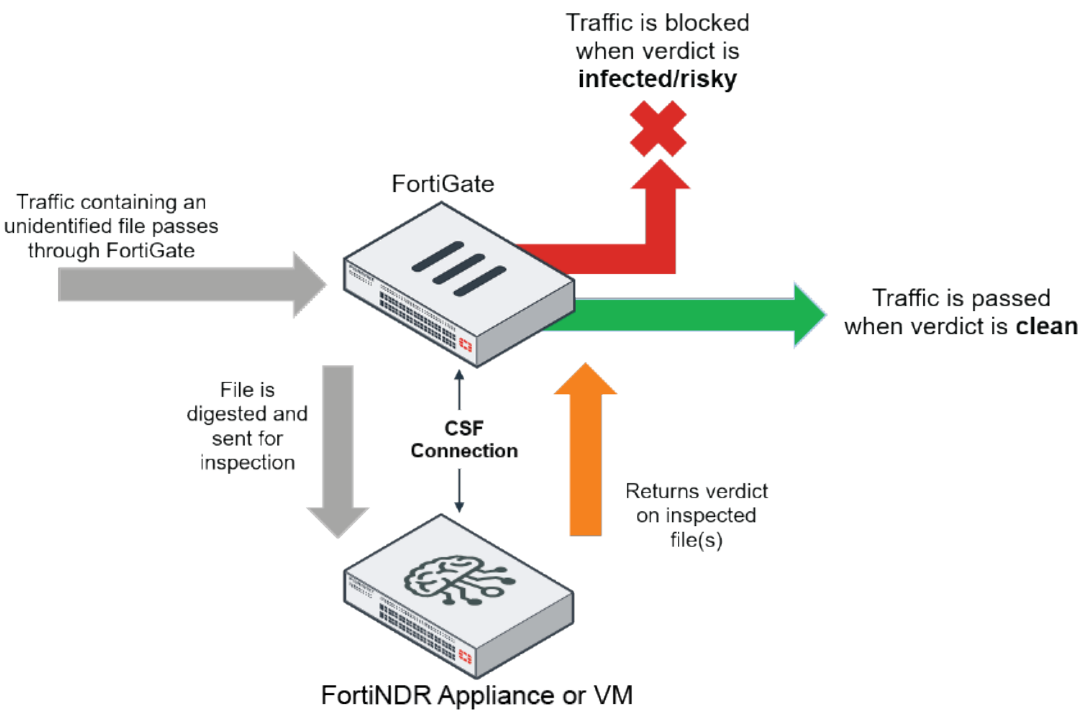
If the administrator decides to take more risk and scan all files traversing HTTP, but log or ignore an inline scanning error or timeout, the profile is modified as follows:

```
config antivirus profile
    edit "av"
        set fortisandbox-error-action {log-only | ignore}
        set fortisandbox-timeout-action {log-only | ignore}
    next
end
```

The AV engine is still used first, followed by FortiSandbox inline scanning. The FortiGate will log or ignore the file if there is an inline scanning error or timeout, and the file is allowed to pass through.

Using FortiNDR inline scanning with antivirus

FortiNDR (formerly FortiAI) can be used with antivirus profiles in proxy inspection mode (flow mode is currently not supported). FortiNDR inspects high-risk files and issues a verdict to the firewall based on how close the file features match those of malware. When enabled, FortiNDR can log, block, ignore, or monitor (allow) the file based on the verdict.



A licensed FortiNDR appliance with version 1.5.1 or later is required to use this feature.

To configure FortiNDR inline inspection with an AV profile:

1. Configure FortiNDR to join a Security Fabric in FortiOS (see Configuring FortiNDR on page 2255).
2. In the FortiNDR CLI, enable inline inspection:

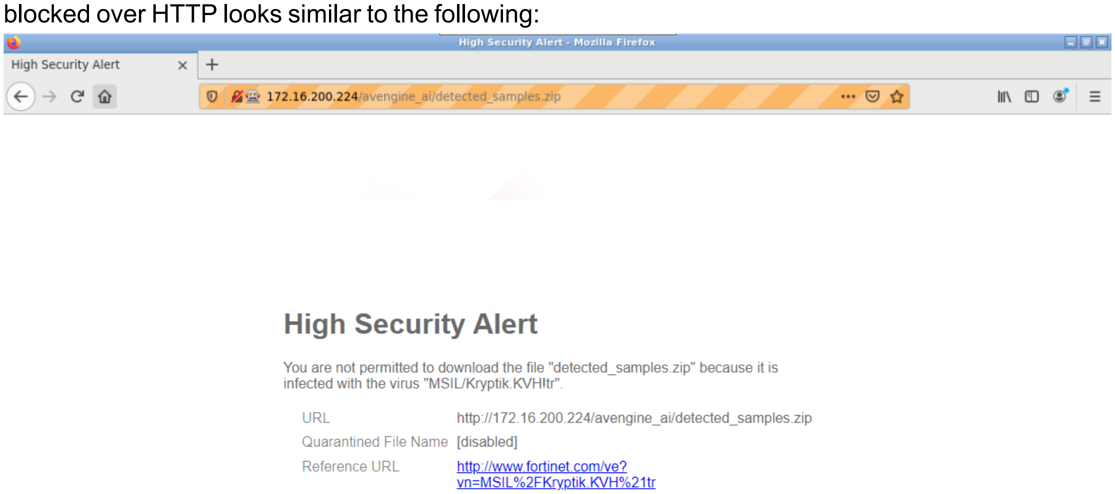
```
config system fortindr
    set status enable
end
```

3. Configure an AV profile in FortiOS to use inline inspection and block detected infections:

```
config antivirus profile
    edit "av"
        set feature-set proxy
        config http
```

```
            set fortindr block
        end
        config ftp
            set fortindr block
        end
        config imap
            set fortindr block
        end
        config pop3
            set fortindr block
        end
        config smtp
            set fortindr block
        end
        config mapi
            set fortindr block
        end
        config nntp
            set fortindr block
        end
        config cifs
            set fortindr block
        end
        config ssh
            set fortindr block
        end
    next
end
```

4. Add the AV profile to a firewall policy. When potential infections are blocked by FortiNDR inline inspection, a replacement message appears (see Replacement messages on page 2134 for more information). An infection blocked over HTTP looks similar to the following:



Sample log

```
date=2021-04-29 time=15:12:07 eventtime=1619734327633022960 tz="-0700" logid="0209008221"
type="utm" subtype="virus" eventtype="fortindr" level="notice" vd="vdom1" policyid=1
msg="Detected by FortiNDR." action="monitored" service="HTTP" sessionid=13312
srcip=10.1.100.221 dstip=172.16.200.224 srcport=50792 dstport=80 srcintf="wan2"
srcintfrole="wan" dstintf="wan1" dstintfrole="wan" proto=6 direction="incoming"
```

```
filename="detected_samples.zip" quarskip="File-was-not-quarantined"
virus="MSIL/Kryptik.KVH!tr" dtype="fortindr"
ref="http://www.fortinet.com/ve?vn=MSIL%2FKryptik.KVH%21tr" virusid=0
url="http://172.16.200.224/avengine_ai/detected_samples.zip" profile="av"
agent="curl/7.68.0" analyticssubmit="false" crscore=50 craction=2 crlevel="critical"
```

FortiNDR inline inspection with other AV inspection methods

The following inspection logic applies when FortiNDR inline inspection is enabled simultaneously with other AV inspection methods. The AV engine inspection and its verdict always takes precedence because of performance. The actual behavior depends on which inspected protocol is used.

HTTP, FTP, SSH, and CIFS protocols:

1. AV engine scan; AV database and FortiSandbox database (if applicable).
   a. FortiNDR inline inspection occurs simultaneously.
2. AV engine machine learning detection for WinPE PUPs (potentially unwanted programs).
   a. FortiNDR inline inspection occurs simultaneously.
3. Outbreak prevention and external hash list resources.
   a. FortiNDR inline inspection occurs simultaneously.

If any AV inspection method returns an infected verdict, the FortiNDR inspection is aborted.

POP3, IMAP, SMTP, NNTP, and MAPI protocols:

1. AV engine scan; AV database and FortiSandbox database (if applicable).
2. AV engine machine learning detection for WinPE PUPs (potentially unwanted programs).
   a. FortiNDR inline inspection occurs simultaneously.
3. Outbreak prevention and external hash list resources.
   a. FortiNDR inline inspection occurs simultaneously.

In an AV profile, use `set fortindr-error-action {log-only | block | ignore}` to configure the action to take if FortiNDR encounters an error.

Accepted file types

The following file types are sent to FortiNDR for inline inspection:

7Z	HTML	RTF
ARJ	JS	TAR
BZIP	LZH	VBA
BZIP2	LZW	VBS
CAB		WinPE (EXE)

ELF	MS Office documents (XML and non-XML)	XZ
GZIP	PDF	ZIP
	RAR	

Web filter

Web filtering restricts or controls user access to web resources and can be applied to firewall policies using either policy-based or profile-based NGFW mode.

In FortiOS, there are three main components of web filtering:

- Web content filter: blocks web pages containing words or patterns that you specify.
- URL filter: uses URLs and URL patterns to block or exempt web pages from specific sources, or block malicious URLs discovered by FortiSandbox.
- FortiGuard Web Filtering service: provides many additional categories you can use to filter web traffic.

These components interact with each other to provide maximum control over what users on your network can view and protect your network from many internet content threats.

Web filters are applied in the following order:

1. URL filter
2. FortiGuard Web Filtering
3. Web content filter
4. Web script filter
5. Antivirus scanning

FortiOS includes three preloaded web filter profiles:

- *default*
- *monitor-all* (monitors and logs all URLs visited, flow-based)
- *wifi-default* (default configuration for offloading WiFi traffic)

You can customize these profiles, or you can create your own to manage network user access.

Some features of this functionality require a subscription to FortiGuard Web Filtering.

The following topics provide information about web filters:

- URL filter on page 1123
- FortiGuard filter on page 1128
- Credential phishing prevention on page 1134
- Additional antiphishing settings on page 1137
- Usage quota on page 1140
- Web content filter on page 1142
- Advanced filters 1 on page 1145

URL filter

The URL filter uses specific URLs with patterns containing text and regular expressions so the FortiGate can process the traffic based on the filter action (exempt, block, allow, monitor) and web pages that match the criteria. Once a URL filter is configured, it can be applied to a firewall policy.

The following filter types are available:

URL filter type	Description
Simple	The FortiGate tries to strictly match the full context. For example, if you enter *www.facebook.com* in the *URL* field, it only matches traffic with *www.facebook.com*. It won't match *facebook.com* or *message.facebook.com*. When the FortiGate finds a match, it performs the selected URL action.
Regular expression/ wildcard	The FortiGate tries to match the pattern based on the rules of regular expressions or wildcards. For example, if you enter **fa** in the *URL* field, it matches all the content that has *fa* such as *www.facebook.com*, *message.facebook.com*, *fast.com*, and so on. When the FortiGate finds a match, it performs the selected URL action.

For more information, see the URL Filter expressions technical note in the Knowledge Base.

The following actions are available:

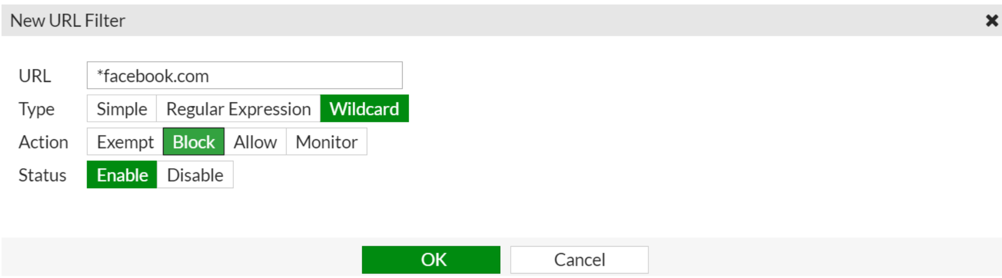
URL filter action	Description
Exempt	The traffic is allowed to bypass the remaining FortiGuard web filters, web content filters, web script filters, antivirus scanning, and DLP proxy operations.
Block	The FortiGate denies or blocks attempts to access any URL that matches the URL pattern. A replacement message is displayed.
Allow	The traffic is passed to the remaining FortiGuard web filters, web content filters, web script filters, antivirus proxy operations, and DLP proxy operations. If the URL does not appear in the URL list, the traffic is permitted.
Monitor	The traffic is processed the same way as the *Allow* action. For the *Monitor* action, a log message is generated each time a matching traffic pattern is established.

In the following example, a URL filter will be created to block the facebook.com URL using a wildcard.

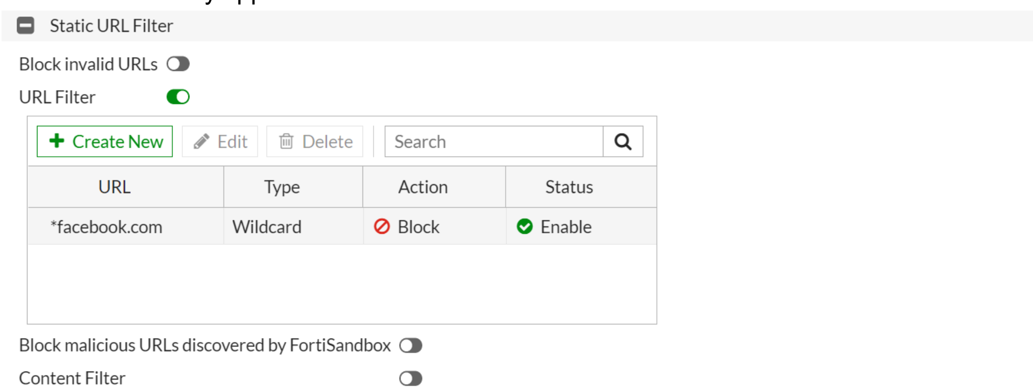
Configuring a URL filter in the GUI

To create a URL filter for Facebook:

1. Go to *Security Profiles* > *Web Filter* and click *Create New*, or edit an existing profile.
2. In the *Static URL Filter* section, enable *URL Filter*.
3. Click *Create New*. The *New URL Filter* pane opens.
4. For URL, enter **facebook.com*, for *Type*, select *Wildcard*, and for *Action*, select *Block*.



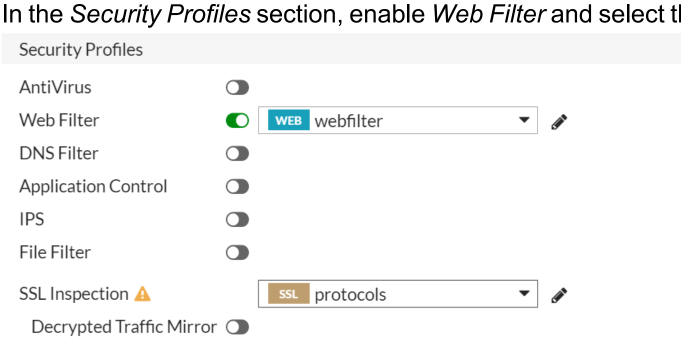
5. Click *OK*. The entry appears in the table.



6. Configure the other settings as needed.
7. Click *OK*.

To apply the web filter profile to a firewall policy:

1. Go to *Policy & Objects* > *Firewall Policy*.
2. Edit a policy, or create a new one.
3. In the *Security Profiles* section, enable *Web Filter* and select the profile you created.



4. Configure the other settings as needed.
5. Click *OK*.

Configuring a URL filter in the CLI

To create a URL filter for Facebook:

```
config webfilter urlfilter
    edit 1
        set name "webfilter"
        config entries
            edit 1
                set url "*facebook.com"
                set type wildcard
                set action block
            next
        end
    next
end
```

To apply the URL filter to a web filter profile:

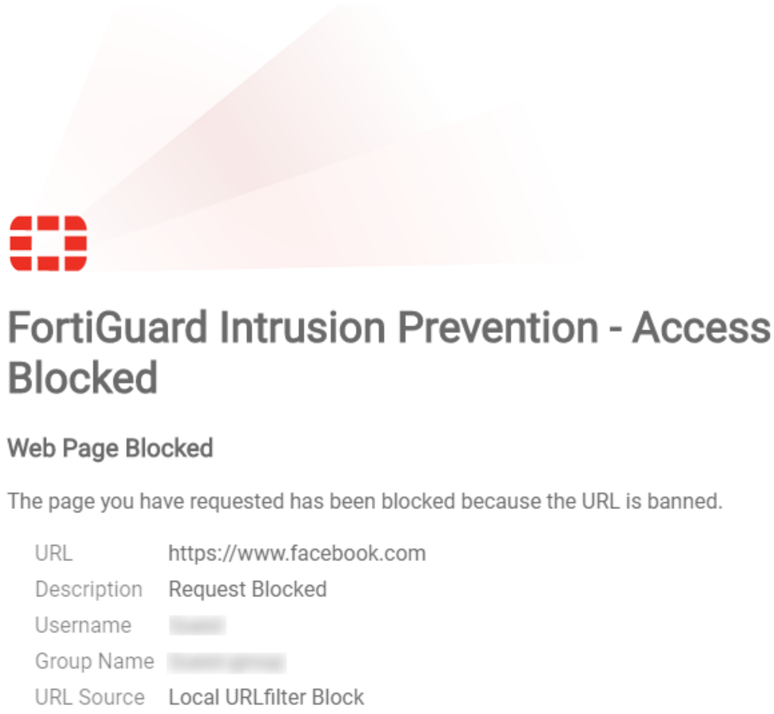
```
config webfilter profile
    edit "webfilter"
        config web
            set urlfilter-table 1
        end
        config ftgd-wf
            ...
        end
    next
end
```

To apply the web filter profile to a firewall policy:

```
config firewall policy
    edit 1
        set name "WF"
        set srcintf "wan2"
        set dstintf "wan1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set inspection-mode proxy
        set logtraffic all
        set webfilter-profile "webfilter"
        set profile-protocol-options "protocol"
        set ssl-ssh-profile "protocols"
        set nat enable
    next
end
```

Verifying the URL filter results

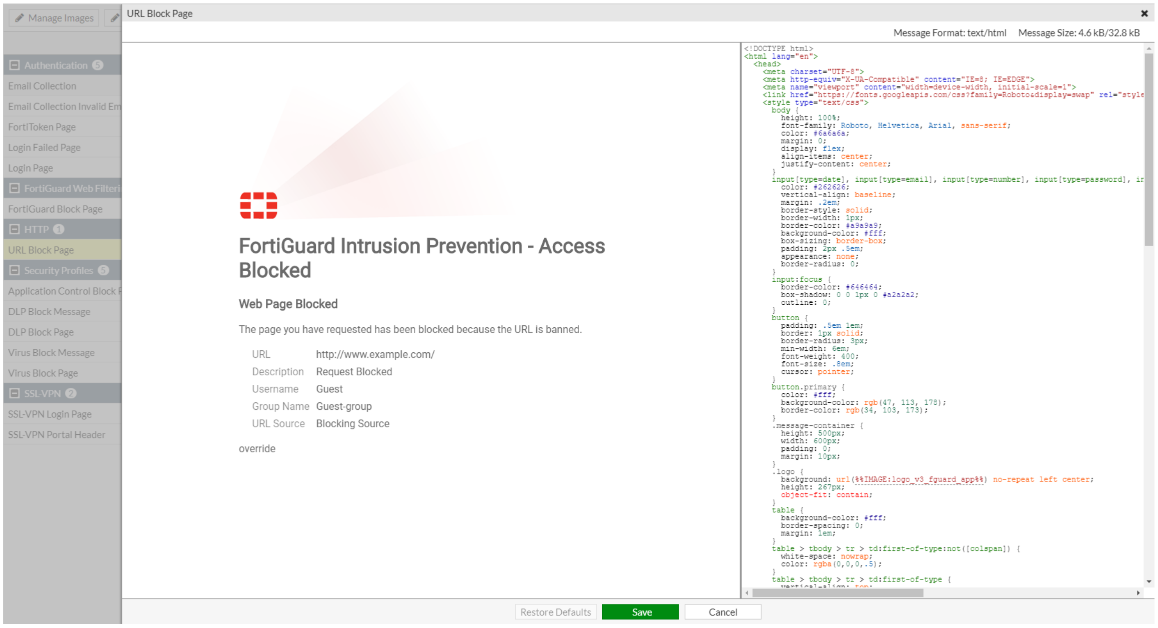
Verify the URL filter results by going to a blocked website. For example, when you go to the Facebook website, the replacement message appears:



To customize the URL web page blocked message:

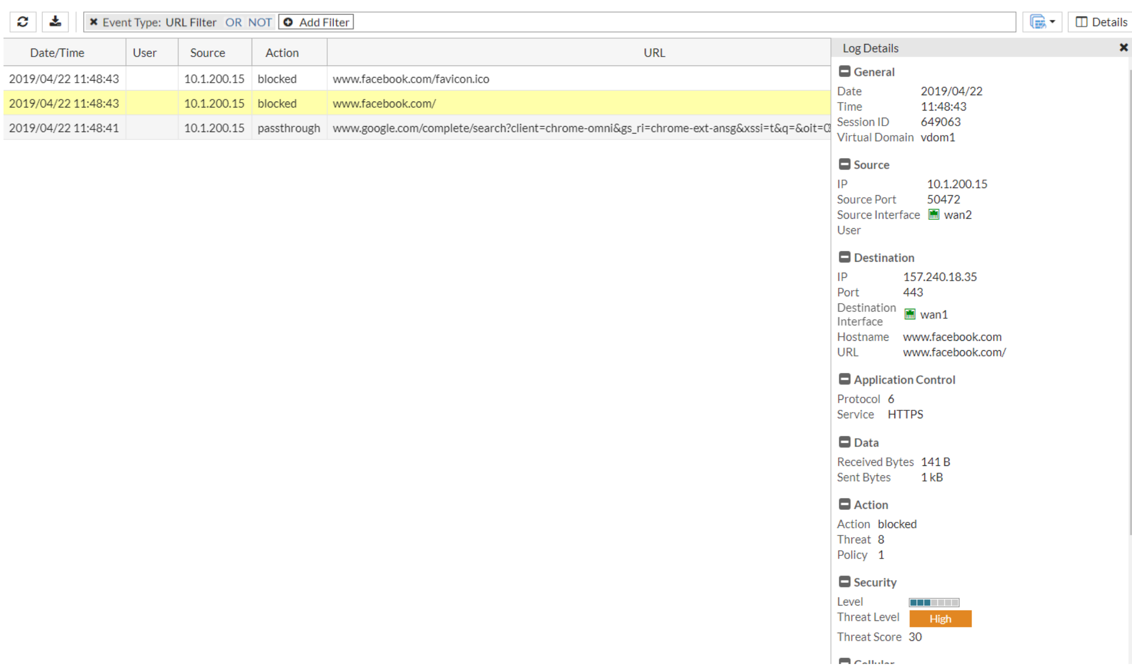
1. Go to *System* > *Replacement Messages*.
2. In the *HTTP* section, select *URL Block Page* and click *Edit*.

3. Edit the HTML to customize the message.



To check web filter logs in the GUI:

1. Go to *Log & Report > Security Events*.
2. Click the *Web Filter* card name.
3. If there are a lot of log entries, click *Add Filter* and select *Event Type > urlfilter* to display logs generated by the URL filter.



To check web filter logs in the CLI:

```
# execute log filter category utm-webfilter
# execute log display

1: date=2019-04-22 time=11:48:43 logid="0315012544" type="utm" subtype="webfilter"
eventtype="urlfilter" level="warning" vd="vdom1" eventtime=1555958923322174610
urlfilteridx=0 urlsource="Local URLfilter Block" policyid=1 sessionid=649063
srcip=10.1.200.15 srcport=50472 srcintf="wan2" srcintfrole="wan" dstip=157.240.18.35
dstport=443 dstintf="wan1" dstintfrole="wan" proto=6 service="HTTPS"
hostname="www.facebook.com" profile="webfilter" action="blocked" reqtype="direct" url="/"
sentbyte=1171 rcvdbyte=141 direction="outgoing" msg="URL was blocked because it is in the
URL filter list" crscore=30 craction=8 crlevel="high"
```

FortiGuard filter

The FortiGuard filter enhances the web filter features by sorting billions of web pages into a wide range of categories that users can allow or block.

The FortiGuard Web Filtering service includes over 45 million individual website ratings that apply to more than two billion pages. When the FortiGuard filter is enabled in a web filter profile and applied to firewall policies, if a request for a web page appears in traffic controlled by one of the firewall policies, the URL is sent to the nearest FortiGuard server. The URL category or rating is returned. If the category is blocked, the FortiGate shows a replacement message in place of the requested page. If the category is not blocked, the page request is sent to the requested URL as normal.

To use this service, you must have a valid FortiGuard license.

The following actions are available:

FortiGuard web filter action	Description
Allow	Permit access to the sites in the category.
Monitor	Permit and log access to sites in the category. User quotas can be enabled for this option (see Usage quota on page 1140).
Block	Prevent access to the sites in the category. Users trying to access a blocked site see a replacement message indicating the site is blocked.
Warning	Display a message to the user allowing them to continue if they choose.
Authenticate	Require the user to authenticate with the FortiGate before allowing access to the category or category group.
Disable	Remove the category from the from the web filter profile. This option is only available for local or remote categories from the right-click menu.

FortiGuard web filter categories

FortiGuard has many web filter categories, including two local categories and a special remote category. Refer to the following table for more information:

FortiGuard web filter category	Where to find more information
All URL categories	See Web Filter Categories.
Local categories	See Web rating override on page 1318.
Remote category	See Threat feeds on page 2504.

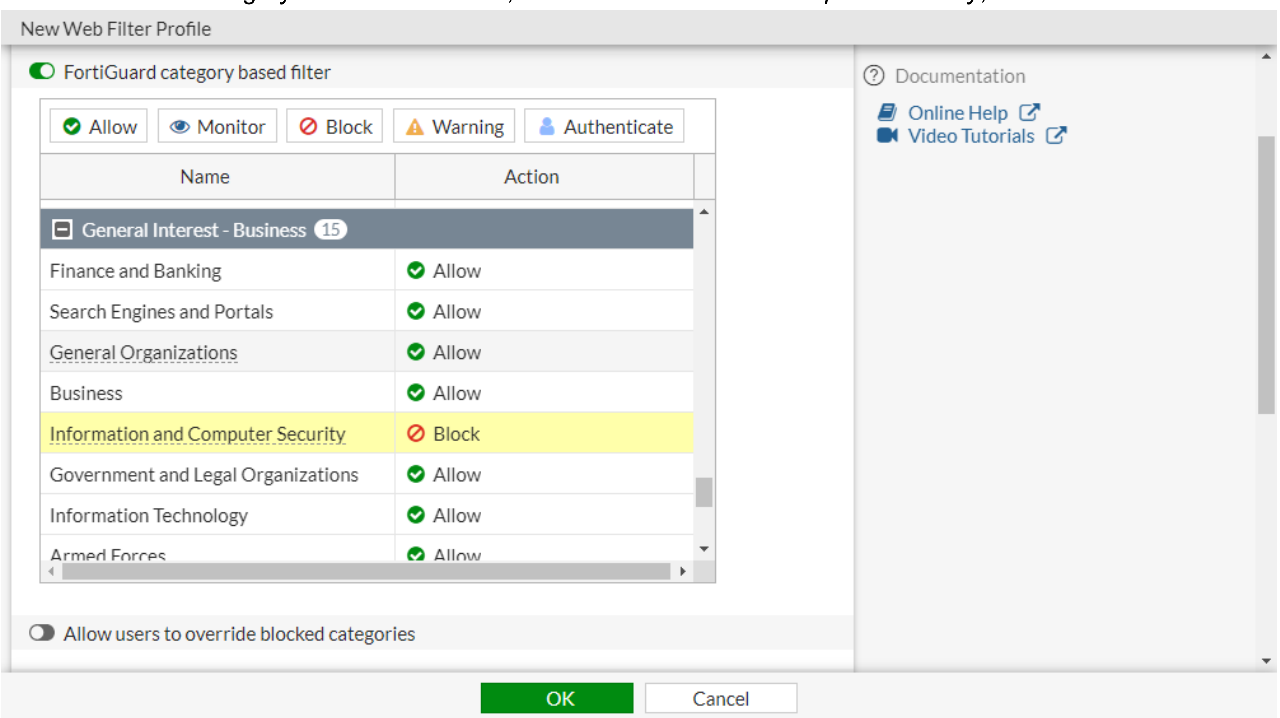
The priority of categories is local category > external category > FortiGuard built-in category. If a URL is configured as a local category, it only follows the behavior of the local category and not the external or FortiGuard built-in category.

Blocking a web category

The following example shows how to block a website based on its category. The information and computer security category (category 52) will be blocked.

To block a category in the GUI:

1. Go to *Security Profiles > Web Filter* and click *Create New*, or edit an existing profile.
2. In the *FortiGuard category based filter* section, select *Information and Computer Security*, then click *Block*.



3. Configure the remaining settings as needed.
4. Click *OK*.

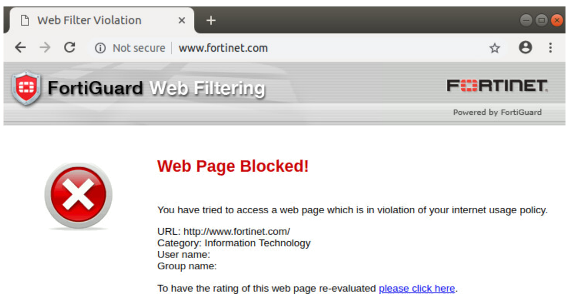
To block a category in the CLI:

```
config webfilter profile
    edit "webfilter"
        config ftgd-wf
            unset options
```

```
            config filters
                edit 1
                    set category 52
                    set action block
                next
            end
        end
    next
end
```

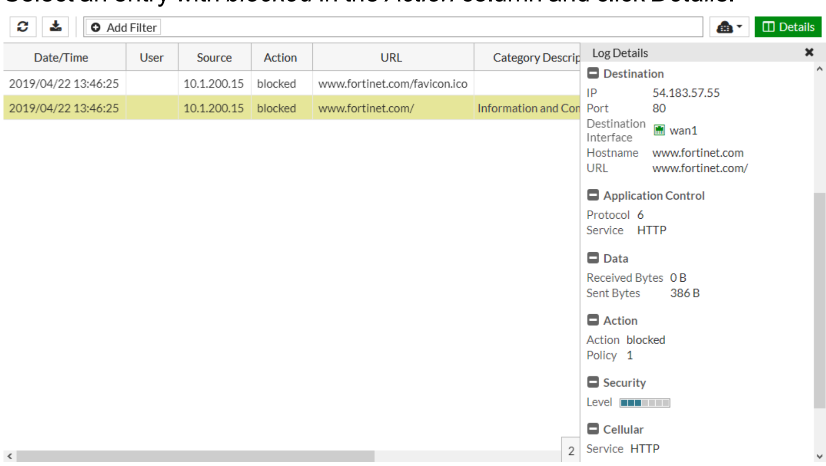
To verify that the category is blocked:

1. Go to a website that belongs to the blocked category, such as www.fortinet.com.
   The page should be blocked and display a replacement message.



To view the log of a blocked website in the GUI:

1. Go to *Log & Report > Security Events*.
2. Click the *Web Filter* card name.
3. Select an entry with *blocked* in the *Action* column and click *Details*.



4.

To view the log of a blocked website in the CLI:

```
# execute log filter category utm-webfilter
# execute log display

1: date=2019-04-22 time=13:46:25 logid="0316013056" type="utm" subtype="webfilter"
eventtype="ftgd_blk" level="warning" vd="vdom1" eventtime=1555965984972459609 policyid=1
sessionid=659263 srcip=10.1.200.15 srcport=49234 srcintf="wan2" srcintfrole="wan"
```

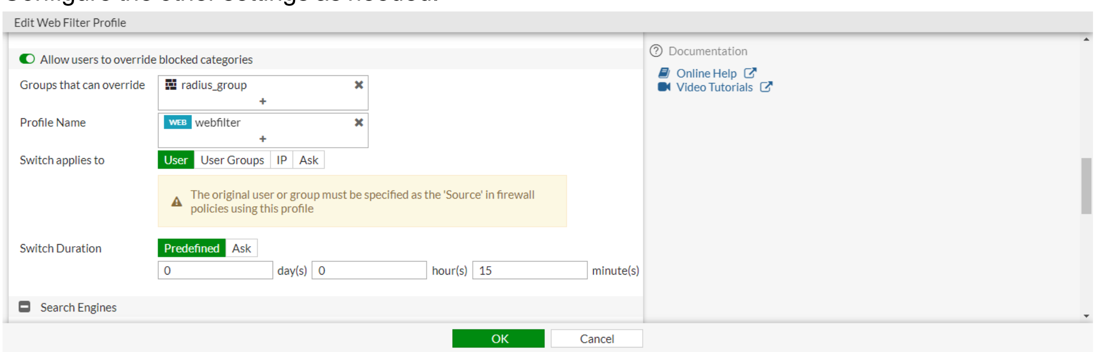
```
dstip=54.183.57.55 dstport=80 dstintf="wan1" dstintfrole="wan" proto=6 service="HTTP"
hostname="www.fortinet.com" profile="webfilter" action="blocked" reqtype="direct" url="/"
sentbyte=386 rcvdbyte=0 direction="outgoing" msg="URL belongs to a denied category in
policy" method="domain" cat=52 catdesc="Information Technology"
```

Allowing users to override blocked categories

There is an option to allow users with valid credentials to override blocked categories.

To allow users to override blocked categories in the GUI:

1. Go to *Security Profiles > Web Filter* and click *Create New*, or edit an existing profile.
2. Enable *Allow users to override blocked categories*.
3. Enter information in the following fields:
   - *Groups that can override*
   - *Profile name*
   - *Switch applies to*
   - *Switch Duration*
4. Configure the other settings as needed.



5. Click *OK*.

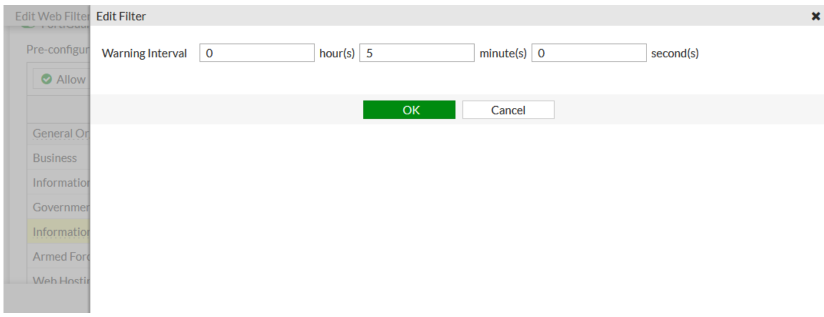
To allow users to override blocked categories in the CLI:

```
config webfilter profile
    edit "webfilter"
        set ovrd-perm bannedword-override urlfilter-override fortiguard-wf-override
contenttype-check-override
        config override
            set ovrd-user-group "radius_group"
            set profile "webfilter"
        end
        config ftgd-wf
            unset options
        end
    next
end
```

Issuing a warning on a web category

The following example shows how to issue a warning when a user visits a website in a specific category (information and computer security, category 52).

To configure a warning for a category in the GUI:

1. Go to *Security Profiles* > *Web Filter* and click *Create New*, or edit an existing profile.
2. In the *FortiGuard category based filter* section, select *Information and Computer Security*, then click *Warning*.
3. Set the *Warning Interval*, then click *OK*.
   The warning interval is the amount of time until the warning appears again after the user proceeds past it.
   

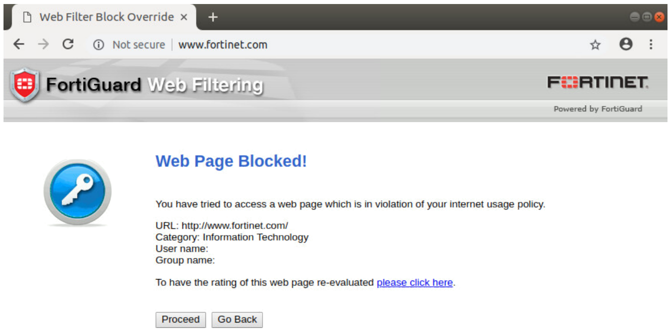
4. Configure the remaining settings as needed.
5. Click *OK*.

To configure a warning for a category in the CLI:

```
config webfilter profile
    edit "webfilter"
        config ftgd-wf
            unset options
            config filters
                edit 1
                    set category 52
                    set action warning
                next
            end
        end
    next
end
```

To verify that the warning works:

1. Go to a website that belongs to the category, such as www.fortinet.com.
2. On the warning page, click *Proceed* or *Go Back*.



Authenticating a web category

The following example shows how to authenticate a website based on its category (information and computer security, category 52).

To authenticate a category in the GUI:

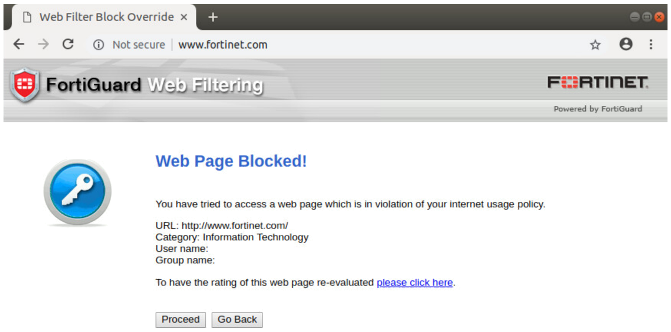
1. Go to *Security Profiles > Web Filter* and edit or create a new web filter profile.
2. In the *FortiGuard category based filter* section, select *Information and Computer Security*, then click *Authenticate*.
3. Set the *Warning Interval* and select one or more user groups, then click *OK*.
4. Configure the remaining settings as needed.
5. Click *OK*.

To authenticate a category in the CLI:

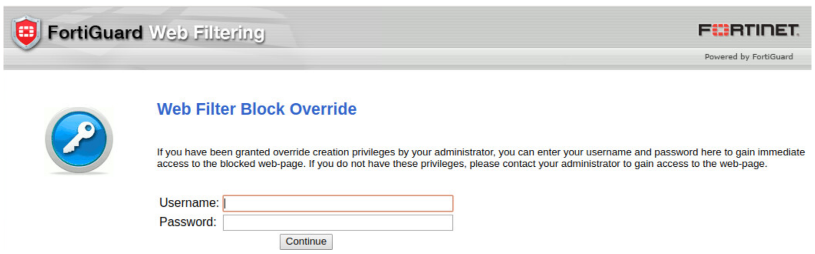
```
config webfilter profile
    edit "webfilter"
        config ftgd-wf
            unset options
            config filters
                edit 1
                    set category 52
                    set action authenticate
                    set auth-usr-grp "local_group"
                next
            end
        end
    next
end
```

To verify that you have configured authentication:

1. Go to a website that belongs to the category, such as www.fortinet.com.
2. On the warning page, click *Proceed*.



3. Enter the username and password for the configured user group, then click *Continue*.



Customizing the replacement message page

When the category action is *Block*, *Warning*, or *Authenticate*, you can customize the replacement message page that a user sees.

To customize the replacement message page:

1. Go to *Security Profiles > Web Filter* and edit or create a new web filter profile.
2. In the *FortiGuard category based filter* section, right-click on a category and select *Customize*.
3. Select a *Replacement Message Group*. See Replacement message groups on page 2137 for details.
4. Optionally, click *Edit FortiGuard Block Page* or *Edit FortiGuard Warning Page* to make modifications.
5. Click *Save*.
6. Configure the remaining settings as needed.
7. Click *OK*.

Credential phishing prevention

When credential phishing prevention is enabled, the FortiGate scans for corporate credentials submitted to external websites and compares them to sensitive credentials stored in the corporate domain controller. Based on the configured antiphishing rules in proxy mode web filter profiles, the FortiGate will block the URL or alert the user if the credentials match ones that are stored on the corporate domain controller.

- The corporate domain controller must be configured in the `domain controller`.
- Credentials can be matched based on sAMAccountName, user principal name (UPN), or down-level logon name.

- The antiphishing profile defines the corporate domain controller, antiphishing check option, default action if no rules match, antiphishing status, and so on.
- Inspection entries in the profile define what action occurs when the submission request matches the specified FortiGuard categories.
- The profile scans for pre-defined and custom username and password fields in the HTTP request, such as `username`, `auth`, and `password`. You can evaluate custom fields by configuring custom patterns.
- The URL filter defines individual URLs that the antiphish action (block or log) is applied to when the URL submission request matches.

Web-based URL filter actions and FortiGuard category-based filtering have higher priority than antiphishing URL filter actions and FortiGuard filtering:

- If a request is blocked by the web-based URL filter or FortiGuard filter, there is no further antiphishing scanning. Antiphishing scanning only happens after the web-based URL filtes and FortiGuard filters allow the traffic.
- If a submission matches an entry in the URL filter table that has an antiphishing action, the defined action is taken. No further FortiGuard category-based rules are applied.
- Like firewall rules, the URL filter table and Fortiguard category-based antiphishing rules use a top-down priority. The rule that matches first is the one that is used.

In this example, URLs that match FortiGuard category 37 (social networking) will be blocked and other categories will be logged.

To configure credential phishing prevention:

1. Configure the corporate domain controller:

```
config user domain-controller
    edit "win2016"
        set hostname "win2016"
        set domain-name "corpserver.local"
        set username "Administrator"
        set password **********
        set ip <server_ip>
    next
end
```

The `hostname` and the `domain-name` are case sensitive.

---

2. Configure the antiphishing profile, which includes the FortiGuard category rule:

```
config webfilter profile
    edit <profile-name>
        set feature-set proxy
        ...
        config web
            ...
        end
        config antiphish
            set status enable
            set domain-controller "win2016"
```

```
            set default-action block
            set check-uri enable
            set check-basic-auth enable
            set max-body-len 65536
            config inspection-entries
                edit "inspect-37"
                    set fortiguard-category 37
                    set action block
                next
                edit "inspect-others"
                    set fortiguard-category all
                    set action log
                next
            end
            config custom-patterns
                edit "customer-name"
                    set category username
                next
                edit "customer-passwd"
                   set category password
                next
            end
        end
        ...
        set web-antiphishing-log enable
    next
end
```

- `check-uri` enables support for scanning HTTP GET URI parameters.
- `check-basic-auth` enables support for scanning the HTTP basic authentication field.

3. Configure the URL filter to scan specific URLs.
The antiphish action is added to the URL filter table entry, and the URL filter is applied to the web filter profile:

```
config webfilter urlfilter
    edit 1
        set name "antiphish-table"
        config entries
            edit 1
                set url "www.example.com"
                set type simple
                set antiphish-action block
                set status enable
                set referrer-host ''
            next
        end
    next
end
config webfilter profile
    edit "<profile-name>"
        config web
            set urlfilter-table 1
        end
        ...
    next
end
```

4. Optionally, define custom patterns to scan fields other than the built-in username and password keywords:

```
config webfilter profile
    edit "<profile-name>"
        config custom-patterns
                edit "customer-name"
                    set category username
                next
                edit "customer-passwd"
                   set category password
                next
            end
        end
    next
end
```

Additional antiphishing settings

The following settings are available for antiphishing:

- Enable DNS service lookup in the domain controller so that the domain controller IP does not need to be configured. The DNS server will resolve the domain controller IP.
- Specify a source IP or port for the fetching domain controller.
- Use an LDAP server as a credential source (only the OpenLDAP server is supported).
- Block or log valid usernames regardless of password match.
- Use literal custom patterns type for username and password.
- Active Directory Lightweight Directory Services (AD LDS) support

Configuration examples

To enable DNS service lookup:

```
config user domain-controller
    edit "win2016"
        set ad-mode ds
        set dns-srv-lookup enable
        set hostname "win2016"
        set username "replicate"
        set password **********
        set domain-name "SMB2016.LAB"
    next
end
```

To specify the source IP and port for the fetching domain controller:

```
config user domain-controller
    edit "win2016"
        set ad-mode ds
        set hostname "win2016"
        set username "replicate"
        set password **********
        set ip-address 172.18.52.188
        set source-ip-address 172.16.100.1
```

```
        set source-port 2000
        set domain-name "SMB2016.LAB"

    next
end
```

To use an LDAP server as a credential store:

1. Configure the LDAP server:

```
config user ldap
    edit "openldap"
        set server "172.18.60.214"
        set cnid "cn"
        set dn "dc=qafsso,dc=com"
        set type regular
        set username "cn=Manager,dc=qafsso,dc=com"
        set password **********
        set antiphish enable
        set password-attr "userPassword"
    next
end
```

2. Configure the web filter profile:

```
config webfilter profile
    edit "webfilter"
        set feature-set proxy
        config ftgd-wf
            unset options
            config filters
                edit 1
                    set action block
                next
            end
        end
        config antiphish
            set status enable
            config inspection-entries
                edit "cat34"
                    set fortiguard-category 34
                    set action block
                next
            end
            set authentication ldap
            set ldap "openldap"
        end
        set log-all-url enable
    next
end
```

To configure username-only credential matching:

```
config webfilter profile
    edit "webfilter"
        set feature-set proxy
        config ftgd-wf
```

```
            unset options
            ...
        end
        config antiphish
            set status enable
            set check-username-only enable
            config inspection-entries
                edit "cat34"
                    set fortiguard-category 34
                    set action block
                next
            end
            set domain-controller "win2016"
        end
        set log-all-url enable
    next
end
```

To configure different custom pattern types for usernames and passwords:

```
config webfilter profile
    edit "webfilter"
        set feature-set proxy
        config ftgd-wf
            unset options
            ...
        end
        config antiphish
            set status enable
            config inspection-entries
                edit "cat34"
                    set fortiguard-category 34
                    set action block
                next
            end
            config custom-patterns
                edit "qwer"
                    set type literal
                next
                edit "[0-6]Dat*"
                next
                edit "dauw9"
                    set category password
                    set type literal
                next
                edit "[0-5]foo[1-4]"
                    set category password
                next
            end
            set domain-controller "win2016"
        end
        set log-all-url enable
    next
end
```

In this example, the `qwer` and `dauw9` entries use the literal type, while `[0-6]Dat*` and `[0-5]foo[1-4]` use the default regex type.

To configure Active Directory in LDS mode:

```
config user domain-controller
    edit "win2016adlds"
        set hostname "win2016adlds"
        set username "foo"
        set password **********
        set ip-address 192.168.10.9
        set domain-name "adlds.local"
        set ad-mode lds
        set adlds-dn "CN=adlds1part1,DC=ADLDS,DC=COM"
        set adlds-ip-address 192.168.10.9
        set adlds-port 3890
    next
end
```

Usage quota

In addition to using category and classification blocks and overrides to limit user access to URLs, you can set a daily quota by category, category group, or classification. Quotas allow access for a specified length of time or a specific bandwidth, and are calculated separately for each user. Quotas are reset daily at midnight.

Quotas can be set for the *Monitor*, *Warning*, or *Authenticate* actions. Once the quota is reached, the traffic is blocked and the replacement message page displays.

Quotas are only available in proxy-based inspection mode.

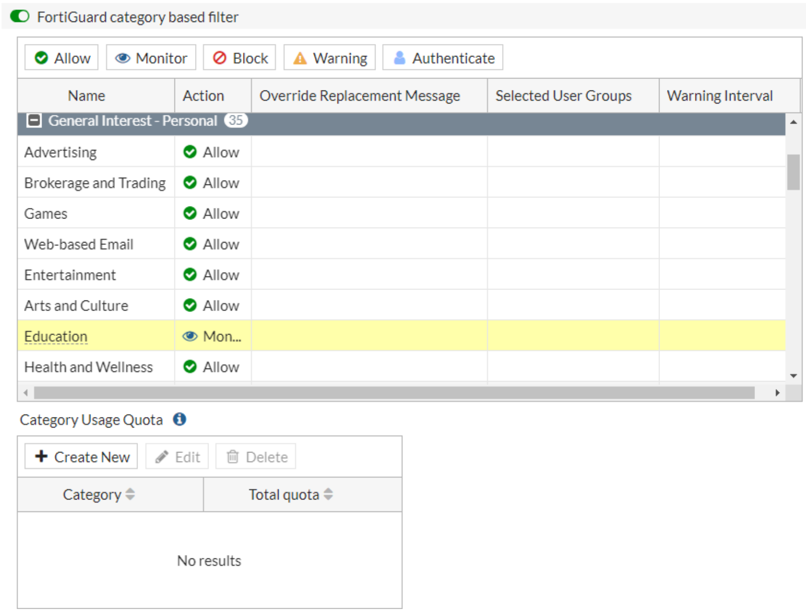
Configuring a quota

The following example shows how to set a time quota for the education category (category 30).

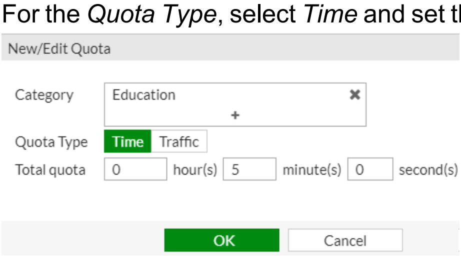
To configure a quota in the GUI:

1. Go to *Security Profiles* > *Web Filter* and click *Create New*, or edit an existing profile.
2. For *Feature set*, select *Proxy-based*.
3. In the *FortiGuard category based filter* section, scroll to the *General Interest - Personal* and click the + to expand the section.

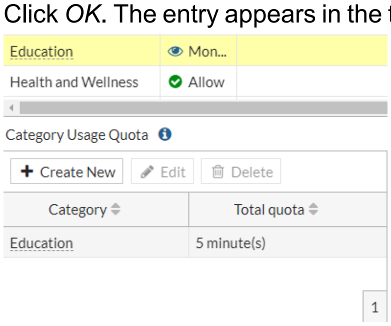
4. Select *Education*, then click *Monitor*.



5. In the *Category Usage Quota* section, click *Create New*.
   The *New/Edit Quota* pane opens.
6. In the *Category* field, select *Education*.
7. For the *Quota Type*, select *Time* and set the *Total quota* to 5 minutes.



8. Click *OK*. The entry appears in the table.



9. Configure the other settings as needed.
10. Click *OK*.

To configure a quota in the CLI:

```
config webfilter profile
    edit "webfilter"
        config ftgd-wf
           unset options
           config filters
```

```
                edit 1
                    set category 30
                next
            end
            config quota
                edit 1
                    set category 30
                    set type time
                    set duration 5m
                next
            end
        end
    next
end
```

To verify the quota usage:

1. Go to a website that belongs to the education category, such https://www.harvard.edu/. You can view websites in that category at the moment.
2. In FortiOS, go to *Dashboard* > *FortiGuard Quota Monitor* to check the used and remaining time.

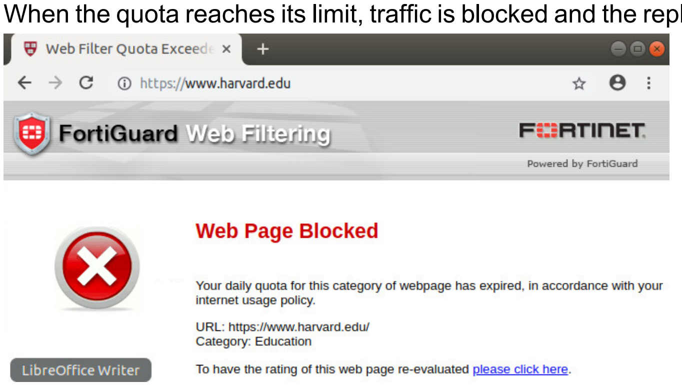
Category Usage Quota

User 10.1.100.11

Web Filter Profile webfilter

Category	Used Quota	Remaining
Education	1 second(s)	4 minute(s) and 59 second(s)

3. When the quota reaches its limit, traffic is blocked and the replacement page displays.



Web content filter

You can control access to web content by blocking webpages containing specific words or patterns. This helps to prevent access to pages with questionable material. You can specify words, phrases, patterns, wildcards, and regular expressions to match content on webpages. You can use multiple web content filter lists and select the best one for each web filter profile. The maximum number of web content patterns in a list is 5000.

When configuring a web content filter list, the following patterns are available:

Web content pattern type	Description
Wildcard	Use this setting to block or exempt one word or text strings of up to 80 characters. You can also use wildcard symbols such as *?* or *** to represent one or more characters. For example, a wildcard expression *forti*.com* matches *fortinet.com* and *fortiguard.com*. The *** represents any character appearing any number of times.
Regular expression	Use this setting to block or exempt patterns of regular expressions that use some of the same symbols as wildcard expressions, but for different purposes. In regular expressions, *** represents the character before the symbol. For example, *forti*.com* matches *fortiii.com* but not *fortinet.com* or *fortiice.com*. In this case, the symbol *** represents *i* appearing any number of times.

Content evaluation

The web content filter scans the content of every webpage that is accepted by a firewall policy. The system administrator can specify banned words and phrases and attach a numerical value (or score) to the importance of those words and phrases. When the web content filter scan detects banned content, it adds the scores of banned words and phrases found on that page. If the sum is higher than a threshold set in the web filter profile, the FortiGate blocks the page.

The default score for web content filter is 10 and the default threshold is 10. This means that by default, a webpage is blocked by a single match. These settings can only be configured in the CLI.

Banned words or phrases are evaluated according to the following rules:

- The score for each word or phrase is counted only once, even if that word or phrase appears many times in the webpage.
- The score for any word in a phrase without quotation marks is counted.
- The score for a phrase in quotation marks is counted only if it appears exactly as written.

The following table is an example of how rules are applied to the webpage contents . For example, a webpage contains only this sentence:

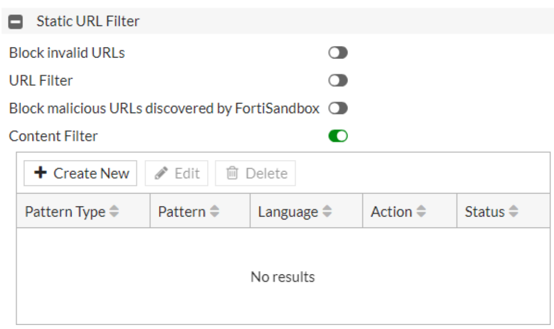
The score for each word or phrase is counted only once, even if that word or phrase appears many times in the webpage.

Banned pattern	Assigned score	Score added to the sum for the entire page	Threshold score	Comment
word	20	20	20	Appears twice but is only counted once. The webpage is blocked.
word phrase	20	40	20	Each word appears twice but is only counted once, giving a total score of 40. The webpage is blocked.
word sentence	20	20	20	*word* appears twice and *sentence* does not appear, but since any word in a phrase without quotation marks is counted, the score for this pattern is 20. The webpage is blocked.

Banned pattern	Assigned score	Score added to the sum for the entire page	Threshold score	Comment
"word sentence"	20	0	20	This phrase does not appear exactly as written. The webpage is allowed.
"word or phrase"	20	20	20	This phrase appears twice but is only counted once. The webpage is blocked.

To configure a web content filter in the GUI:

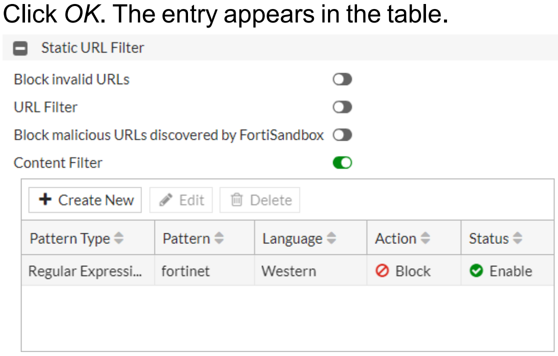
1. Go to *Security Profiles* > *Web Filter* and click *Create New*, or edit an existing profile.
2. In the *Static URL Filter* section, enable *Content Filter*.



3. Click *Create New*. The *New Web Content Filter* pane opens.
4. Configure the following settings:

Pattern Type	Regular Expression
Pattern	fortinet
Language	Western
Action	Block
Status	Enable

5. Click *OK*. The entry appears in the table.



6. Configure the other settings as needed.
7. Click *OK*.

To configure a web content filter in the CLI:

1. Create the content (banned word) table:

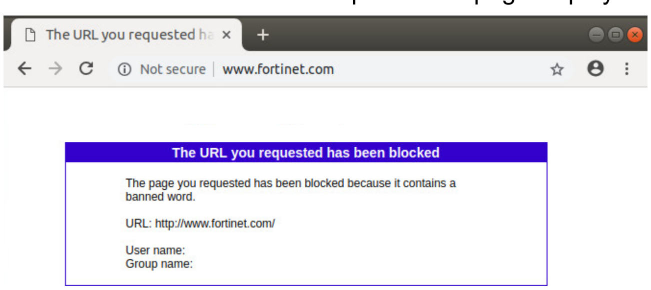
```
config webfilter content
   edit 1
      set name "webfilter"
      config entries
         edit "fortinet"
           set pattern-type regexp
           set status enable
           set lang western
           set score 10
           set action block
         next
      end
   next
end
```

2. Apply the content table to the web filter profile:

```
config webfilter profile
   edit "webfilter"
      config web
         set bword-threshold 10
         set bword-table 1
      end
      config ftgd-wf
         unset options
      end
   next
end
```

To verify the content filter:

1. Go to a website with the word *fortinet*, such as www.fortinet.com.
   The website is blocked and a replacement page displays:



Advanced filters 1

This topic gives examples of the following advanced filter features:

- Block malicious URLs discovered by FortiSandbox on page 1146
- Allow websites when a rating error occurs on page 1146
- Rate URLs by domain and IP address on page 1147
- Block invalid URLs on page 1147

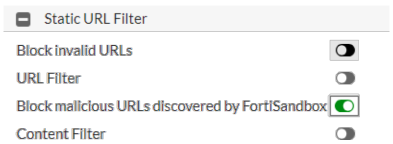
Block malicious URLs discovered by FortiSandbox

This setting blocks malicious URLs that FortiSandbox finds. Your FortiGate must be connected to a registered FortiSandbox.

For information on configuring FortiSandbox, see Using FortiSandbox post-transfer scanning with antivirus on page 1111 and Using FortiSandbox inline scanning with antivirus on page 1113.

To block malicious URLs discovered by FortiSandbox in the GUI:

1. Go to *Security Profiles > Web Filter* and click *Create New*, or edit an existing profile.
2. In the *Static URL Filter* section, enable *Block malicious URLs discovered by FortiSandbox*.



3. Click *OK*.

To block malicious URLs discovered by FortiSandbox in the CLI:

```
config webfilter profile
    edit "webfilter"
        config web
            set blocklist enable
        end
    next
end
```

Allow websites when a rating error occurs

If you do not have a FortiGuard license, but you have enabled services that need a FortiGuard license (such as FortiGuard filter), then you will get a rating error message.

Use this setting to allow access to websites that return a rating error from the FortiGuard Web Filter service.

To allow websites with rating errors in the GUI:

1. Go to *Security Profiles > Web Filter* and click *Create New*, or edit an existing profile.
2. In the *Rating Options* section, enable *Allow websites when a rating error occurs*.
3. Click *OK*.

To allow websites with rating errors in the CLI:

```
config webfilter profile
    edit "webfilter"
        config ftgd-wf
            set options error-allow
        end
    next
end
```

Rate URLs by domain and IP address

If you enable this setting, in addition to only sending domain information to FortiGuard for rating, the FortiGate always sends both the URL domain name and the TCP/IP packet's IP address (except for private IP addresses) to FortiGuard for the rating.

The FortiGuard server might return a different category of IP address and URL domain. If they are different, the FortiGate uses the rating weight of the IP address or domain name to determine the rating result and decision. This rating weight is hard-coded in FortiOS.

For example, if we use a spoof IP of Google as www.irs.gov, the FortiGate will send both the IP address and domain name to FortiGuard to get the rating. We get two different ratings: one is the search engine and portals that belong to the Google IP, the second is the government and legal organizations that belongs to www.irs.gov. Because the search engine and portals rating has a higher weight than government and legal organizations, the traffic is rated as search engine and portals.

To rate URLs by domain and IP address in the GUI:

1. Go to *Security Profiles* > *Web Filter* and click *Create New*, or edit an existing profile.
2. In the *Rating Options* section, enable *Rate URLs by domain and IP address*.
3. Click *OK*.

To rate URLs by domain and IP address in the CLI:

```
config webfilter profile
    edit "webfilter"
        config ftgd-wf
            set options rate-server-ip
        end
    next
end
```

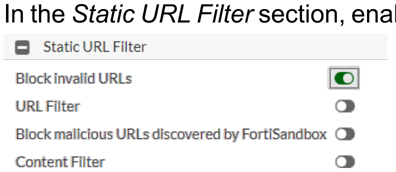
Block invalid URLs

Use this setting to block websites when their SSL certificate CN field does not contain a valid domain name.

This option also blocks URLs that contains spaces. If there is a space in the URL, it must be written as *%20* in the URL path.

To block invalid URLs in the GUI:

1. Go to *Security Profiles* > *Web Filter* and click *Create New*, or edit an existing profile.
2. In the *Static URL Filter* section, enable *Block invalid URLs* .



3. Click *OK*.

To block invalid URLs in the CLI:

```
config webfilter profile
    edit "webfilter"
        set options block-invalid-url
    next
end
```

Advanced filters 2

This topic gives examples of the following advanced filter features:

- Safe search on page 1148
- Restrict YouTube access on page 1149
- Log all search keywords on page 1150
- Restrict Google account usage to specific domains on page 1150
- HTTP POST action on page 1151
- Remove Java applets, ActiveX, and cookies on page 1151

These advanced filters are only available in proxy-based inspection mode.

Safe search

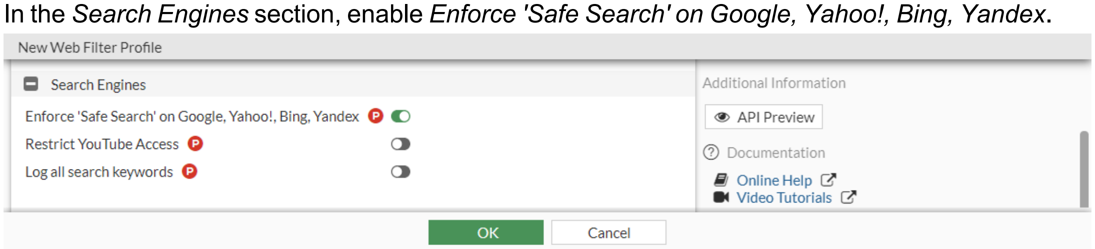
This setting applies to popular search sites and prevents explicit websites and images from appearing in search results.

The supported search sites are:

- Google
- Yahoo
- Bing
- Yandex

To enable safe search in the GUI:

1. Go to *Security Profiles > Web Filter* and click *Create New*, or edit an existing profile.
2. In the *Search Engines* section, enable *Enforce 'Safe Search' on Google, Yahoo!, Bing, Yandex*.



3. Click *OK*.

To enable safe search in the CLI:

```
config webfilter profile
    edit "webfilter"
        config web
            set safe-search url header
        end
    next
end
```

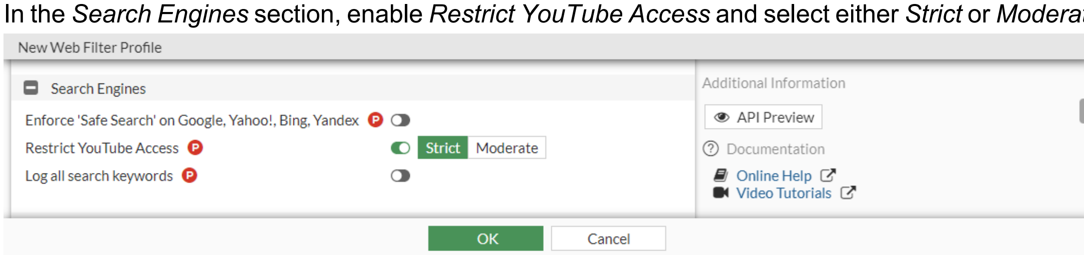
Restrict YouTube access

The *Restrict YouTube access* setting in the video filter profile adds the HTTP header `YouTube-Restrict: Strict` or `YouTube-Restrict: Moderate` into the HTTP request when enabled. When YouTube reads this header, it applies the appropriate content restriction based on the selected mode. YouTube Restricted Mode is an optional setting that filters out potentially mature videos while leaving a large number of videos still available (see Restrict YouTube content available to users and Manage your organization's YouTube settings for more information). Google defines the restricted YouTube access modes as follows:

- Strict Restricted YouTube access: this setting is the most restrictive. Strict Restricted Mode does not block all videos, but works as a filter to screen out many videos based on an automated system, while leaving some videos still available for viewing.
- Moderate Restricted YouTube access: this setting is similar to Strict Restricted Mode but makes a much larger collection of videos available.

To restrict YouTube access in the GUI:

1. Go to *Security Profiles > Web Filter* and click *Create New*, or edit an existing profile.
2. In the *Search Engines* section, enable *Restrict YouTube Access* and select either *Strict* or *Moderate*.



3. Click *OK*.

To restrict YouTube access in the CLI:

```
config webfilter profile
    edit <name>
        config web
            set set youtube-restrict {none | strict | moderate}
        end
    next
end
```

Vimeo access

The file filter profile includes a setting to restrict Vimeo access, which can only be configured in the CLI.

To restrict Vimeo access:

```
config webfilter profile
    edit <name>
        config web
            set vimeo-restrict {7 | 134}
        end
    next
end
```

`vimeo-restrict {7 | 134}`	Set the Vimeo restriction: • 7: do not show mature content • 134: do not show unrated and mature content

Log all search keywords

Use this setting to log all search phrases.

To enable logging search keywords in the GUI:

1. Go to *Security Profiles > Web Filter* and click *Create New*, or edit an existing profile.
2. In the *Search Engines* section, enable *Log all search keywords*.
3. Click *OK*.

To enable logging search keywords in the CLI:

```
config webfilter profile
    edit "webfilter"
        config web
            set log-search enable
        end
    next
end
```

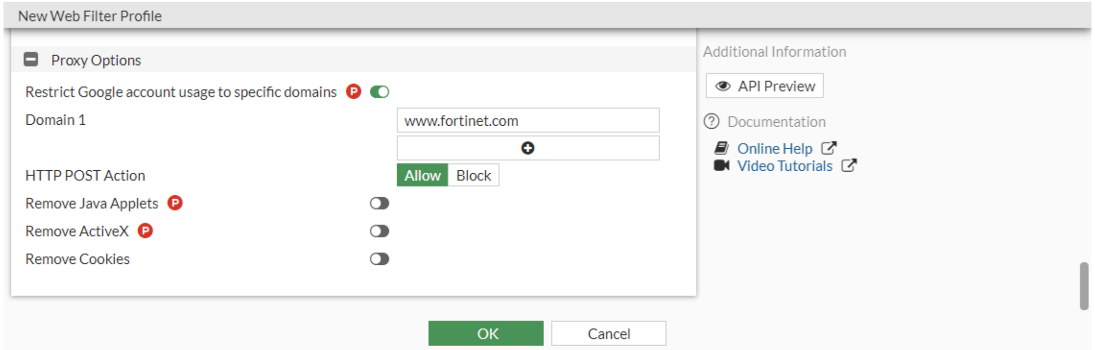
Restrict Google account usage to specific domains

Use this setting to block access to certain Google accounts and services, while allowing access to accounts with domains in the exception list.

To enable Google account restriction:

1. Go to *Security Profiles > Web Filter* and click *Create New*, or edit an existing profile.
2. In the *Proxy Options* section, enable *Restrict Google account usage to specific domains*.

3. Click the + and enter the domains that Google can access, such as www.fortinet.com.



4. Click *OK*.

When you try to use Google services like Gmail, only traffic from the domain of www.fortinet.com can go through. Traffic from other domains is blocked.

HTTP POST action

Use this setting to select the action to take with HTTP POST traffic. HTTP POST is the command used by the browser when you send information, such as a completed form or a file you are uploading to a web server. The action options are allow or block. The default is allow.

To configure HTTP POST in the GUI:

1. Go to *Security Profiles > Web Filter* and click *Create New*, or edit an existing profile.
2. In the *Proxy Options* section, for *HTTP POST Action*, select *Allow* or *Block*.
3. Click *OK*.

To configure HTTP POST in the CLI:

```
config webfilter profile
    edit "webfilter"
        set post-action {normal | block}
        config ftgd-wf
            unset options
        end
    next
end
```

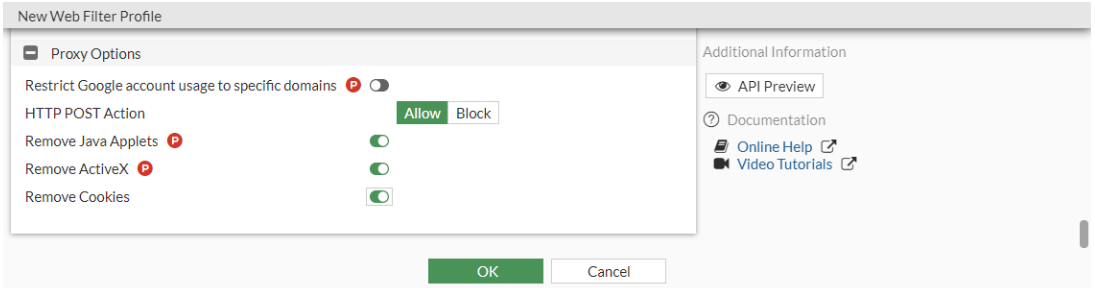
Remove Java applets, ActiveX, and cookies

Web filter profiles have settings to filter Java applets, ActiveX, and cookies from web traffic. Note that if these filters are enabled, websites using Java applets, ActiveX, and cookies might not function properly.

To enable these filters in the GUI:

1. Go to *Security Profiles > Web Filter* and click *Create New*, or edit an existing profile. and go to the *Proxy Options* section.

2. In the *Proxy Options* section, enabled the filters you want to use: *Remove Java Applets*, *Remove ActiveX*, or *Remove Cookies*.



To enable these filters in the CLI:

```
config webfilter profile
   edit "webfilter"
      set options {activexfilter cookiefilter javafilter}
      config ftgd-wf
         unset options
      end
   next
end
```

Web filter statistics

FortiOS provides diagnostics commands to view web filter statistics reports, which are either proxy-based or flow-based. The commands are available in both VDOM and global command lines.

Proxy-based web filter statistics report

Use the `diagnose wad filter vd {<VDOM> | global}` command to filter for per-VDOM or global statistics reports.

In the following example, there are two VDOMs (root and vdom1) using proxy-based policies that have web filter profiles enabled.

To view per-VDOM statistics reports:

```
(global) # diagnose wad filter vd root
Drop_unknown_session is enabled.

(global) # diagnose wad stats filter list
filtering of vdom root
  dlp          = 0
  content-type = 0
  urls:
    examined = 6
    allowed = 3
    blocked = 0
    logged = 0
    overridden = 0
```

```
(global) # diagnose wad filter vd vdom1
(global) # diagnose wad stats filter list
filtering of vdom vdom1
  dlp          = 0
  content-type = 0
  urls:
    examined = 13
    allowed = 2
    blocked = 9
    logged = 8
    overridden = 0
```

```
(global) # diagnose wad filter vd ALL
(global) # diagnose wad stats filter list
filtering of all accessible vdoms
  dlp          = 0
  content-type = 0
  urls:
    examined = 19
    allowed = 5
    blocked = 9
    logged = 8
    overridden = 0
```

Flow-based web filter statistics report

Use the `diagnose webfilter stats list {<VDOM> | global}` command to check the flow-based web filter statistics.

In the following example, the VDOM is using flow-based policies that have web filter profiles enabled.

To view web filter statistics:

```
# diagnose webfilter stats list root
Proxy/flow URL filter stats:
request:    9474
blocked:    8606
allowed:    868
overridden:0
logged:     8606
pending:    0
```

URL certificate blocklist

As increasing numbers of malware have started to use SSL to attempt to bypass IPS, maintaining a fingerprint-based certificate blocklist is useful to block botnet communication that relies on SSL.

This feature adds a dynamic package that is distributed by FortiGuard and is part of the Web Filtering service. It is enabled by default for SSL/SSH profiles, and can be configured using the following CLI commands:

```
config vdom
    edit <vdom>
        config firewall ssl-ssh-profile
            edit "certificate-inspection"
```

```
                set block-blocklisted-certificates enable
            next
            edit "deep-inspection"
                set block-blocklisted-certificates enable
            next
        end
    next
end
```

Websense Integrated Services Protocol

Websense Integrated Services Protocol (WISP) servers can be used server, which allows the FortiGate to send traffic to the third-party web filtering service for rating and approval checking.

When WISP is enabled, the FortiGate maintains a pool of TCP connections to the WISP server. The TCP connections are used to forward HTTP request information and log information to the WISP server and receive policy decisions.

When a WISP server is used in a web filter profile, in flow or proxy mode, the following web filter scanning priority sequence is used:

1. Local URL filter
2. Websense web filtering service
3. FortiGuard web filtering service

The following example uses a WISP server configured in a flow mode web filter profile.

To use a WISP server in flow mode:

1. Configure the WISP servers:

```
config web-proxy wisp
    edit "wisp1"
        set server-ip 10.2.3.4
    next
    edit "wisp2"
        set server-ip 10.2.3.5
    next
    edit "wisp3"
        set server-ip 192.168.1.2
    next
    edit "wisp4"
        set server-ip 192.168.3.4
    next
end
```

2. Configure the web filter profile:

```
config webfilter profile
    edit "webfilter_flowbase"
        set feature-set flow
        config ftgd-wf
            unset options
            config filters
                edit 64
                    set category 64
                    set action block
```

```
                    next
                end
            end
            set wisp enable
            set wisp-servers "wisp1" "wisp2"
            set wisp-algorithm {primary-secondary | round-robin | auto-learning}
            set log-all-url enable
        next
    end
```

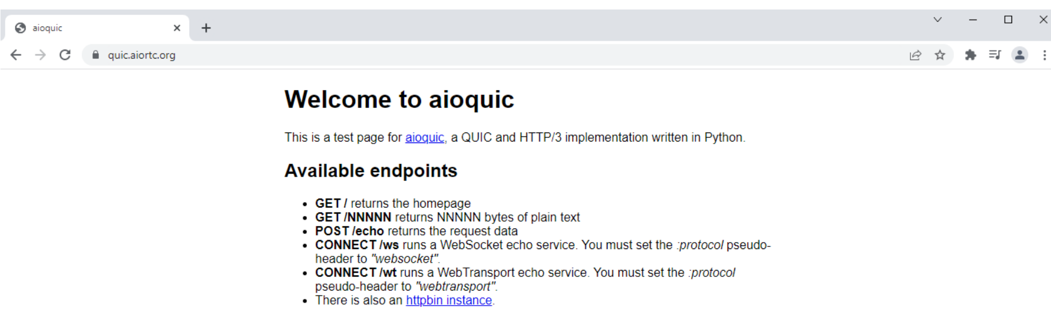
Inspecting HTTP3 traffic

HTTP/3 traffic can be inspected on the FortiGate in flow mode inspection.

When using Chrome, the browser may switch the HTTP/3 connection to HTTP/2 when deep inspection is applied, due to its sensitivity to delays caused by deep inspection.

Example

In this example, a web filter profile is created to block the words *Welcome to aioquic*, which appear in a website that uses HTTP/3.



To block content in HTTP/3 traffic:

1. Configure the web filter banned word table:

```
config webfilter content
    edit 1
        set name "aioquic"
        config entries
            edit "Welcome to aioquic"
                set status enable
            next
        end
    next
end
```

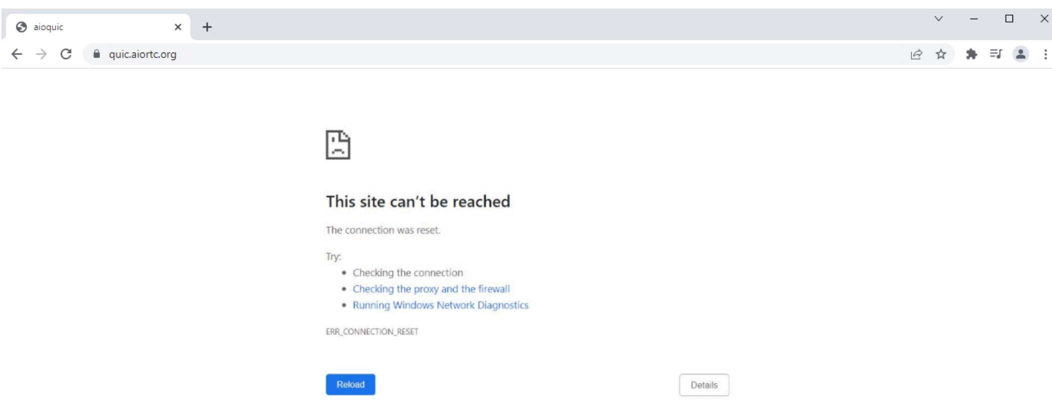
2. Apply the banned word table in the web filter profile:

```
config webfilter profile
    edit "flow-webfilter"
        config web
            set bword-table 1
        end
        config ftgd-wf
            unset options
        end
    next
end
```

3. Configure the firewall policy:

```
config firewall policy
    edit 1
        set utm-status enable
        set ssl-ssh-profile "deep-inspection"
        set webfilter-profile "flow-webfilter"
        set logtraffic all
        set nat enable
    next
end
```

4. Access the website using a supported HTTP/3 client, such as Chrome or Firefox. The website is blocked by the FortiGate.



Video filter

With the video filter profile, you can filter YouTube videos based on FortiGuard categories or by channel ID for a more granular override of a single channel, user, or video. The video filter profile is currently supported in proxy-based policies and requires SSL deep inspection. The FortiGuard Video filtering service is based on a valid FortiGuard web filter license.

The following topics provide information about video filters:

- Filtering based on FortiGuard categories on page 1157
- Filtering based on YouTube channel on page 1161

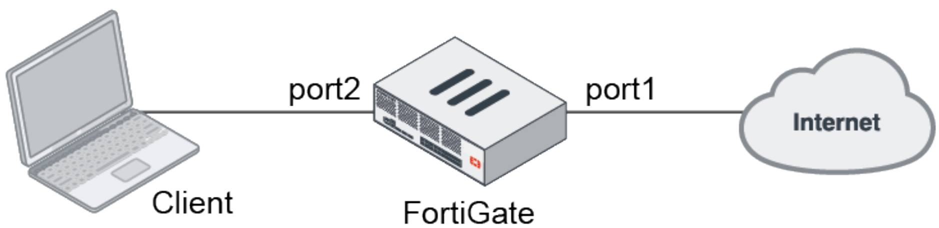
Filtering based on FortiGuard categories

Video filtering is only proxy-based and uses the WAD daemon to inspect the video in four phases:

1. When the WAD receives a video query from a client, it extracts the video ID (`vid`) and tries to check the category and channel from the local cache.
2. If there is no match from the local cache, it connects to the FortiGuard video rating server to query the video category.
3. If the FortiGuard rating fails, it uses the `videofilter.youtube-key` to communicate with the Google API server to get its category and channel ID. This is the API query setting and it requires the user's own YouTube API key string. This configuration is optional.
4. If all steps fail to match the video, the WAD calls on the IPS engine to match the video ID and channel ID from the application signature database.

The FortiGuard anycast service must be enabled to use this feature.

In the following example, a new video filter profile is created to block the Knowledge category.



In the firewall policy settings, the default application control profile is recommended because it blocks QUIC traffic. Many Google services use the QUIC protocol on UDP/443. By blocking QUIC, YouTube will use standard HTTPS TCP/443 connections.

To configure a video filter based on FortiGuard categories in the GUI:

1. Create the video filter profile:
   a. Go to *Security Profiles > Video Filter* and click *Create New*.
   b. Enter a name (*category_filter*).
   c. In the *FortiGuard Category Based Filter* section, set the *Knowledge* category *Action* to *Block*.
   d. Click *OK*.

2. Create the firewall policy:
   a. Enter the following:

Incoming Interface	port2
Outgoing Interface	port1
Source	All
Destination	All
Service	All
Inspection Mode	Proxy-based
NAT	Enable
Video Filter	Enable and select *category_filter*
Application Control	Enable and select *default*
SSL Inspection	deep-inspection
Log Allowed Traffic	All Sessions

   b. Configure the other settings as needed and click *OK*.

To configure a video filter based on FortiGuard categories in the CLI:

1. Create the video filter profile:

```
config videofilter profile
    edit "category_filter"
        config fortiguard-category
            edit 5
                set action block
                set category-id 4
                set log enable
            next
        end
    next
end
```

2. Create the firewall policy:

```
config firewall policy
    edit 10
        set name "client_yt_v4"
        set srcintf "port2"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set inspection-mode proxy
        set ssl-ssh-profile "deep-inspection"
        set application-list "default"
```

```
            set videofilter-profile "category_filter"
            set logtraffic all
            set nat enable
        next
    end
```

To configure the YouTube API key (optional):

```
config videofilter youtube-key
    edit 1
        set key ********
        set status enable
    next
end
```

Verifying that the video is blocked

When a user browses to YouTube and selects a video based in the Knowledge category, a replacement message will appear. This replacement message says the URL is blocked, and displays the URL of the YouTube video. On the FortiGate, verify the forward traffic and web filter logs.

Sample forward traffic log

```
2: date=2021-04-27 time=15:27:13 eventtime=1619562433424944288 tz="-0700" logid="0000000013"
type="traffic" subtype="forward" level="notice" vd="vdom1" srcip=10.1.100.11 srcport=60628
srcintf="port2" srcintfrole="undefined" dstip=172.217.3.206 dstport=443 dstintf="port1"
dstintfrole="undefined" srccountry="Reserved" dstcountry="United States" sessionid=8230
proto=6 action="client-rst" policyid=10 policytype="policy" poluuid="a5e991ba-a799-51eb-
4efe-ce32b9f70b75" policyname="client_yt_v4" service="HTTPS" trandisp="snat"
transip=172.16.200.1 transport=60628 duration=95 sentbyte=3546 rcvdbyte=21653 sentpkt=24
rcvdpkt=34 appcat="unscanned" wanin=2152 wanout=2290 lanin=2000 lanout=2000
utmaction="block" countweb=3 utmref=65532-0
```

Sample web filter log

```
1: date=2021-04-27 time=15:25:37 eventtime=1619562338128550236 tz="-0700" logid="0347013664"
type="utm" subtype="webfilter" eventtype="videofilter-category" level="warning" vd="vdom1"
msg="Video category is blocked." policyid=10 sessionid=8230 srcip=10.1.100.11
dstip=172.217.3.206 srcport=60628 dstport=443 srcintf="port2" srcintfrole="undefined"
dstintf="port1" dstintfrole="undefined" proto=6 service="HTTPS" action="blocked"
videoinfosource="Cache" profile="category_filter" videoid="EAyo3_zJj5c" videocategoryid=4
hostname="www.youtube.com" url="https://www.youtube.com/watch?v=EAyo3_zJj5c"
```

Troubleshooting and debugging

To verify if the FortiGuard video filtering license is valid:

```
# get system fortiguard

fortiguard-anycast  : enable
fortiguard-anycast-source: debug
protocol            : https
port                : 443
```

```
...
webfilter-license   : Contract
webfilter-expiration: Fri Dec 13 2030
...
videofilter-license : Contract
videofilter-expiration: Fri Dec 13 2030
```

The `videofilter` license should be synchronized with the `webfilter` license.

To verify the WAD worker is running:

```
# diagnose test app wad 1000
Process [0]: WAD manager type=manager(0) pid=232 diagnosis=yes.
Process [1]: type=worker(2) index=0 pid=294 state=running
              diagnosis=no debug=enable valgrind=supported/disabled
...
Process [6]: type=YouTube-filter-cache-service(9) index=0 pid=290 state=running
              diagnosis=no debug=enable valgrind=unsupported/disabled
...
```

To display and debug video filter cache:

```
# diagnose test app wad ?
....
        321:  Display Video Filter Cache stats.
        322:  Reset Video Filter Cache stats.
        323:  Flush Video Filter Cache entries.
        324:  Display Video Filter module stats.
        325:  Request category list from Youtube API.
        326:  Display FTGD agent module stats.
        327:  Reset FTGD agent module stats.
        328:  Toggle Video Filter Cache Check.
        329:  Toggle Video Filter FTGD Query.
        330:  Toggle Video Filter API Check.
```

To enable real-time WAD debugs:

```
# diagnose wad debug enable level verbose
# diagnose wad debug enable category video
# diagnose debug enable
```

Sample output

```
[p:274][s:8754][r:186] wad_http_req_exec_video_filter_check(167): hreq=0x7f1184f288e0, check
video filter check videofilter
[p:274][s:8754][r:186] wad_vf_req_submit(1869): node=0x7f1186694640, ctx=0x7f118502d1f8,
youtube_channel_filter_id=0
[p:274][s:8754][r:186] wad_vf_match_pattern_cb(1551): ctx=0x7f118502d1f8 matched type video
[p:274][s:8754][r:186] wad_vf_extract_video_id(297): str='v=EAyo3_zJj5c', start='v=',
end='&'
[p:274][s:8754][r:186] wad_vf_extract_video_id(297): str='v=EAyo3_zJj5c', start='v=', end=''
[p:274][s:8754][r:186] wad_vf_extract_video_id(322): video-id: start=2, end=13
[p:274][s:8754][r:186] wad_vf_sync_task_trigger_async_task(1602): extracted vid=EAyo3_zJj5c
ctx=0x7f118502d1f8
[p:274][s:8754][r:186] wad_vf_sync_task_trigger_async_task(1622): video filter
```

```
ctx=0x7f118502d1f8 creates new task=0x7f118657e7a0
[p:274][s:8754][r:186] wad_vfc_client_lookup(159): oid=15194313278609724406
[p:274][s:8754][r:186] wad_vfc_core_lookup(277): youtube-filter-cache core(0x7f11864d2078)
found the item!
[p:274][s:8754][r:186] wad_vfc_client_lookup(174): local lookup: ret=0 result=hit, hit_
cnt=51
local hit item, item's value:
  oid=15194313278609724406
  vid="EAyo3_zJj5c"
  category="4"
  title="Youtube Data API V3 Video Search Example"
  channel="UCR6d0EiC3G4WA8-Rqji6a8g"
  desc(first 100 characters)="Youtube Data API V3 Video Search Example

Welcome Folks My name is Gautam and Welcome to Coding Shik......"
[p:274][s:8754][r:186] wad_vf_task_proc_cache_resp(1048): vf filter cache hit,
item=0x7f116dacc060
[p:274][s:8754][r:186] wad_vf_async_task_run(1491): end of async task ret=0
[p:274][s:8754][r:186] wad_vf_sync_task_proc_async_result(1686): task=0x7f118657e7a0
item=0x7f116dacc060
[p:274][s:8754][r:186] wad_vf_sync_task_proc_async_result(1721): ctx(0x7f118502d1f8) channel
UCR6d0EiC3G4WA8-Rqji6a8g not match
[p:274][s:8754][r:186] wad_vf_sync_task_proc_async_result(1733): ctx(0x7f118502d1f8)
category result is block
[p:274][s:8754][r:186] wad_vfc_client_add(230): oid=15194313278609724406
```

Filtering based on YouTube channel

Video filtering can be configured to filter specific YouTube channels. When a video matches a YouTube channel, the video will take the corresponding action of allow, monitor, or block. Video filtering is only supported in proxy-based inspection mode, and deep inspection must be enabled in the firewall policy.

By default, when the FortiGuard category-based filter and YouTube channel override are used together, a video will be blocked if it matches either category or YouTube channel and the action is set to block.

Identifying the YouTube channel ID

The following table lists how to identify the YouTube channel ID based on different YouTube video URLs formats:

Video URL	Channel ID
www.youtube.com/channel/<channel-id>	<channel-id> indicates the ID for the channel.
www.youtube.com/user/<user-id>	Open the page source and locate: `<meta itemprop="channelId" content="<channel-id>">` <channel-id> indicates the channel ID for the user page.
www.youtube.com/watch?v=<string>	Open the page source and locate: `<meta itemprop="channelId" content="<channel-id>">` <channel-id> indicates the channel ID for the video.

In a YouTube channel filter profile, the default action is set to monitor when there is no match. Logging is also disabled by default.

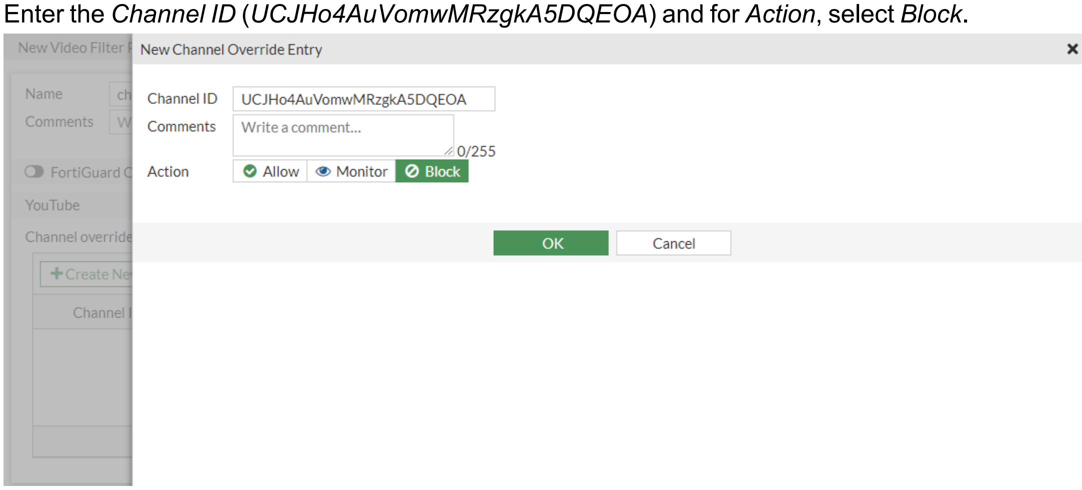
```
config videofilter youtube-channel-filter
    edit <id>
        set default-action {block | monitor | allow}
        set log {enable | disable}
    next
end
```

Example

In the following example, the Fortinet YouTube channel ID (UCJHo4AuVomwMRzgkA5DQEOA) is blocked, and the video filter is applied to a policy.

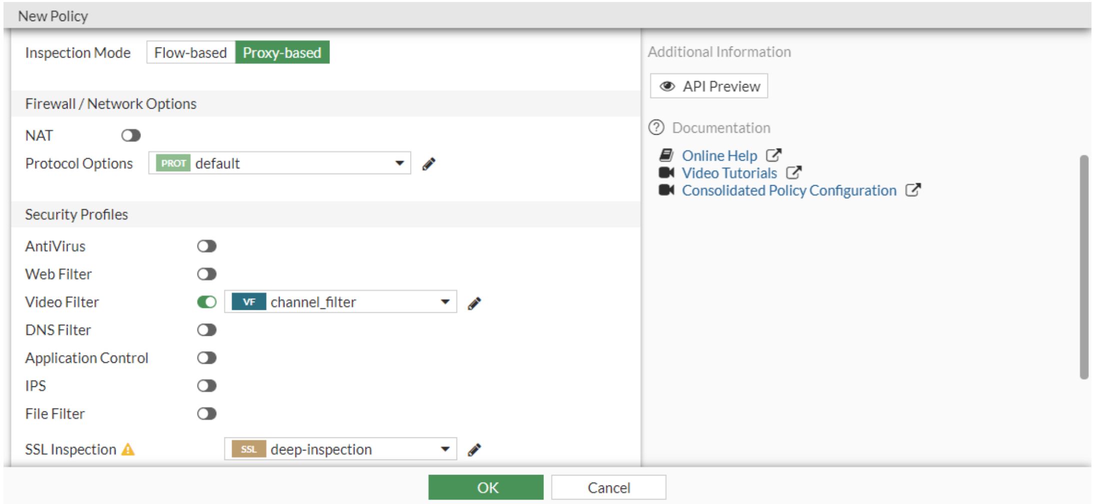
To configure a video filter based on a YouTube channel in the GUI:

1. Go to *Security Profiles > Video Filter* and click *Create New*.
2. In the *Channel override list* section, click *Create New*. The *New Channel Override Entry* pane opens.
   a. Enter the *Channel ID* (*UCJHo4AuVomwMRzgkA5DQEOA*) and for *Action*, select *Block*.



   b. Click *OK*.
3. Click *OK*.
4. Create the firewall policy:
   a. Go to *Policy & Objects > Firewall Policy* and click *Create New*.
   b. For *Inspection Mode*, select *Proxy-based*.
   c. Enable *Video Filter* and select the profile you created.

d. For *SSL Inspection*, select *deep-inspection*.



e. Configure the other settings as needed and click *OK*.

To configure a video filter based on a YouTube channel in the CLI:

1. Create the channel filter:

```
config videofilter youtube-channel-filter
    edit 1
        set name "channel_filter"
        set default-action monitor
        set log enable
        config entries
            edit 1
                set action block
                set channel-id "UCJHo4AuVomwMRzgkA5DQEOA"
            next
        end
    next
end
```

2. Create the video filter profile:

```
config videofilter profile
    edit "channel_filter"
        set youtube-channel-filter 1
    next
end
```

3. Create the firewall policy:

```
config firewall policy
    edit 1
        set name "video-filter"
        set srcintf "port2"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
```

```
        set service "ALL"
        set utm-status enable
        set inspection-mode proxy
        set ssl-ssh-profile "deep-inspection"
        set videofilter-profile "channel_filter"
        set nat disable
    next
end
```

DNS filter

You can apply DNS category filtering to control user access to web resources. You can customize the default profile, or create your own to manage network user access and apply it to a firewall policy, or you can add it to a DNS server on a FortiGate interface. For more information about configuring DNS, see DNS on page 202.

DNS filtering has the following features:

- FortiGuard Filtering: filters the DNS request based on the FortiGuard domain rating.
- Botnet C&C domain blocking: blocks the DNS request for the known botnet C&C domains.
- External dynamic category domain filtering: allows you to define your own domain category.
- DNS safe search: enforces Google, Bing, and YouTube safe addresses for parental controls.
- Local domain filter: allows you to define your own domain list to block or allow.
- External IP block list: allows you to define an IP block list to block resolved IPs that match this list.
- DNS translation: maps the resolved result to another IP that you define.

Some DNS filter features require a subscription to FortiGuard Web Filtering.

DNS filtering connects to the FortiGuard secure DNS server over anycast by default. For more information about this configuration, see DNS over TLS and HTTPS on page 214.

The IPS engine handles the DNS filter in flow mode policies and queries the FortiGuard web filter server for FortiGuard categories. In proxy mode, the DNS proxy daemon handles the DNS filter and queries the FortiGuard SDNS server for FortiGuard categories. When a DNS filter profile is enabled in `config system dns-server`, the DNS proxy daemon handles the traffic.

DNS filter profiles cannot be used in firewall policies when the FortiGate is in NGFW policy-based mode; see NGFW policy on page 772 for more information. They can be used in the DNS server; see FortiGate DNS server on page 206 for more information.

A DNS filter profile can be applied in a policy to scan DNS traffic traversing the FortiGate (see Configuring a DNS filter profile on page 1166), or applied on the DNS server interface (see Applying DNS filter to FortiGate DNS server on page 1183).

DNS filter behavior in proxy mode

In cases where the DNS proxy daemon handles the DNS filter (described in the preceding section) and if DNS caching is enabled (this is the default setting), then the FortiGate will respond to subsequent DNS queries using the result in the DNS cache and will not forward these queries to a real DNS server.

There are two options to disable this behavior:

- Disable DNS caching globally.
- Remove the DNS filter profile from the proxy mode firewall policy or from the DNS server configured on a FortiGate interface.

To disable DNS caching globally:

```
config system dns
    set dns-cache-limit 0
end
```

There will be a performance impact to DNS queries since each query will not be cached, and will be forwarded to a real DNS server.

FortiGuard DNS rating service

DNS over TLS connections to the FortiGuard secure DNS server is supported. The CLI options are only available when `fortiguard-anycast` is enabled. DNS filtering connects to the FortiGuard secure DNS server over anycast by default.

To configure DoT to the secure DNS server in the CLI:

```
config system fortiguard
    set fortiguard-anycast enable
    set fortiguard-anycast-source fortinet
    set anycast-sdns-server-ip 0.0.0.0
    set anycast-sdns-server-port 853
end
```

The following topics provide information about DNS filters:

- Configuring a DNS filter profile on page 1166
- FortiGuard category-based DNS domain filtering on page 1168
- Botnet C&C domain blocking on page 1171
- DNS safe search on page 1175
- Local domain filter on page 1177
- DNS translation on page 1180
- Applying DNS filter to FortiGate DNS server on page 1183
- DNS inspection with DoT and DoH on page 1184
- Troubleshooting for DNS filter on page 1188

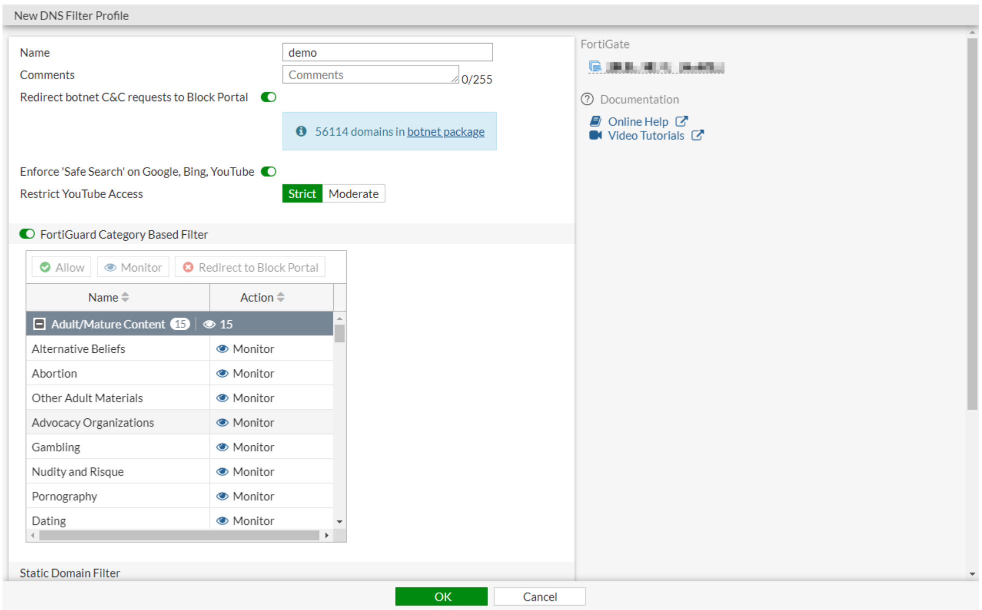
Configuring a DNS filter profile

Once a DNS filter is configured, it can be applied to a firewall policy. This example scans DNS traffic traversing the FortiGate.

When a FortiGate DNS server has been configured, refer to the steps in Applying DNS filter to FortiGate DNS server on page 1183.

To configure DNS Filter profile in the GUI:

1. Go to *Security Profiles > DNS Filter* and click *Create New*, or edit an existing profile.
2. Configure the settings as needed.



3. Click *OK*.

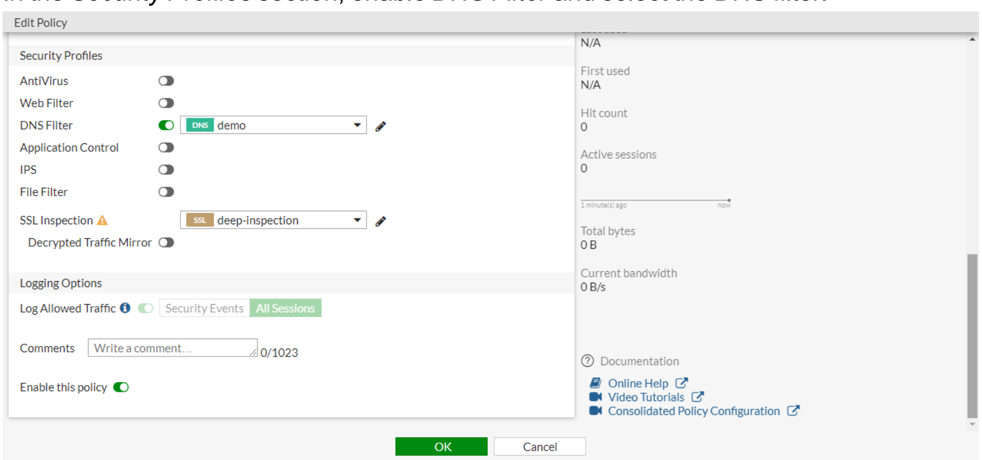
To create or configure DNS Filter profile in the CLI:

```
config dnsfilter profile
    edit "demo"
        set comment ''
        config domain-filter
            unset domain-filter-table
        end
        config ftgd-dns
            set options error-allow
            config filters
                edit 2
                    set category 2
                    set action monitor
                next
                edit 7
                    set category 7
                    set action block
                next
                ...
```

```
                edit 22
                    set category 0
                    set action monitor
                next
            end
        end
        set log-all-domain enable
        set sdns-ftgd-err-log enable
        set sdns-domain-log enable
        set block-action redirect
        set block-botnet enable
        set safe-search enable
        set redirect-portal 93.184.216.34
        set youtube-restrict strict
    next
end
```

To apply DNS Filter profile to the policy in the GUI:

1. Go to *Policy & Objects* > *Firewall Policy* and click *Create New*, or edit an existing policy.
2. In the *Security Profiles* section, enable *DNS Filter* and select the DNS filter.



3. Configure the other settings as needed.
4. Click *OK*.

To apply DNS Filter profile to the policy in the CLI:

```
config firewall policy
    edit 1
        set name "Demo"
        set srcintf "port10"
        set dstintf "port9"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set inspection-mode proxy
        set logtraffic all
        set fsso disable
```

```
        set dnsfilter-profile "demo"
        set profile-protocol-options "default"
        set ssl-ssh-profile "deep-inspection"
        set nat enable
    next
end
```

DNS filter behavior in proxy mode

In cases where the DNS proxy daemon handles the DNS filter (described in DNS filter on page 1164) and if DNS caching is enabled (this is the default setting), then the FortiGate will respond to subsequent DNS queries using the result in the DNS cache and will not forward these queries to a real DNS server.

There are two options to disable this behavior:

- Disable DNS caching globally.
- Remove the DNS filter profile from the proxy mode firewall policy or from the DNS server configured on a FortiGate interface.

To disable DNS caching globally:

```
config system dns
    set dns-cache-limit 0
end
```

There will be a performance impact to DNS queries since each query will not be cached, and will be forwarded to a real DNS server.

FortiGuard category-based DNS domain filtering

You can use the FortiGuard category-based DNS domain filter to inspect DNS traffic. This makes use of FortiGuard's continuously updated domain rating database for more reliable protection.

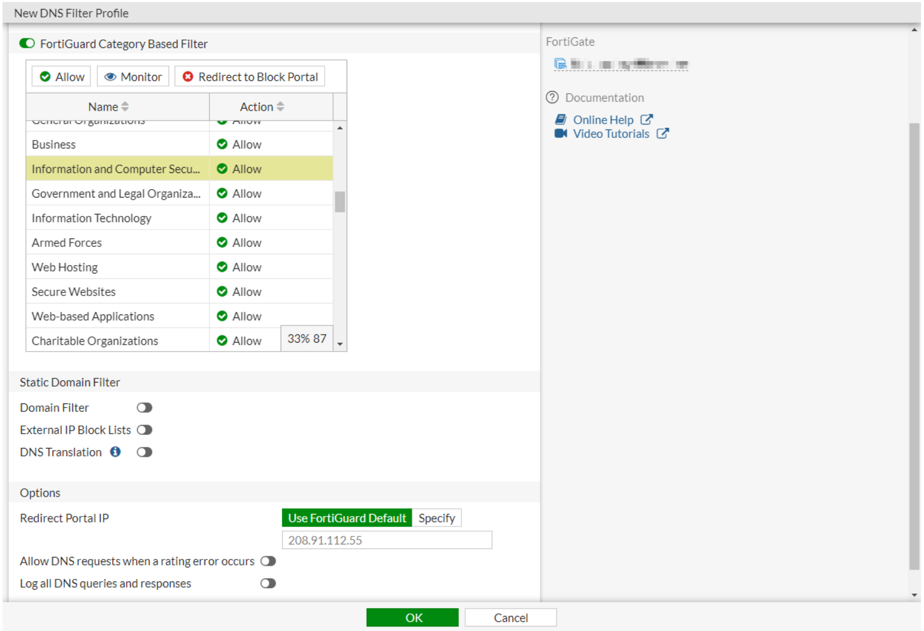
A DNS filter profile can be applied in a policy to scan DNS traffic traversing the FortiGate (see Configuring a DNS filter profile on page 1166), or applied on the DNS server interface (see Applying DNS filter to FortiGate DNS server on page 1183).

The FortiGate must have a FortiGuard Web Filter license to use the FortiGuard category-based filter.

To configure FortiGuard category-based DNS domain filtering in the GUI:

1. Go to *Security Profiles > DNS Filter* and click *Create New*, or edit an existing profile.
2. Enable *FortiGuard Category Based Filter*.
3. Select the category and then select *Allow*, *Monitor*, or *Redirect to Block Portal* for that category.
4. In the *Options* section, select a setting for *Redirect Portal IP*. Select either *Use FortiGuard Default* (208.91.112.55) or click *Specify* and enter another portal IP. The FortiGate will use the portal IP to replace the resolved IP in the DNS

response packet.



5. Click *OK*.

To configure FortiGuard category-based DNS domain filtering in the CLI:

```
config dnsfilter profile
   edit "demo"
      set comment ''
      config domain-filter
         unset domain-filter-table
      end
      config ftgd-dns
         set options error-allow
         config filters
             edit 2
                 set category 2
                 set action monitor
             next
             edit 7
                 set category 7
                 set action monitor
             next
            ...
             edit 22
                 set category 0
                 set action monitor
             next
         end
      end
      set log-all-domain enable
      set sdns-ftgd-err-log enable
      set sdns-domain-log enable
      set block-action {redirect | block}
      set block-botnet enable
```

```
        set safe-search enable
        set redirect-portal 93.184.216.34
        set youtube-restrict strict
    next
end
```

Verifying the logs

From your internal network PC, use a command line tool, such as dig or nslookup, to do a DNS query for some domains. For example:

```
#dig www.example.com
;; ->>HEADER<<- opcode: QUERY; status: NOERROR; id: 61252
;; Flags: qr rd ra; QUERY: 1; ANSWER: 1; AUTHORITY: 13; ADDITIONAL: 11

;; QUESTION SECTION:
;; www.example.com.             IN      A

;; ANSWER SECTION:
www.example.com.        17164   IN      A       93.184.216.34

;; AUTHORITY SECTION:
com.                    20027   IN      NS      h.gtld-servers.net.
com.                    20027   IN      NS      i.gtld-servers.net.
com.                    20027   IN      NS      f.gtld-servers.net.
com.                    20027   IN      NS      d.gtld-servers.net.
com.                    20027   IN      NS      j.gtld-servers.net.
com.                    20027   IN      NS      l.gtld-servers.net.
com.                    20027   IN      NS      e.gtld-servers.net.
com.                    20027   IN      NS      a.gtld-servers.net.
com.                    20027   IN      NS      k.gtld-servers.net.
com.                    20027   IN      NS      g.gtld-servers.net.
com.                    20027   IN      NS      m.gtld-servers.net.
com.                    20027   IN      NS      c.gtld-servers.net.
com.                    20027   IN      NS      b.gtld-servers.net.

;; ADDITIONAL SECTION:
a.gtld-servers.net.     21999   IN      A       192.5.6.30
a.gtld-servers.net.     21999   IN      AAAA    2001:503:a83e::2:30
b.gtld-servers.net.     21997   IN      A       192.33.14.30
b.gtld-servers.net.     21997   IN      AAAA    2001:503:231d::2:30
c.gtld-servers.net.     21987   IN      A       192.26.92.30
c.gtld-servers.net.     20929   IN      AAAA    2001:503:83eb::30
d.gtld-servers.net.     3340    IN      A       192.31.80.30
d.gtld-servers.net.     3340    IN      AAAA    2001:500:856e::30
e.gtld-servers.net.     19334   IN      A       192.12.94.30
e.gtld-servers.net.     19334   IN      AAAA    2001:502:1ca1::30
f.gtld-servers.net.     3340    IN      A       192.35.51.30

;; Received 509 B
;; Time 2019-04-05 09:39:33 PDT
;; From 172.16.95.16@53(UDP) in 3.8 ms
```

To check the DNS filter log in the GUI:

1. Go to *Log & Report > Security Events*.
2. Click the *DNS Query* card name. There are logs for the DNS traffic that just passed through the FortiGate with the FortiGuard rating for the domain name.

Add Filter | Details

Date/Time	Sub Type	Source	Domain Name	Query Type	Policy	Message	Domain Filter List	Category	Category Description	Domain Filter Index	#
2019/04/05 09:39:34	dns	10.1.100.18	www.example.com	A	1	Domain is monitored		52	Information Technology		1
2019/04/05 09:39:34	dns	10.1.100.18	www.example.com	A	1						2

To check the DNS filter log in the CLI:

```
# execute log filter category utm-dns

# execute log display
2 logs found.
2 logs returned.

1: date=2019-04-05 time=09:39:34 logid="1501054802" type="utm" subtype="dns" eventtype="dns-
response" level="notice" vd="vdom1" eventtime=1554482373 policyid=1 sessionid=50868
srcip=10.1.100.18 srcport=34308 srcintf="port10" srcintfrole="undefined" dstip=172.16.95.16
dstport=53 dstintf="port9" dstintfrole="undefined" proto=17 profile="demo" xid=17647
qname="www.example.com" qtype="A" qtypeval=1 qclass="IN" ipaddr="93.184.216.34" msg="Domain
is monitored" action="pass" cat=52 catdesc="Information Technology"

2: date=2019-04-05 time=09:39:34 logid="1500054000" type="utm" subtype="dns" eventtype="dns-
query" level="information" vd="vdom1" eventtime=1554482373 policyid=1 sessionid=50868
srcip=10.1.100.18 srcport=34308 srcintf="port10" srcintfrole="undefined" dstip=172.16.95.16
dstport=53 dstintf="port9" dstintfrole="undefined" proto=17 profile="demo" xid=17647
qname="www.example.com" qtype="A" qtypeval=1 qclass="IN"
```

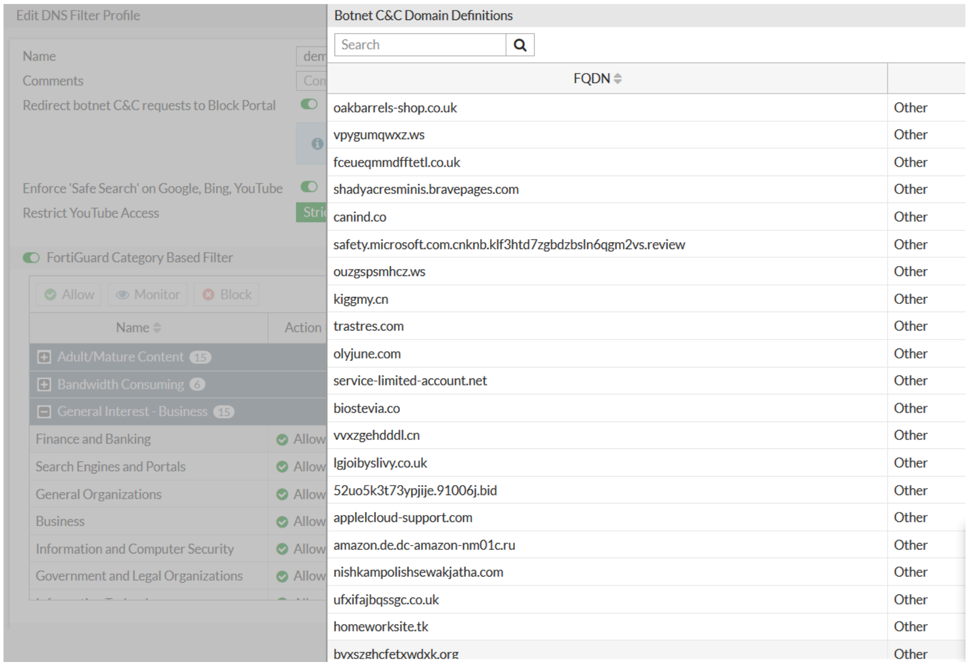
Botnet C&C domain blocking

FortiGuard Service continually updates the botnet C&C domain list. The botnet C&C domain blocking feature can block the botnet website access at the DNS name resolving stage. This provides additional protection for your network.

A DNS filter profile can be applied in a policy to scan DNS traffic traversing the FortiGate (see Configuring a DNS filter profile on page 1166), or applied on the DNS server interface (see Applying DNS filter to FortiGate DNS server on page 1183).

To configure botnet C&C domain blocking in the GUI:

1. Go to *Security Profiles > DNS Filter* and click *Create New*, or edit an existing profile.
2. Enable *Redirect botnet C&C requests to Block Portal*.
3. Optionally, click the *botnet package* link. The *Botnet C&C Domain Definitions* pane opens, which displays the latest list.



4. Configure the other settings as needed.
5. Click *OK*.

To configure botnet C&C domain blocking in the CLI:

```
config dnsfilter profile
   edit "demo"
      set comment ''
      config domain-filter
         unset domain-filter-table
      end
      config ftgd-dns
         set options error-allow
         config filters
            ...
         end
      end
      set log-all-domain enable
      set sdns-ftgd-err-log enable
      set sdns-domain-log enable
      set block-action block
      set block-botnet enable
      set safe-search enable
      set redirect-portal 208.91.112.55
      set youtube-restrict strict
   next
end
```

Verifying the logs

Select a botnet domain from that list. From your internal network PC, use a command line tool, such as dig or nslookup, to send a DNS query to traverse the FortiGate. For example:

```
#dig canind.co
;; ->>HEADER<<- opcode: QUERY; status: NOERROR; id: 997
;; Flags: qr rd; QUERY: 1; ANSWER: 1; AUTHORITY: 0; ADDITIONAL: 0

;; QUESTION SECTION:
;; canind.co.                   IN      A

;; ANSWER SECTION:
canind.co.              60      IN      A       208.91.112.55

;; Received 43 B
;; Time 2019-04-05 09:55:21 PDT
;; From 172.16.95.16@53(UDP) in 0.3 ms
```

The botnet domain query was blocked and redirected to the portal IP (208.91.112.55) .

To check the DNS filter log in the GUI:

1. Go to *Log & Report* > *Security Events*.
2. Click the *DNS Query* card name to view the DNS query blocked as a botnet domain.

Date/Time	Sub Type	Source	Domain Name	Query Type	Policy	Message	Domain Filter List	Category	Category Description
2019/04/04 16:43:59	dns	10.1.100.18	canind.co	A	1	Domain was blocked by dns botnet C&C			
2019/04/04 16:43:59	dns	10.1.100.18	canind.co	A	1				

To check the DNS filter log in the CLI:

```
(vdom1) # execute log filter category utm-dns

(vdom1) # execute log display
2 logs found.
2 logs returned.

1: date=2019-04-04 time=16:43:59 logid="1501054601" type="utm" subtype="dns" eventtype="dns-
response" level="warning" vd="vdom1" eventtime=1554421439 policyid=1 sessionid=14135
srcip=10.1.100.18 srcport=57447 srcintf="port10" srcintfrole="undefined" dstip=172.16.95.16
dstport=53 dstintf="port9" dstintfrole="undefined" proto=17 profile="demo" xid=24339
qname="canind.co" qtype="A" qtypeval=1 qclass="IN" msg="Domain was blocked by dns botnet
C&C" action="redirect" botnetdomain="canind.co"

2: date=2019-04-04 time=16:43:59 logid="1500054000" type="utm" subtype="dns" eventtype="dns-
query" level="information" vd="vdom1" eventtime=1554421439 policyid=1 sessionid=14135
srcip=10.1.100.18 srcport=57447 srcintf="port10" srcintfrole="undefined" dstip=172.16.95.16
dstport=53 dstintf="port9" dstintfrole="undefined" proto=17 profile="demo" xid=24339
qname="canind.co" qtype="A" qtypeval=1 qclass="IN"
```

Botnet C&C IPDB blocking

FortiOS also maintains a botnet C&C IP address database (IPDB). If a DNS query response IP address (resolved IP address) matches an entry inside the botnet IPDB, this DNS query is blocked by the DNS filter botnet C&C.

To view the botnet IPDB list in the CLI:

```
(global) # diagnose sys botnet list 9000 10
9000. proto=TCP ip=103.228.28.166, port=80, rule_id=7630075, name_id=3, hits=0
```

```
9001. proto=TCP ip=5.9.32.166, port=481, rule_id=4146631, name_id=7, hits=0
9002. proto=TCP ip=91.89.44.166, port=80, rule_id=48, name_id=96, hits=0
9003. proto=TCP ip=46.211.46.166, port=80, rule_id=48, name_id=96, hits=0
9004. proto=TCP ip=77.52.52.166, port=80, rule_id=48, name_id=96, hits=0
9005. proto=TCP ip=98.25.53.166, port=80, rule_id=48, name_id=96, hits=0
9006. proto=TCP ip=70.120.67.166, port=80, rule_id=48, name_id=96, hits=0
9007. proto=TCP ip=85.253.77.166, port=80, rule_id=48, name_id=96, hits=0
9008. proto=TCP ip=193.106.81.166, port=80, rule_id=48, name_id=96, hits=0
9009. proto=TCP ip=58.13.84.166, port=80, rule_id=48, name_id=96, hits=0
```

Select an IP address from the IPDB list and use a reverse lookup service to find its corresponding domain name. From your internal network PC, use a command line tool, such as dig or nslookup, to query this domain and verify that it is blocked by the DNS filter botnet C&C. For example:

```
# dig cpe-98-25-53-166.sc.res.rr.com
;; ->>HEADER<<- opcode: QUERY; status: NOERROR; id: 35135
;; Flags: qr rd; QUERY: 1; ANSWER: 1; AUTHORITY: 0; ADDITIONAL: 0

;; QUESTION SECTION:
;; cpe-98-25-53-166.sc.res.rr.com.              IN      A

;; ANSWER SECTION:
cpe-98-25-53-166.sc.res.rr.com. 60      IN      A       208.91.112.55

;; Received 64 B
;; Time 2019-04-05 11:06:47 PDT
;; From 172.16.95.16@53(UDP) in 0.6 ms
```

Since the resolved IP address matches the botnet IPDB, the query was blocked and redirected to the portal IP (208.91.112.55) .

To check the DNS filter log in the GUI:

1. Go to *Log & Report > DNS Query* to view the DNS query blocked by botnet C&C IPDB.

Date/Time	Sub Type	Source	Domain Name	Query Type	Policy	Message	Domain Filter List	Category	Category Description
2019/04/05 11:06:48	dns	10.1.100.18	cpe-98-25-53-166.sc.res.rr.com	A	1	Domain was blocked by dns botnet C&C			
2019/04/05 11:06:48	dns	10.1.100.18	cpe-98-25-53-166.sc.res.rr.com	A	1				

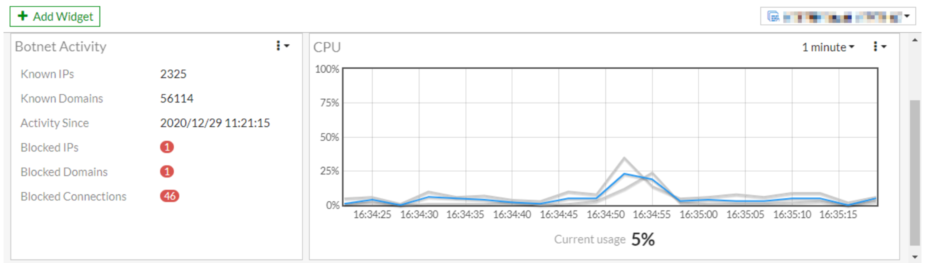
To check the DNS filter log in the CLI:

```
(global) # execute log filter category utm-dns

(global) # execute log display
2 logs found.
2 logs returned.

1: date=2019-04-05 time=11:06:48 logid="1501054600" type="utm" subtype="dns" eventtype="dns-response" level="warning" vd="vdom1" eventtime=1554487606 policyid=1 sessionid=55232 srcip=10.1.100.18 srcport=60510 srcintf="port10" srcintfrole="undefined" dstip=172.16.95.16 dstport=53 dstintf="port9" dstintfrole="undefined" proto=17 profile="demo" xid=16265 qname="cpe-98-25-53-166.sc.res.rr.com" qtype="A" qtypeval=1 qclass="IN" ipaddr="93.184.216.34" msg="Domain was blocked by dns botnet C&C" action="redirect" botnetip=98.25.53.166

2: date=2019-04-05 time=11:06:48 logid="1500054000" type="utm" subtype="dns" eventtype="dns-query" level="information" vd="vdom1" eventtime=1554487606 policyid=1 sessionid=55232
```

```
srcip=10.1.100.18 srcport=60510 srcintf="port10" srcintfrole="undefined" dstip=172.16.95.16
dstport=53 dstintf="port9" dstintfrole="undefined" proto=17 profile="demo" xid=16265
qname="cpe-98-25-53-166.sc.res.rr.com" qtype="A" qtypeval=1 qclass="IN"
```

To check botnet activity:

1. Go to *Dashboard* > *Status* and locate the *Botnet Activity* widget.



2. If you do not see the widget, click *Add Widget*, and add the *Botnet Activity* widget.

DNS safe search

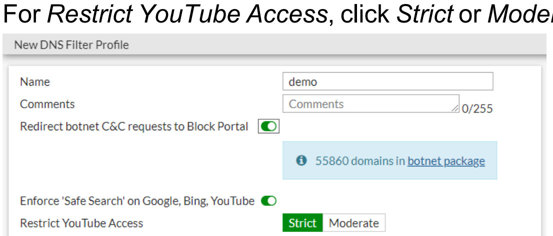
The DNS safe search option helps avoid explicit and inappropriate results in the Google, Bing, and YouTube search engines. The FortiGate responds with content filtered by the search engine.

For individual search engine safe search specifications, refer to the documentation for Google, Bing, and YouTube.

A DNS filter profile can be applied in a policy to scan DNS traffic traversing the FortiGate (see Configuring a DNS filter profile on page 1166), or applied on the DNS server interface (see Applying DNS filter to FortiGate DNS server on page 1183).

To configure safe search in the GUI:

1. Go to *Security Profiles* > *DNS Filter* and click *Create New*, or edit an existing profile.
2. Enable *Enforce 'Safe search' on Google, Bing, YouTube*.
3. For *Restrict YouTube Access*, click *Strict* or *Moderate*.



4. Configure the other settings as needed.
5. Click *OK*.

To configure safe search in the CLI:

```
config dnsfilter profile
    edit "demo"
```

```
        config ftgd-dns
            set options error-allow
            config filters
                edit 2
                    set category 2
                next
                ...
            end
        end
        set log-all-domain enable
        set block-botnet enable
        set safe-search enable
        set youtube-restrict strict
    next
end
```

Verifying the logs

From your internal network PC, use a command line tool, such as dig or nslookup, and perform a DNS query on www.bing.com. For example:

```
# dig www.bing.com
;; ->>HEADER<<- opcode: QUERY; status: NOERROR; id: 46568
;; Flags: qr rd ra; QUERY: 1; ANSWER: 2; AUTHORITY: 0; ADDITIONAL: 0

;; QUESTION SECTION:
;; www.bing.com.                IN      A

;; ANSWER SECTION:
www.bing.com.           103     IN      CNAME   strict.bing.com
strict.bing.com.        103     IN      A       204.79.197.220

;; Received 67 B
;; Time 2019-04-05 14:34:52 PDT
;; From 172.16.95.16@53(UDP) in 196.0 ms
```

The DNS query for www.bing.com returns with a CNAME strict.bing.com, and an A record for the CNAME. The user's web browser then connects to this address with the same search engine UI, but any explicit content search is filtered out.

To check the DNS filter log in the GUI:

1. Go to *Log & Report > Security Events*.
2. Click the *DNS Query* card name.

Date/Time	Sub Type	Source	Domain Name	Query Type	Policy	Message	Domain Filter List	Category	Category Description
2019/04/05 14:34:53	dns	10.1.100.18	www.bing.com	A	1	DNS Safe Search enforced		41	Search Engines and Portals
2019/04/05 14:34:53	dns	10.1.100.18	www.bing.com	A	1				

The DNS filter log in FortiOS shows a message of *DNS Safe Search enforced*.

To check the DNS filter log in the CLI:

```
# execute log filter category utm-dns
# execute log display
2 logs found.
2 logs returned.
```

```
1: date=2019-04-05 time=14:34:53 logid="1501054804" type="utm" subtype="dns" eventtype="dns-
response" level="notice" vd="vdom1" eventtime=1554500093 policyid=1 sessionid=65955
srcip=10.1.100.18 srcport=36575 srcintf="port10" srcintfrole="undefined" dstip=172.16.95.16
dstport=53 dstintf="port9" dstintfrole="undefined" proto=17 profile="demo" xid=59573
qname="www.bing.com" qtype="A" qtypeval=1 qclass="IN" ipaddr="204.79.197.220" msg="DNS Safe
Search enforced" action="pass" sscname="strict.bing.com" cat=41 catdesc="Search Engines and
Portals"

2: date=2019-04-05 time=14:34:53 logid="1500054000" type="utm" subtype="dns" eventtype="dns-
query" level="information" vd="vdom1" eventtime=1554500092 policyid=1 sessionid=65955
srcip=10.1.100.18 srcport=36575 srcintf="port10" srcintfrole="undefined" dstip=172.16.95.16
dstport=53 dstintf="port9" dstintfrole="undefined" proto=17 profile="demo" xid=59573
qname="www.bing.com" qtype="A" qtypeval=1 qclass="IN"
```

Local domain filter

In addition to the FortiGuard category-based domain filter, you can define a local static domain filter to allow or block specific domains.

In a DNS filter profile, the local domain filter has a higher priority than FortiGuard category-based domain filter. DNS queries are scanned and matched first with the local domain filter. If an entry matches and the local filter action is set to block, then that DNS query is blocked and redirected.

If the local domain filter list has no match, then the FortiGuard category-based domain filter is used. If a DNS query domain name rating belongs to the block category, the query is blocked and redirected. If the FortiGuard category-based filter has no match, then the original resolved IP address is returned to the client DNS resolver.

If the local domain filter action is set to allow and an entry matches, it will skip the FortiGuard category-based domain filter and directly return to the client DNS resolver. If the local domain filter action is set to monitor and an entry matches, it will go to the FortiGuard category-based domain filter for scanning and matching.

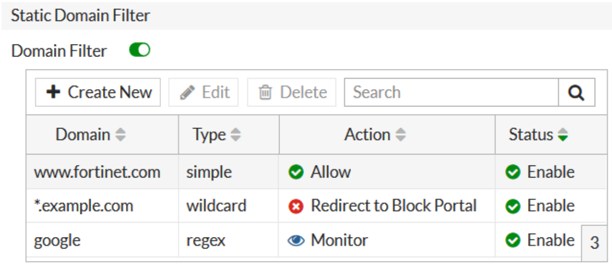
A DNS filter profile can be applied in a policy to scan DNS traffic traversing the FortiGate (see Configuring a DNS filter profile on page 1166), or applied on the DNS server interface (see Applying DNS filter to FortiGate DNS server on page 1183).

To configure the local domain filter in the GUI:

1. Go to *Security Profiles > DNS Filter* and click *Create New*, or edit an existing profile.
2. In the *Static Domain Filter* section, enable *Domain Filter*.
3. Click *Create New*. The *Create Domain Filter* pane opens.
4. Enter a domain, and select a *Type* and *Action*. This example has three filters:

Domain	Type	Action
www.fortinet.com	Simple	Allow
*.example.com	Wildcard	Redirect to Block Portal
google	Reg. Expression	Monitor

5. Click *OK*. The entry appears in the table.



6. Configure the other settings as needed.
7. Click *OK*.

To configure the local domain filter in the CLI:

```
config dnsfilter domain-filter
    edit 1
        set name "demo"
        set comment ''
        config entries
            edit 1
                set domain "www.fortinet.com"
                set type simple
                set action allow
                set status enable
            next
            edit 2
                set domain "*.example.com"
                set type wildcard
                set action block
                set status enable
            next
            edit 3
                set domain "google"
                set type regex
                set action monitor
                set status enable
            next
        end
    next
end
```

Wildcard entries are converted to regular expressions by FortiOS. As a result, wildcards will match any suffix, as long as there is a word boundary following the search term.

For example:

```
config entries
    edit 1
        set domain "*.host"
        set type wildcard
    next
end
```

will match `wp36.host` and `wp36.host.pressdns.com`, but not `wp36.host123.pressdnds.com`.

To avoid this, use an explicit regular expression search string:

```
config entries
    edit 1
        set domain "^.*\\.host$"
        set type regexp
    next
end
```

To check the DNS filter log in the GUI:

1. Go to *Log & Report > Security Events*.
2. Click the *DNS Query* card name.

Date/Time	Sub Type	Source	Domain Name	Query Type	Policy	Message	Domain Filter List	Category	Category Description
2019/04/05 15:37:06	dns	10.1.100.18	www.google.com	A	1	Domain belongs to a denied category in policy		41	Search Engines and Port
2019/04/05 15:37:06	dns	10.1.100.18	www.google.com	A	1				
2019/04/05 15:36:59	dns	10.1.100.18	www.example.com	A	1	Domain was blocked because it is in the domain-filter list	demo		
2019/04/05 15:36:59	dns	10.1.100.18	www.example.com	A	1				
2019/04/05 15:36:51	dns	10.1.100.18	www.fortinet.com	A	1	Domain was allowed because it is in the domain-filter list	demo		
2019/04/05 15:36:51	dns	10.1.100.18	www.fortinet.com	A	1				

Since the local domain filter for *google* is set to monitor, it is blocked by the FortiGuard category-based domain filter because the policy action is deny.

To check the DNS filter log in the CLI:

```
# execute log filter category utm-dns
# execute log display
...
7: date=2019-04-05 time=15:37:06 logid="1501054803" type="utm" subtype="dns" eventtype="dns-
response" level="warning" vd="vdom1" eventtime=1554503826 policyid=1 sessionid=69132
srcip=10.1.100.18 srcport=49832 srcintf="port10" srcintfrole="undefined" dstip=172.16.95.16
dstport=53 dstintf="port9" dstintfrole="undefined" proto=17 profile="demo" xid=4612
qname="www.google.com" qtype="A" qtypeval=1 qclass="IN" ipaddr="208.91.112.55" msg="Domain
belongs to a denied category in policy" action="redirect" cat=41 catdesc="Search Engines and
Portals"

8: date=2019-04-05 time=15:37:06 logid="1500054000" type="utm" subtype="dns" eventtype="dns-
query" level="information" vd="vdom1" eventtime=1554503826 policyid=1 sessionid=69132
srcip=10.1.100.18 srcport=49832 srcintf="port10" srcintfrole="undefined" dstip=172.16.95.16
```

```
dstport=53 dstintf="port9" dstintfrole="undefined" proto=17 profile="demo" xid=4612
qname="www.google.com" qtype="A" qtypeval=1 qclass="IN"

9: date=2019-04-05 time=15:36:59 logid="1501054400" type="utm" subtype="dns" eventtype="dns-
response" level="warning" vd="vdom1" eventtime=1554503818 policyid=1 sessionid=69121
srcip=10.1.100.18 srcport=40659 srcintf="port10" srcintfrole="undefined" dstip=172.16.95.16
dstport=53 dstintf="port9" dstintfrole="undefined" proto=17 profile="demo" xid=24730
qname="www.example.com" qtype="A" qtypeval=1 qclass="IN" msg="Domain was blocked because it
is in the domain-filter list" action="redirect" domainfilteridx=1 domainfilterlist="demo"

10: date=2019-04-05 time=15:36:59 logid="1500054000" type="utm" subtype="dns"
eventtype="dns-query" level="information" vd="vdom1" eventtime=1554503818 policyid=1
sessionid=69121 srcip=10.1.100.18 srcport=40659 srcintf="port10" srcintfrole="undefined"
dstip=172.16.95.16 dstport=53 dstintf="port9" dstintfrole="undefined" proto=17
profile="demo" xid=24730 qname="www.example.com" qtype="A" qtypeval=1 qclass="IN"

11: date=2019-04-05 time=15:36:51 logid="1501054401" type="utm" subtype="dns"
eventtype="dns-response" level="information" vd="vdom1" eventtime=1554503810 policyid=1
sessionid=69118 srcip=10.1.100.18 srcport=33461 srcintf="port10" srcintfrole="undefined"
dstip=172.16.95.16 dstport=53 dstintf="port9" dstintfrole="undefined" proto=17
profile="demo" xid=53801 qname="www.fortinet.com" qtype="A" qtypeval=1 qclass="IN"
ipaddr="13.56.55.78, 54.183.57.55" msg="Domain was allowed because it is in the domain-
filter list" action="pass" domainfilteridx=1 domainfilterlist="demo"

12: date=2019-04-05 time=15:36:51 logid="1500054000" type="utm" subtype="dns"
eventtype="dns-query" level="information" vd="vdom1" eventtime=1554503810 policyid=1
sessionid=69118 srcip=10.1.100.18 srcport=33461 srcintf="port10" srcintfrole="undefined"
dstip=172.16.95.16 dstport=53 dstintf="port9" dstintfrole="undefined" proto=17
profile="demo" xid=53801 qname="www.fortinet.com" qtype="A" qtypeval=1 qclass="IN"
```

DNS translation

This setting allows you to translate a DNS resolved IP address to another IP address you specify on a per-policy basis.

For example, website A has a public address of 1.2.3.4. However, when your internal network users visit this website, you want them to connect to the internal host 192.168.3.4. You can use DNS translation to translate the DNS resolved address 1.2.3.4 to 192.168.3.4. Reverse use of DNS translation is also applicable. For example, if you want a public DNS query of your internal server to get a public IP address, then you can translate a DNS resolved private IP to a public IP address.

A DNS filter profile can be applied in a policy to scan DNS traffic traversing the FortiGate (see Configuring a DNS filter profile on page 1166), or applied on the DNS server interface (see Applying DNS filter to FortiGate DNS server on page 1183).

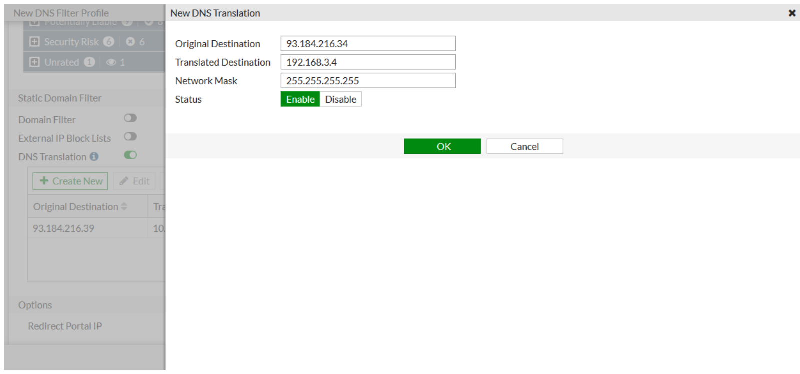
Sample configuration

This configuration forces the DNS filter profile to translate 93.184.216.34 (www.example.com) to 192.168.3.4. When internal network users perform a DNS query for www.example.com, they do not get the original www.example.com IP address of 93.184.216.34. Instead, it is replaced with 192.168.3.4.

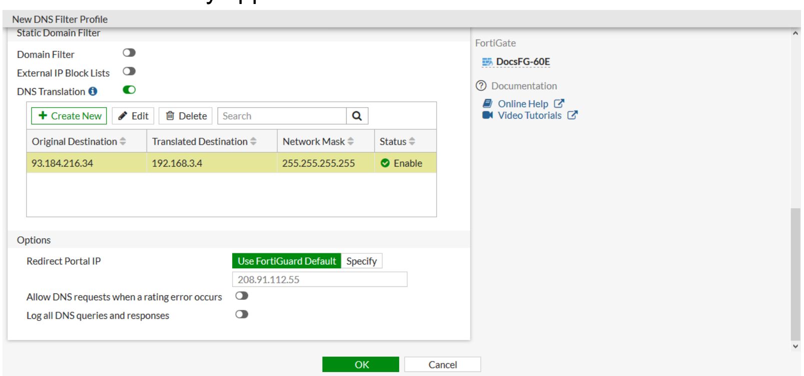
To configure DNS translation in the GUI:

1. Go to *Security Profiles > DNS Filter* and click *Create New*, or edit an existing profile.
2. In the *Static Domain Filter* section, enable *DNS Translation*.

3. Click *Create New*. The *New DNS Translation* pane opens.
4. Enter the *Original Destination* (the domain's original IP address), the *Translated Destination* IP address, and the *Network Mask*.



5. Click *OK*. The entry appears in the table.



6. Configure the other settings as needed.
7. Click *OK*.

To configure DNS translation in the CLI:

```
config dnsfilter profile
    edit "demo"
        set comment ''
        ...
        config dns-translation
            edit 1
                set src 93.184.216.34
                set dst 192.168.3.4
                set netmask 255.255.255.255
            next
        end
        set redirect-portal 0.0.0.0
        set redirect-portal6 ::
        set youtube-restrict strict
    next
end
```

To check DNS translation using a command line tool before DNS translation:

```
# dig www.example.com
;; ->>HEADER<<- opcode: QUERY; status: NOERROR; id: 27030
;; Flags: qr rd ra; QUERY: 1; ANSWER: 1; AUTHORITY: 2; ADDITIONAL: 0

;; QUESTION SECTION:
;; www.example.com.            IN      A

;; ANSWER SECTION:
www.example.com.        33946   IN      A       93.184.216.34

;; AUTHORITY SECTION:
example.com.            18578   IN      NS      b.iana-servers.net.
example.com.            18578   IN      NS      a.iana-servers.net.

;; Received 97 B
;; Time 2019-04-08 10:47:26 PDT
;; From 172.16.95.16@53(UDP) in 0.5 ms
```

To check DNS translation using a command line tool after DNS translation:

```
# dig www.example.com
;; ->>HEADER<<- opcode: QUERY; status: NOERROR; id: 62060
;; Flags: qr rd ra; QUERY: 1; ANSWER: 1; AUTHORITY: 2; ADDITIONAL: 0

;; QUESTION SECTION:
;; www.example.com.            IN      A

;; ANSWER SECTION:
www.example.com.        32491   IN      A       192.168.3.4

;; AUTHORITY SECTION:
example.com.            17123   IN      NS      b.iana-servers.net.
example.com.            17123   IN      NS      a.iana-servers.net.

;; Received 97 B
;; Time 2019-04-08 11:11:41 PDT
;; From 172.16.95.16@53(UDP) in 0.5 ms
```

DNS translation network mask

The following is an example of DNS translation that uses a network mask:

To configure DNS translation in the CLI:

```
config dns-translation
        edit 1
           set src 93.184.216.34
           set dst 1.2.3.4
           set netmask 255.255.224.0
        next
    end
```

To check DNS translation using a command line tool after DNS translation:

```
# dig www.example.com
;; ->>HEADER<<- opcode: QUERY; status: NOERROR; id: 6736
;; Flags: qr rd ra; QUERY: 1; ANSWER: 1; AUTHORITY: 2; ADDITIONAL: 0

;; QUESTION SECTION:
;; www.example.com.             IN      A

;; ANSWER SECTION:
www.example.com.        29322   IN      A       1.2.24.34

;; AUTHORITY SECTION:
example.com.            13954   IN      NS      a.iana-servers.net.
example.com.            13954   IN      NS      b.iana-servers.net.

;; Received 97 B
;; Time 2019-04-08 12:04:30 PDT
;; From 172.16.95.16@53(UDP) in 2.0 ms
```

The binary arithmetic to convert 93.184.216.34 to 1.2.3.4 with the subnet mask is as follows:

1. AND src(Original IP) with negative netmask (93.184.216.34 & ~255.255.224.0):

```
01011101.10111000.11011000.00100010 93.184.216.34
00000000.00000000.00011111.11111111 ~255.255.224.0
-------------------------------------------------------- &
00000000.00000000.00011000.00100010 0.0.24.34
```

2. AND dst(Translated IP) with netmask:

```
00000001.00000010.00000011.00000100 1.2.3.4
11111111.11111111.11100000.00000000 255.255.224.0
-------------------------------------------------------- &
00000001.00000010.00000000.00000000 1.2.0.0
```

3. Final step 2 bitwise-OR 3:

```
00000000.00000000.00011000.00100010 0.0.24.34
00000001.00000010.00000000.00000000 1.2.0.0
-------------------------------------------------------- |
00000001.00000010.00011000.00100010 1.2.24.34
```

Applying DNS filter to FortiGate DNS server

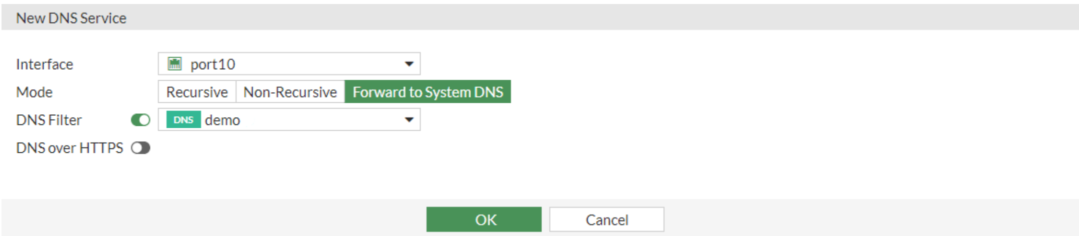
You can configure a FortiGate as a DNS server in your network. When you enable DNS service on a specific interface, the FortiGate will listen for DNS service on that interface.

Depending on the configuration, DNS service works in three modes: *Recursive*, *Non-Recursive*, or *Forward to System DNS* (server). For details on how to configure the FortiGate as a DNS server and configure the DNS database, see FortiGate DNS server on page 206.

You can apply a DNS filter profile to *Recursive* and *Forward to System DNS* mode. This is the same as the FortiGate working as a transparent DNS proxy for DNS relay traffic.

To configure DNS service in the GUI:

1. Go to *Network > DNS Servers* (if this option is not available, go to *System > Feature Visibility* and enable *DNS Database*).
2. In the *DNS Service on Interface* section, click *Create New* and select an *Interface* from the dropdown.
3. For *Mode*, select *Forward to System DNS*.
4. Enable *DNS Filter* and select a profile from the dropdown.



5. Click *OK*.

To configure DNS service in the CLI:

```
config system dns-server
    edit "port10"
        set mode forward-only
        set dnsfilter-profile "demo"
    next
end
```

To check DNS service with a DNS filter profile using a command line tool:

In this example, port10 is enabled as a DNS service with the DNS filter profile demo. The IP address of port10 is 10.1.100.5 , and the DNS filter profile is configured to block category 52 (information technology). From your internal network PC, use a command line tool, such as dig or nslookup, to perform a DNS query. For example:

```
# dig @10.1.100.5 www.fortinet.com
;; ->>HEADER<<- opcode: QUERY; status: NOERROR; id: 52809
;; Flags: qr rd; QUERY: 1; ANSWER: 1; AUTHORITY: 0; ADDITIONAL: 0

;; QUESTION SECTION:
;; www.fortinet.com.            IN      A

;; ANSWER SECTION:
www.fortinet.com.       60      IN      A       208.91.112.55

;; Received 50 B
;; Time 2019-04-08 14:36:34 PDT
;; From 10.1.100.5@53(UDP) in 13.6 ms
```

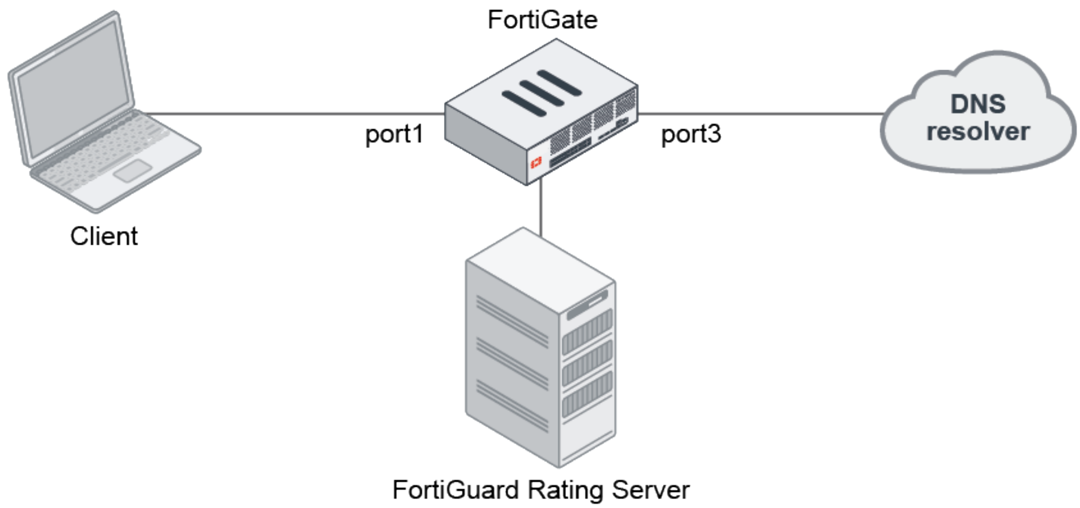
The relay DNS traffic was filtered based on the DNS filter profile configuration. It was blocked and redirected to the portal IP (208.91.112.55).

DNS inspection with DoT and DoH

DNS over TLS (DoT) and DNS over HTTPS (DoH) are supported in DNS inspection. Prior to 7.0, DoT and DoH traffic silently passes through the DNS proxy. In 7.0. the WAD is able to handle DoT and DoH, and redirect DNS queries to the

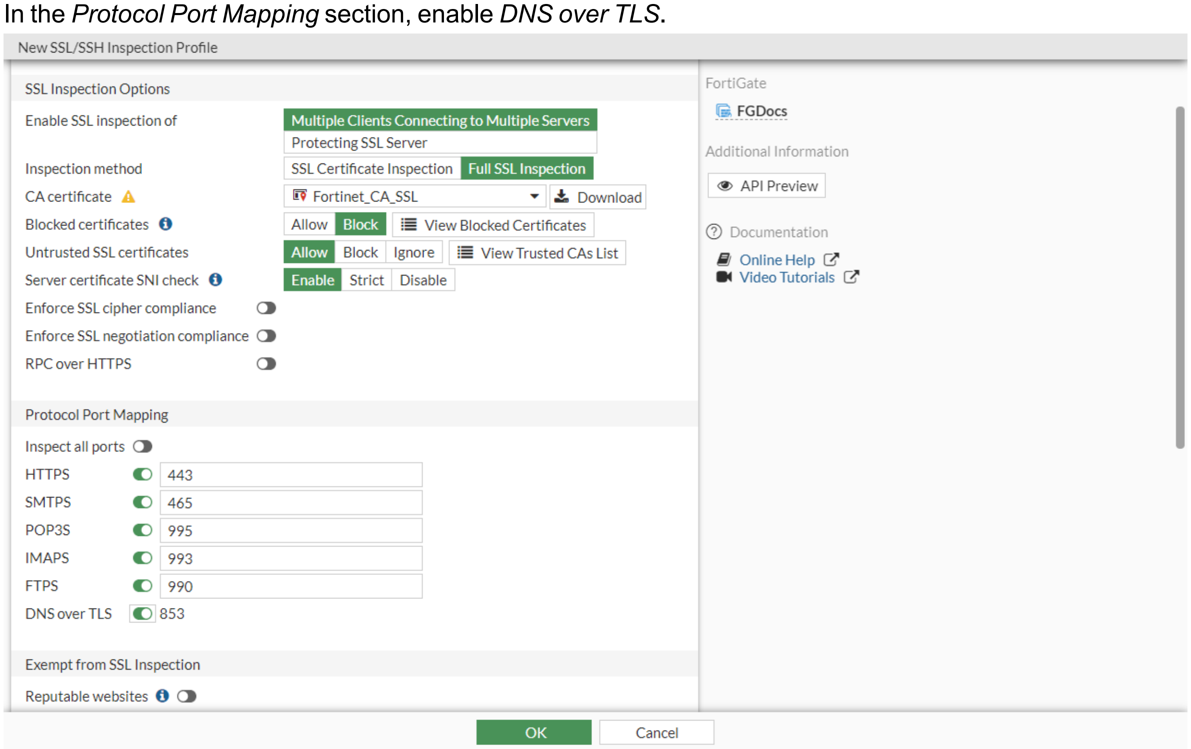
DNS proxy for further inspection.

In the following examples, the FortiGate inspects DNS queries made over DoT and DoH to a Cloudflare DNS server. The DNS filter profile blocks the education category.



To configure DNS inspection of DoT and DoH queries in the GUI:

1. Configure the SSL-SSH profile:
   a. Go to *Security Profiles > SSL/SSH Inspection* and click *Create New*.
   b. Set *Inspection method* to *Full SSL Inspection*. DoT and DoH can only be inspected using doing deep inspection.
   c. In the *Protocol Port Mapping* section, enable *DNS over TLS*.



   d. Configure the other settings as needed.
   e. Click *OK*.
2. Configure the DNS filter profile:
   a. Go to *Security Profiles > DNS Filter* and click *Create New*.
   b. Enable *Redirect botnet C&C requests to Block Portal*.
   c. Enable *FortiGuard Category Based Filter* and set the *Action* for the *Education* category to *Redirect to Block Portal*.
   d. Configure the other settings as needed.
   e. Click *OK*.
3. Configure the firewall policy:
   a. Go to *Policy & Objects > Firewall Policy* and click *Create New*.
   b. Enable *DNS Filter* and select the profile you created.
   c. For *SSL Inspection*, select the profile you created.
   d. Configure the other settings as needed.
   e. Click *OK*.

To configure DNS inspection of DoT and DoH queries in the CLI:

1. Configure the SSL-SSH profile:

```
config firewall ssl-ssh-profile
    edit "ssl"
        config dot
            set status deep-inspection
            set client-certificate bypass
            set unsupported-ssl-cipher allow
            set unsupported-ssl-negotiation allow
            set expired-server-cert block
            set revoked-server-cert block
            set untrusted-server-cert allow
            set cert-validation-timeout allow
            set cert-validation-failure block
        end
    next
end
```

2. Configure the DNS filter profile:

```
config dnsfilter profile
    edit "dnsfilter"
        config ftgd-dns
            config filters
                edit 1
                    set category 30
                    set action block
                next
            end
        end
        set block-botnet enable
    next
end
```

3. Configure the firewall policy:

```
config firewall policy
    edit 1
        set srcintf "port1"
        set dstintf "port3"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set inspection-mode proxy
        set profile-protocol-options "protocol"
        set ssl-ssh-profile "ssl"
        set webfilter-profile "webfilter"
        set dnsfilter-profile "dnsfilter"
        set nat enable
    next
end
```

Testing the connection

To query DNS over TLS:

1. Send a DNS query over TLS to the Cloudflare server 1.1.1.1 (this example uses kdig on an Ubuntu client). The www.ubc.ca domain belongs to the education category:

```
~$ kdig -d @1.1.1.1 +tls-ca +tls-host=cloudflare-dns.com www.ubc.ca
;; DEBUG: Querying for owner(www.ubc.ca.), class(1), type(1), server(1.1.1.1), port
(853), protocol(TCP)
;; DEBUG: TLS, imported 128 system certificates
;; DEBUG: TLS, received certificate hierarchy:
;; DEBUG:  #1, C=US,ST=California,L=San Francisco,O=Cloudflare\, Inc.,CN=cloudflare-
dns.com
;; DEBUG:      SHA-256 PIN: elpYCnCs9ZtkQBI4+cb2QtZcyOl5UI9jMkSvbTsTad0=
;; DEBUG:  #2, C=US,ST=California,L=Sunnyvale,O=Fortinet,OU=Certificate
Authority,CN=FG3H1E5818903681,EMAIL=support@fortinet.com
;; DEBUG:      SHA-256 PIN: s48VtdODlNZfAG2g/92hMLhitU51qsP9pkHAUtTJ+f4=
;; DEBUG: TLS, skipping certificate PIN check
;; DEBUG: TLS, The certificate is trusted.
;; TLS session (TLS1.3)-(ECDHE-SECP256R1)-(ECDSA-SECP256R1-SHA256)-(AES-256-GCM)
;; ->>HEADER<<- opcode: QUERY; status: NOERROR; id: 56850
;; Flags: qr rd; QUERY: 1; ANSWER: 1; AUTHORITY: 0; ADDITIONAL: 0

;; QUESTION SECTION:
;; www.ubc.ca.                  IN      A

;; ANSWER SECTION:
www.ubc.ca.             60      IN      A       208.91.112.55

;; Received 44 B
;; Time 2021-03-12 06:53:37 UTC
;; From 1.1.1.1@853(TCP) in 6.0 ms
```

In this query, the FortiGate inspects the DNS query to the Cloudflare DNS server. It replaces the result with the IP of the FortiGuard block page, which successfully blocks the query.

To query DNS over HTTPS:

1. In your browser, enable DNS over HTTPS.
2. Go to www.ubc.ca. The website is redirected to the block page.

Troubleshooting for DNS filter

If you have trouble with the DNS filter profile in your policy, start with the following troubleshooting steps:

- Check the connection between the FortiGate and FortiGuard DNS rating server (SDNS server).
- Check that the FortiGate has a valid FortiGuard web filter license.
- Check the FortiGate DNS filter configuration.

Checking the connection between the FortiGate and FortiGuard SDNS server

You need to ensure the FortiGate can connect to the FortiGuard SDNS server. By default, the FortiGate uses UDP port 53 to connect to the SDNS server.

To check the connection between the FortiGate and SDNS server:

1. Verify the FortiGuard SDNS server information:

```
# diagnose test application dnsproxy 3
...
FDG_SERVER:208.91.112.220:53
FGD_CATEGORY_VERSION:8
SERVER_LDB: gid=6f00, tz=-420, error_allow=0
FGD_REDIR:208.91.112.55
```

   The SDNS server IP address might be different depending on location (in this example, it is 208.91.112.220:53).

2. In the management VDOM, check the communication between the FortiGate and the SDNS server:

```
#execute ping 208.91.112.220
```

3. Optionally, you can check the communication using a PC on the internal network (this example uses dig).
   a. Disable the DNS filter profile so that it does not affect your connection check.
   b. Ping your ISP or a public DNS service provider's DNS server, for example, Google's public DNS server of 8.8.8.8:

```
#dig @8.8.8.8 www.fortinet.com
```

   Or, specify the SDNS server as a DNS server:

```
#dig @208.91.112.220 www.fortinet.com
```

c. Verify that you can get a domain www.fortinet.com A record from the DNS server. This shows that the UDP port 53 connection path is not blocked.

```
#dig @8.8.8.8 www.fortinet.com
;; ->>HEADER<<- opcode: QUERY; status: NOERROR; id: 35121
;; Flags: qr rd ra; QUERY: 1; ANSWER: 3; AUTHORITY: 0; ADDITIONAL: 0

;; QUESTION SECTION:
;; www.fortinet.com.            IN      A

;; ANSWER SECTION:
www.fortinet.com.       289     IN      CNAME   fortinet-prod4-858839915.us-west-
1.elb.amazonaws.com.
fortinet-prod4-858839915.us-west-1.elb.amazonaws.com. 51        IN      A
52.8.142.247
fortinet-prod4-858839915.us-west-1.elb.amazonaws.com. 51        IN      A
13.56.55.78

;; Received 129 B
;; Time 2019-04-29 14:13:18 PDT
;; From 8.8.8.8@53(UDP) in 13.2 ms
```

Checking the FortiGuard DNS rating service license

The FortiGuard DNS rating service shares the license with the FortiGuard web filter, so you must have a valid web filter license for the DNS rating service to work. While the license is shared, the DNS rating service uses a separate connection mechanism from the web filter rating.

To check the DNS rating service license in the CLI:

1. View the DNS settings:

```
# diagnose test application dnsproxy 3
```

2. Look for the `FGD_DNS_SERVICE_LICENSE` line and check that the license has not expired:

```
FGD_DNS_SERVICE_LICENSE:
server=208.91.112.220:53, expiry=2022-10-03, expired=0, type=2
```

3. Check the `sdns-server` lines to show the functioning servers:

```
sdns-server:208.91.112.220:53 tz=-480 tls=0 req=0 to=0 res=0 rt=4 ready=1 timer=0
probe=0 failure=0 last_failed=0
```

Checking the FortiGate DNS filter profile configuration

To check the DNS filter profile configuration:

1. In FortiOS, create a local domain filter and set the *Action* to *Redirect to Block Portal* (see Local domain filter on page 1177).
2. Apply this DNS filter profile to the policy.
3. From the client PC, perform a DNS query on this domain. If you get the profile's redirected portal address, this means that the DNS filter profile works as expected.

Additional troubleshooting

Use `diagnose test application dnsproxy <test level>` to troubleshoot further DNS proxy information, where:

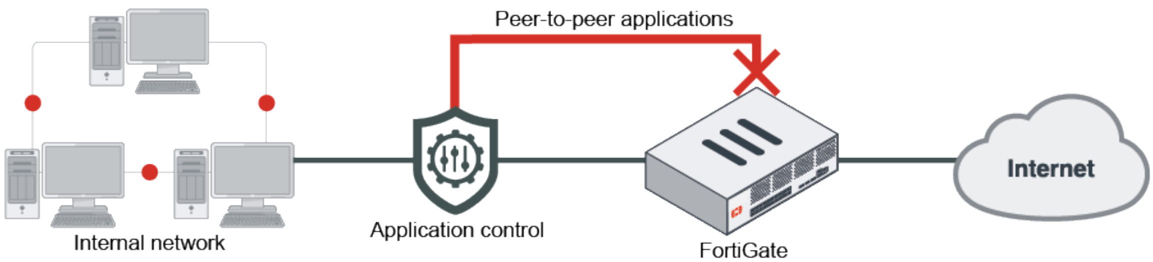
Test level	Action
1	Clear DNS cache
2	Show statistics
3	Dump DNS setting
4	Reload FQDN
5	Requery FQDN
6	Dump FQDN
7	Dump DNS cache
8	Dump DNS database
9	Reload DNS database
10	Dump secure DNS policy/profile
11	Dump botnet domain
12	Reload secure DNS setting
13	Show hostname cache
14	Clear hostname cache
15	Show SDNS rating cache
16	Clear SDNS rating cache
17	Show DNS debug bit mask
18	Show DNS debug object members
99	Restart the dnsproxy worker

To debug DNS proxy details:

```
#diagnose debug application dnsproxy -1
#diagnose debug {enable | disable}
```

Application control

FortiGates can recognize network traffic generated by a large number of applications. Application control sensors specify what action to take with the application traffic. Application control uses IPS protocol decoders that can analyze network traffic to detect application traffic, even if the traffic uses non-standard ports or protocols. Application control supports traffic detection using the HTTP protocol (versions 1.0, 1.1, and 2.0).



FortiOS includes three preloaded application sensors:

- *default* (monitors all applications)
- *wifi-default* (default configuration for offloading WiFi traffic)
- *block-high-risk*

You can customize these sensors, or you can create your own to log and manage the applications on your network.

Once configured, you can add the application sensor to a firewall policy.

This functionality requires a subscription to FortiGuard Application Control.

The following topics provide information about application control:

- Basic category filters and overrides on page 1191
- Excluding signatures in application control profiles on page 1194
- Port enforcement check on page 1196
- Protocol enforcement on page 1197
- SSL-based application detection over decrypted traffic in a sandwich topology on page 1199
- Matching multiple parameters on application control signatures on page 1200
- Application signature dissector for DNP3 on page 1202

Basic category filters and overrides

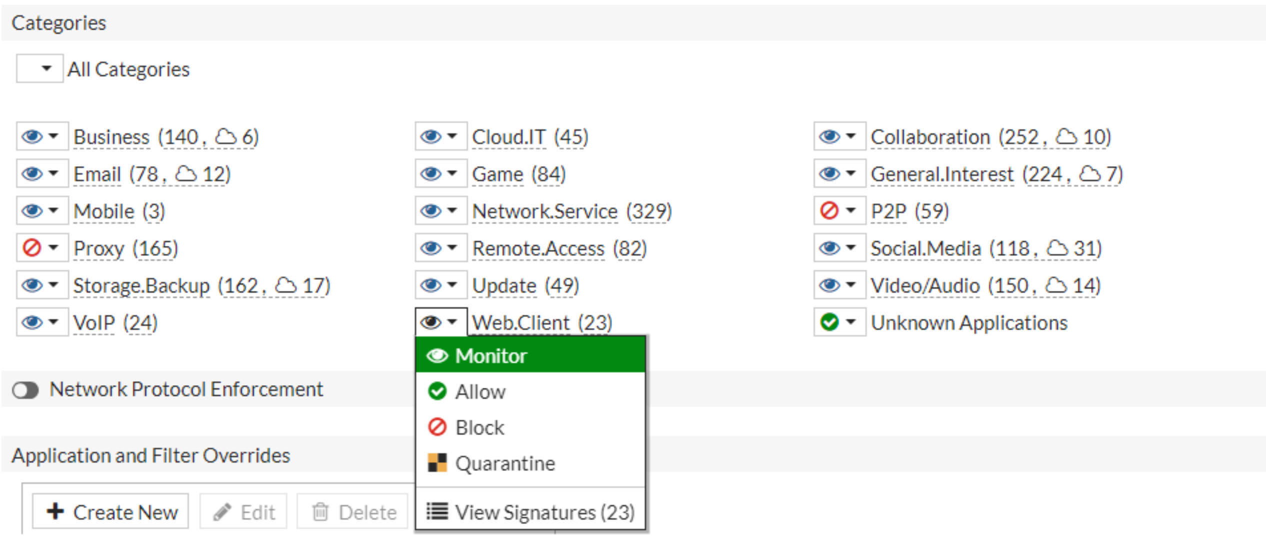
Once you have created an application sensor, you can define the applications that you want to control. You can add applications and filters using categories, application overrides, and/or filter overrides with designated actions (monitor, allow, block, or quarantine).

Configuring category filters

Categories allow you to choose groups of signatures based on a category type. Applications belonging to the category trigger the action that is set for the category. For a list of application control categories, refer to the FortiGuard Labs website.

To configure category filters in the GUI:

1. Go to *Security Profiles > Application Control* and click *Create New*, or edit an existing sensor.
2. Under *Categories*, click the icon next to the category name to set the action or view the application signatures.



3. Click *OK*.

To configure category filters in the CLI:

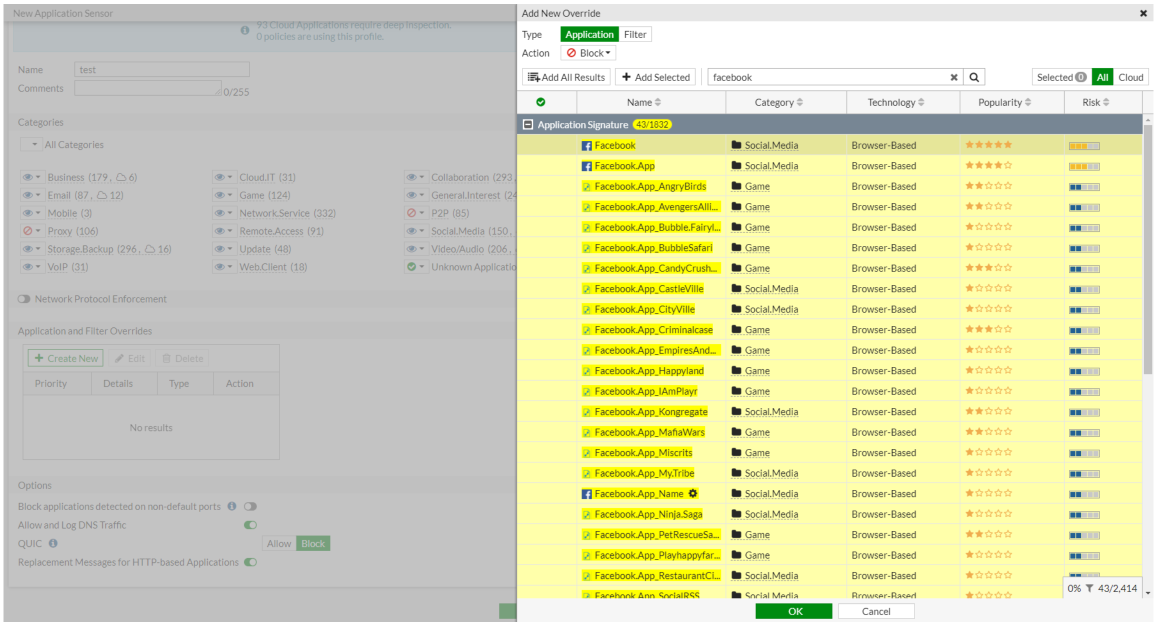
```
config application list
    edit <name>
        config entries
            edit <id>
                set category <id>
                set action {pass | block | reset}
                set log {enable | disable}
            next
        end
    next
end
```

Configuring application and filter overrides

Multiple application signatures can be added for one sensor with a designated action. Filters can be added based on behavior, application category, popularity, protocol, risk, technology, or vendor subtypes.

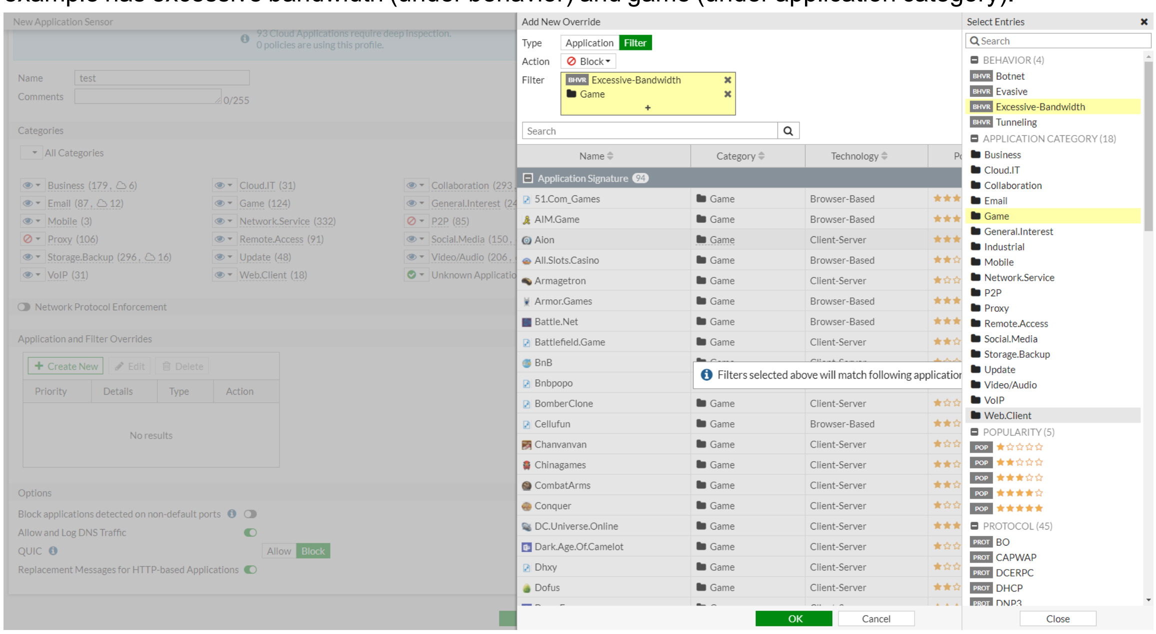
To configure overrides in the GUI:

1. Go to *Security Profiles > Application Control* and click *Create New*, or edit an existing sensor.
2. In the *Application and Filter Overrides* table, click *Create New*.
3. Add an application:
    a. For *Type*, select *Application*.
    b. Select an *Action* from the dropdown.
    c. In the *Search* box, enter an application name and press `Enter`.
    d. In the search results, select desired the applications (you can select multiple applications) and click *Add Selected*.



e. Click *OK*.

4. Add a filter:
   a. In the *Application and Filter Overrides* table, click *Create New*.
   b. For *Type*, select *Filter*.
   c. Select an *Action* from the dropdown.
   d. In the *Filter* field, click the + . The *Select Entries* pane opens, and you can search based on filter subtypes. This example has excessive bandwidth (under behavior) and game (under application category).



   e. Click *OK*.

5. Click *OK*.

To configure overrides in the CLI:

```
config application list
    edit <name>
        config entries
            edit <id>
                set protocols <integer>
                set risk <integer>
                set vendor <id>
                set technology <id>
                set behavior <id>
                set popularity <integer>
                set action {pass | block | reset}
                set log {enable | disable}
            next
        end
    next
end
```

`protocols <integer>`	Application protocol filter (0 - 47, or all).
`risk <integer>`	Risk or impact of allowing traffic from this application to occur (1 - 5; low (1), elevated (2), medium (3), high (4), and critical (5)).
`vendor <id>`	Application vendor filter (0 - 25, or all).
`technology <id>`	Application technology filter: • all • 0 (network-protocol) • 1 (browser-based) • 2 (client-server) • 4 (peer-to-peer)
`behavior <id>`	Application behavior filter: • all • 2 (botnet) • 3 (evasive) • 5 (excessive bandwidth) • 6 (tunneling) • 9 (cloud)
`popularity <integer>`	Application popularity filter (1 - 5, from least to most popular).
`action {pass | block | reset}`	Pass/block traffic or reset the connection for traffic from this application (default = block).
`log {enable | disable}`	Enable/disable logging for this application list (default = enable).

Excluding signatures in application control profiles

In an application control list, the exclusion option allows users to specify a list of applications they wish to exclude from an entry filtered by category, technology, or others. By excluding the signature, the application is no longer processed on the entry in which it is excluded, but may match subsequent entries that exist.

To configure signature exclusion:

```
config application list
    edit <name>
        config entries
            edit <id>
                set category <id>
                set exclusion <application id>
                set action {pass | block | reset}
            next
        end
    next
end
```

Sample configurations

In the following example, category 23 (social media) is blocked in the entries, and signature 34527 (Instagram) is excluded from this entry. Traffic to Instagram will pass because the signature is removed from entry 1 and the action of `other-application-action` is set to `pass`.

To configure signature exclusion:

```
config application list
    edit "test"
        set other-application-action pass
        set unknown-application-action pass
        set other-application-log enable
        set unknown-application-log enable
            config entries
                edit 1
                    set category 23
                    set exclusion 34527
                    set action block
                next
            end
    next
end
```

In the following example, entry 1 is configured so that category 23 (social media) is set to pass and signature 34527 (Instagram) is excluded. In entry 2, application 34527 (Instagram) is blocked, so the traffic to Instagram will be blocked, even though it is excluded in entry 1. Traffic to other signatures in category 23, such as Facebook, will still pass.

To configure signature exclusion:

```
config application list
    edit "test"
        set other-application-action pass
        set unknown-application-action pass
        set other-application-log enable
        set unknown-application-log enable
            config entries
                edit 1
```

```
                    set category 23
                    set exclusion 34527
                    set action pass
                next
                edit 2
                    set application 34527
                    set action block
                next
            end
    next
end
```

In the following example, an explicit proxy is behind the FortiGate with an excluded signature for 107347980 (Proxy.HTTP) and category 6 (proxy) is set to block. The client will allow normal proxy traffic to pass, but it will discard all proxy application traffic (such as KProxy, Tor, and so on).

To configure signature exclusion:

```
config application list
    edit "test"
        set other-application-action pass
        set unknown-application-action pass
        set other-application-log enable
        set unknown-application-log enable
            config entries
                edit 1
                    set category 6
                    set exclusion 107347980
                    set action block
                next
            end
    next
end
```

Port enforcement check

Most networking applications run on specific ports. For example, SSH runs on port 22, and Facebook runs on ports 80 and 443.

If the default network service is enabled in the application control profile, a port enforcement check is done at the application profile level, and any detected application signatures running on the non-standard TCP/IP port are blocked. This means that each allowed application runs on its default port.

To configure port enforcement check:

```
config application list
    edit <name>
        set enforce-default-app-port enable
        config entries
            edit 1
                set application 15896
                set action pass
            next
        end
```

```
    next
end
```

For example, when applying this application control sensor, FTP traffic (application 15896) with the standard port (port 21) is allowed, while the non-standard port (port 2121) is blocked.

Protocol enforcement

Protocol enforcement allows you to configure networking services (e.g. FTP, HTTP, HTTPS) on known ports (e.g. 21, 80, 443). For protocols that are not allowlisted under select ports, the IPS engine performs the violation action to block, allow, or monitor that traffic.

This feature can be used in the following scenarios:

- When one protocol dissector confirms the service of network traffic, protocol enforcement can check whether the confirmed service is allowlisted under the server port. If it is not allowlisted, the traffic is considered a violation and IPS can take the action specified in the configuration (block or monitor it).
- When there is no confirmed service for the network traffic, the traffic is considered a service violation if IPS dissectors rule out all of the services enforced under its server port.

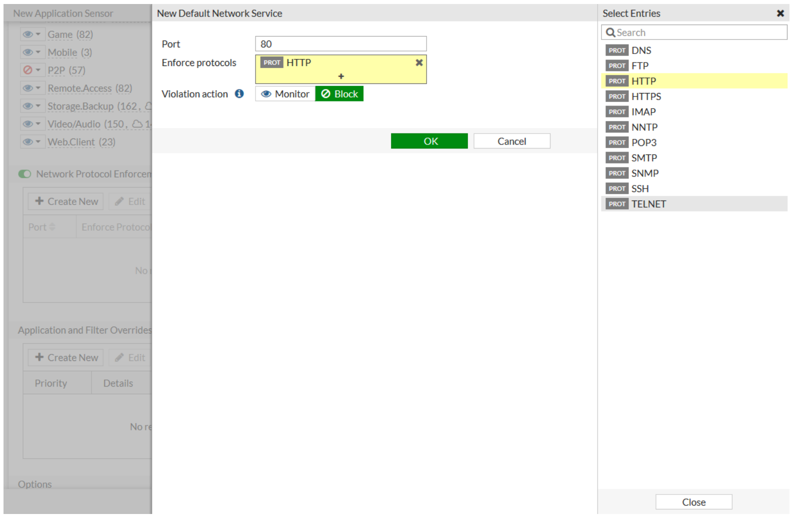
In an applicable profile, a default network service list can be created to associate well known ports with accepted services.

In the following example, an application sensor is configured to enforce HTTP on port 80 (block), and DNS on port 53 (monitor).

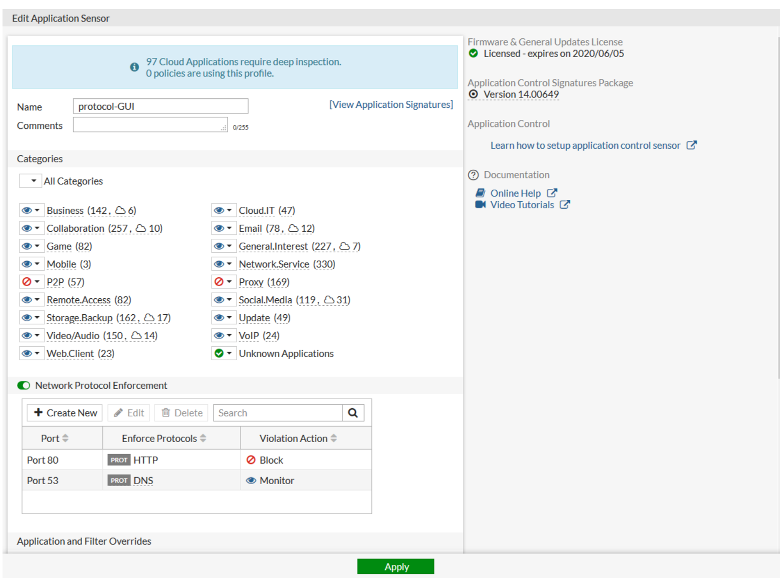
To configure protocol enforcement in the GUI:

1. Go to *Security Profiles > Application Control*.
2. Create a new application sensor or edit an existing one.
3. Enable *Network Protocol Enforcement*.
   Enforcement entries can be created, edited, or deleted to configure network services on certain ports and determine the violation action.
4. In the *Network Protocol Enforcement* table, click *Create New*.
5. Configure the entry for HTTP:
   a. For *Port*, enter 80.
   b. For *Enforced protocols*, select *HTTP*.
   c. For *Violation action*, select *Block*.

d. Click *OK*.



6. Configure the entry for DNS:
   a. Click *Create New*, then for *Port*, enter 53.
   b. For *Enforced protocols*, select *DNS*.
   c. For *Violation action*, select *Monitor*.
   d. Click *OK*.
   The entries are displayed in the table.



7. Click *OK*.

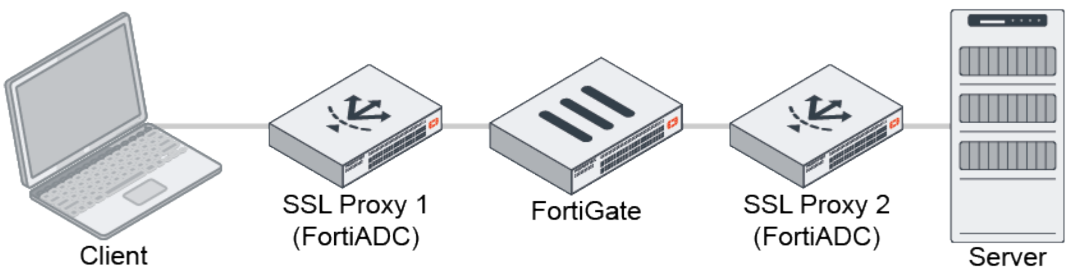
To configure protocol enforcement in the CLI:

```
config application list
    edit "protocol-GUI"
        set other-application-log enable
        set control-default-network-services enable
        config default-network-services
            edit 1
```

```
                set port 80
                set services http
                set violation-action block
            next
            edit 2
                set port 53
                set services dns
                set violation-action monitor
            next
        end
    next
end
```

SSL-based application detection over decrypted traffic in a sandwich topology

When a FortiGate is sandwiched between SSL encryption and decryption devices, the FortiGate can process the decrypted traffic that passes between those devices. This feature adds support for decrypted traffic in application control. In some pre-defined signatures, the signature is pre-marked with the *require_ssl_di* tag. The `force-inclusion-ssl-di-sigs` option under `application list` allows users to control the inspection of dissected traffic. When this option is enabled, the IPS engine forces the pre-marked SSL-based signatures to be applied to the decrypted traffic of the respective applications. In the following topology, SSL Proxy 1 handles the client connection and SSL Proxy 2 handles the server connection, leaving the content unencrypted as traffic passes through the FortiGate.



To configure SSL-based application detection over decrypted traffic:

```
config application list
    edit "test"
        set force-inclusion-ssl-di-sigs {enable | disable}
    next
end
```

Example pre-marked SSL-based signature:

F-SBID(--vuln_id 15722; --attack_id 42985; --name "Facebook_Chat"; --group im; --protocol tcp; --default_action pass; --revision 4446; --app_cat 23; --vendor 3; --technology 1; --behavior 9; --pop 4; --risk 2; --language "Multiple"; --weight 20; --depend-on 15832; --depend-on 38468; ***--require_ssl_di "Yes"****; --casi 1; --casi 8; --parent 15832; --app_port "TCP/443"; --severity info; --status hidden; --service http; --flow from_client; --pattern "/pull?"; --context uri; --no_case; --pattern ".facebook.com"; --context host; --no_case; --tag set,Tag.Facebook.Pull; --tag quiet; --scan-range 10m,all; --date 20190301;)*

All signatures that include the *require_ssl_di* tag are pre-defined and cannot be customized.

Matching multiple parameters on application control signatures

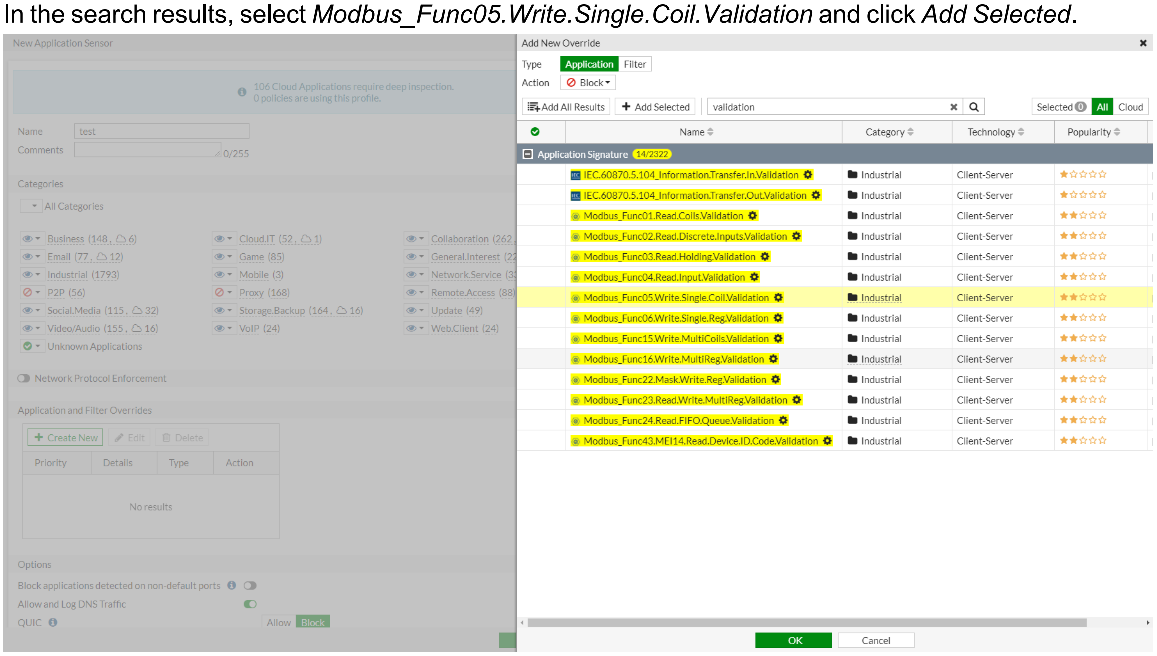
Application control signatures that support parameters (such as SCADA protocols) can have multiple parameters grouped together and matched at the same time. Multiple application parameter groups can be added to an override. Traffic will be flagged if it matches at least one parameter group.

This example uses the Modbus_Func05.Write.Single.Coil.Validation signature. This is an industrial signature, so ensure that no signatures are excluded:

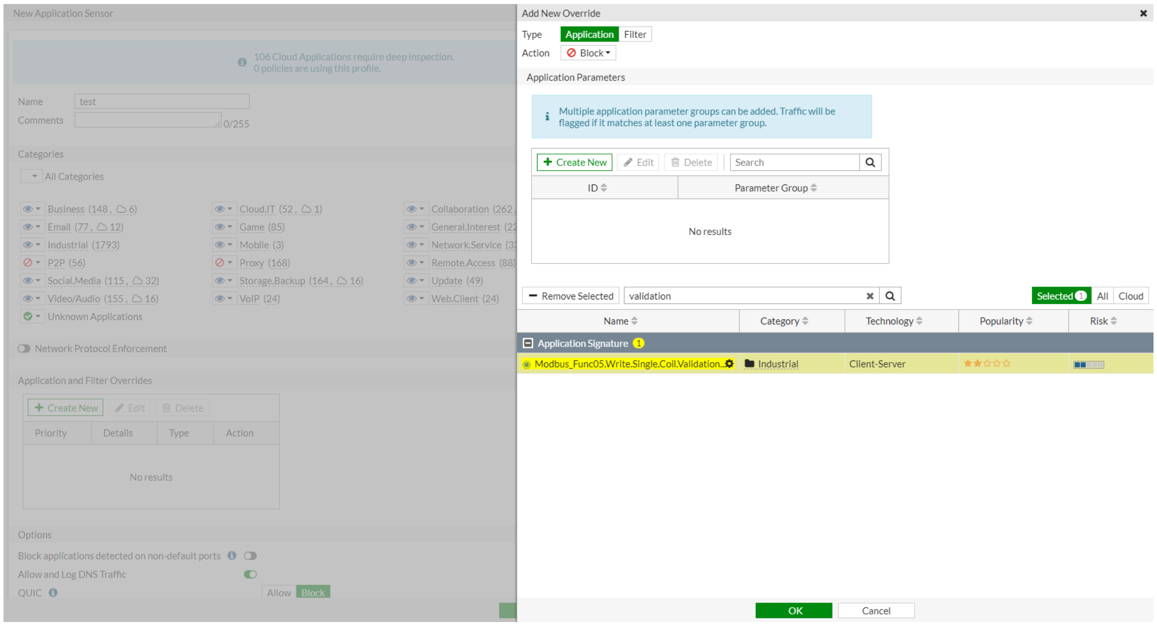
```
config ips global
    set exclude-signatures none
end
```

To configure an application sensor with multiple parameters in the GUI:

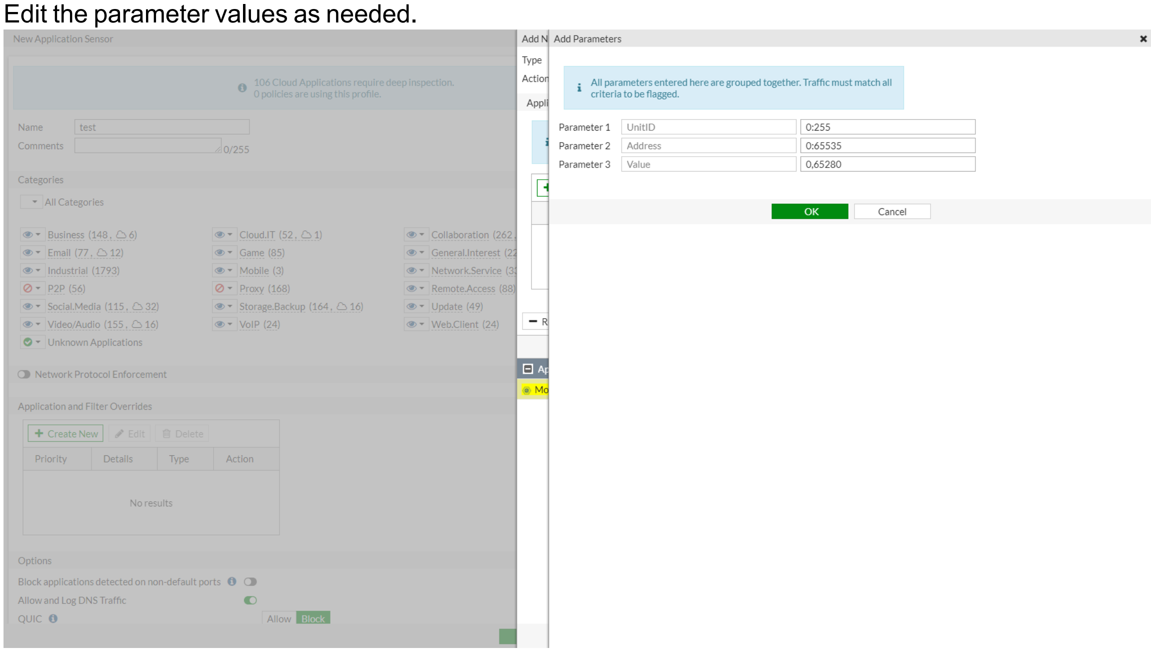
1. Go to *Security Profiles > Application Control* and click *Create New*, or edit an existing sensor.
2. In the *Application and Filter Overrides* table, click *Create New*.
3. Search for *Modbus_Func05.Write.Single.Coil.Validation* and press `Enter`. A gear icon beside the signature name indicates it has configurable application parameters.
4. In the search results, select *Modbus_Func05.Write.Single.Coil.Validation* and click *Add Selected*.



5. Click the *Selected* tab. In the *Application Parameters* section, click *Create New*.



6. Edit the parameter values as needed.



7. Click *OK*.
8. Add more signatures if needed.
9. Click *OK*.

To configure an application sensor with multiple parameters in the CLI:

```
config application list
    edit "test"
        set other-application-log enable
        config entries
            edit 1
```

```
                set application 48885
                config parameters
                    edit 1
                        config members
                            edit 1
                                set name "UnitID"
                                set value "0:255"
                            next
                            edit 2
                                set name "Address"
                                set value "0:65535"
                            next
                            edit 3
                                set name "Value"
                                set value "0,65280"
                            next
                        end
                    next
                end
            next
            edit 2
                set category 2 6
            next
        end
    next
end
```

Application signature dissector for DNP3

The DNP3 application signature dissector supports detecting DNP3 traffic that is encapsulated by the RealPort protocol (Net.CX). DNP3 is used in industrial solutions over serial ports, USB ports, printers, and so on. RealPort encapsulation allows transportation of the underlying protocols over TCP/IP. The FortiGate industrial signatures must be enabled to use RealPort.DNP3 signatures:

```
config ips global
    set exclude-signatures none
end
```

IPS engine version 7.0015 and later supports RealPort.DNP3 dissectors.

Sample logs

```
119: date=2021-03-09 time=18:56:35 eventtime=1615344995698958507 tz="-0800"
logid="1059028704" type="utm" subtype="app-ctrl" eventtype="signature" level="information"
vd="vd1" appid=49890 srcip=10.1.100.191 dstip=172.16.200.159 srcport=43946 dstport=771
srcintf="port10" srcintfrole="undefined" dstintf="port9" dstintfrole="undefined" proto=6
service="RLDNP3" direction="incoming" policyid=1 sessionid=1204 applist="test" action="pass"
appcat="Industrial" app="RealPort.DNP3" incidentserialno=88083610 msg="Industrial:
RealPort.DNP3," apprisk="elevated"
```

```
1: date=2021-03-09 time=18:56:08 eventtime=1615344968811546102 tz="-0800" logid="1059028704"
type="utm" subtype="app-ctrl" eventtype="signature" level="information" vd="vd1" appid=49899
srcip=10.1.100.191 dstip=172.16.200.159 srcport=43946 dstport=771 srcintf="port10"
srcintfrole="undefined" dstintf="port9" dstintfrole="undefined" proto=6 service="RLDNP3"
```

```
direction="outgoing" policyid=1 sessionid=1204 applist="test" action="pass"
appcat="Industrial" app="RealPort.DNP3_Confirm" incidentserialno=88083404 msg="Industrial:
RealPort.DNP3_Confirm," clouduser="34 -> 34" filename="Null" apprisk="elevated"
cloudaction="others"
```

Intrusion prevention

Intrusion Prevention System (IPS) detects network attacks and prevents threats from compromising the network, including protected devices. IPS can be in the form of a standalone appliance, or part of the feature set of a Next Generation Firewall (NGFW), such as FortiGate. IPS utilizes signatures, protocol decoders, heuristics (or behavioral monitoring), threat intelligence (such as FortiGuard Labs), and advanced threat detection in order to prevent exploitation of known and unknown zero-day threats. FortiGate IPS is even capable of performing deep packet inspection to scan encrypted payloads in order to detect and prevent threats from attackers.

Networks and devices are often exploited through vulnerabilities. Software vulnerabilities are one such example where a bug or inherent weakness in the code provides attackers an opportunity to gain access to the software. More severe vulnerabilities allow unauthorized access, data leakage, and execution of malicious code. Exploitation of these vulnerabilities can cause damage to the machine and infect others. While the best solution is to patch vulnerabilities as soon as patches are available, IPS signatures offer a solution to detect and block exploitation of many vulnerabilities before they enter the network.

IPS signatures

Fortinet's solution combines industry-leading threat intelligence from FortiGuard Labs with the FortiGate NGFW to identify the latest threats and prevent them from entering your network. IPS signatures are one such method for delivering the latest protection. FortiGuard Labs uses AI and Machine Learning (ML) to analyze billions of events every day. The FortiGuard Labs research team also proactively performs threat research to discover new vulnerabilities and exploitation, and produces signatures to identify such threats. These IPS signatures are delivered to each FortiGate daily, so that the IPS engine is armed with the latest databases to match the latest threats.

IPS sensors

A FortiGate IPS sensor is a collection of IPS signatures and filters that define the scope of what the IPS engine will scan when the IPS sensor is applied. An IPS sensor can have multiple sets of signatures and/or filters. A set of IPS signatures consists of manually selected signatures, while a set of IPS filters consists of filters based on signature attributes like target, severity, protocol, OS, and application. Each signature has predefined attributes and an action, such as block, allow, monitor (pass), quarantine, and reset. It is also possible to create custom IPS signatures to apply to an IPS sensor.

From the *Security Profiles > Intrusion Prevention* pane, you can create new IPS sensors and view a list of predefined sensors.

FortiOS includes the following predefined IPS sensors with associated predefined signatures:

Predefined IPS sensors	Description
all_default	Filters all predefined signatures, and sets action to the signature's default action.
all_default_pass	Filters all predefined signatures, and sets action to pass/monitor.

Predefined IPS sensors	Description
default	Filters all predefined signatures with severity of Critical/High/Medium. Sets action to signature's default action.
high_security	Filters all predefined signatures with severity of Critical/High/Medium, and sets action to Block. For Low severity signatures, sets action to signature's default action.
protect_client	Protects against client-side vulnerabilities by filtering on `Target=Client`. Sets action to signature's default action.
protect_email_server	Protects against email server-side vulnerabilities by filtering on `Target=Server` and `Protocol=IMAP`, `POP3` or `SMTP`. Sets action to signature's default action.
protect_http_server	Protects against HTTP server-side vulnerabilities by filtering on `Target=Server` and `Protocol=HTTP`. Sets action to signature's default action.
wifi-default	Filters all predefined signatures with severity of Critical/High/Medium. Sets action to signature's default action. Used in profile for offloading WiFi traffic.

DDoS attacks

Besides protecting against threats and exploitation of vulnerabilities, the IPS engine is also responsible for mitigating Denial of Service (DoS) attacks where attackers attempt to bring a service down by flooding the target with traffic from distributed systems. Using anomaly-based defense, FortiGate can detect a variety of L3 and L4 anomalies and take action against these attacks. This can be configured under IPv4 and IPv6 DoS Policies, which is discussed in detail under DoS policy on page 790.

This section contains the following topics:

- Signature-based defense on page 1204
- IPS configuration options on page 1208

This section also provides the following examples about IPS sensors:

- IPS signature filter options on page 1212
- IPS with botnet C&C IP blocking on page 1216
- IPS signatures for the industrial security service on page 1220
- IPS sensor for IEC 61850 MMS protocol on page 1221
- SCTP filtering capabilities on page 1223

Signature-based defense

Signature-based defense is used against known attacks or vulnerability exploits. These often involve an attacker attempting to gain access to your network. The attacker must communicate with the host in an attempt to gain access, and this communication includes commands or sequences of commands and variables. The IPS signatures include these command sequences, allowing the FortiGate unit to detect and stop the attack.

This section describes the following components used in signature-based defense:

- IPS signatures on page 1205
- Protocol decoders on page 1205
- IPS engine on page 1205

- IPS sensors on page 1205
- IPS filters on page 1206
- Custom and predefined signature entries on page 1207
- Policies on page 1208

IPS signatures

IPS signatures are the basis of signature-based intrusion prevention. Every attack can be reduced to a particular string of commands or a sequence of commands and variables. Signatures include this information, and FortiGate uses the information to detect and stop attacks.

Signatures also include characteristics about the attack they describe. These characteristics include the network protocol associated with the attack, the vulnerable operating system, and the vulnerable application.

To view the complete list of signatures, go to *Security Profiles > IPS Signatures*. The list of signatures includes predefined and custom signatures. You can hover over the name of the IPS signature to display a pop-up window that includes an ID number. You can click the ID number to display the FortiGuard page.

Protocol decoders

Before examining network traffic for attacks, the IPS engine uses protocol decoders to identify each protocol appearing in the traffic. Attacks are protocol-specific, so your FortiGate unit conserves resources by looking for attacks only in the protocols used to transmit them. For example, the FortiGate unit will only examine HTTP traffic for the presence of a signature describing an HTTP attack.

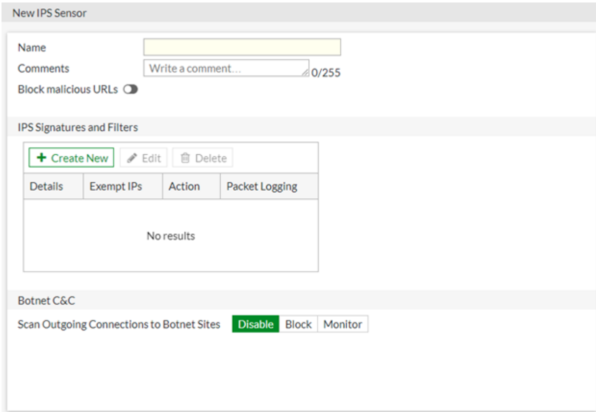
IPS engine

Once the protocol decoders separate the network traffic by protocol, the IPS engine examines the network traffic for the attack signatures by using IPS sensors.

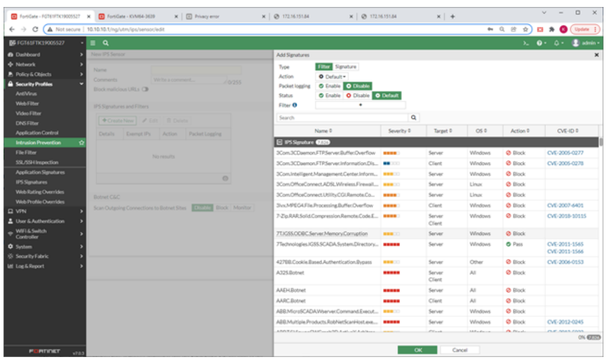
IPS sensors

The IPS engine does not examine network traffic for all signatures. The IPS engine examines network traffic for signatures specified in IPS sensors. You must first create an IPS sensor, and then you can specify what signatures the IPS sensor will use. You can add individual signatures to IPS sensors, or you can add filters to IPS sensors, and the filters automatically include the applicable signatures.

To view IPS sensors, go to *Security Profiles > Intrusion Prevention*. To create a new sensor, click *Create New*.



An IPS sensor is composed of IPS signatures and filters. Under *IPS Signatures and Filters*, click *Create New* to create a set of IPS signatures or a set of IPS filters.



You can create IPS sensors for specific types of traffic, and then select the IPS sensors in firewall policies designed to handle the same type of traffic. For example, you can specify all of the web-server related signatures in an IPS sensor, and select the IPS sensor in a firewall policy that controls all traffic to and from a web server that is protected by the FortiGate unit.

The FortiGuard Service periodically adds new predefined signatures to counter new threats. New predefined signatures are automatically included in IPS sensors that are configured to use filters when the new signatures match existing filter specifications. For example, if you have an IPS sensor with a filter that includes all signatures for the Windows operating system, your filter will automatically incorporate new Windows signatures that the FortiGuard Service adds to the database.

IPS signature and filter entries are checked from top down. When a signature is found in a set of signatures or filters, the action defined for the signature is taken.

IPS filters

IPS sensors can contain one or more IPS filters. A filter is a collection of signature attributes that you specify. The signatures that have all of the attributes specified in a filter are included in the IPS filter.

Following are the attribute groups:

- Target
- Severity
- Protocol

- OS
- Application

Starting in FortiOS 6.4.2, you can also filter by CVE ID or CVE pattern by using the CLI. See FortiOS 6.4 New Features > IPS signature filter options.

When selecting multiple attributes within the same group, the selections are combined by using a logical `OR`. When selecting multiple attributes between attribute groups, each attribute group is combined by using a logical `AND`.

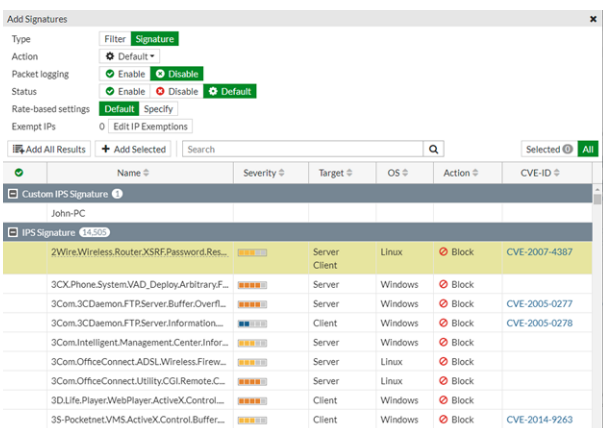
Once you select filters in the GUI, the filtered list of IPS signatures are displayed. Adjust your filters accordingly to construct a suitable list for your needs.

For example, if your FortiGate unit protects a Linux server running the Apache web server software, you could create a new filter to protect it. By setting *OS* filter attribute to *Linux*, and the filter attribute *Application* to *Apache*, the filter will include only the signatures that apply to both Linux and Apache. If you wanted to scan for all the Linux signatures and all the Apache signatures, you would create two filters, one for each.

To view the filters in an IPS sensor, go to *Security Profiles > Intrusion Prevention*, select the IPS sensor, and click *Edit*.

Custom and predefined signature entries

Signature entries allow you to add individual, custom or predefined IPS signatures to an IPS sensor. If you need only one signature, or you want to manually select multiple signatures that don't fall into the criteria for an IPS filter, adding a signature entry to an IPS sensor is the easiest way. Signature entries are also the only way to include custom signatures in an IPS sensor.



To select an individual signature, click a signature, and select *Add Selected*. The signature moves to the *Selected* list.

To select multiple signatures, use the *Search* bar to perform a keyword search, and then click *Add All Results* to move all entries to the *Selected* list.

Overriding the default action

Each IPS signature comes with a default action such as *Block* and *Pass*. In some scenarios, you may want to override this action. You can override a set of IPS filter or signatures. By default, a set of IPS filter or signatures has an action of *Default*, which applies a signature's default action when the signature is matched. By changing the action, you can override the setting for all signatures within the filter or signature set.

Policies

You must select an IPS sensor in a security policy or an interface policy to apply the IPS sensor to traffic. An IPS sensor that it not selected in a policy is not applied to network traffic.

IPS configuration options

Besides configuring an IPS filter or selecting IPS signatures for an IPS sensor, you can configure additional IPS options for each sensor or globally for all sensors. This topic introduces the following available configuration options:

- Malicious URL database for drive-by exploits detection on page 1208
- IPS signature rate count threshold on page 1208
- Botnet C&C on page 1209
- Hardware acceleration for flow-based security profiles (NTurbo and IPSA) on page 1209
- Extended IPS database on page 1210
- IPS engine-count on page 1210
- Industrial signature database on page 1210
- Fail-open on page 1211
- IPS buffer size on page 1211
- Session count accuracy on page 1211
- Protocol decoders on page 1211

Malicious URL database for drive-by exploits detection

This feature uses a local malicious URL database on the FortiGate to assist in detection of drive-by exploits, such as adware that allows automatic downloading of a malicious file when a page loads without the user's detection. The database contains all malicious URLs active in the last one month, and all drive-by exploit URLs active in the last three months. The number of URLs controlled are in the one million range.

This feature can be enabled from a IPS Sensor in the GUI by going to *Security Profiles > Intrusion Prevention* and editing or creating an IPS Sensor. Then enable *Block malicious URLs*.

From the CLI:

```
config ips sensor
   edit <profile>
      set block-malicious-url [enable | disable]
   next
end
```

Blocking malicious URLs is not supported on some FortiGate models, such as FortiGate 51E, 50E, or 30E.

IPS signature rate count threshold

You can use the IPS signature rate-based settings to specify a rate count threshold that must be met before the signature is triggered. A rate count threshold provides a more controlled recording of attack activity. For example, if multiple login attempts produce a failed result over a short period of time, then an alert would be sent and traffic might be blocked, which is a more manageable response than sending an alert every time a login fails.

This can be configured from the GUI by going to *Security Profiles > Intrusion Prevention*. Create or edit an IPS sensor. Within the sensor, edit the IPS signatures and filters. Only IPS signatures have the rate-based settings option. IPS filters do not.

Some settings are only available from CLI.

The syntax for this configuration is as follows:

```
config ips sensor
   edit default
      config entries
         edit <Filter ID number>
            set rule <*id>
            set rate-count <integer between 1 - 65535>
            set rate-duration <integer between 1 - 65535>
```

The value of the rate-duration is an integer for the time in seconds.

```
set rate-mode <continuous | periodical>
```

The rate-mode refers to how the count threshold is met.

If the setting is "`continuous`", and the action is set to block, the action is engaged as soon as the rate-count is reached. For example, if the count is 10, the traffic would be blocked as soon as the signature is triggered 10 times.

If the setting is "`periodical`", the FortiGate allows up to the value of the rate-count incidents where the signature is triggered during the rate-duration. For example, if the rate count is 100 and the duration is 60, the signature would need to be triggered 100 times in 60 seconds for the action to be engaged.

```
set rate-track <dest-ip | dhcp-client-mac | dns-domain | none | src-ip>
```

This setting allows the tracking of one of the protocol fields within the packet.

Botnet C&C

See IPS with botnet C&C IP blocking on page 1216.

Hardware acceleration for flow-based security profiles (NTurbo and IPSA)

Some FortiGate models support a feature call NTurbo that can offload flow-based firewall sessions to network processors. See also Hardware Acceleration > NTurbo offloads flow-based processing. For IPSA enhanced pattern matching, see Hardware Acceleration > IPSA offloads flow-based advanced pattern matching.

Some FortiGate models also support offloading enhanced pattern matching for flow-based security profiles to CP8 or CP9 content processors. You can use the following command to configure NTurbo and IPSA:

```
config ips global
   set np-accel-mode {none | basic}
   set cp-accel-mode {none | basic | advanced}
end
```

If the `np-accel-mode` option is available, your FortiGate supports NTurbo. The `none` option disables NTurbo, and `basic` (the default) enables NTurbo.

If the `cp-accel-mode` option is available, your FortiGate supports IPSA. The `none` option disables IPSA, and `basic` enables basic IPSA, and `advanced` enables enhanced IPSA, which can offload more types of pattern matching than basic IPSA. The `advanced` option is only available on FortiGate models with two or more CP8 processors, or one or more CP9 processors.

Extended IPS database

Some models have access to an extended IPS Database. Because the extended database may affect FortiGate performance, the extended database package may be disabled by default on some models, such as desktop models.

You can only enable the extended IPS database by using the CLI.

```
config ips global
   set database extended
end
```

FortiGate models with the CP9 SPU receive the IPS full extended database, and the other physical FortiGate models receive a slim version of the extended database. The slim-extended database is a smaller version of the full extended database that contains top active IPS signatures. It is designed for customers who prefer performance.

Customers with non-CP9 SPU models need to upgrade to a CP9 SPU model (physical FortiGate) in order to get full IPS signature coverage. All FortiGate models 200 (E and F) and higher have a CP9 SPU.

See Determining the content processor in your FortiGate unit in the FortiOS Hardware Acceleration Guide to check if your device has a CP9 SPU.

IPS engine-count

FortiGate units with multiple processors can run one or more IPS engine concurrently. The engine-count CLI command allows you to specify how many IPS engines to use at the same time:

```
config ips global
   set engine-count <int>
end
```

The recommended and default setting is `0`, which allows the FortiGate unit to determine the optimum number of IPS engines.

Industrial signature database

Industrial signatures are defined to protect Industrial Control Systems (ICS), Operational Technology (OT) and SCADA systems, which are critical infrastructure used by manufacturing industries. These signatures are enabled by default, but can be configured by using the following CLI:

```
config ips global
   set exclude-signatures {none* | industrial}
end
```

Fail-open

A fail-open scenario is triggered when IPS raw socket buffer is full. Therefore IPS engine has no space in memory to create more sessions and needs to decide whether to drop the sessions or bypass the sessions without inspection.

```
config ips global
   set fail-open {enable | disable}
end
```

The default setting is `disable`, so sessions are dropped by IPS engine when the system enters fail-open mode.

When enabled, the IPS engine fails open, and it affects all protocols inspected by FortiOS IPS protocol decoders, including but not limited to HTTP, HTTPS, FTP, SMTP, POP3, IMAP, and so on. When the IPS engine fails open, traffic continues to flow without IPS scanning.

Sessions offloaded to Nturbo do not support fail-open. When Nturbo data path is overloaded, traffic is dropped regardless of fail-open setting.

IPS buffer size

If system enters fail-open mode frequently, it is possible to increase the IPS socket buffer size to allow more data buffering, which reduces the chances of overloading the IPS engine. You can set the size of the IPS buffer.

```
config ips global
   set socket-size <int>
end
```

The default socket size and maximum configurable value varies by model. In short, socket-size determines how much data the kernel passes to the IPS engine each time the engine samples packets.

Take caution when modifying the default value. If the socket-size is too large, the higher memory used by the IPS engine may cause the system to enter conserve mode more frequently. If set too low, the system may enter IPS fail-open mode too frequently.

Session count accuracy

The IPS engine can track the number of open session in two ways. An accurate count uses more resources than a less accurate heuristic count.

```
config ips global
   set session-limit-mode {accurate | heuristic}
end
```

The default is `heuristic`.

Protocol decoders

The FortiGate Intrusion Prevention system uses protocol decoders to identify the abnormal traffic patterns that do not meet the protocol requirements and standards. For example, the HTTP decoder monitors traffic to identify any HTTP packets that do not meet the HTTP protocol standards.

To change the ports a decoder examines, you must use the CLI. In this example, the ports examined by the DNS decoder are changed from the default 53 to 100, 200, and 300.

```
config ips decoder dns_decoder
     config parameter "port_list"
     set value "100,200,300"
   end
end
```

You cannot assign specific ports to decoders that are set to *auto* by default. These decoders can detect their traffic on any port. Specifying individual ports is not necessary.

IPS signature filter options

IPS signature filter options include hold time, CVE pattern, and IPS sensor attributes.

Hold time

The hold time option allows you to set the amount of time that signatures are held after a FortiGuard IPS signature update per VDOM. During the holding period, the signature's mode is monitor. The new signatures are enabled after the hold time to avoid false positives.

The hold time can be from 0 days and 0 hours (default) up to 7 days, in the format `##d##h`.

To configure the amount of time to hold and monitor IPS signatures:

```
config system ips
    set signature-hold-time 3d12h
    set override-signature-hold-by-id enable
end
```

When a signature that is on hold is matched, the log will include the message `signature is on hold`:

```
date=2010-07-06 time=00:00:57 logid="0419016384" type="utm" subtype="ips"
eventtype="signature" level="alert" vd="vd1" eventtime=1278399657778481842 tz="-0700"
severity="info" srcip=10.1.100.22 srccountry="Reserved" dstip=172.16.200.55 srcintf="port13"
srcintfrole="undefined" dstintf="port14" dstintfrole="undefined" sessionid=3620
action="detected" proto=6 service="HTTP" policyid=1 attack="Eicar.Virus.Test.File"
srcport=52170 dstport=80 hostname="172.16.200.55" url="/virus/eicar" direction="incoming"
attackid=29844 profile="test" ref="http://www.fortinet.com/ids/VID29844"
incidentserialno=25165825 msg="file_transfer: Eicar.Virus.Test.File, (signature is on hold)"
```

To view signatures being held by rule ID 29844 on the VDOM:

```
# diagnose ips signature on-hold vd1 29844
Rule: 29844, attack_id: 58886, last updated: 20170411
Rule: 29844, attack_id: 59517, last updated: 20170411
Rule: 29844, attack_id: 60105, last updated: 20170411
...
```

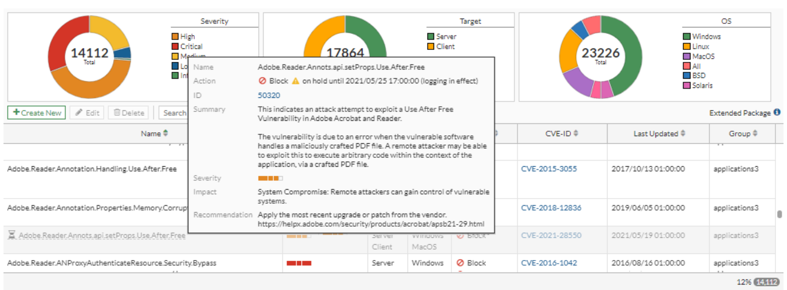
To view all help signatures on the VDOM:

```
# diagnose ips signature on-hold vd1
Rule: 17541, attack_id: 20899, last updated: 20140423
Rule: 17557, attack_id: 20934, last updated: 20140423
Rule: 17559, attack_id: 20932, last updated: 20140423
Rule: 17560, attack_id: 20933, last updated: 20140423
Rule: 17562, attack_id: 20928, last updated: 20170908
Rule: 17677, attack_id: 21187, last updated: 20171106
Rule: 17713, attack_id: 43756, last updated: 20140424
Rule: 17759, attack_id: 21298, last updated: 20140423
...
```

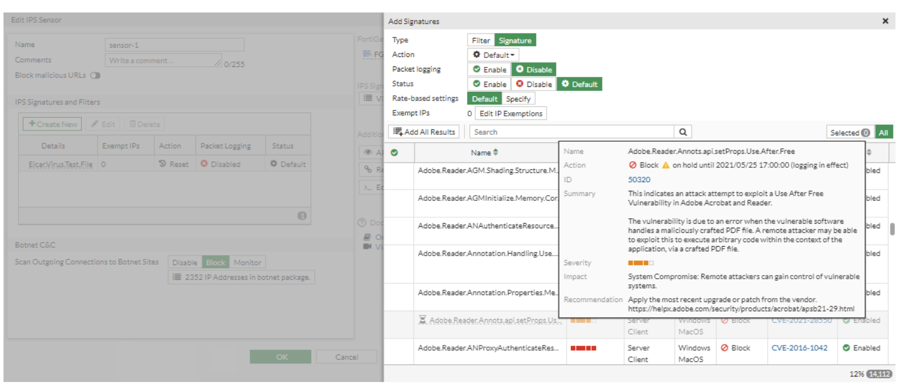
Viewing on hold information in the GUI

On hold signatures are grayed out in the GUI with an hourglass icon beside the signature name. A tooltip displays the on hold expiry time and other details.

On the *Security Profiles > IPS Signatures* page, for example, the *Adobe.Reader.Annots.api.setProps.Use.After.Free* signature is on hold. Hover over the grayed-out entry to view the tooltip, which includes the action and hold time expiry. On this page, all on hold signatures are displayed as on hold regardless of whether `override-signature-hold-by-id` is enabled.



The same tooltip is available on the *Edit IPS Sensor* (*Security Profiles > Intrusion Prevention*) page when creating or editing the IPS signatures. In the *Add Signatures* pane when the *Type* is *Signature*, signatures on hold are only displayed as on hold if `override-signature-hold-by-id` is enabled.



You can still use on hold signatures in an IPS sensor profile; however, the profile will not block matching traffic. It will monitor it instead (logging in effect) until the on hold time expires.

CVE pattern

The CVE pattern option allows you to filter IPS signatures based on CVE IDs or with a CVE wildcard, ensuring that any signatures tagged with that CVE are automatically included.

To configure CVE patterns for CVE-2010-0177 and all CVE-2017 CVEs:

```
config ips sensor
    edit "cve"
        set comment "cve"
        config entries
            edit 1
                set cve "cve-2010-0177"
                set status enable
                set log-packet enable
                set action block
            next
            edit 2
                set cve "cve-2017"
                set action reset
            next
        end
    next
end
```

For example, the CVE of the IPS signature *Mozilla.Firefox.PluginArray.NsMimeType.Code.Execution* is CVE-2010-0177. This matches the CVE filter in the IPS sensor, so traffic is blocked and logged:

```
date=2020-07-13 time=15:44:56 logid="0419016384" type="utm" subtype="ips"
eventtype="signature" level="alert" vd="vd1" eventtime=1594593896666145871 tz="-0700"
severity="critical" srcip=10.1.100.22 srccountry="Reserved" dstip=172.16.200.55
srcintf="port2" srcintfrole="undefined" dstintf="port1" dstintfrole="undefined"
sessionid=1638 action="dropped" proto=6 service="HTTPS" policyid=1
attack="Mozilla.Firefox.PluginArray.NsMimeType.Code.Execution" srcport=58298 dstport=443
hostname="172.16.200.55" url="/Mozilla" direction="incoming" attackid=20853 profile="sensor-
1" ref="http://www.fortinet.com/ids/VID20853" incidentserialno=124780667 msg="web_client:
Mozilla.Firefox.PluginArray.NsMimeType.Code.Execution," crscore=50 craction=4096
crlevel="critical"
```

IPS sensor attributes

When configuring IPS sensor profiles, IPS signatures can be filtered based on the attributes: default status, default action, vulnerability type, and the last update date. When monitoring the specific, filtered signatures, logs are not generated for other, irrelevant signatures.

This avoids generating a lot of false positives due to many signatures having the pass action, which is never logged.

To configure filters in an IPS sensor profile:

```
config ips sensor
    edit "test_default"
        config entries
            edit 1
                set default-action pass
```

```
                set default-status enable
                set vuln-type 12
                set last-modified before 2020/02/02
            next
        end
    next
end
```

`default-action {pass	block	all}`	Filter by signatures' default actions (default = all).
`default-status {enable	disable	all}`	Filter by signatures' default statuses (default = all).
`vuln-type <integer> ... <integer>`	Filter by signatures' vulnerability types.
`last-modified {before	after	between} <date> [end-date]`	Filter by signatures' last modified date (default = before 00/00/00). The date format is `yyyy/mm/dd`. The year range is 2001 - 2050.

When the IPS profile is used in a firewall profile and then the EICAR virus test file signature is triggered, the signature matches the values set in the filter and logs are generated:

```
1:date=2022-02-15 time=14:07:03 eventtime=1644962823303491048 tz="-0800" logid="0419016384"
type="utm" subtype="ips" eventtype="signature" level="alert" vd="vd1" severity="info"
srcip=10.1.100.11 srccountry="Reserved" dstip=172.16.200.55 dstcountry="Reserved"
srcintf="port38" srcintfrole="undefined" dstintf="port37" dstintfrole="undefined"
sessionid=1171 action="detected" proto=6 service="HTTP" policyid=1 poluuid="623d2d28-8ea7-
51ec-00ef-7549685a77c2" policytype="policy" attack="Eicar.Virus.Test.File" srcport=47230
dstport=80 hostname="172.16.200.55" url="/virus/eicar" direction="incoming" attackid=29844
profile="test_default" ref="http://www.fortinet.com/ids/VID29844" incidentserialno=103809025
msg="file_transfer: Eicar.Virus.Test.File"
```

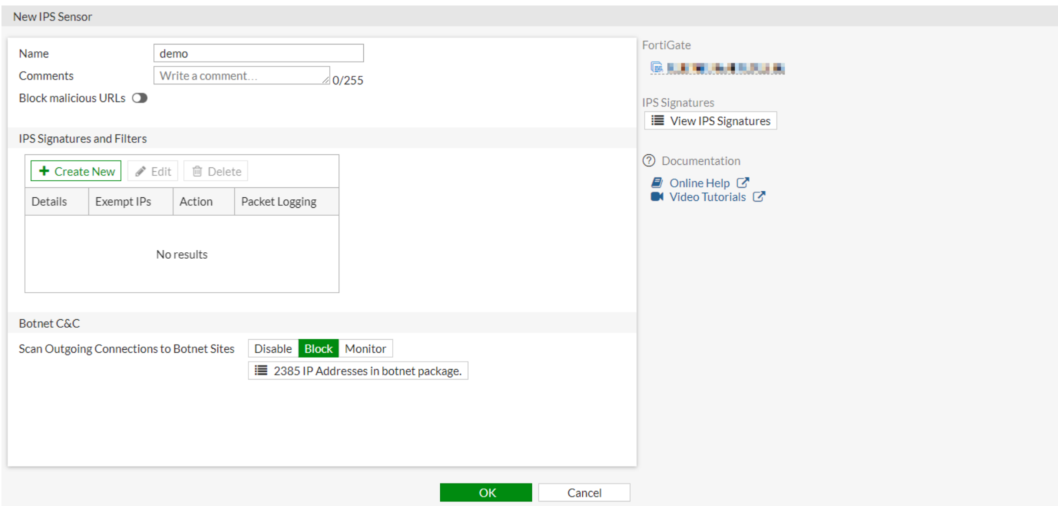
```
# get ips rule status | grep Eicar.Virus.Test.File -A 18
rule-name: "Eicar.Virus.Test.File"
rule-id: 29844
rev: 10.111
date: 1491926400
action: pass
status: enable
log: disable
log-packet: disable
severity: 0.info
service: TCP, HTTP, FTP, SMTP, POP3, IMAP, NNTP
location: server, client
os: All
application: Other
rate-count: 0
rate-duration: 0
rate-track: none
rate-mode: continuous
vuln_type: Anomaly
```

IPS with botnet C&C IP blocking

The *Botnet C&C* section consolidates multiple botnet options in the IPS profile. This allows you to enable botnet blocking across all traffic that matches the policy by configuring one setting in the GUI, or by the `scan-botnet-connections` option in the CLI.

To configure botnet C&C IP blocking in the GUI:

1. Go to *Security Profiles > Intrusion Prevention*, and click *Create New* to create a new IPS sensor, or double-click an existing IPS sensor to open it for editing.
2. Navigate to the *Botnet C&C* section.
3. For *Scan Outgoing Connections to Botnet Sites*, select *Block* or *Monitor*.



4. Configure the other settings as needed.
5. Click *OK* to save the IPS sensor.
6. Add the IPS sensor to a firewall policy.
   The IPS engine will scan outgoing connections to botnet sites. If you access a botnet IP address, an IPS log is generated for this attack.
7. Go to *Log & Report > Security Events* and click the *Intrusion Prevention* card to view the log.

To configure botnet C&C IP blocking in the CLI:

```
config ips sensor
   edit "Demo"
      set scan-botnet-connections {disable | block | monitor}
   next
end
```

The `scan-botnet-connections` option is no longer available in the following CLI commands:

- `config firewall policy`
- `config firewall interface-policy`
- `config firewall proxy-policy`
- `config firewall sniffer`

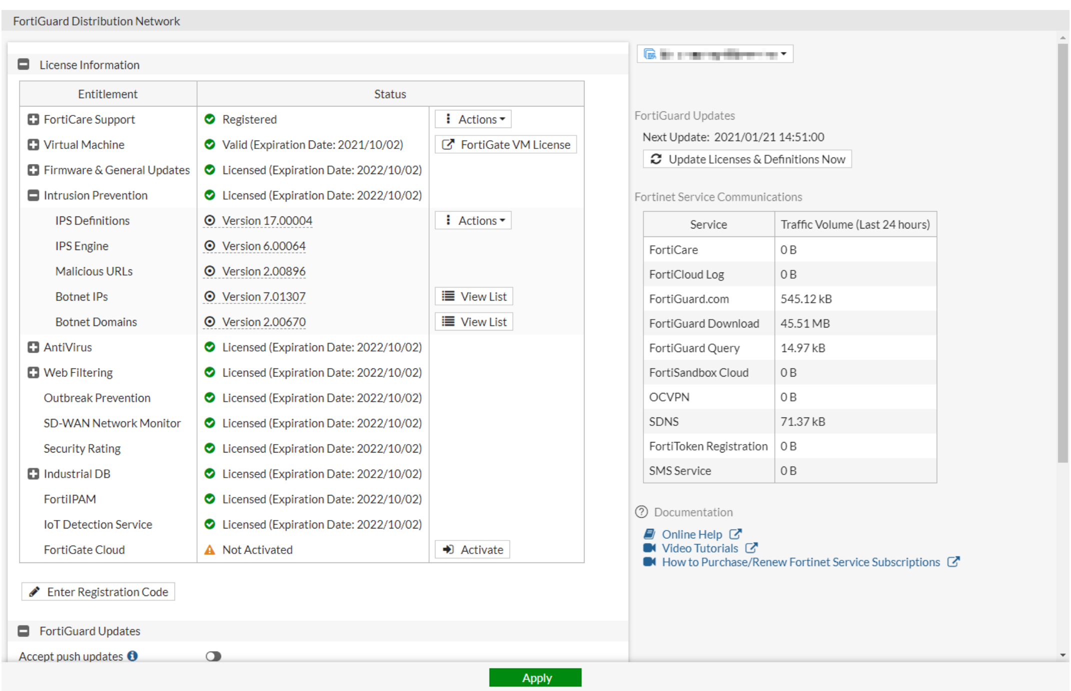
Sample log

```
# execute log filter category 4
# execute log display
1 logs found.
1 logs returned.

1: date=2022-04-28 time=16:18:34 eventtime=1651187914585406621 tz="-0700" logid="0422016400"
type="utm" subtype="ips" eventtype="botnet" level="warning" vd="vd1" msg="Botnet C&C
Communication." severity="critical" srcip=10.1.100.11 srccountry="Reserved"
dstip=2.58.149.169 srcintf="port13" srcintfrole="undefined" dstintf="port14"
dstintfrole="undefined" sessionid=894198 action="dropped" srcport=41798 dstport=80 proto=6
service="HTTP" policyid=1 profile="sensor-1" direction="outgoing" attack="Loki"
attackid=7630239 ref="http://www.fortinet.com/be?bid=7630239" crscore=50 craction=4
crlevel="critical"
```

Botnet IPs and domains lists

To view botnet IPs and domains lists:

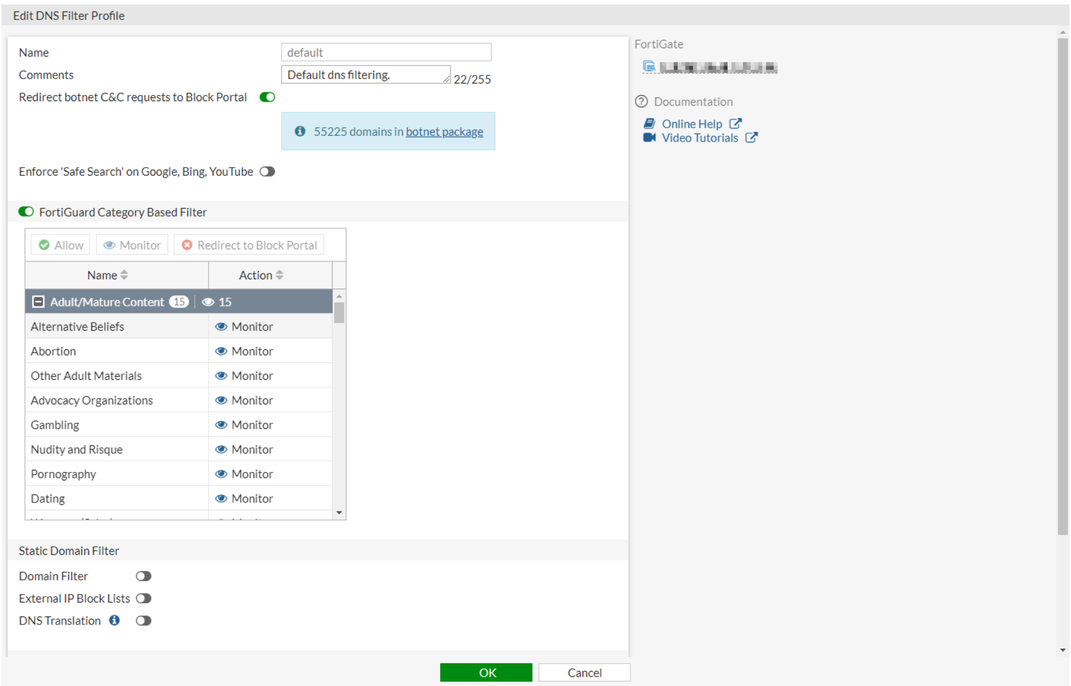
1. Go to *System > FortiGuard*.
2. Expand *License Information > Intrusion Prevention* to view *Botnet IPs* and *Botnet Domains* information.
3. Click *View List* for more details.



Botnet C&C domain blocking

To block connections to botnet domains:

1. Go to *Security Profiles > DNS Filter*, and click *Create New*, or double-click an existing filter to open it for editing.
2. Enable *Redirect botnet C&C requests to Block Portal*.



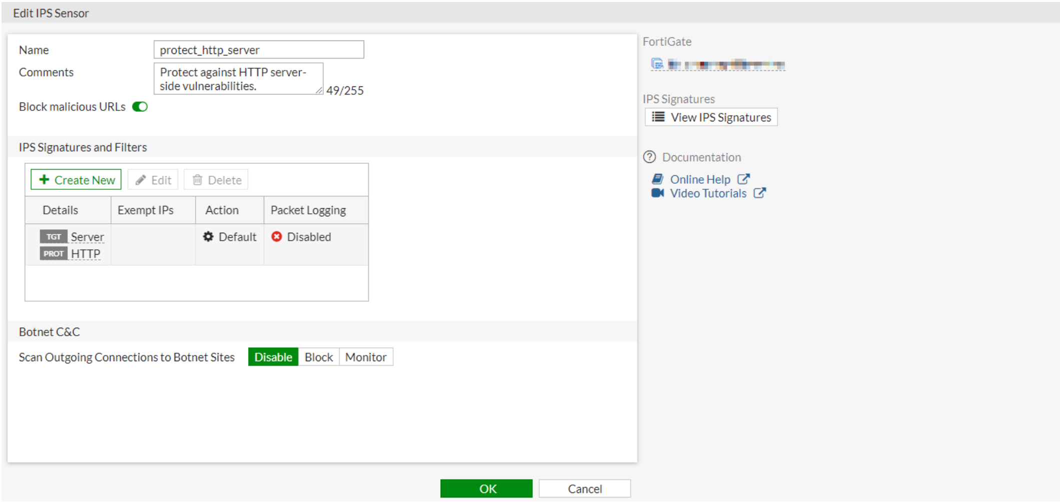
3. Configure the other settings as needed.
4. Click *OK*.
5. Add the filter profile to a firewall policy.

Botnet C&C URL blocking

To block malicious URLs:

1. Go to *Security Profiles > Intrusion Prevention*, and click *Create New*, or double-click an existing filter to open it for editing.

2. Enable *Block malicious URLs*.



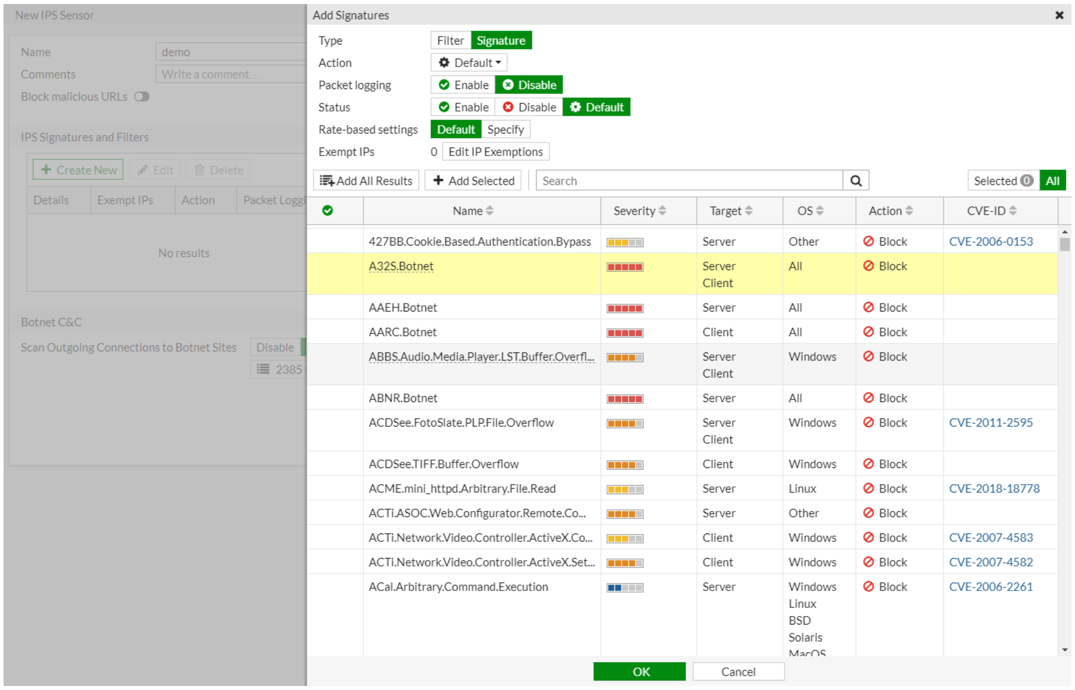
3. Configure the other settings as needed.
4. Click *OK*.
5. Add the sensor to a firewall policy.

Botnet C&C signature blocking

To add IPS signatures to a sensor:

1. Go to *Security Profiles > Intrusion Prevention*, and click *Create New*, or double-click an existing sensor to open it for editing.
2. In the *IPS Signatures and Filters* section, click *Create New*. A list of available signatures appears.
3. For *Type*, select *Signature*. Select the signatures you want to include from the list.
4. Configure the other settings as needed.

5. Click *Add Selected*.



6. Click *OK* to add the IPS signatures to the IPS sensor.
7. Click *OK* to save the IPS sensor.
8. Add the sensor to a firewall policy to detect or block attacks that match the IPS signatures.

IPS signatures for the industrial security service

The FortiGuard Industrial Security Service (ISS) includes both application control and intrusion prevention signatures for industrial applications and protocols. The industrial database attack definitions are only updated if the FortiGate has a valid ISS license and an IPS security profile is used in a policy.

By default, industrial signatures are excluded from the signature lists in the GUI.

To verify that the FortiGate has a valid ISS license:

1. Go to *System > FortiGuard*.
2. In the *License Information* table, check the license status of *Industrial DB*.
3. Expand the *Industrial DB* entry to see the current *Industrial Attack Definitions* version.

To force the industrial DB attack definitions to update:

1. Optionally, create an IPS profile:
   a. Go to *Security Profiles > Intrusion Prevention* and click *Create New*.
   b. Enter a name for the profile.
   c. In the *IPS Signatures and Filters* table click *Create New*.
   d. Click *OK*.
   e. Click *OK*.

See Intrusion prevention on page 1203 for more information.

2. Use the IPS profile in a policy:
   a. Go to *Policy & Objects > Firewall Policy*.
   b. Edit an existing policy, or click *Create New* to create a new policy.
   c. Under *Security Profiles*, enable *IPS* and select an IPS profile.
   d. Configure the remaining settings as needed, then click *OK*.
3. Go to *System > FortiGuard* and either click *Update Licenses & Definitions Now*, or wait for the next automatic update. The update could take a few minutes.
4. Refresh the page, then check the *Industrial Attack Definitions* version to confirm that they have been updated.

To make ISS IPS and application control signatures available in the GUI:

```
config ips global
    set exclude-signatures none
end
```

To view the signatures in the GUI:

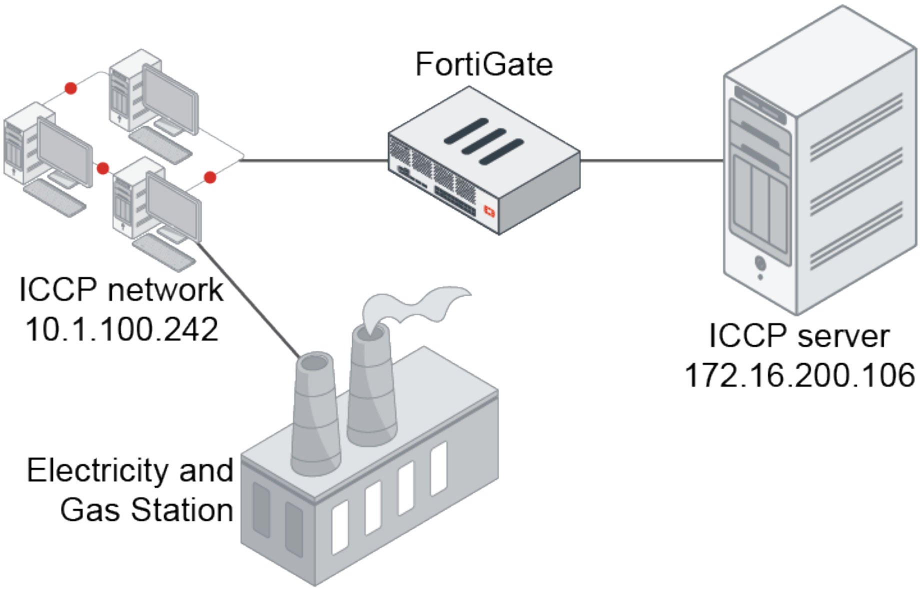
1. Go to *Security Profiles > Application Signatures* and search for *industrial* to find signatures that identify industrial protocols.
2. Go to *Security Profiles > IPS Signatures* to find signatures that detect networks attacks that target industrial assets.

IPS sensor for IEC 61850 MMS protocol

IEC 61850 is a SCADA protocol whose services are mapped to a number of protocols, including MMS services. MMS/ICCP detection is supported in IPS. The purpose of the MMS dissectors is to identify every IEC 61850 service to distinguish different MMS/ICCP messages. IPS engine 6.0.12 and later support MMS dissectors.

The following scenarios are also supported:

- Multiple MMS PDUs are transferred in one TCP payload, and the IPS engine identifies individuals.
- An MMS message is split over multiple TCP segments, where MMS runs over COTP segments.
- ICCP/TASE.2 that also uses MMS transport (ISO transport over TCP for ICCP) is detected.



Industrial signatures must be enabled in the global IPS settings to receive MMS/ICCP signatures. By default, industrial signatures are excluded.

```
config ips global
    set exclude-signatures none
end
```

Below are some industrial signatures for MMS/ICCP messages that can be detected by the IPS engine. This is not an exhaustive list.

- MMS_GetNameList.Request
- MMS_GetNamedVariableListAttributes.Request
- MMS_GetVariableAccessAttributes.Request
- MMS_Identify.Request
- MMS_Initiate.Request
- MMS_Read.Request
- MMS_Reset.Request
- ICCP_Transfer.Reporting
- ICCP_Create.Dataset
- ICCP_Abort
- ICCP_Start.Transfer.DSTransferSet
- ICCP_Get.Dataset.Element.Values
- ICCP_Get.Next.DSTransfer.Set.Value
- ICCP_Delete.Dataset
- ICCP_Start.Transfer.IMTransferSet

Diagnose command

The COTP dissector adds support for identifying every MMS PDU, and let the IPS engine separate them, like the Modbus and IEC-104 services for example.

```
# diagnose ips debug enable all
# diagnose debug enable
```

```
[284@78]ips_l7_dsct_processor: serial=8142 create: cotp
[284@78]ips_l7_dsct_processor: serial=8142 create: iec104
[284@78]ips_l7_dsct_processor: serial=8142 create: modbus
```

Log samples

MMS dissectors can be triggered, and MMS/ICCP signatures can be monitored and logged.

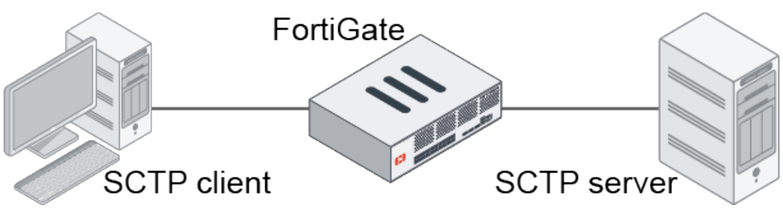
Log samples:

```
date=2020-03-26 time=15:51:10 logid="1059028704" type="utm" subtype="app-ctrl"
eventtype="signature" level="information" vd="vd1" eventtime=1585263070836106492 tz="-0700"
appid=43699 srcip=10.1.100.242 dstip=172.16.200.106 srcport=50963 dstport=102
srcintf="port13" srcintfrole="undefined" dstintf="port14" dstintfrole="undefined" proto=6
service="tcp/26112" direction="outgoing" policyid=1 sessionid=2711 applist="test"
action="pass" appcat="Industrial" app="MMS_Read.Request" incidentserialno=376610508
msg="Industrial: MMS_Read.Request," apprisk="elevated"
```

```
date=2020-03-26 time=16:15:45 logid="1059028704" type="utm" subtype="app-ctrl"
eventtype="signature" level="information" vd="vd1" eventtime=1585091746264983273 tz="-0700"
appid=44684 srcip=10.1.100.242 dstip=172.16.200.106 srcport=41665 dstport=102
srcintf="port13" srcintfrole="undefined" dstintf="port14" dstintfrole="undefined" proto=6
service="tcp/26112" direction="incoming" policyid=1 sessionid=194463 applist="test"
action="pass" appcat="Industrial" app="ICCP_Transfer.Reporting" incidentserialno=762763993
msg="Industrial: ICCP_Transfer.Reporting," apprisk="elevated"
```

SCTP filtering capabilities

A Stream Control Transmission Protocol (SCTP) dissector and Payload Protocol Identifier (PPID) filter can be used to either terminate the SCTP session, or replace the offending data chunk with zeros to keep the client and server sequence numbers synchronized. The SCTP filter action can also pass the data chunk.



To configure and test an SCTP filter:

1. Configure an SCTP filter profile that uses the reset action:

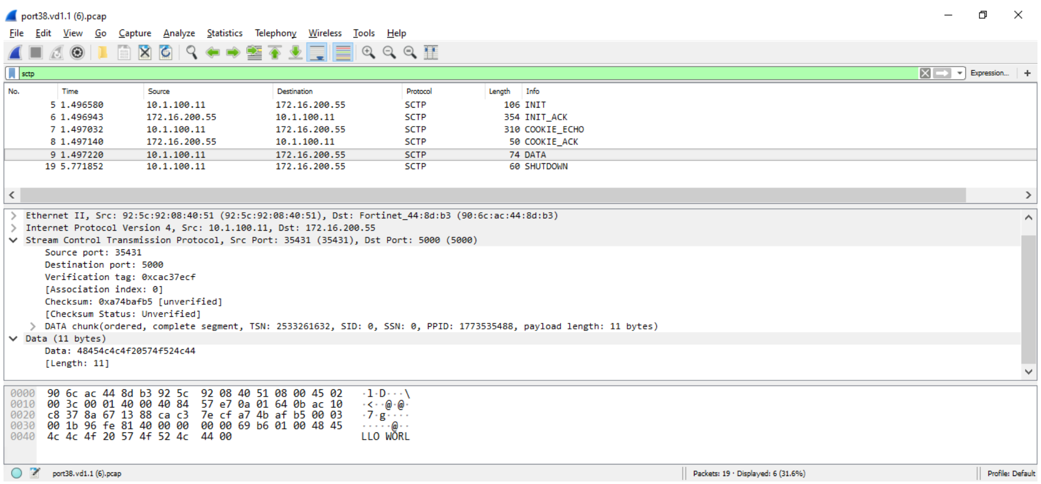
```
config sctp-filter profile
    edit "sctp"
        set comment "Demo profile"
        config ppid-filters
            edit 1
                set ppid 112233
                set action reset
                set comment "test chunk"
            next
        end
```

```
    next
end
```

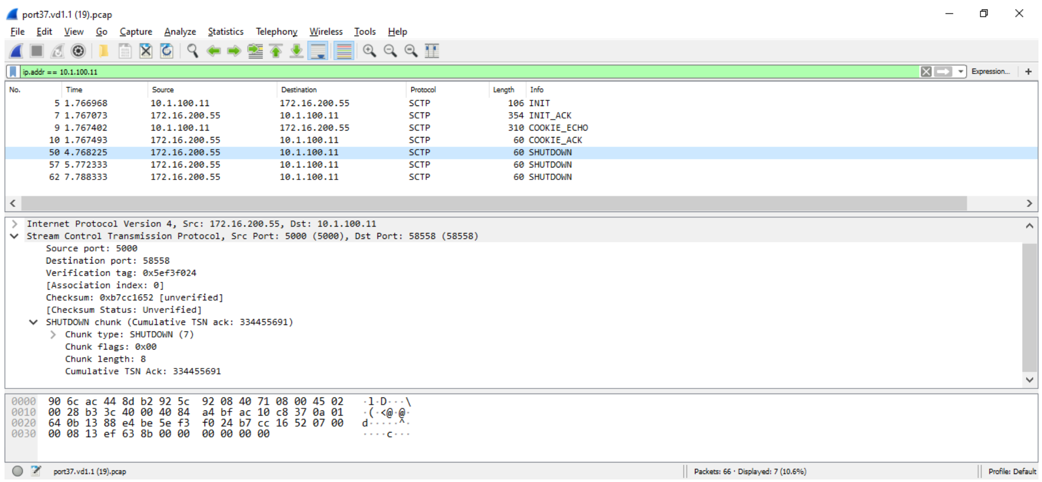
2. Use the SCTP filter profile in a firewall policy:

```
config firewall policy
    edit 1
        set name "1"
        set srcintf "port38"
        set dstintf "port37"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set ssl-ssh-profile "new-deep-inspection"
        set sctp-filter-profile "sctp"
        set logtraffic all
    next
end
```

3. On the SCTP client, confirm that the connection works and send a data chunk with PPID 112233.



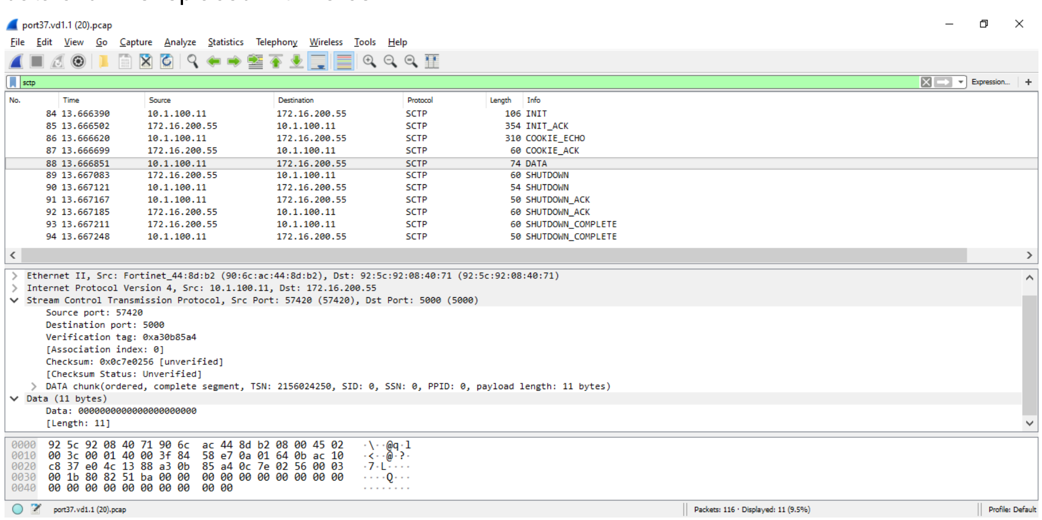
4. The IPS engine detects the data chunk. The PPID matches the PPID filter, and the filter action is reset, so the data chunk is not received on the server, and the session is terminated.



5. Change the filter action to replace:

```
config sctp-filter profile
    edit "sctp"
        config ppid-filters
            edit 1
                set action replace
            next
        end
    next
end
```

6. Resend the data chunk.
7. The IPS engine detects the data chunk. The PPID matches the PPID filter, and the filter action is replace, so the data chunk is replaced with zeros.



File filter

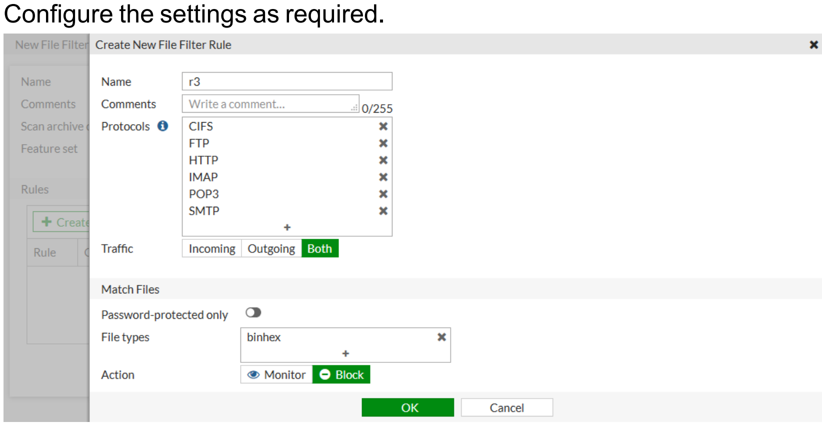
The file filter can be applied directly to firewall policies and supports various traffic protocols in proxy or flow mode.

Protocol	Proxy mode	Flow mode
CIFS	Yes	Yes
FTP	Yes	Yes
HTTP	Yes	Yes
IMAP	Yes	Yes
MAPI	Yes	No
POP3	Yes	Yes
SMTP	Yes	Yes
SSH	Yes	No

Prior to FortiOS 6.4.1, file filter was embedded in the web filter, email filter, SSH inspection, and CIFS profiles.

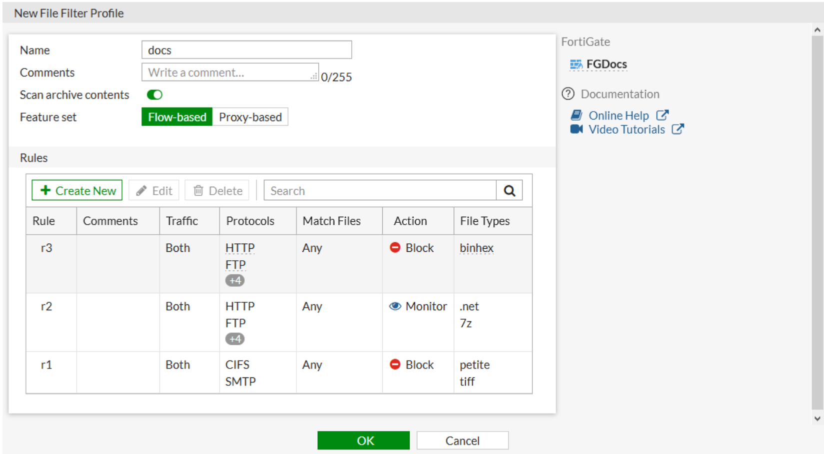
To configure a file filter in the GUI:

1. Configure the filter profile:
   a. Go to *Security Profiles > File Filter* and click *Create New*.
   b. Select a *Feature set*.
   c. In the *Rules* table, click *Create New*.
   d. Configure the settings as required.



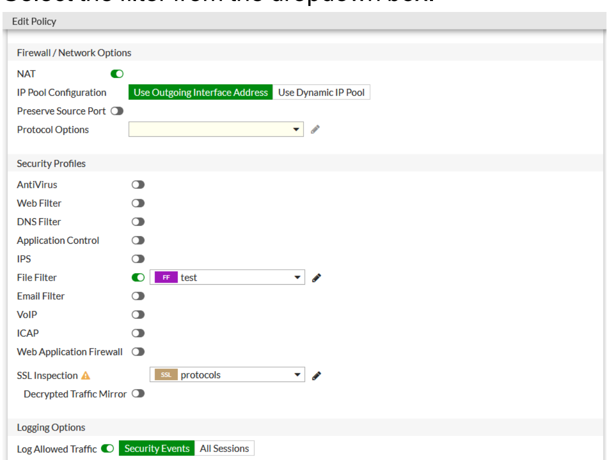
   e. Click *OK* to save the rule.

f. Optionally, create more rules.



g. Click *OK* to save the filter profile.

2. Apply the filter to a policy:
   a. Go to *Policy & Objects > Firewall Policy* and edit an existing policy or create a new one.
   b. In the *Security Profiles* section, enable *File Filter*.
   c. Select the filter from the dropdown box.



   d. Configure the other settings as needed.
   e. Click *OK*.

To configure a file filter in the CLI:

1. Configure the file filter profile:

```
config file-filter profile
    edit "test"
        set comment ''
        set feature-set flow
        set replacemsg-group ''
        set log enable
        set scan-archive-contents enable
        config rules
            edit "r2"
                set comment ''
```

```
                set protocol http ftp smtp imap pop3 cifs
                set action block
                set direction outgoing
                set password-protected any
                set file-type "sis" "tar" "tiff" "torrent" "upx" "uue" "wav" "wma" "xar"
"xz" "zip"
            next
            edit "r1"
                set comment ''
                set protocol http ftp smtp imap pop3 cifs
                set action log-only
                set direction any
                set password-protected any
                set file-type ".net" "7z" "activemime" "arj" "aspack" "avi" "base64"
"bat" "binhex" "bmp" "bzip" "bzip2"
            next
            edit "r3"
                set comment ''
                set protocol http ftp smtp imap pop3
                set action block
                set direction any
                set password-protected any
                set file-type "binhex"
            next
        end
    next
end
```

2. Apply the filter to a policy:

```
config firewall policy
    edit 1
        set name "filefilter-policy"
        set srcintf "port10"
        set dstintf "port9"
        set srcaddr "all"
        set dstaddr "all"
        set srcaddr6 "all"
        set dstaddr6 "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set profile-protocol-options "protocol"
        set ssl-ssh-profile "protocols"
        set file-filter-profile "test"
        set auto-asic-offload disable
        set np-acceleration disable
        set nat enable
    next
end
```

Logs

Go to *Log & Report* > *Security Events* and select the *File Filter* card to view the file filter logs.

Log samples

```
date=2020-04-21 time=17:04:02 logid="1900064000" type="utm" subtype="file-filter"
eventtype="file-filter" level="warning" vd="root" eventtime=1587513843211612684 tz="-0700"
policyid=1 sessionid=1751 srcip=10.1.100.22 srcport=57382 srcintf="port21"
srcintfrole="undefined" dstip=172.16.200.44 dstport=445 dstintf="port23"
dstintfrole="undefined" proto=6 service="CIFS" profile="filefilter" direction="incoming"
action="blocked" filtername="1" filename="sample\\putty.exe" filesize=454656 filetype="exe"
msg="File was blocked by file filter."
```

```
date=2020-04-21 time=17:03:54 logid="1900064000" type="utm" subtype="file-filter"
eventtype="file-filter" level="warning" vd="root" eventtime=1587513834376811325 tz="-0700"
policyid=1 sessionid=1742 srcip=10.1.100.22 srcport=36754 srcintf="port21"
srcintfrole="undefined" dstip=172.16.200.44 dstport=22 dstintf="port23"
dstintfrole="undefined" proto=6 service="SSH" subservice="SCP" profile="filefilter"
direction="incoming" action="blocked" filtername="1" filename="test.pdf" filesize=571051
filetype="pdf" msg="File was blocked by file filter."
```

```
date=2020-04-21 time=17:00:30 logid="1900064000" type="utm" subtype="file-filter"
eventtype="file-filter" level="warning" vd="root" eventtime=1587513630482716465 tz="-0700"
policyid=1 sessionid=1684 srcip=10.1.100.22 srcport=58524 srcintf="port21"
srcintfrole="undefined" dstip=172.16.200.44 dstport=143 dstintf="port23"
dstintfrole="undefined" proto=6 service="IMAP" profile="filefilter" direction="incoming"
action="blocked" from="pc4user1@qa.fortinet.com" to="pc4user2@qa.fortinet.com"
recipient="pc4user2" subject="QA Test" filtername="1" filename="test.JPG" filesize=48079
filetype="jpeg" msg="File was blocked by file filter."
```

```
date=2020-04-21 time=16:59:58 logid="1900064000" type="utm" subtype="file-filter"
eventtype="file-filter" level="warning" vd="root" eventtime=1587513598866551739 tz="-0700"
policyid=1 sessionid=1674 srcip=10.1.100.22 srcport=39854 srcintf="port21"
srcintfrole="undefined" dstip=172.16.200.44 dstport=110 dstintf="port23"
dstintfrole="undefined" proto=6 service="POP3" profile="filefilter" direction="incoming"
action="blocked" from="pc4user1@qa.fortinet.com" to="pc4user2@qa.fortinet.com"
recipient="pc4user2" subject="QA Test" filtername="1" filename="test.JPG" filesize=48079
filetype="jpeg" msg="File was blocked by file filter."
```

```
date=2020-04-21 time=16:58:31 logid="1900064000" type="utm" subtype="file-filter"
eventtype="file-filter" level="warning" vd="root" eventtime=1587513511516745955 tz="-0700"
policyid=1 sessionid=1619 srcip=10.1.100.22 srcport=53144 srcintf="port21"
srcintfrole="undefined" dstip=172.16.200.44 dstport=25 dstintf="port23"
dstintfrole="undefined" proto=6 service="SMTP" profile="filefilter" direction="outgoing"
action="blocked" from="pc4user1@qa.fortinet.com" to="pc4user2@qa.fortinet.com"
sender="pc4user1@qa.fortinet.com" recipient="pc4user2@qa.fortinet.com" subject="QA Test"
filtername="1" filename="test.PNG" filesize=65173 filetype="png" msg="File was blocked by
file filter."
```

```
date=2020-04-21 time=16:58:14 logid="1900064000" type="utm" subtype="file-filter"
eventtype="file-filter" level="warning" vd="root" eventtime=1587513494608988795 tz="-0700"
policyid=1 sessionid=1605 srcip=10.1.100.22 srcport=43186 srcintf="port21"
srcintfrole="undefined" dstip=172.16.200.44 dstport=21 dstintf="port23"
dstintfrole="undefined" proto=6 service="FTP" profile="filefilter" direction="incoming"
action="blocked" filtername="1" filename="index.html" filesize=21 filetype="html" msg="File
was blocked by file filter."
```

Supported file types

File filter allows the FortiGate to block files passing through based on file type based on the file's meta data only, and not on file size or file content. A DLP profile must be configured to block files based on size or content, such as SSN numbers, credit card numbers, or regexp.

The following file types are supported in file filter and DLP profiles:

Type	Description
.net	Match .NET files
7z	Match 7-Zip files
activemime	Match ActiveMime files
arj	Match ARJ compressed files
aspack	Match ASPack files
avi	Match AVI files
base64	Match Base64 files
bat	Match Windows batch files
binhex	Match BinHex files
bmp	Match BMP files
bzip	Match Bzip files
bzip2	Match Bzip2 files
cab	Match Windows CAB files
chm	Match Windows compiled HTML help files
class	Match CLASS files
cod	Match COD files
crx	Match Chrome extension files
dmg	Match Apple disk image files
elf	Match ELF files
exe	Match Windows executable files
flac	Match FLAC files
fsg	Match FSG files
gif	Match GIF files
gzip	Match Gzip files
hlp	Match Windows help files
hta	Match HTA files

Type	Description
html	Match HTML files
iso	Match ISO archive files
jad	Match JAD files
javascript	Match JavaScript files
jpeg	Match JPEG files
lzh	Match LZH compressed files
mach-o	Match Mach object files
mime	Match MIME files
mov	Match MOV files
mp3	Match MP3 files
mpeg	Match MPEG files
msi	Match Windows Installer MSI Bzip files
msoffice	Match MS-Office files. For example, DOC, XLS, PPT, and so on.
msofficex	Match MS-Office XML files. For example, DOCX, XLSX, PPTX, and so on.
pdf	Match PDF files
petite	Match Petite files
png	Match PNG files
rar	Match RAR archives
rm	Match RM files
sis	Match SIS files
tar	Match TAR files
tiff	Match TIFF files
torrent	Match torrent files
unknown*	Match unknown files
upx	Match UPX files
uue	Match UUE files
wav	Match WAV files
wma	Match WMA files
xar	Match XAR archive files
xz	Match XZ files
zip	Match ZIP files

* This file type is only available in DLP profiles.

Email filter

Email filters can be configured to perform spam detection and filtering. You can customize the default profile, or create your own and apply it to a firewall policy.

Two kinds of filtering can be defined in a single profile, and they will act independent of one another.

Filter options can be organized according to the source of the decision:

- Local options: the FortiGate qualifies the email based on local conditions, such as block/allowlists, banned words, or DNS checks using FortiGuard Antispam.
- FortiGuard-based options: the FortiGate qualifies the email based on the score or verdict returned from FortiGuard Antispam.
- Third-party options: the FortiGate qualifies the email based on information from a third-party source (like an ORB list).

Local and FortiGuard block/allowlists can be enabled and combined in a single profile. When combined, the local block/allowlist has a higher priority than the FortiGuard block list during a decision making process. For example, if a client IP address is blocklisted in the FortiGuard server, but you want to override this decision and allow the IP to pass through the filter, you can define the IP address or subnet in a local block/allowlist with the clear action. Because the information coming from the local list has a higher priority than the FortiGuard service, the email will be considered clean.

Some features of this functionality require a subscription to FortiGuard Antispam.

Protocol comparison between email filter inspection modes

The following table indicates which email filters are supported by their designated inspection modes.

	SMTP	POP3	IMAP	MAPI
Proxy	Yes	Yes	Yes	Yes
Flow	Yes	Yes	Yes	No

The following topics provide information about email filter profiles:

- Local-based filters on page 1233
- FortiGuard-based filters on page 1240
- Third-party-based filters on page 1242
- Filtering order on page 1242
- Protocols and actions on page 1244
- Configuring webmail filtering on page 1245

Local-based filters

There are six types of local spam filters:

- HELO DNS lookup
- Return email DNS check
- Block/allow list
- Banned words*
- Trusted IP addresses*
- MIME header*

* These filters can only be configured in the CLI.

By default, HELO DNS and return email DNS checks are done before the block/allow list check. In some situations, such as when configuring a block/allow list to clear an email from performing further filtering, configure the following to give precedence to the block/allow list:

```
config emailfilter profile
    edit <name>
        config smtp
            set local-override enable
        next
    end
end
```

HELO DNS lookup

Whenever a client opens an SMTP session with a server, the client sends a HELO command with the client domain name. The FortiGate takes the domain name specified by the client in the HELO and performs a DNS lookup to determine if the domain exists. If the lookup fails, the FortiGate determines that any emails delivered during the SMTP session are spam. The HELO DNS lookup is only available for SMTP traffic.

Return email DNS check

The FortiGate performs a DNS lookup on the return field. If no such record exists, the email is treated as spam. When return email DNS checking is enabled, the FortiGate takes the domain in the reply-to email address and reply-to domain, and checks the DNS servers to see if there is an A or MX record for the domain. If the domain does not exist, the FortiGate treats the email as spam.

Block/allow list

Block/allow lists can be made from emails or IP subnets to forbid or allow them to send or receive emails. The following table summarizes the configurable options in a block/allow list.

<table>
<tr><th>Type</th><th>Description</th><th>Pattern</th><th>Action</th></tr>
<tr><td>IP/Netmask and IPv6/Netmask</td><td>The FortiGate compares the IP address of the client delivering the email to the addresses in the IP address block/allow list specified in the email filter profile.
If a match is found, the FortiGate takes the action configured for the matching block/allow list entry against all delivered email.
By default the <code>hdrip</code> setting under <code>config smtp</code> is disabled. If enabled, the FortiGate checks all the IP addresses in the header of SMTP email against the specified IP address block/allow list.</td><td>The filter is an IP address with a subnet mask.</td><td>• Mark as Reject: the email is dropped before reaching its destination.
• Mark as Spam: the email is allowed through, but it will be tagged with an indicator marking the email as spam.
• Mark as Clear: the email is allowed to go through to its destination on the assumption that it is not spam.</td></tr>
<tr><td>Recipient Address[*]</td><td>The FortiGate compares the sender email address to the contents of the RCPT TO envelope header and To: mail header to the specified pattern. If a match is found, the FortiGate takes the action configured for the matching block/allow list entry.</td><td rowspan="3">• Wildcard: the filter is an email address with a wildcard symbol in place of the variable characters (such as *.example.com or fred@*.com).
• Regular Expression: the filter is a regular expression. For example, ^ [_a-z0-9-]+(\.[_a-z0-9-]+)*@(example|xmple|examp).(com|org|net) can be used to filter based on a number of email domain name combinations.</td><td rowspan="3">• Mark as Spam: the email is allowed through, but it will be tagged with an indicator marking the email as spam.
• Mark as Clear: the email is allowed to go through to its destination on the assumption that it is not spam.</td></tr>
<tr><td>Sender Address[*]</td><td>The FortiGate compares the sender email address to the contents of the MAIL FROM envelope header, From: mail header, and Sender: mail header to the specified pattern. If a match is found, the FortiGate takes the action configured for the matching block/allow list entry.</td></tr>
<tr><td>Subject[*]</td><td>The FortiGate compares the sender email address to the contents of the Subject: mail header to the specified pattern. If a match is found, the FortiGate takes the action configured for the matching block/allow list entry.</td></tr>
</table>

[*] This type is only supported in proxy mode.

Banned words

When banned word checking is enabled, the FortiGate examines emails for words that appear in the banned word list specified in the email filter profile.

The banned word pattern can be either wildcard or Perl regular expression, which could include part of a word, a whole word, a phrase, multiple words, or multiple phrases.

Each time the banned word filter detects a pattern in an email, it adds the pattern score to the sum of scores for the message. The score is set when creating a new pattern to block content (`set score`). Higher scores indicate more offensive content. If the total score of the discovered banned words in the email exceeds the threshold value set in the email filter profile, then the FortiGate treats the email as spam. The score for each pattern is counted only once, even if that pattern appears many times in the email. The default score for banned word patterns is 10, and the default threshold in the email filter is 10. This means that by default, an email message is blocked by a single match.

For example, if the FortiGate scans an email containing only this sentence: “The score for each word or phrase is counted only once, even if that word or phrase appears many times in the email message.” and the banned word list contains the following patterns:

Banned word pattern	Pattern type	Assigned score	Score added to sum for entire page	Comments
word	Wildcard	20	20	The pattern appears twice, but it is counted once.
word phrase	Wildcard	20	0	Both words appear in the email, but they do not appear together as specified in the pattern. There are no matches.
word*phrase	Wildcard	20	20	A match occurs as long as “word” appears before “phrase” regardless of what is in between them. The pattern appears twice, but it is counted once.
mail*age	Wildcard	20	20	This pattern is a match because “email message” appears in the email.

The email would be treated as spam if the banned word threshold is set to 60 or less.

To apply a banned word filter to an email filter profile:

1. Configure the banned words list:

```
config emailfilter bword
    edit 1
        set name "banned"
        config entries
            edit 23
                set pattern-type {wildcard | regexp}
                set pattern <string>
                set score <1 - 99999>
            next
        end
```

```
    next
end
```

2. Configure the email filter profile:

```
config emailfilter profile
    edit "myBannedWordsProfile"
        set spam-filtering enable
        set options bannedword
        set spam-bword-threshold <0 - 2147483647>
        set spam-bword-table 23
    next
end
```

Once a banned word list is configured in the CLI and applied to an email filter profile, some settings can be edited in the GUI for that particular email filter profile. A banned word profile can be selected, and its *Threshold* (`spam-bword-threshold`) can be edited.

Trusted IP addresses

When the FortiGate creates a list of trusted IP addresses, any incoming email traffic from these IP address is exempt from having IP-based checks, such as DNSBL, RBL, FortiGuard Antispam service, or locally-defined IP block lists.

If the FortiGate sits behind a company's mail transfer units, it may be unnecessary to check email IP addresses because they are internal and trusted. In this case, only external IP addresses would be checked. In some cases, external IP addresses may be added to the list if they are known to not be spam sources.

To configure a trusted IP address list:

1. Define the IP address list:

```
config emailfilter iptrust
    edit 1
        set name "trustedIP"
        config entries
            edit 33
                set addr-type {ipv4 | ipv6}
                set ipv4-subnet <IPv4_classnet>
                set ipv6-subnet <IPv6_network>
            next
        end
    next
end
```

2. Add the list to the email filter profile:

```
config emailfilter profile
    edit "email_filter_profile"
        set spam-iptrust-table 1
    next
end
```

MIME header

This feature filters by the MIME header.

To configure a MIME header check:

1. Define the header content:

```
config emailfilter mheader
    edit 100
        set name "mheader"
        config entries
            edit 1
                set fieldname <string>
                set fieldbody <string>
                set pattern-type {wildcard | regexp}
                set action {spam | clear}
            next
        end
    next
end
```

2. Add the header to the email filter profile:

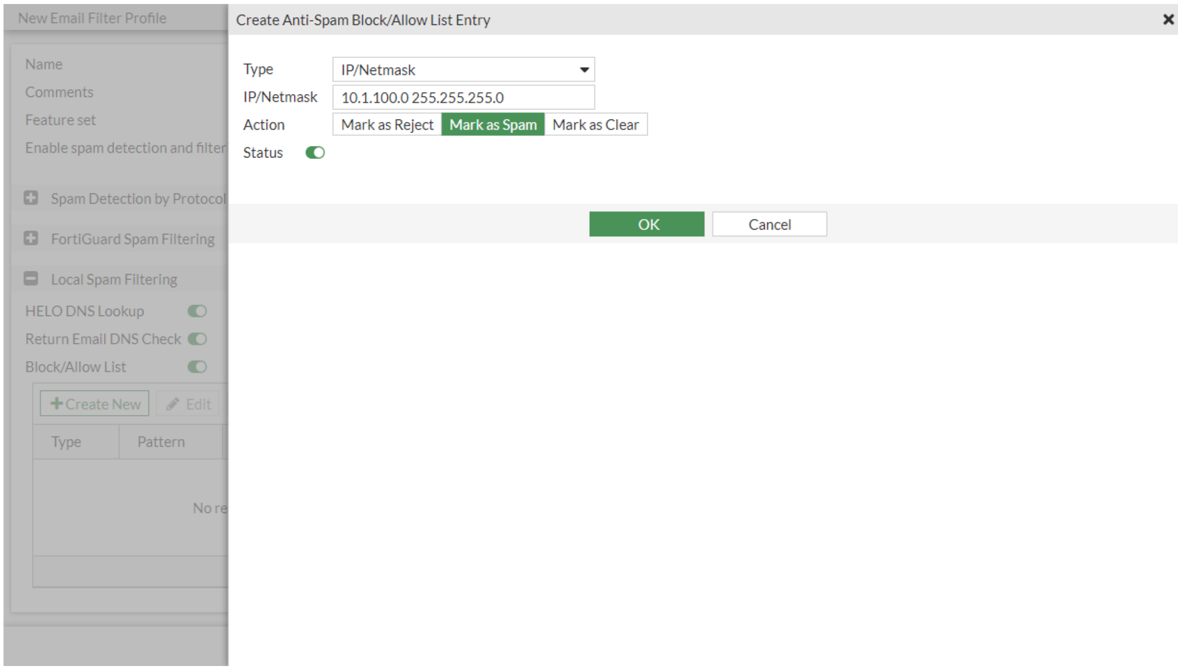
```
config emailfilter profile
    edit "email_filter_profile"
        set options spamhdrcheck
        set spam-mheader-table 100
    next
end
```

Configuring a local-based email filter

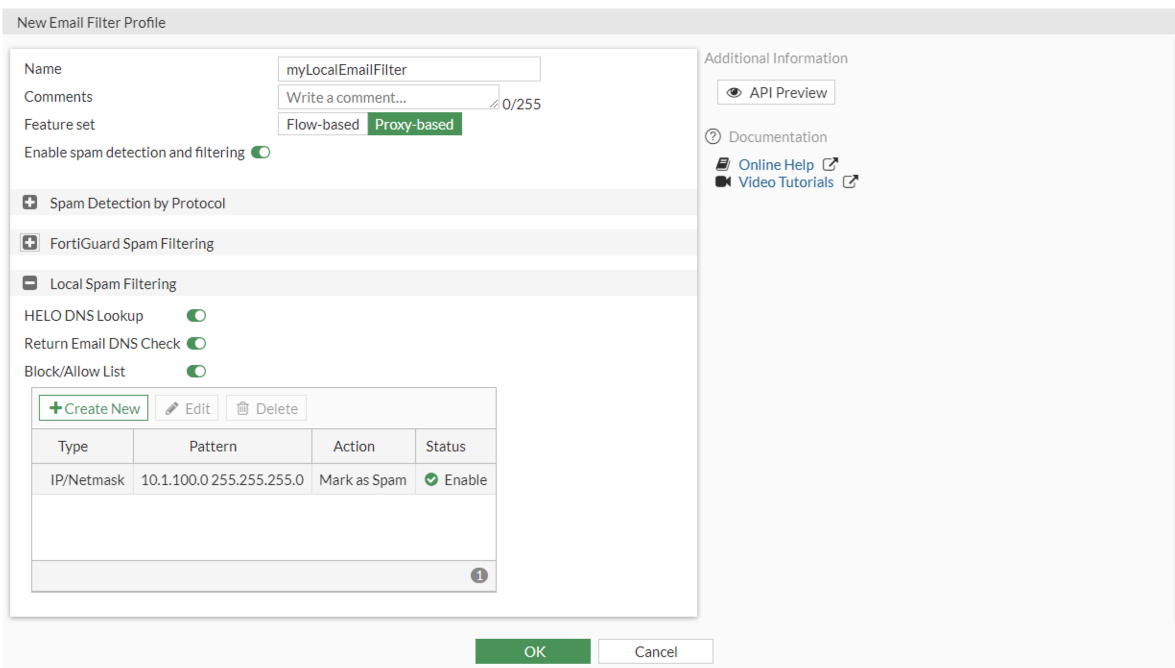
To configure a local-based email filter in the GUI:

1. Configure the email filter profile:
   a. Go to *Security Profiles* > *Email Filter* and click *Create New*, or edit an existing profile.
   b. Select a *Feature set* (*Proxy-based* is used in this example) and enable *Enable spam detection and filtering*.
   c. In the *Local Spam Filtering* section, enable the desired filters (*HELO DNS Lookup*, *Return Email DNS Check*, *Block/Allow List*).
   d. In the *Block/Allow List* table, click *Create New*. The *Create Anti-Spam Block/Allow List Entry* pane opens.
   e. Set the *Type* to *IP/Netmask* and enter an *IP/Netmask*.

f. Select an *Action*.



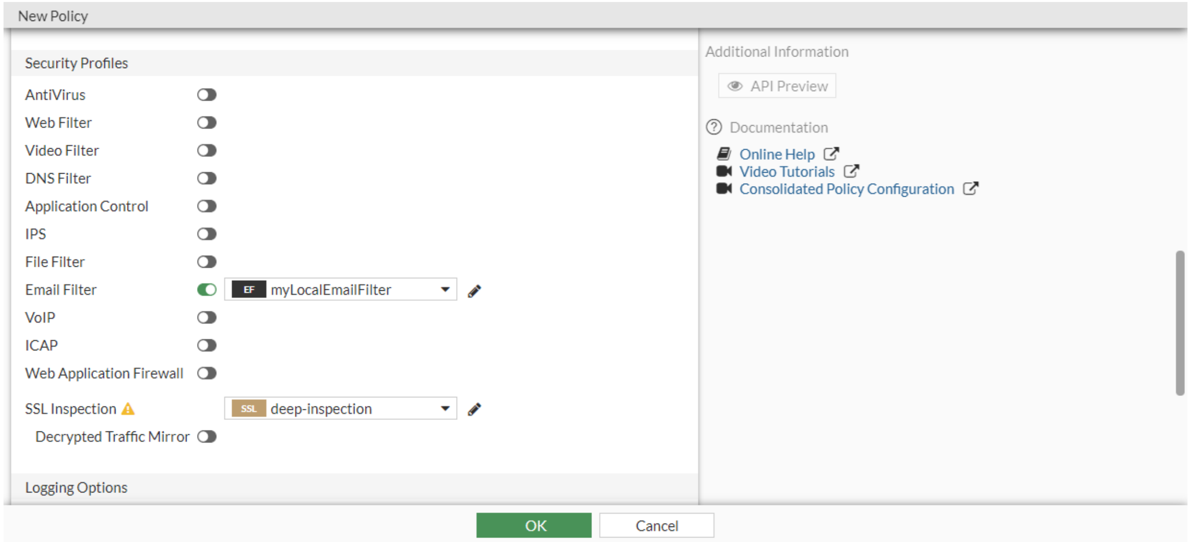
g. Click *OK* to save the block/allow list.



h. Click *OK* save the email filter profile.

2. Configure the firewall policy:
   a. Go to *Policy & Objects > Firewall Policy* and click *Create New*, or edit an existing policy.
   b. Set the inspection-mode to *Proxy-based*.

c. Enable the *Email Filter* option and select the previously created profile.



d. Set *SSL Inspection* to a profile that has deep SSL inspection enabled.
Deep inspection is required to filter SMTP, POP3, IMAP, or any SSL/TLS encapsulated protocol.

e. Configure the other settings as needed.

f. Click *OK*.

To configure a local-based email filter in the CLI:

1. Configure a block/allow list:

```
config emailfilter block-allow-list
    edit 1
        set name "myBAL"
        config entries
            edit 1
                set status enable
                set type ip
                set action spam
                set addr-type ipv4
                set ip4-subnet 10.1.100.0 255.255.255.0
            next
        end
    next
end
```

2. Configure an email filter profile:

```
config emailfilter profile
    edit "myLocalEmailFilter"
        set spam-filtering enable
        set options spambal spamhelodns spamraddrdns
        config smtp
            set action tag
        end
        set spam-bal-table 1
    next
end
```

3. Use the profile in a firewall policy:

```
config firewall policy
    edit 1
        set inspection-mode proxy
        set ssl-ssh-profile "deep-inspection"
        set emailfilter-profile "myLocalEmailFilter"
    next
end
```

FortiGuard-based filters

The FortiGate consults FortiGuard servers to help identify spammer IP address or emails, known phishing URLs, known spam URLs, known spam email checksums, and others. For more information, refer to the FortiGuard website.

There are five FortiGuard spam filtering options:

- IP address check
- URL check
- Detect phishing URLs in email (requires URL check to be enabled)
- Email checksum check
- Spam submission

IP address check

The FortiGate queries the FortiGuard Antispam service to determine if the IP address of the client delivering the email is in the block list. If there is a match, the FortiGate treats delivered emails as spam.

URL check

The FortiGate submits all URLs that appear in the email body to the FortiGuard service for checking. If a URL exists in the FortiGuard URL block list, the FortiGate treats the email as spam.

Detect phishing URLs in email

The FortiGate submits all URL hyperlinks that appear in the email body to the FortiGuard service for checking. If a URL exists in the FortiGuard URL phishing list, the FortiGate removes the hyperlink from the message. The URL remains in place, but it is no longer a clickable hyperlink.

Email checksum check

The FortiGate submits a checksum of each email to the FortiGuard service for checking. If a checksum exists in the FortiGuard checksum block list, the FortiGate treats the email as spam.

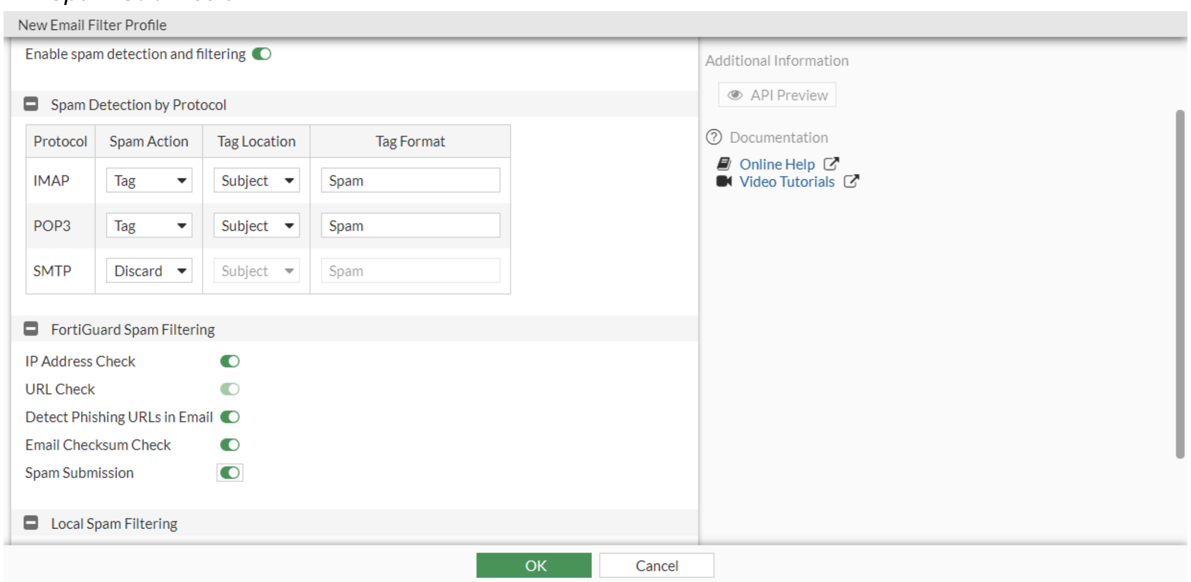
Spam submission

Spam submission is a way to inform the FortiGuard Antispam service of non-spam messages incorrectly marked as spam. When enabled, the FortiGate adds a link to the end of every email marked as spam. Click the link to notify the FortiGuard Antispam service if an email is marked incorrectly.

Configuring FortiGuard filters

To configure FortiGuard filters in the GUI:

1. Go to *Security Profiles > Email Filter* and click *Create New*.
2. Enable *Enable spam detection and filtering*.
3. In the *FortiGuard Spam Filtering Spam Filtering* section, enable the following as needed:
   - *IP Address Check*
   - *URL Check*
   - *Detect Phishing URLs in Email*
   - *Email Checksum Check*
   - *Spam Submission*



4. Click *OK*.

To configure FortiGuard filters in the CLI:

```
config emailfilter profile
    edit <name>
        set spam-filtering enable
        set options spamfsip spamfsurl spamfsphish spamfschksum spamfssubmit
    next
end
```

Option	Description
`spamfsip`	Check email IP addresses
`spamfsurl`	Check email content URLs
`spamfsphish`	Check email content phishing URLs
`spamfschksum`	Check email checksums
`spamfssubmit`	Add FortiGuard Antispam spam submission text

Third-party-based filters

In addition to local and FortiGuard filters, FortiOS can leverage third-party sources, which are known as DNS-based blackhole lists (DNSBL) or Open Relay Behavior-modification Systems (ORBS). These are maintained lists of IP addresses that have been identified as associated with spamming.

The following example demonstrates how to configure a DNSBL. The `config emailfilter dnsbl` command is used to configure either DNSBL or ORBS.

To configure a DNSBL:

1. Define the server to get the DNSBL list from:

```
config emailfilter dnsbl
    edit 100
        set name "dnsbl"
        config entries
            edit 1
                set status enable
                set server <IP address or server name>
                set action {reject | spam}
            next
        end
    next
end
```

2. Add the DNSBL list to an email filter profile:

```
config emailfilter profile
    edit "email_filter_profile"
        set options spamrbl
        set spam-rbl-table 100
    next
end
```

Filtering order

The FortiGate checks for spam using various filtering techniques. The filtering order used by the FortiGate depends on which mail protocol is used.

Filters requiring a query to a server and a reply (FortiGuard Antispam service and DNSBL/ORDBL) are run simultaneously. To avoid delays, queries are sent while other filters are running. The first reply to trigger a spam action takes effect as soon as the reply is received.

Each spam filter passes the email to the next if no matches or problems are found. If the action in the filter is *Mark as Spam*, the FortiGate tags the email as spam according to the settings in the email filter profile. If the action in the filter is *Mark as Reject*, the email session is dropped. If the action in the filter is *Mark as Clear*, the email is exempt from any remaining filters. For SMTP and SMTPS, if the action is *Discard*, the email is discarded or dropped.

SMTP and SMTPS spam filtering order

The FortiGate scans SMTP and SMTPS email for spam in a specific order, which depends on whether or not the local override feature is enabled. This feature is disabled by default, but enabling it gives priority to local spam filters.

You can enable local override (`set local-override`) in an email filter profile to override SMTP or SMTPS remote checks, which includes checks for IP RBL, IP FortiGuard AntiSpam, and HELO DNS with the locally defined antispam block and/or allow lists.

SMTPS spam filtering is available on FortiGates that support SSL content scanning and inspection.

To configure local override of an antispam filter:

```
config emailfilter profile
    edit <name>
        set spam-filtering enable
        set options spambal spamfsip spamfsurl spamhelodns spamfsphish
        config smtp
            set local-override {enable | disable}
        end
        set spam-bal-table 1
    next
end
```

Local override disabled	Local override enabled
1. HELO DNS lookup, last hop IP check against ORDBL **2.** Return email DNS check, FortiGuard email checksum check, FortiGuard URL check, FortiGuard IP address check, phishing URLs detection **3.** Last hop IP checks local block/allow list **4.** Envelope address checks local block/allow list **5.** Headers IPs local block/allow list **6.** Headers email address local block/allow list, MIME header checks based on local list of patterns (`mheader`) **7.** Banned words (subject first, then body) based on local block/allow list (`bword`)	**1.** Last hop IP checks local block/allow list **2.** Envelope address checks local block/allow list **3.** Headers IPs local block/allow list, MIME header checks based on local list of patterns (`mheader`) **4.** Headers email address local block/allow list **5.** Banned words (subject first, then body) based on local list of patterns (`bword`) **6.** HELO DNS lookup, last hop IP check against ORDBL **7.** Return email DNS check, FortiGuard email checksum check, FortiGuard URL check, FortiGuard IP address check, phishing URLs detection

IMAP, IMAPS, POP3, and POP3S spam filtering order

The FortiGate scans IMAP, IMAPS, POP3, and POP3S email for spam in the following order:

1. MIME headers check, email address block/allow list check
2. Banned word check on email subject
3. IP block/allow list check
4. Banned word check on email body
5. Return email DNS check, FortiGuard email checksum check, FortiGuard URL check, DNSBL and ORDBL checks

IMAPS and POP3S spam filtering are available on FortiGates that support SSL content scanning and inspection.

Protocols and actions

In an email filter profile, there are options to configure settings for SMTP, POP3, IMAP, and MAPI protocols. For each protocol, you can set an action to either discard (block), tag, or pass the log for that protocol. The action options vary per protocol. For the tag action, the spam email can be tagged with configured text in the subject or header.

MAPI is only configurable in the CLI and with the proxy feature set.

To configure protocols in an email filer:

```
config emailfilter profile
    edit <name>
        set feature-set {flow | proxy}
        set spam-filtering enable
        set options {bannedword spambal spamfsip spamfssubmit spamfschksum spamfsurl
spamhelodns spamraddrdns spamrbl spamhdrcheck spamfsphish}
        config smtp
            set log-all {enable | disable}
            set action {pass | tag | discard}
            set tag-type {subject | header | spaminfo}
            set tag-msg <string>
            set hdrip {enable | disable}
            set local-override {enable | disable}
        end
        config imap
            set log-all {enable | disable}
            set action {pass | tag}
            set tag-type {subject | header | spaminfo}
            set tag-msg <string>
        end
        config pop3
            set log-all {enable | disable}
            set action {pass | tag}
```

```
            set tag-type {subject | header | spaminfo}
            set tag-msg <string>
        end
        config mapi
            set log-all {enable | disable}
            set action {pass | discard}
        end
    next
end
```

`options ...`	The following options are available: • bannedword: content block. • spambal: block/allow list. • spamfsip: email IP address FortiGuard antispam block list check. • spamfssubmit: add FortiGuard antispam spam submission text. • spamfschksum: email checksum FortiGuard antispam check. • spamfsurl: email content URL FortiGuard antispam check. • spamhelodns: email HELO/EHLO domain DNS check. • spamraddrdns: email return address DNS check. • spamrbl: email DNSBL and ORBL check. • spamhdrcheck: email MIME header check. • spamfsphish: email content phishing URL FortiGuard antispam check.
`tag-type {subject | header | spaminfo}`	Set the tag type: • subject: prepend text to the spam email subject. • header: append a user-defined MIME header to the spam email. • spaminfo: append spam information to the spam email header.
`tag-msg <string>`	Subject text or header added to the spam email.
`hdrip {enable | disable}`	Enable/disable SMTP email header IP checks for spamfsip, spamrbl, and spambal filters.
`local-override {enable | disable}`	Enable/disable local filter to override SMTP remote check result.

For more information, see config emailfilter profile in the FortiOS CLI Reference.

Configuring webmail filtering

You can configure an email filter to detect and log emails sent by Gmail and Hotmail. These interfaces do not use standard email protocols (SMTP, POP3, or IMAP) and use HTTPS instead. However, you can still configure the email filter to detect emails that pass through the FortiGate.

The FortiGate only detects and logs the emails, it does not discard or tag them.

To configure webmail filtering:

```
config emailfilter profile
    edit <name>
        set spam-filtering enable
        config msn-hotmail
            set log-all enable
        end
        config gmail
            set log-all enable
        end
    next
end
```

Data leak prevention

The FortiGate data leak prevention (DLP) system prevents sensitive data from leaving or entering your network by archiving some or all of the content that passes through the FortiGate. DLP archiving is configured per filter, which allows a single profile to archive only the required data. There are two forms of DLP archiving.

- Summary only: a summary of all the activity detected by the profile is recorded. For example, when an email message is detected, the sender, recipient, message subject, and total size are recorded. When a user accesses a web browser, every URL that they visit is recorded.
- Full: detailed records of all the activity detected by the profile is recorded. For example, when an email message is detected, the message itself, including any attachments, is recorded. When a user accesses a web browser, every page that they visit is archived.

You can configure the DLP archiving protocol in the DLP profile (see config dlp profile). You can customize the default DLP profile or create your own by adding individual filters based on:

- Data types (keyword, regex, hex, credit card, social security number, or custom)
- Dictionaries
- Sensors
- File patterns
- Known files using DLP fingerprinting
- Known files using DLP watermarking

Once configured, you can apply the DLP profile to a firewall policy. Data matching defined sensitive data patterns is blocked, logged, or allowed when it passes through the FortiGate. DLP can only be configured in the CLI.

Filters are ordered, but there is no precedence between the possible actions.

Protocol comparison between DLP inspection modes

The following table indicates which protocols can be inspected by DLP based on the specified inspection modes.

	HTTP	FTP	IMAP	POP3	SMTP	NNTP	MAPI	CIFS	SFTP/SCP
Proxy	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Flow	Yes	Yes	Yes	Yes	Yes	No	No	Yes	No

Logging and blocking files by file name

Sometimes, file names are not accurately recorded in DLP logs, even though the files are blocked correctly based on the DLP profile. This is particularly apparent on cloud-based services, such as Google Drive or SharePoint.

For HTTP file uploads, some cloud services use proprietary encodings and APIs to transfer files and exchange metadata, instead of standard HTTP mechanisms, requiring custom handling of the proprietary API. If a cloud service changes the API without notice, the custom handling becomes outdated and file names might not be logged properly. Due to this, special consideration must be taken when using DLP to block files by file pattern. To block a specific file type, it is better to block by file type, and not by file name pattern.

The following topics provide information about DLP:

- Basic DLP settings on page 1247
- DLP fingerprinting on page 1254

Basic DLP settings

DLP settings can be configured for data types, dictionaries, sensors, file patterns, and profiles. This topic includes three examples that incorporate several DLP settings.

- Block HTTPS upload traffic that includes credit card or social security number (SSN) information.
- Log FTP upload traffic with a specific pattern.
- Block HTTPS downloads of EXE files and log HTTPS downloads of files larger than 500 KB.

DLP data type

This configuration includes five pre-defined data types to match for keyword, regex, hex, credit card, and social security number (SSN). Custom data types can be added.

```
config dlp data-type
    edit "keyword"
        set pattern "built-in"
    next
    edit "regex"
        set pattern "built-in"
    next
    edit "hex"
        set pattern "built-in"
    next
    edit "credit-card"
```

```
        set pattern "\\b([2-6]{1}\\d{3})[- ]?(\\d{4})[- ]?(\\d{2})[- ]?(\\d{2})[- ]?(\\d
{2,4})\\b"
        set verify "built-in"
        set look-back 20
        set transform "\\b\\1[- ]?\\2[- ]?\\3[- ]?\\4[- ]?\\5\\b"
    next
    edit "ssn-us"
        set pattern "\\b(\\d{3})-(\\d{2})-(\\d{4})\\b"
        set verify "(?<!-)\\b(?!666|000|9\\d{2})\\d{3}-(?!00)\\d{2}-(?!0{4})\\d{4}\\b(?!-)"
        set look-back 12
        set transform "\\b\\1-\\2-\\3\\b"
    next
end
```

To add a custom DLP data type:

```
config dlp data-type
    edit <name>
        set pattern <string>
        set verify <string>
        set transform <string>
    next
end
```

`pattern <string>`	Enter a regular expression pattern string without a look around.
`verify <string>`	Enter a regular expression pattern string used to verify the data type.
`transform <string>`	Enter the template to transform user input to a pattern using the capture group from `pattern`.

DLP dictionary

A DLP dictionary is a collection of data type entries.

To configure a DLP dictionary:

```
config dlp dictionary
    edit <name>
        config entries
            edit 1
                set type {credit-card | hex | keyword | regex | ssn-us}
                set pattern <string>
                set repeat {enable | disable}
                set status {enable | disable}
            next
        end
    next
end
```

DLP sensor

A DLP sensor defines which dictionary to check. It counts the number of dictionary matches to trigger the sensor.

To configure a DLP sensor:

```
config dlp sensor
    edit <name>
        set match-type {match-all | match-any | match-eval}
        set eval <string>
        config entries
            edit <id>
                set dictionary <dlp_dictionary>
                set count <integer>
                set status {enable | disable}
            next
        end
    next
end
```

DLP file pattern

A DLP file pattern can block, allow, log, or quarantine a file based on the specified file type in the file filter list (see Supported file types on page 1230).

To configure a DLP file pattern:

```
config dlp filepattern
    edit <id>
        set name <name>
        config entries
            edit <name>
                set filter-type {type | pattern}
                set file-type <file_type>
            next
        end
    next
end
```

DLP profile

A DLP profile allows for filtering by size and file type. DLP profiles can be applied in firewall policies.

To configure a DLP profile:

```
config dlp profile
    edit <name>
        set feature-set {flow | proxy}
        config rule
            edit <id>
                set proto <protocol> <protocol> ...
                set sensor <dlp_sensor>
                set action {allow | log-only | block | quarantine-ip}
            next
        end
    next
end
```

Example 1

This configuration will block HTTPS upload traffic that includes credit card or social security number (SSN) information. The pre-defined data types for `credit-card` and `ssn-us` are used in the dictionary.

To block HTTPS upload traffic that includes credit card or SSN information:

1. Configure the DLP dictionary:

```
config dlp dictionary
    edit "dic-case1-cc-ssn"
        config entries
            edit 1
                set type "credit-card"
            next
            edit 2
                set type "ssn-us"
            next
        end
    next
end
```

2. Configure the DLP sensor:

```
config dlp sensor
    edit "sensor-case1-cc-ssn"
        config entries
            edit 1
                set dictionary "dic-case1-cc-ssn"
            next
        end
    next
end
```

3. Configure the DLP profile:

```
config dlp profile
    edit "profile-case1-cc-ssn"
        config rule
            edit 1
                set proto http-post
                set sensor "sensor-case1-cc-ssn"
                set action block
            next
        end
    next
end
```

4. Add the DLP profile to a firewall policy:

```
config firewall policy
    edit 1
        set srcintf "port2"
        set dstintf "port1"
        set action accept
        set srcaddr "all"
        set dstaddr "all"
        set srcaddr6 "all"
```

```
        set dstaddr6 "all"
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set inspection-mode proxy
        set ssl-ssh-profile "custom-deep-inspection"
        set dlp-profile "profile-case1-cc-ssn"
        set logtraffic all
        set nat enable
    next
end
```

When a credit card or SSN is included in HTTP POST traffic, a replacement message appears because it is blocked. A DLP log is generated.

Sample log

```
5: date=2022-02-15 time=09:49:04 eventtime=1644947344512841971 tz="-0800" logid="0954024576"
type="utm" subtype="dlp" eventtype="dlp" level="warning" vd="root" filteridx=1
dlpextra="sensor-case1-cc-ssn " filtertype="rule" filtercat="file" severity="medium"
policyid=1 poluuid="905fb604-7ed4-51ec-0853-79e498591bf8" policytype="policy" sessionid=9290
epoch=64494265 eventid=0 srcip=10.1.100.106 srcport=64006 srccountry="Reserved"
srcintf="port2" srcintfrole="undefined" srcuuid="358d0f56-7ed4-51ec-50f7-a5e4525a641d"
dstip=35.209.241.59 dstport=443 dstcountry="United States" dstintf="port1"
dstintfrole="undefined" dstuuid="358d0f56-7ed4-51ec-50f7-a5e4525a641d" proto=6
service="HTTPS" filetype="unknown" direction="outgoing" action="block"
hostname="dlptest.com" url="https://dlptest.com/https-post/" agent="Mozilla/5.0 (Windows NT
10.0; Win64; x64) AppleWebKit/537.36 (KH" filename="item_meta[6]" filesize=19
profile="profile-case1-cc-ssn"
```

Example 2

This configuration will log FTP upload traffic with the following patterns:

- keyword = demo
- regex = demo(regex){1,5}
- hex = e6b58be8af95

The dictionary entries have repeat match enabled. The DLP sensor is set so this is repeated five times.

To log FTP upload traffic that has specific keyword, regex, and hex patterns repeated for five times:

1. Configure the DLP dictionary:

```
config dlp dictionary
    edit "dic-case2-keyword-regex-hex"
        config entries
            edit 1
                set type "keyword"
                set pattern "demo"
                set repeat enable
            next
            edit 2
                set type "regex"
                set pattern "demo(regex){1,5}"
```

```
                set repeat enable
            next
            edit 3
                set type "hex"
                set pattern "e6b58be8af95"
                set repeat enable
            next
        end
    next
end
```

2. Configure the DLP sensor:

```
config dlp sensor
    edit "sensor-case2-keyword-regex-hex"
        config entries
            edit 1
                set dictionary "dic-case2-keyword-regex-hex"
                set count 5
            next
        end
    next
end
```

3. Configure the DLP profile:

```
config dlp profile
    edit "profile-case2-keyword-regex-hex"
        config rule
            edit 1
                set proto ftp
                set sensor "sensor-case2-keyword-regex-hex"
                set action log-only
            next
        end
    next
end
```

4. Add the DLP profile to a firewall policy:

```
config firewall policy
    edit 1
        set srcintf "port2"
        set dstintf "port1"
        set action accept
        set srcaddr "all"
        set dstaddr "all"
        set srcaddr6 "all"
        set dstaddr6 "all"
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set inspection-mode proxy
        set ssl-ssh-profile "custom-deep-inspection"
        set dlp-profile "profile-case2-keyword-regex-hex"
        set logtraffic all
        set nat enable
```

```
        next
    end
```

5. Upload a Word document that contains "demo, demo, demo, demoregexregex," using FTP.

A DLP log is generated after the FTP traffic passes.

Sample log

```
3: date=2022-02-15 time=10:42:34 eventtime=1644950554735620032 tz="-0800" logid="0954024577"
type="utm" subtype="dlp" eventtype="dlp" level="notice" vd="root" filteridx=1
dlpextra="sensor-case2-keyword-regex-hex " filtertype="rule" filtercat="file"
severity="medium" policyid=1 poluuid="905fb604-7ed4-51ec-0853-79e498591bf8"
policytype="policy" sessionid=10551 epoch=64494633 eventid=0 srcip=10.1.100.106
srcport=55647 srccountry="Reserved" srcintf="port2" srcintfrole="undefined"
srcuuid="358d0f56-7ed4-51ec-50f7-a5e4525a641d" dstip=35.163.228.146 dstport=1048
dstcountry="United States" dstintf="port1" dstintfrole="undefined" dstuuid="358d0f56-7ed4-
51ec-50f7-a5e4525a641d" proto=6 service="FTP" filetype="msofficex" direction="outgoing"
action="log-only" filename="dlp-test.docx" filesize=11627 profile="profile-case2-keyword-
regex-hex" infectedfilename="word/document.xml" infectedfilesize=2448
infectedfiletype="html" infectedfilelevel=1
```

Example 3

This configuration will block HTTPS downloads of EXE files and log HTTPS downloads of files larger than 500 KB.

To block HTTPS download of EXE files and log downloads larger than 500 KB:

1. Configure the DLP file pattern:

```
config dlp filepattern
    edit 3
        set name "case3-exe"
        config entries
            edit "exe"
                set filter-type type
                set file-type exe
            next
        end
    next
end
```

2. Configure the DLP profile:

```
config dlp profile
    edit "profile-case3-type-size"
        config rule
            edit 1
                set proto http-get
                set filter-by none
                set file-type 3
                set action block
            next
            edit 2
                set proto http-get
                set filter-by none
```

```
                set file-size 500
                set action log-only
            next
        end
    next
end
```

3. Add the DLP profile to a firewall policy:

```
config firewall policy
    edit 1
        set srcintf "port2"
        set dstintf "port1"
        set action accept
        set srcaddr "all"
        set dstaddr "all"
        set srcaddr6 "all"
        set dstaddr6 "all"
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set inspection-mode proxy
        set ssl-ssh-profile "custom-deep-inspection"
        set dlp-profile "profile-case3-type-size"
        set logtraffic all
        set nat enable
    next
end
```

4. Download an EXE file using HTTPS. The download is blocked, a replacement message appears, and a DLP log is generated.

Sample log

```
1: date=2022-02-15 time=11:54:29 eventtime=1644954869682887856 tz="-0800" logid="0954024577"
type="utm" subtype="dlp" eventtype="dlp" level="notice" vd="root" filteridx=2 dlpextra="500
kB" filtertype="none" filtercat="file" severity="medium" policyid=1 poluuid="905fb604-7ed4-
51ec-0853-79e498591bf8" policytype="policy" sessionid=12082 epoch=901683674 eventid=0
srcip=10.1.100.18 srcport=59520 srccountry="Reserved" srcintf="port2"
srcintfrole="undefined" srcuuid="358d0f56-7ed4-51ec-50f7-a5e4525a641d" dstip=51.81.186.201
dstport=443 dstcountry="United States" dstintf="port1" dstintfrole="undefined"
dstuuid="358d0f56-7ed4-51ec-50f7-a5e4525a641d" proto=6 service="HTTPS" direction="incoming"
action="log-only" hostname="2.na.dl.wireshark.org"
url="https://2.na.dl.wireshark.org/win64/Wireshark-win64-3.6.2.exe" agent="curl/7.61.1"
filename="Wireshark-win64-3.6.2.exe" filesize=10502090 profile="profile-case3-type-size"
```

DLP fingerprinting

DLP fingerprinting can be used to detect sensitive data. The file that the DLP profile filters is uploaded and the FortiGate generates and stores a checksum fingerprint. The FortiGate generates a fingerprint for all the files that are detected in network traffic, and compares all the checksums stored in its database. If a match is found, the configured action is taken. Any type of file can be detected by DLP fingerprinting, and fingerprints can be saved for each revision of a file as it is updated.

Using fingerprinting requires:

1. Selecting the files to be fingerprinted by targeting a document source.
2. Adding fingerprinting filters to DLP profiles.
3. Adding the profiles to firewall policies that accept traffic that the fingerprinting will be applied on.

The document fingerprint feature requires a FortiGate that has internal storage.

To configure a DLP fingerprint document:

```
config dlp fp-doc-source
    edit <name>
        set server-type smb
        set server <string>
        set period {none | daily | weekly | monthly}
        set vdom {mgmt | current}
        set scan-subdirectories {enable | disable}
        set remove-deleted {enable | disable}
        set keep-modified {enable | disable}
        set username <string>
        set password <password>
        set file-path <string>
        set file-pattern <string>
        set sensitivity <Critical | Private | Warning>
        set tod-hour <integer>
        set tod-min <integer>
        set weekday {sunday | monday | tuesday | wednesday | thursday | friday | saturday}
        set date <integer>
    next
end
```

Command	Description
`server-type smb`	Set the protocol used to communicate with document server. Only Samba (SMB) servers are supported.
`server <string>`	Enter the IPv4 or IPv6 address of the server.
`period {none | daily | weekly | monthly}`	Set the frequency that the FortiGate checks the server for new or changed files.
`vdom {mgmt | current}`	Enter the VDOM that can communicate with the file server.
`scan-subdirectories {enable | disable}`	Enable/disable scanning subdirectories to find files.
`remove-deleted {enable | disable}`	Enable/disable keeping the fingerprint database up to date when a file is deleted from the server.
`keep-modified {enable | disable}`	Enable/disable keeping the old fingerprint and adding a new one when a file is changed on the server.
`username <string>`	Enter the user name required to log into the file server.
`password <password>`	Enter the password required to log into the file server.

Command	Description
file-path <string>	Enter the path on the server to the fingerprint files.
file-pattern <string>	Enter the pattern for matching files on the server to be fingerprinted.
sensitivity <Critical | Private | Warning>	Set the sensitivity or threat level for matches with this fingerprint database.
tod-hour <integer>	Set the hour of the day. This option is only available when period is not none.
tod-min <integer>	Set the minute of the hour. This option is only available when period is not none.
weekday {sunday | monday | tuesday | wednesday | thursday | friday | saturday}	Set the day of the week. This option is only available when period is weekly.
date <integer>	Set the day of the month. This option is only available when period is monthly.

To configure a DLP fingerprint profile:

```
config dlp profile
    edit <name>
        config filter
            edit <id>
                set proto {smtp | pop3 | imap http-get | http-post | ftp | nntp | mapi}
                set filter-by fingerprint
                set sensitivity {Critical | Private | Warning}
                set match-percentage <integer>
                set action {allow | log-only | block | quarantine-ip}
            next
        end
    next
end
```

Command	Description
proto {smtp | pop3 | imap http-get | http-post | ftp | nntp | mapi}	Set the protocol to inspect.
filter-by fingerprint	Set to match against a fingerprint sensitivity.
sensitivity {Critical | Private | Warning}	Set the DLP file pattern sensitivity to match.
match-percentage <integer>	Set the percentage of the checksum required to match before the profile is triggered.
action {allow | log-only | block | quarantine-ip}	Set the action to take with content that matches the DLP profile.

View the DLP fingerprint database on the FortiGate

Use `diagnose test application dlpfingerprint <integer>` to display the fingerprint information that is on the FortiGate.

Integer	Function
1	Show the fingerprint daemon menu
2	Dump the database
3	Dump all files
5	Dump all chunks
6	Refresh all document sources in all VDOMs
7	Show the database file size and limit
9	Display statistics
10	Clear statistics
99	Restart this daemon

To dump all fingerprinted files:

```
# diagnose test application dlpfingerprint 3
DLPFP diag_test_handler called
File DB:
---------------------------------------
id, filename,                             vdom, archive, deleted, scanTime,   docSourceSrvr,
sensitivity, chunkCnt, reviseCnt,
1,  /fingerprint/upload/1.txt,            vdom1,  0,       0,       1494868196,   1,       2,
1,    0,
2,  /fingerprint/upload/30percentage.xls,   vdom1,  0,       0,       1356118250,   1,       2,
13,      0,
3,  /fingerprint/upload/50.pdf,            vdom1,  0,       0,       1356118250,   1,       2,
122,  0,
4,  /fingerprint/upload/50.pdf.tar.gz,      vdom1,  0,       0,       1356118250,   1,       2,
114,     0,
5,  /fingerprint/upload/check-list_AL-SIP_HA.xls,   vdom1,  0,       0,       1356118251,     1,
2,       32,      0,
6,  /fingerprint/upload/clean.zip,          vdom1,  0,       0,       1356118251,   1,       2,
1,    0,
7,  /fingerprint/upload/compare.doc,        vdom1,  0,       0,       1522097410,   1,       2,
18,      0,
8,  /fingerprint/upload/dlpsensor-watermark.pdf,    vdom1,  0,       0,       1356118250,     1,
2,       11,      0,
9,  /fingerprint/upload/eicar.com,          vdom1,  0,       0,       1356118250,   1,       2,
1,    0,
10, /fingerprint/upload/eicar.zip,          vdom1,  0,       0,       1356118250,   1,       2,
1,    0,
11, /fingerprint/upload/EMAIL-CONTENT-ARCHIVE.ppt,  vdom1,  0,       0,       1356118250,     1,
2,       11,      0,
12, /fingerprint/upload/encrypt.zip,        vdom1,  0,       0,       1356118250,   1,       2,
77,      0,
13, /fingerprint/upload/extension_7_8_1.crx,        vdom1,  0,       0,       1528751781,     1,
2,       2720,    0,
14, /fingerprint/upload/fingerprint.txt,    vdom1,  0,       0,       1498582679,   1,       2,
```

```
37,     0,
15, /fingerprint/upload/fingerprint90.txt,  vdom1,  0,      0,      1498582679,   1,      2,
37,     0,
16, /fingerprint/upload/fo2.pdf,            vdom1,  0,      0,      1450488049,   1,      2,
1,    0,
17, /fingerprint/upload/foo.doc,            vdom1,  0,      0,      1388538131,   1,      2,
9,    0,
18, /fingerprint/upload/fortiauto.pdf,      vdom1,  0,      0,      1356118251,   1,      2,
146,    0,
19, /fingerprint/upload/image.out,          vdom1,  0,      0,      1531802940,   1,      2,
5410, 0,
20, /fingerprint/upload/jon_file.txt,       vdom1,  0,      0,      1536596091,   1,      2,    1,
      0,
21, /fingerprint/upload/machotest,          vdom1,  0,      0,      1528751955,   1,      2,
19,   0,
22, /fingerprint/upload/nntp-server.doc,    vdom1,  0,      0,      1356118250,   1,      2,
17,     0,
23, /fingerprint/upload/notepad++.exe,      vdom1,  0,      0,      1456090734,   1,      2,
1061,   0,
24, /fingerprint/upload/nppIExplorerShell.exe,      vdom1,  0,      0,      1438559930,     1,
2,      5,      0,
25, /fingerprint/upload/NppShell_06.dll,    vdom1,  0,      0,      1456090736,   1,      2,
111,    0,
26, /fingerprint/upload/PowerCollections.chm,       vdom1,  0,      0,      1533336889,     1,
2,      728,    0,
27, /fingerprint/upload/reflector.dmg,      vdom1,  0,      0,      1533336857,   1,      2,
21117,  0,
28, /fingerprint/upload/roxio.iso,          vdom1,  0,      0,      1517531765,   1,      2,
49251,0,
29, /fingerprint/upload/SciLexer.dll,       vdom1,  0,      0,      1456090736,   1,      2,
541,    0,
30, /fingerprint/upload/screen.jpg,         vdom1,  0,      0,      1356118250,   1,      2,
55,   0,
31, /fingerprint/upload/Spec to integrate FASE into FortiOS.doc,    vdom1,  0,    0,
1356118251,    1,      2,      31,     0,
32, /fingerprint/upload/subdirectory1/subdirectory2/subdirectory3/hibun.aea,       vdom1,  0,
0,      1529019743,     1,      2,      1,      0,
33, /fingerprint/upload/test.pdf,           vdom1,  0,      0,      1356118250,   1,      2,
5,    0,
34, /fingerprint/upload/test.tar,           vdom1,  0,      0,      1356118251,   1,      2,
3,    0,
35, /fingerprint/upload/test.tar.gz,        vdom1,  0,      0,      1356118250,   1,      2,    1,
      0,
36, /fingerprint/upload/test1.txt,          vdom1,  0,      0,      1540317547,   1,      2,
1,    0,
37, /fingerprint/upload/thousand-files.zip, vdom1,  0,      0,      1536611774,   1,      2,
241,    0,
38, /fingerprint/upload/Thumbs.db,          vdom1,  0,      0,      1445878135,   1,      2,
3,    0,
39, /fingerprint/upload/widget.pdf,         vdom1,  0,      0,      1356118251,   1,      2,
18,   0,
40, /fingerprint/upload/xx00-xx01.tar,      vdom1,  0,      0,      1356118250,   1,      2,    5,
      0,
41, /fingerprint/upload/xx02-xx03.tar.gz,   vdom1,  0,      0,      1356118251,   1,      2,    1,
      0,
```

VoIP solutions

You can configure VoIP profiles to allow SIP and SCCP traffic and to protect your network from SIP- and SCCP-based attacks.

FortiOS includes two preloaded VoIP profiles:

- *default*
- *strict*

You can customize these profiles, or you can create your own and add them to firewall policies that allow VoIP.

VoIP profiles cannot be used NGFW policy-based mode. See NGFW policy on page 772 for more information.

The following topics provide information about VoIP profiles:

General use cases

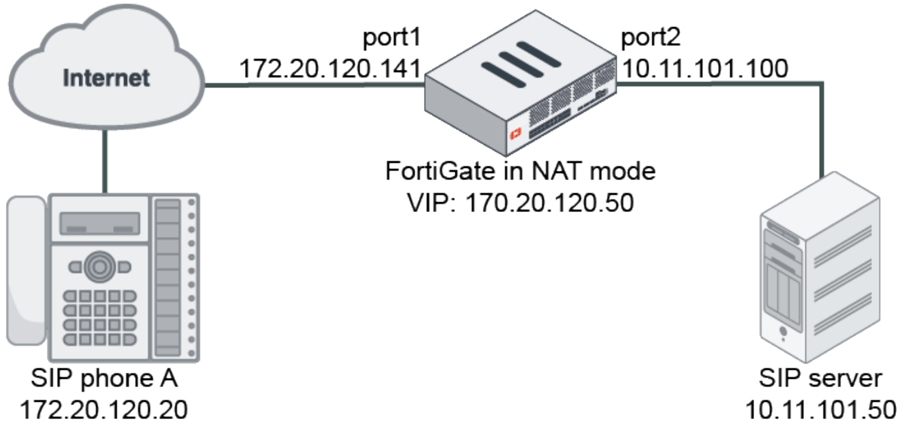
There are three scenarios in which the FortiOS session initiation protocol (SIP) solution is usually deployed:

1. The SIP server is in a private network that is protected from the internet by a FortiGate.
2. The SIP clients are in a private network that is protected from the internet by a FortiGate.
3. The SIP server is in a private network, such as a corporation's internal network or an ISP's network, that is protected from the internet by a FortiGate. The SIP clients are in a remote private network, such as a SOHO network, and behind a NAT device that is not aware of SIP applications.

The following VIP, NAT, and HNT examples show configurations for these common scenarios.

VIP

A FortiGate with SIP Application Layer Gateway (ALG) or SIP session helper protects the SIP server from the internet, while SIP phones from the internet need to register to the SIP server and establish calls through it.



A VIP needs to be configured for the SIP server, and the VIP must be applied in a firewall policy for the phones to send REGISTER messages through the FortiGate from port1 to port2.

Only one firewall policy needs to be configured for all SIP phones on both the internet and private network to register to the SIP server through port1 and set up SIP calls. This example assumes either SIP ALG or SIP session helper is enabled.

To configure the VIP for the SIP server:

```
config firewall vip
    edit "VIP_for_SIP_Server"
        set extip 172.20.120.50
        set extintf "port1"
        set mappedip "10.11.101.50"
    next
end
```

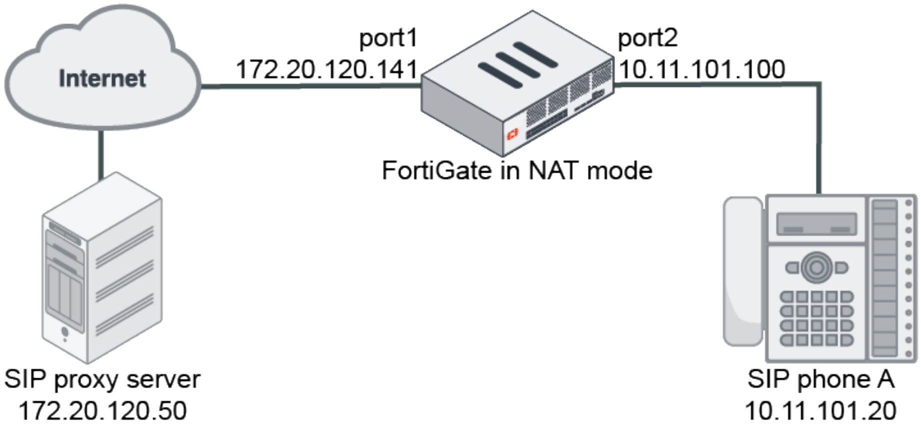
To configure the firewall policy:

```
config firewall policy
    edit 1
        set srcintf "port1"
        set dstintf "port2"
        set srcaddr "all"
        set dstaddr "VIP_for_SIP_Server"
        set action accept
        set schedule "always"
        set service "SIP"
    next
end
```

Setting `service` to `SIP` and not `all` in the firewall policy can improve protection by restricting the data traffic passing through the FortiGate to the SIP call traffic only.

NAT

A FortiGate with SIP ALG or SIP session helper protects the SIP phones and the internal network from the internet, while SIP phones in the internal network need to register to the SIP server installed on the internet and establish calls through it.



One firewall policy needs to be configured with NAT enabled for SIP phones to send REGISTER messages through the FortiGate from port2 to port1. This example assumes either SIP ALG or SIP session helper is enabled.

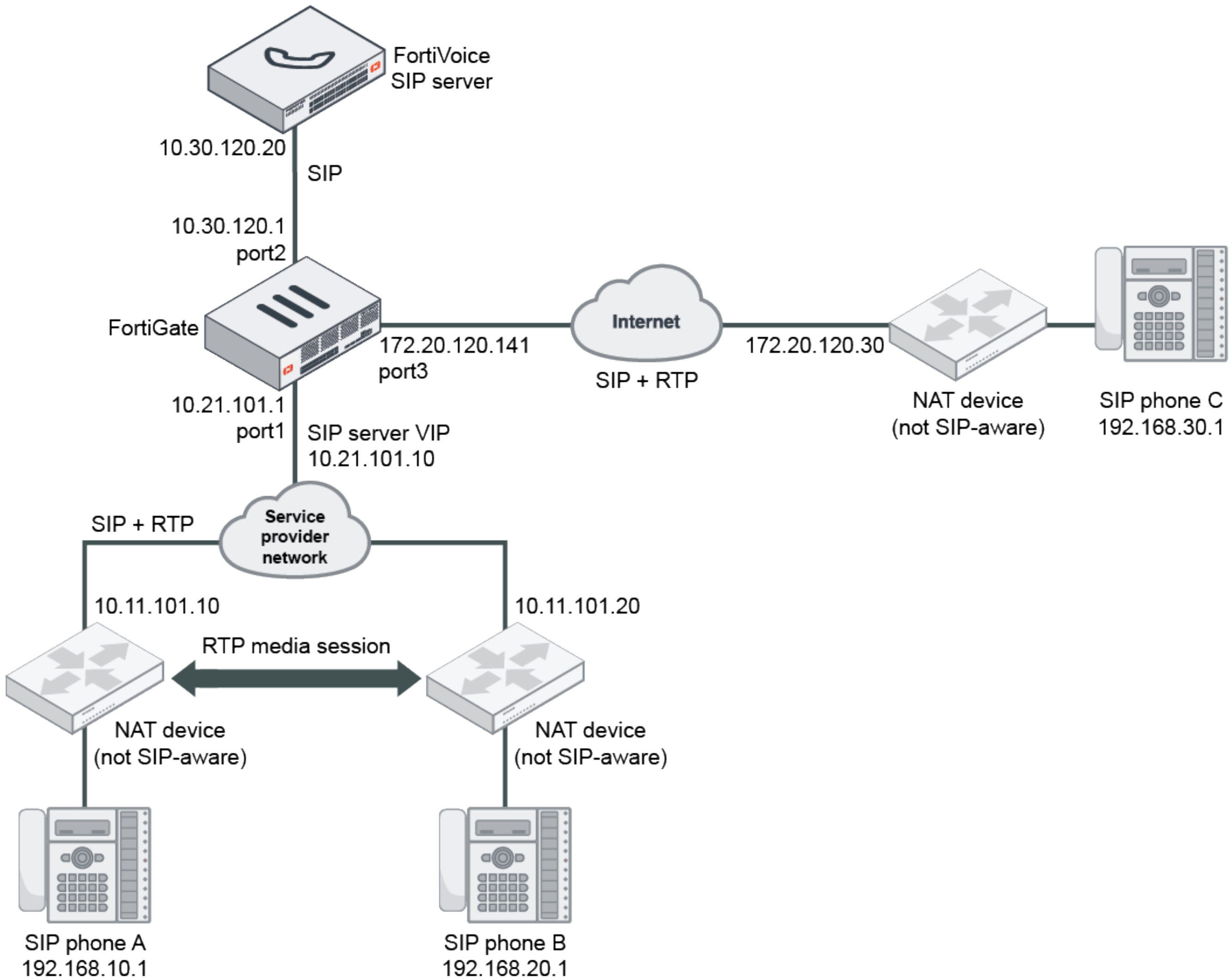
To configure the firewall policy:

```
config firewall policy
    edit 1
        set srcintf "port2"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "SIP"
        set nat enable
    next
end
```

HNT

A FortiGate with SIP ALG protects the SIP server from the internet, while SIP phones are in remote private networks behind NAT devices that are not aware of the SIP application. This is only supported in proxy mode.

In this example, the SIP server is located in an ISP's service cloud that is protected by the FortiGate SIP ALG, and the SIP phones are installed in the home networks of the ISP's customers.



The SIP messages traversing the remote NAT devices might have their IP addresses translated by the NAT device at the network layer, but untranslated at the SIP application layer because those NAT devices are not aware of the SIP applications. This causes problems in a SIP session initiated process. Special configurations for the hosted NAT traversal (HNT) are required to resolve this issue.

To configure the FortiGate with HNT support for SIP phones A and B to set up calls with each other:

1. Identify port1 as the external interface:

```
config system interface
    edit "port1"
        set external enable
    next
end
```

2. Configure the VIP for the SIP server:

```
config firewall vip
    edit "VIP_for_SIP_Server"
        set extip 10.21.101.10
```

```
        set extintf "port1"
        set mappedip "10.30.120.20"
    next
end
```

3. Configure a VoIP profile with HNT enabled:

```
config voip profile
    edit "hnt"
        config sip
            set hosted-nat-traversal enable
            set hnt-restrict-source-ip enable
        end
    next
end
```

`hosted-nat-traversal` must be enabled. `hnt-restrict-source-ip` does not have to be enabled, but can be enabled to restrict the RTP packets' source IP to be the same as the SIP packets' source IP.

4. Apply the VoIP profile and VIP in a firewall policy for phone A and B to register and set up SIP calls through the FortiGate and SIP server:

```
config firewall policy
    edit 1
        set srcintf "port1"
        set dstintf "port2"
        set srcaddr "all"
        set dstaddr "VIP_for_SIP_Server"
        set action accept
        set schedule "always"
        set service "SIP"
        set utm-status enable
        set voip-profile "hnt"
        set nat enable
    next
end
```

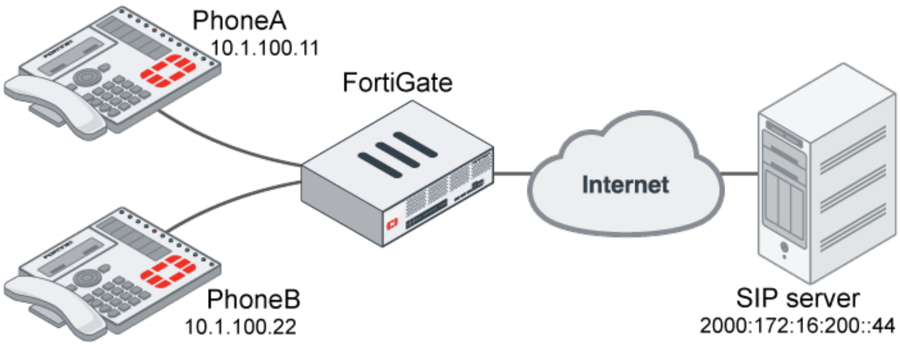
`nat` must be enabled in the firewall policy.

NAT46 and NAT64 for SIP ALG

NAT46 and NAT64 are supported for SIP ALG. A mix of IPv4 and IPv6 networks can use SIP ALG, allowing for proper call handling.

NAT46 example

In this example, SIP phones on the internal network use IPv4, and the SIP server on an external network uses IPv6. NAT46 is used with SIP ALG to allow for seamless communication. A VoIP profile, `sip`, has already been created.



To configure the FortiGate:

1. Configure a firewall VIP with NAT46 enabled:

```
config firewall vip
    edit "vip46_server_asterisk"
        set extip 10.1.100.100
        set nat44 disable
        set nat46 enable
        set extintf "port1"
        set ipv6-mappedip 2000:172:16:200::44
    next
end
```

2. Configure an IPv6 pool:

```
config firewall ippool6
    edit "client_server_nat46"
        set startip 2000:172:16:200::200
        set endip 2000:172:16:200::207
        set nat46 enable
    next
end
```

3. Configure a firewall policy:

```
config firewall policy
    edit 1
        set name "policy46-1"
        set srcintf "port1"
        set dstintf "port9"
        set action accept
        set nat46 enable
        set srcaddr "all"
        set dstaddr "vip46_server_asterisk"
        set srcaddr6 "all"
        set dstaddr6 "all"
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set inspection-mode proxy
        set voip-profile "sip"
        set logtraffic all
        set auto-asic-offload disable
        set ippool enable
```

```
        set poolname6 "client_server_nat46"
    next
end
```

To check the SIP calls and session lists when the phones are registering to the SIP server:

1. View the SIP proxy SIP calls:

```
# diagnose sys sip-proxy calls
sip calls
  vdom 3 (vdom1) vrf 0 call 7f64bf044b00
    call-id: 1513782757
    txn 7f64bf048f00 (REGISTER)
      cseq 2 dir 0 state 5 status 200 expiry 868 HA 0
      i_session: 7f64bf045e00  r_session: 7f64bf045e00
      register: present
      from: sip:2002@10.1.100.100
      to: sip:2002@10.1.100.100
      src: 10.1.100.22:5060
      dst: [2000:172:16:200::44]:5060

  vdom 3 (vdom1) vrf 0 call 7f64bf076700
    call-id: 1490871789
    txn 7f64bf047a00 (REGISTER)
      cseq 2 dir 0 state 5 status 200 expiry 861 HA 0
      i_session: 7f64bf045000  r_session: 7f64bf045000
      register: present
      from: sip:2001@10.1.100.100
      to: sip:2001@10.1.100.100
      src: 10.1.100.11:5060
      dst: [2000:172:16:200::44]:5060
```

2. View the IPv4 session list:

```
# diagnose sys session list

orgin->sink: org pre->post, reply pre->post dev=9->52/52->9 gwy=10.1.100.100/10.1.100.11
hook=pre dir=org act=noop 10.1.100.11:5060->10.1.100.100:5060(0.0.0.0:0)
hook=post dir=reply act=noop 10.1.100.100:5060->10.1.100.11:5060(0.0.0.0:0)
peer=2000:172:16:200::203:65476->2000:172:16:200::44:5060 naf=1
hook=pre dir=org act=noop 2000:172:16:200::203:65476->2000:172:16:200::44:5060(:::0)
hook=post dir=reply act=noop 2000:172:16:200::44:5060->2000:172:16:200::203:65476(:::0)

orgin->sink: org pre->post, reply pre->post dev=9->52/52->9 gwy=10.1.100.100/10.1.100.22
hook=pre dir=org act=noop 10.1.100.22:5060->10.1.100.100:5060(0.0.0.0:0)
hook=post dir=reply act=noop 10.1.100.100:5060->10.1.100.22:5060(0.0.0.0:0)
peer=2000:172:16:200::200:65476->2000:172:16:200::44:5060 naf=1
hook=pre dir=org act=noop 2000:172:16:200::200:65476->2000:172:16:200::44:5060(:::0)
hook=post dir=reply act=noop 2000:172:16:200::44:5060->2000:172:16:200::200:65476(:::0)
```

3. View the IPv4 expectation session list:

```
# diagnose sys session list expectation

orgin->sink: org pre->post, reply pre->post dev=9->0/52->0 gwy=0.0.0.0/0.0.0.0
hook=pre dir=org act=noop 10.1.100.100:0->10.1.100.11:5060(0.0.0.0:0)
hook=pre dir=org act=noop 0.0.0.0:0->0.0.0.0:0(0.0.0.0:0)
peer=:::0->:::0 naf=2
```

```
orgin->sink: org pre->post, reply pre->post dev=9->0/52->0 gwy=0.0.0.0/0.0.0.0
hook=pre dir=org act=noop 10.1.100.100:0->10.1.100.22:5060(0.0.0.0:0)
hook=pre dir=org act=noop 0.0.0.0:0->0.0.0.0:0(0.0.0.0:0)
peer=:::0->:::0 naf=2
```

4. View the IPv6 session list:

```
# diagnose sys session6 list

hook=pre dir=org act=noop 2000:172:16:200::203:65476->2000:172:16:200::44:5060(:::0)
hook=post dir=reply act=noop 2000:172:16:200::44:5060->2000:172:16:200::203:65476(:::0)
peer=10.1.100.100:5060->10.1.100.11:5060 naf=2
hook=pre dir=org act=noop 10.1.100.11:5060->10.1.100.100:5060(0.0.0.0:0)
hook=post dir=reply act=noop 10.1.100.100:5060->10.1.100.11:5060(0.0.0.0:0)

hook=pre dir=org act=noop 2000:172:16:200::200:65476->2000:172:16:200::44:5060(:::0)
hook=post dir=reply act=noop 2000:172:16:200::44:5060->2000:172:16:200::200:65476(:::0)
peer=10.1.100.100:5060->10.1.100.22:5060 naf=2
hook=pre dir=org act=noop 10.1.100.22:5060->10.1.100.100:5060(0.0.0.0:0)
hook=post dir=reply act=noop 10.1.100.100:5060->10.1.100.22:5060(0.0.0.0:0)
```

5. View the IPv6 expectation session list:

```
# diagnose sys session6 list expectation

orgin->sink: org pre->post, reply pre->post dev=17->0/52->0
hook=post dir=org act=noop 2000:172:16:200::44:0->2000:172:16:200::200:65476(:::0)
hook=pre dir=org act=noop :::0->:::0(:::0)
peer=10.1.100.100:0->10.1.100.22:5060 naf=1

orgin->sink: org pre->post, reply pre->post dev=17->0/52->0
hook=post dir=org act=noop 2000:172:16:200::44:0->2000:172:16:200::203:65476(:::0)
hook=pre dir=org act=noop :::0->:::0(:::0)
peer=10.1.100.100:0->10.1.100.11:5060 naf=1
```

To check the SIP calls and session lists when one phone is calling another phone:

1. View the SIP proxy SIP calls:

```
# diagnose sys sip-proxy calls

sip calls
  vdom 3 (vdom1) vrf 0 call 7f64bf057a00
    call-id: 217ac4733f80ac766c7e0f3a69d317a1@[2000:172:16:200::44]:5060
    txn 7f64bf038800 (INVITE)
      cseq 103 dir 1 state 11 status 200 expiry 252 HA 0
      i_session: 7f64bf036500  r_session: 7f64bf036500
      register: not-present
      contact[0]:  factory 7f64bf057900/4 expectation 7f64bf02cf00/2 session
7f64bf036500
      contact[1]:  factory 7f64bf057700/3 expectation 7f64bf02ca00/3 session
7f64bf036500
      from: sip:2001@[2000:172:16:200::44]
      to: sip:2002@[2000:172:16:200::200]:65476;o=10.1.100.22;line=28c59e086cac7c9
      src: [2000:172:16:200::44]:5060
      dst: 10.1.100.22:5060
```

```
  vdom 3 (vdom1) vrf 0 call 7f64bf057a00
    call-id: 217ac4733f80ac766c7e0f3a69d317a1@[2000:172:16:200::44]:5060
    txn 7f64bf038100 (INVITE)
      cseq 102 dir 1 state 11 status 200 expiry 252 HA 0
      i_session: 7f64bf036500  r_session: 7f64bf036500
      register: not-present
      contact[0]:  factory 7f64bf057900/4 expectation 7f64bf02cf00/2 session
7f64bf036500
      contact[1]:  factory 7f64bf057700/3 expectation 7f64bf02ca00/3 session
7f64bf036500
      from: sip:2001@[2000:172:16:200::44]
      to: sip:2002@[2000:172:16:200::200]:65476;o=10.1.100.22;line=28c59e086cac7c9
      src: [2000:172:16:200::44]:5060
      dst: 10.1.100.22:5060

  vdom 3 (vdom1) vrf 0 call 7f64bf057600
    call-id: 1876706695
    txn 7f64bf037300 (REGISTER)
      cseq 2 dir 0 state 5 status 200 expiry 856 HA 0
      i_session: 7f64bf036500  r_session: 7f64bf036500
      register: present
      from: sip:2002@10.1.100.100
      to: sip:2002@10.1.100.100
      src: 10.1.100.22:5060
      dst: [2000:172:16:200::44]:5060

  vdom 3 (vdom1) vrf 0 call 7f64bf057400
    call-id: 1372246794
    txn 7f64bf035e00 (REGISTER)
      cseq 2 dir 0 state 5 status 200 expiry 853 HA 0
      i_session: 7f64bf035000  r_session: 7f64bf035000
      register: present
      from: sip:2001@10.1.100.100
      to: sip:2001@10.1.100.100
      src: 10.1.100.11:5060
      dst: [2000:172:16:200::44]:5060

  vdom 3 (vdom1) vrf 0 call 7f64bf057800
    call-id: 16530657
    txn 7f64bf038f00 (INVITE)
      cseq 102 dir 1 state 11 status 200 expiry 252 HA 0
      i_session: 7f64bf035000  r_session: 7f64bf035000
      register: not-present
      contact[0]:  factory 7f64bf057900/4 expectation 7f64bf02cc80/2 session
7f64bf035000
      contact[1]:  factory 7f64bf057500/3 expectation 7f64bf02c780/3 session
7f64bf035000
      from: sip:2002@[2000:172:16:200::44]
      to: sip:2001@[2000:172:16:200::44]
      src: [2000:172:16:200::44]:5060
      dst: 10.1.100.11:5060

  vdom 3 (vdom1) vrf 0 call 7f64bf057800
    call-id: 16530657
    txn 7f64bf037a00 (INVITE)
      cseq 21 dir 0 state 11 status 200 expiry 252 HA 0
```

```
      i_session: 7f64bf035000  r_session: 7f64bf035000
      register: not-present
      contact[0]:  factory 7f64bf057500/3 expectation 7f64bf02c780/3 session
7f64bf035000
      contact[1]:  factory 7f64bf057900/4 expectation 7f64bf02cc80/2 session
7f64bf035000
      from: sip:2001@10.1.100.100
      to: sip:2002@10.1.100.100
      src: 10.1.100.11:5060
      dst: [2000:172:16:200::44]:5060
```

2. View the IPv6 session list:

```
# diagnose sys session6 list

hook=pre dir=org act=noop 2000:172:16:200::203:17078->2000:172:16:200::44:17090(:::0)
hook=post dir=reply act=noop 2000:172:16:200::44:17090->2000:172:16:200::203:17078(:::0)
peer=10.1.100.100:17090->10.1.100.11:17078 naf=2

hook=pre dir=org act=noop 2000:172:16:200::200:17078->2000:172:16:200::44:17082(:::0)
hook=post dir=reply act=noop 2000:172:16:200::44:17082->2000:172:16:200::200:17078(:::0)
peer=10.1.100.100:17082->10.1.100.22:17078 naf=2
hook=pre dir=org act=noop 10.1.100.22:17078->10.1.100.100:17082(0.0.0.0:0)
hook=post dir=reply act=noop 10.1.100.100:17082->10.1.100.22:17078(0.0.0.0:0)

hook=pre dir=org act=noop 2000:172:16:200::203:65476->2000:172:16:200::44:5060(:::0)
hook=post dir=reply act=noop 2000:172:16:200::44:5060->2000:172:16:200::203:65476(:::0)
peer=10.1.100.100:5060->10.1.100.11:5060 naf=2
hook=pre dir=org act=noop 10.1.100.11:5060->10.1.100.100:5060(0.0.0.0:0)
hook=post dir=reply act=noop 10.1.100.100:5060->10.1.100.11:5060(0.0.0.0:0)

hook=pre dir=org act=noop 2000:172:16:200::200:65476->2000:172:16:200::44:5060(:::0)
hook=post dir=reply act=noop 2000:172:16:200::44:5060->2000:172:16:200::200:65476(:::0)
peer=10.1.100.100:5060->10.1.100.22:5060 naf=2
hook=pre dir=org act=noop 10.1.100.22:5060->10.1.100.100:5060(0.0.0.0:0)
hook=post dir=reply act=noop 10.1.100.100:5060->10.1.100.22:5060(0.0.0.0:0)
```

3. View the IPv6 expectation session list:

```
# diagnose sys session6 list expectation

hook=post dir=org act=noop 2000:172:16:200::44:0->2000:172:16:200::203:65476(:::0)
hook=pre dir=org act=noop :::0->:::0(:::0)
peer=10.1.100.100:0->10.1.100.11:5060 naf=1
```

4. View the IPv4 session list:

```
# diagnose sys session list

orgin->sink: org pre->post, reply pre->post dev=9->52/52->9 gwy=10.1.100.100/10.1.100.22
hook=pre dir=org act=noop 10.1.100.22:17078->10.1.100.100:17082(0.0.0.0:0)
hook=post dir=reply act=noop 10.1.100.100:17082->10.1.100.22:17078(0.0.0.0:0)
peer=2000:172:16:200::200:17078->2000:172:16:200::44:17082 naf=1
hook=pre dir=org act=noop 2000:172:16:200::200:17078->2000:172:16:200::44:17082(:::0)
hook=post dir=reply act=noop 2000:172:16:200::44:17082->2000:172:16:200::200:17078(:::0)

orgin->sink: org pre->post, reply pre->post dev=9->52/52->9 gwy=10.1.100.100/10.1.100.22
hook=pre dir=org act=noop 10.1.100.22:5060->10.1.100.100:5060(0.0.0.0:0)
```

```
hook=post dir=reply act=noop 10.1.100.100:5060->10.1.100.22:5060(0.0.0.0:0)
peer=2000:172:16:200::200:65476->2000:172:16:200::44:5060 naf=1
hook=pre dir=org act=noop 2000:172:16:200::200:65476->2000:172:16:200::44:5060(:::0)
hook=post dir=reply act=noop 2000:172:16:200::44:5060->2000:172:16:200::200:65476(:::0)

orgin->sink: org pre->post, reply pre->post dev=9->52/52->9 gwy=10.1.100.100/10.1.100.11
hook=pre dir=org act=noop 10.1.100.11:5060->10.1.100.100:5060(0.0.0.0:0)
hook=post dir=reply act=noop 10.1.100.100:5060->10.1.100.11:5060(0.0.0.0:0)
peer=2000:172:16:200::203:65476->2000:172:16:200::44:5060 naf=1
hook=pre dir=org act=noop 2000:172:16:200::203:65476->2000:172:16:200::44:5060(:::0)
hook=post dir=reply act=noop 2000:172:16:200::44:5060->2000:172:16:200::203:65476(:::0)
```

5. View the IPv4 expectation session list:

```
# diagnose sys session list expectation

orgin->sink: org pre->post, reply pre->post dev=9->0/52->0 gwy=0.0.0.0/0.0.0.0
hook=pre dir=org act=noop 10.1.100.100:0->10.1.100.11:5060(0.0.0.0:0)
hook=pre dir=org act=noop 0.0.0.0:0->0.0.0.0:0(0.0.0.0:0)
peer=:::0->:::0 naf=2

orgin->sink: org pre->post, reply pre->post dev=9->0/52->0 gwy=0.0.0.0/0.0.0.0
hook=pre dir=org act=noop 10.1.100.100:0->10.1.100.22:17078(0.0.0.0:0)
peer=:::0->:::0 naf=2

orgin->sink: org pre->post, reply pre->post dev=9->0/52->0 gwy=0.0.0.0/0.0.0.0
hook=pre dir=org act=noop 10.1.100.100:0->10.1.100.22:17079(0.0.0.0:0)
peer=:::0->:::0 naf=2

orgin->sink: org pre->post, reply pre->post dev=9->0/52->0 gwy=0.0.0.0/0.0.0.0
hook=post dir=org act=noop 10.1.100.22:0->10.1.100.100:17083(0.0.0.0:0)
peer=2000:172:16:200::200:17085->2000:172:16:200::44:17903 naf=1
```

Log messages

When the phones are registering to the SIP server:

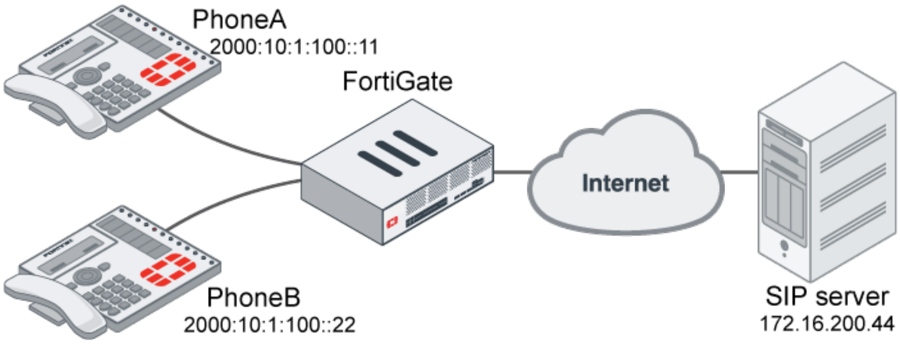
```
date=2022-02-17 time=16:44:47 eventtime=1645145087805236720 tz="-0800" logid="0814044032"
type="utm" subtype="voip" eventtype="voip" level="information" vd="vdom1" session_id=924
epoch=0 event_id=9 srcip=10.1.100.11 src_port=5060 dstip=2000:172:16:200::44 dst_port=5060
proto=17 src_int="port1" dst_int="port9" policy_id=1 profile="sip" voip_proto="sip"
kind="register" action="permit" status="authentication-required" duration=0 dir="session_
origin" call_id="1868762230" from="sip:2001@10.1.100.100" to="sip:2001@10.1.100.100"
```

When one phone is calling another phone:

```
date=2022-02-17 time=16:44:53 eventtime=1645145093351288241 tz="-0800" logid="0814044032"
type="utm" subtype="voip" eventtype="voip" level="information" vd="vdom1" session_id=924
epoch=0 event_id=11 srcip=10.1.100.11 src_port=5060 dstip=2000:172:16:200::44 dst_port=5060
proto=17 src_int="port1" dst_int="port9" policy_id=1 profile="sip" voip_proto="sip"
kind="call" action="permit" status="start" duration=0 dir="session_origin" call_
id="133636365" from="sip:2001@10.1.100.100" to="sip:2002@10.1.100.100"
```

NAT64 example

In this example, SIP phones on the internal network use IPv6, and the SIP server on an external network uses IPv4. NAT64 is used with SIP ALG to allow for seamless communication. A VoIP profile, `sip`, has already been created.



To configure the FortiGate:

1. Configure a firewall VIP with NAT64 enabled:

```
config firewall vip
    edit "vip64-1-asterisk"
        set extip 2000:10:1:100::100
        set nat66 disable
        set nat64 enable
        set ipv4-mappedip 172.16.200.44
    next
end
```

2. Configure an IP pool:

```
config firewall ippool
    edit "client_server_nat46"
        set startip 172.16.200.2
        set endip 172.16.200.3
        set nat64 enable
    next
end
```

3. Configure a firewall policy:

```
config firewall policy
    edit 1
        set name "policy64-1"
        set srcintf "port1"
        set dstintf "port9"
        set action accept
        set nat64 enable
        set srcaddr "all"
        set dstaddr "all"
        set srcaddr6 "all"
        set dstaddr6 "vip64-1-asterisk"
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set inspection-mode proxy
```

```
            set voip-profile "sip"
            set logtraffic all
            set auto-asic-offload disable
            set ippool enable
            set poolname "client_server_nat64"
        next
    end
```

SIP message inspection and filtering

SIP ALG provides users with security features to inspect and control SIP messages that are transported through the FortiGate, including:

- Verifying the SIP message syntax.
- Blocking particular types of SIP requests.
- Restricting the rate of particular SIP requests.

These can be performed in both proxy-based or flow-based firewall policies. In 7.0, flow-based SIP inspection is done by the IPS engine. This optimizes memory and CPU usage when VoIP profiles with SIP inspection are configured with other UTM profiles in a flow-based firewall policy because inspection is done entirely by the IPS engine.

These features are configured in the VoIP profile:

```
config voip profile
    edit <name>
        set feature-set {proxy | flow}
        config sip
            set ...
            ...
        end
    next
end
```

For more information, see config voip profile in the FortiOS CLI Reference.

The VoIP profile can then be applied to a firewall policy to process the SIP call traffic. The firewall policy's inspection mode decides whether inspection happens on the SIP ALG proxy or on the IPS engine.

```
config firewall policy
    edit <id>
        set inspection-mode {proxy | flow}
        set voip-profile <name>
    next
end
```

SIP message syntax inspection

For syntax verification, the following attributes are available for configuration in the VoIP profile to determine what action is taken when a specific syntax error or attack based on invalid syntax is detected. For example, the action can be set to pass or discard it.

```
malformed-request-line
malformed-header-via
malformed-header-from
malformed-header-to
malformed-header-call-id
malformed-header-cseq
malformed-header-rack
malformed-header-rseq
malformed-header-contact
malformed-header-record-route
malformed-header-route
malformed-header-expires
malformed-header-content-type
malformed-header-content-length
malformed-header-max-forwards
malformed-header-allow
malformed-header-p-asserted-identity
malformed-header-sdp-v
malformed-header-sdp-o
malformed-header-sdp-s
malformed-header-sdp-i
malformed-header-sdp-c
malformed-header-sdp-b
malformed-header-sdp-z
malformed-header-sdp-k
malformed-header-sdp-a
malformed-header-sdp-t
malformed-header-sdp-r
malformed-header-sdp-m
malformed-header-no-require*
malformed-header-no-proxy-require*
```

`*` = only available in flow mode

SIP message blocking

The following options are available in the VoIP profile to block SIP messages:

```
block-long-lines
block-unknown
block-ack
block-bye
block-cancel
block-info
block-invite
block-message
block-notify
block-options
block-prack
block-publish
block-refer
block-register
block-subscribe
block-update
block-geo-red-options**
```

`**` = only available in proxy mode

SIP message rate limiting

The rate of certain types of SIP requests that are passing through the SIP ALG can be restricted:

```
register-rate
invite-rate
subscribe-rate
message-rate
notify-rate
refer-rate
update-rate
options-rate
ack-rate
prack-rate
info-rate
publish-rate
bye-rate
cancel-rate
```

Additionally, flow-based SIP supports the following rate tracking features:

```
register-rate-track none
invite-rate-track none
subscribe-rate-track none
message-rate-track none
notify-rate-track none
refer-rate-track none
update-rate-track none
options-rate-track none
ack-rate-track none
prack-rate-track none
info-rate-track none
publish-rate-track none
bye-rate-track none
cancel-rate-track none
```

SIP pinholes

When SIP ALG processes a SIP call, it usually opens pinholes for SIP signaling and RTP/RTCP packets. NAT usually takes place during the process at both the network and SIP application layers. SIP ALG ensures that, with NAT happening, corresponding SIP and RTP/RTCP pinholes are created during the process when it is necessary for call sessions to be established through FortiOS devices.

By default, SIP ALG manages pinholes automatically, but some special configurations can be used to restrict the pinholes if required.

SIP pinhole restriction

The `strict-register` attribute is enabled by default. When enabled, after a SIP endpoint registers to the SIP server through a firewall policy on the FortiGate, only the SIP messages sent from the same IP address as the SIP server are allowed to pass through the SIP pinhole that is created in the FortiGate to reach the SIP endpoints. If the attribute is disabled, SIP messages from any IP addresses can pass through the pinhole created after the registration.

SIP pinhole restriction is only supported by SIP ALG and in proxy mode.

To configure registrar connection ability:

```
config voip profile
    edit <name>
        config sip
            set strict-register {enable | disable}
        end
    next
end
```

RTP/RTCP pinhole restriction

The `nat-port-range` setting is used to specify a port range in the VoIP profile to restrict the NAT port range for Real-time Transport Protocol/Real-time Transport Control Protocol (RTP/RTCP) packets in a Session Initiation Protocol (SIP) call session that is handled by the SIP application layer gateway (ALG) in a FortiGate.

When NAT is enabled, or VIP is used in a firewall policy for SIP ALG to handle a SIP call session established through a FortiGate, the SIP ALG can perform NAT to translate the ports used for the RTP/RTCP packets when they are flowing through the device between the external and internal networks.

To edit the translated port range for RTP/RTCP packets:

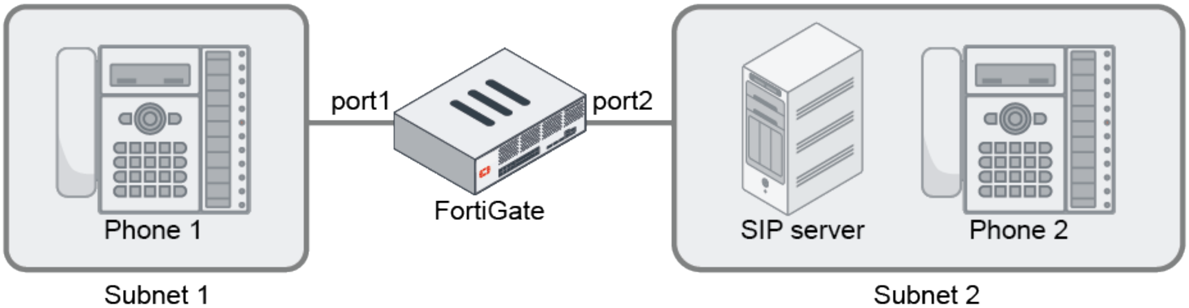
```
config voip profile
    edit <name>
        config sip
            set nat-port-range <start_port_number>-<end_port_number>
        end
    next
end
```

`nat-port-range <start_port_number>-<end_port_number>`	Enter the NAT port range (minimum port number = 5117, default = 5117-65535).

Example

In this example, Phone 1 is in Subnet 1, and the SIP server and Phone 2 are in Subnet 2. All SIP signaling messages and RTP/RTCP packets go through the SIP server. The RTP/RTCP ports on Phone 1 are configured as 17078/17079.

The FortiGate administrator wants to use NAT for the port 17078/17079 to 30000/30001. If Phone 1 and Phone 2 are registered to the SIP server, and they establish a call session between them through the FortiGate and the SIP server, then the RTP/RTCP ports 17078/17079 of Phone 1 will be translated to ports 30000/30001. All RTP/RTCP packets going out of port2 have source ports of 30000/30001, and all RTP/RTCP packets going into port2 also have destination ports of 30000/30001.



To configure the custom port range:

1. Edit the VoIP profile:

```
config voip profile
    edit "natPortRange"
        config sip
            set nat-port-range 30000-30001
        end
    next
end
```

It is best practice to configure the starting port as an even number and the ending port as an odd number.

2. Configure the firewall policy:

```
config firewall policy
    edit 1
        set srcintf port1
        set dstintf port2
        set srcaddr all
        set dstaddr all
        set service SIP
        set action accept
        set schedule always
        set voip-profile natPortRange
        set nat enable
    next
end
```

SIP over TLS

Some SIP phones and servers can communicate using TLS to encrypt the SIP signaling traffic. To allow SIP over TLS calls to pass through the FortiGate, the encrypted signaling traffic must be unencrypted and inspected. The FortiGate SIP ALG intercepts, unencrypts, and inspects the SIP packets, which are then re-encrypted and forwarded to their destination.

The SIP ALG only supports full mode TLS. This means that the SIP traffic between SIP phones and the FortiGate, and between the FortiGate and the SIP server, is always encrypted. The highest TLS version supported by SIP ALG is TLS 1.2.

To enable SIP over TLS support, the SSL mode in the VoIP profile must be set to `full`. The SSL server and client certificates can be provisioned so that the FortiGate can use them to establish connections to SIP phones and servers, respectively.

This configuration is only supported in proxy mode.

To configure SIP over TLS:

1. Configure a VoIP profile with SSL enabled:

```
config voip profile
    edit "tls"
        config sip
            set ssl-mode full
            set ssl-client-certificate "ssl_client_cert"
            set ssl-server-certificate "ssl_server_cert"
        end
    next
end
```

   The `ssl_server_cert`, `ssl_client_cert`, and key files can be generated using a certification tool, such as OpenSSL, and imported to the local certificate store of the FortiGate from *System* > *Certificates* in the GUI. Existing local certificates in the certificate store can also be used. As always for TLS connections, the certificates used must be verified and trusted at the other end of the connection when required.

   For example, the CA certificate of the SIP server's certificate should be imported to the FortiGate as an external CA certification, so that the FortiGate can use it to verify the SIP server's certificate when setting up the TLS connection. The CA certificate configured as the `ssl_server_cert` should be installed as the trusted certificate on the SIP phones. The deployment of the certificates across the network depends on the SIP client and server devices that are used in the system.

2. Apply the profile to the firewall policy:

```
config firewall policy
    edit 1
        set srcintf "port1"
        set dstintf "port2"
        set srcaddr "all"
        set dstaddr "vip_sip_server"
        set action accept
        set schedule "always"
        set service "SIP"
        set utm-status enable
        set voip-profile "tls"
    next
end
```

Voice VLAN auto-assignment

You can leverage LLDP-MED to assign voice traffic to the desired voice VLAN. After detection and setup, the IP phone on the network is segmented to its own VLAN for policy, prioritization, and reporting. The LLDP reception capabilities in

FortiOS include LLDP-MED assignment for voice, voice signaling, guest, guest voice signaling, softphone, video conferencing, streaming video, and video signaling.

You can configureVLAN auto-assignment using the following steps:

1. Set up the VLAN for the voice device
2. Set up the DHCP server for the voice VLAN
3. Sett up the LLDP network policy
4. Enable LLDP on the physical interface that the VLAN belongs to
5. Apply the LLDP network policy on the physical interface
6. Confirm that the VLAN was assigned

To set up the VLAN for the voice device:

```
config system interface
    edit "vlan_100"
        set vdom "root"
        set ip 192.168.1.99 255.255.255.0
        set alias "voice_vlan"
        set device-identification enable
        set role lan
        set snmp-index 25
        set interface "port10"
        set vlanid 100
    next
end
```

To set up the DHCP server for the voice VLAN:

```
config system dhcp server
    edit 1
        set dns-service default
        set default-gateway 192.168.1.99
        set netmask 255.255.255.0
        set interface "vlan_100"
        config ip-range
            edit 1
                set start-ip 192.168.1.110
                set end-ip 192.168.1.210
            next
        end
    next
end
```

To set up the LLDP network policy:

```
config system lldp network-policy
    edit "1"
        config voice
            set status enable
            set tag dot1q
            set vlan 100
        end
    next
end
```

To enable LLDP on the physical interface that the VLAN belongs to:

```
config system interface
    edit "port10"
        set vdom "root"
        set type physical
        set lldp-reception enable
        set lldp-transmission enable
        set snmp-index 14
    next
end
```

To apply the LLDP network policy on the physical interface:

```
config system interface
    edit "port10"
        set lldp-network-policy "1"
    next
end
```

To confirm that the VLAN was assigned as expected:

1. Connect an IP phone to the network.
2. Check the IP address on the phone.
   The IP address should belong to the voice VLAN.
3. Sniff on the FortiGate incoming interface to see if traffic from the IP phone has the desired VLAN tag.
   In this example, the voice traffic from the IP phone should be in VLAN 100.

Scanning MSRP traffic

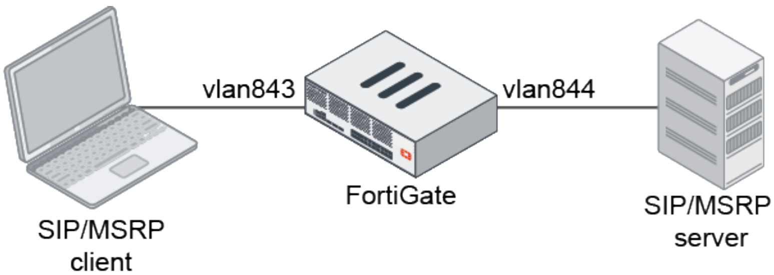
An MSRP (Message Session Relay Protocol) decoder in the IPS engine scans for IPS signatures against the application data. Malicious payload in the text message can be blocked. A VoIP profile using flow inspection mode must be configured in the firewall policy. An IPS profile must be configured in the firewall policy to inspect the payload.

```
config voip profile
    edit <name>
        set feature-set flow
        config msrp
            set status {enable | disable}
            set log-violations {enable | disable}
            set max-msg-size <integer>
            set max-msg-size-action {pass | block | reset | monitor}
        end
    next
end
```

`status {enable | disable}`	Enable/disable MSRP.
`log-violations {enable | disable}`	Enable/disable logging of MSRP violations.
`max-msg-size <integer>`	Maximum allowable MSRP message size, in bytes (0 - 65535, default = 0).

`max-msg-size-action {pass | block | reset | monitor}`	Action for violating maximum MSRP message size: • pass: pass or allow matching traffic (default) • block: block or drop matching traffic • reset: reset sessions for matching traffic • monitor: pass and log matching traffic

Examples



In this first example, MSRP messages larger than 10 bytes will be blocked. The client sends an oversized MSRP message to the server. Message Automation & Protocol Simulation (MAPS™) is used, and a client-server model was configured to use the software to send MSRP traffic from vlan843 (client) to vlan844 (server) with plain text placed in the message field. The software uses the content of the MsrpInputMessage.txt file located in the default folder, where anything in that file will be sent by MSRP. The following text is used:

> GL's Message Automation & Protocol Simulation (MAPS™) is a protocol simulation and conformance test tool that supports a variety of protocols such as SIP, MEGACO, MGCP, SS7, ISDN, GSM, MAP, CAS, LTE, UMTS, SS7 SIGTRAN, ISDN SIGTRAN, SIP I, GSM AoIP, Diameter and others. This message automation tool covers solutions for both protocol simulation and protocol analysis. The application includes various test plans and test cases to support the testing of real-time entities. Along with automation capability, the application gives users the unlimited ability to edit messages and control scenarios (message sequences).

To configure MSRP traffic scanning:

1. Configure the VoIP profile:

```
config voip profile
    edit msrp_test
        set feature-set flow
        config msrp
            set status enable
            set log-violations enable
            set max-msg-size 10
            set max-msg-size-action block
        end
    next
end
```

2. Configure the firewall policy:

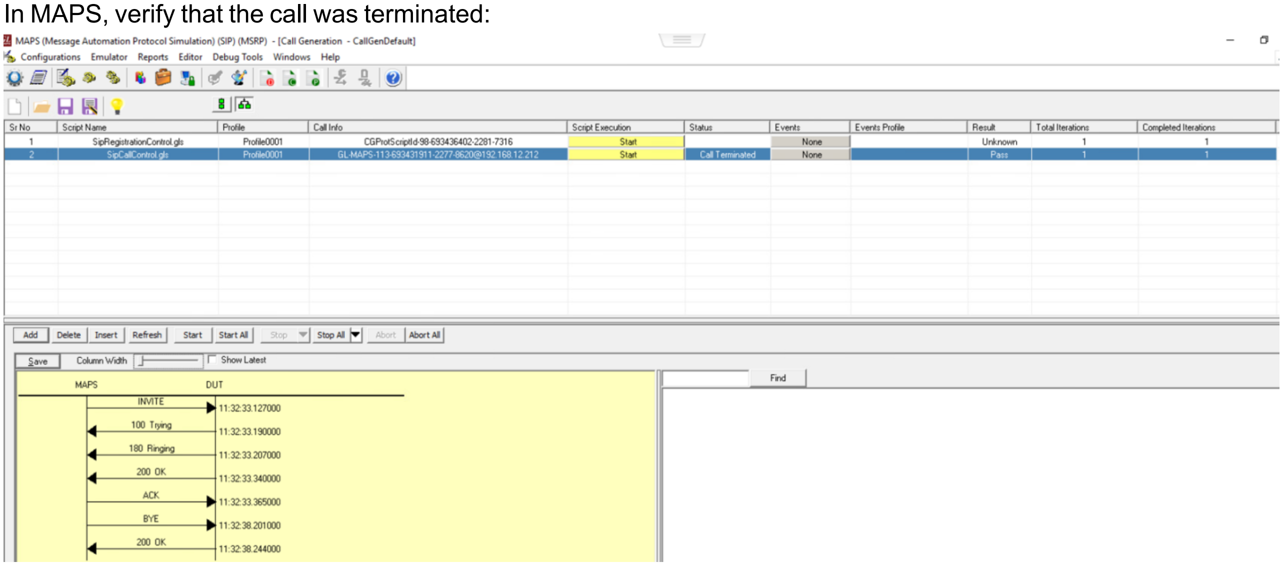
```
config firewall policy
    edit 1
        set name "vdom3"
        set srcintf "vlan843"
```

```
            set dstintf "vlan844"
            set action accept
            set srcaddr "all"
            set dstaddr "all"
            set schedule "always"
            set service "ALL"
            set utm-status enable
            set ssl-ssh-profile "certificate-inspection"
            set voip-profile "msrp_test"
            set logtraffic all
        next
    end
```

3. Verify the log:

```
# execute log filter category 4
# execute log display
1 logs found.
1 logs returned.

1: date=2021-06-10 time=17:21:19 eventtime=1623370879840284165 tz="-0700"
logid="0419016384" type="utm" subtype="ips" eventtype="signature" level="alert"
vd="vdom3" severity="info" srcip=192.168.12.212 srccountry="Reserved"
dstip=192.168.12.213 srcintf="vlan843" srcintfrole="lan" dstintf="vlan844"
dstintfrole="lan" sessionid=27700 action="dropped" proto=6 service="MSRP" policyid=1
attack="MSRP.Max.Message.Size.Exceeded" srcport=20036 dstport=20036 direction="outgoing"
attackid=1000000 profile="g-default" ref="http://www.fortinet.com/ids/VID1000000"
incidentserialno=189792275 psrcport=0 pdstport=0 msg="msrp_decoder:
MSRP.Max.Message.Size.Exceeded, msg_size=270 exceeds config maximum=10"
```

4. In MAPS, verify that the call was terminated:



In this second example, malicious files will be blocked. The client sends an EICAR test sample to the server in an MSRP message. Message Automation & Protocol Simulation (MAPS™) is used, and a client-server model was configured to use the software to send MSRP traffic from vlan843 (client) to vlan844 (server) with a plain text EICAR file containing a virus in the message field. The following text is used:

X5O!P%@AP[4\PZX54(P^)7CC)7}$EICAR-STANDARD-ANTIVIRUS-TEST-FILE!$H+H*

To configure MSRP traffic scanning:

1. Configure the VoIP profile:

```
config voip profile
    edit msrp_test
        set feature-set flow
        config msrp
            set status enable
            set log-violations enable
            set max-msg-size 0
            set max-msg-size-action pass
        end
    next
end
```

2. Configure the IPS profile:

```
config ips sensor
    edit "msrp"
        set extended-log enable
        config entries
            edit 1
                set rule 7470 29844
                set status enable
                set action block
            next
        end
    next
end
```

3. Configure the firewall policy:

```
config firewall policy
    edit 1
        set name "vdom3"
        set srcintf "vlan843"
        set dstintf "vlan844"
        set action accept
        set srcaddr "all"
        set dstaddr "all"
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set ssl-ssh-profile "certificate-inspection"
        set ips-sensor "msrp"
        set voip-profile "msrp_test"
        set logtraffic all
    next
end
```

4. Verify the log:

```
# execute log filter category 4
# execute log display
1 logs found.
1 logs returned.

1: date=2021-09-16 time=11:29:48 eventtime=1631816988947762597 tz="-0700"
```

```
logid="0419016384" type="utm" subtype="ips" eventtype="signature" level="alert"
vd="vdom3" severity="info" srcip=192.168.12.212 srccountry="Reserved"
dstip=192.168.12.213 srcintf="vlan843" srcintfrole="lan" dstintf="vlan844"
dstintfrole="lan" sessionid=41344 action="dropped" proto=6 service="MSRP" policyid=1
attack="Eicar.Virus.Test.File" srcport=20069 dstport=20069 direction="outgoing"
attackid=29844 profile="msrp" ref="http://www.fortinet.com/ids/VID29844"
incidentserialno=123731970 psrcport=0 pdstport=0 msg="file_transfer:
Eicar.Virus.Test.File,"
```

ICAP

Internet Content Adaptation Protocol (ICAP) is an application layer protocol that is used to offload tasks from the firewall to separate, specialized servers. For more information see RFC 3507.

ICAP profiles can only be applied to policies that use proxy-based inspection. If you enable ICAP in a policy, HTTP and HTTPS (if HTTPS inspection is supported) traffic that is intercepted by the policy is transferred to the ICAP server specified by the selected ICAP profile. Responses from the ICAP server are returned to the FortiGate, and then forwarded to their destination.

By default, *ICAP* is not visible in the GUI. See Feature visibility on page 2163 for instructions on making it visible.

ICAP filter profiles cannot be used in NGFW policy-based mode. See NGFW policy on page 772 for more information.

To configure ICAP:

1. Set up your ICAP server.
2. On the FortiGate, add an ICAP server.
3. Create an ICAP profile.
4. Use the ICAP profile in a firewall policy that covers the traffic that needs to be offloaded to the ICAP server.

The following topics provide information about ICAP:

- ICAP configuration example on page 1283
- ICAP response filtering on page 1285
- Secure ICAP clients on page 1287
- ICAP scanning with SCP and FTP on page 1288

TCP connection pool for connections to ICAP server

A TCP connection pool can maintain local-out TCP connections to the external ICAP server due to a backend update in FortiOS. TCP connections will not be terminated once data has been exchanged with the ICAP server, but instead are reused in the next ICAP session to maximize efficiency.

For example, consider a scenario where an ICAP profile is used as a UTM profile in an explicit web proxy policy, and a client visits web servers through this proxy policy.

Once the WAD is initialized, when a HTTP request is sent from the client to the server through the FortiGate with an ICAP profile applied to the matched proxy policy, a TCP connection is established between the FortiGate and the ICAP server to exchange data.

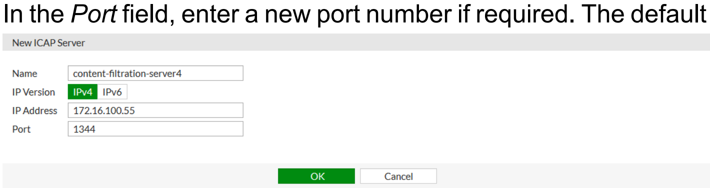
When an ICAP session is finished, the TCP connection is kept in the WAD connection pool. When another ICAP session needs to be established, the WAD will check if there are any idle connections available in the connection pool. If an idle connection is available, then it will be reused; otherwise, a new TCP connection is established for the ICAP session. This process can be checked in the WAD debug log.

ICAP configuration example

In this example, the ICAP server performs proprietary content filtering on HTTP and HTTPS requests. If the content filter is unable to process a request, then the request is blocked. Streaming media is not considered by the filter, so it is allowed through and is not processed.

To configure the ICAP setup in the GUI:

1. Add the ICAP server:
   a. Go to *Security Profiles > ICAP Servers* and click *Create New*.
   b. In the *Name* field, enter a name for the ICAP server, such as *content-filtration-server4*.
   c. Select the *IP Version*.
   d. In the *IP Address* field, enter the IP address of the ICAP server.
   e. In the *Port* field, enter a new port number if required. The default value is *1344*.

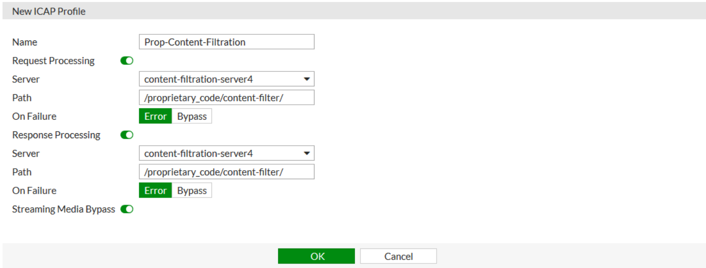


   f. Click *OK*.

The maximum number of concurrent connections to ICAP server can be configured in the CLI (`set max-connections`). The default setting is 100 connections.

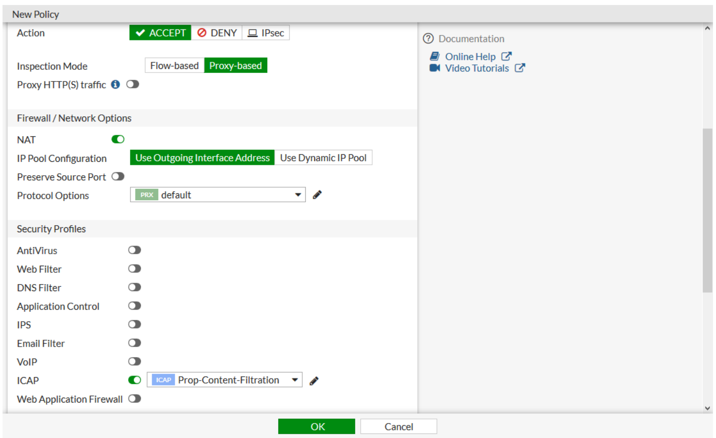
2. Create the ICAP profile:
   a. Go to *Security Profiles > ICAP* and click *Create New*.
   b. In the *Name* field, enter a name for the ICAP profile, such as *Prop-Content-Filtration*.
   c. Enable *Request Processing* and set the following:
      - *Server*: select the ICAP server (*content-filtration-server4*).
      - *Path*: enter the path to the processing component on the server, such as */proprietary_code/content-filter/*.
      - *On Failure*: select *Error* to block the request. If the message cannot be processed, it will not be blocked.
   d. Enable *Response Processing* and set the following:
      - *Server*: select the ICAP server (*content-filtration-server4*).
      - *Path*: enter the path to the processing component on the server, such as */proprietary_code/content-filter/*.
      - *On Failure*: select *Error* to block the request. If the message cannot be processed, it will not be blocked.

e. Enable *Streaming Media Bypass* to not offload streaming media to the ICAP server.



f. Click *OK*.

3. Add the ICAP profile to a policy:
   a. Go to *Policy & Objects > Firewall Policy* and click *Create New*.
   b. Set *Inspection Mode* to *Proxy-based*.
   c. Under *Security Profiles*, enable *ICAP* and select the ICAP server.



   d. Configure the other settings as needed.
   e. Click *OK*.

To configure the ICAP setup in the CLI:

1. Add the ICAP server:

```
config icap server
    edit "content-filtration-server4"
        set ip-version 4
        set ip-address 172.16.100.55
        set port 1344
        set max-connections 200
    next
end
```

2. Create the ICAP profile:

```
config icap profile
    edit "Prop-Content-Filtration"
        set request enable
        set response enable
        set streaming-content-bypass enable
        set request-server "content-filtration-server4"
        set response-server "content-filtration-server4"
```

```
        set request-failure error
        set response-failure error
        set request-path "/proprietary_code/content-filter/"
        set response-path "/proprietary_code/content-filter/"
        set methods delete get head options post put trace other
    next
end
```

3. Add the ICAP profile to a policy:

```
config firewall policy
    edit 5
        set name "icap_filter3"
        set srcintf "virtual-wan-link"
        set dstintf "virtual-wan-link"
        set srcaddr "FABRIC_DEVICE"
        set dstaddr "FABRIC_DEVICE"
        set dstaddr-negate enable
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set inspection-mode proxy
        set ssl-ssh-profile "certificate-inspection"
        set icap-profile "Prop-Content-Filtration"
        set logtraffic disable
        set fsso disable
        set nat enable
    next
end
```

ICAP response filtering

ICAP HTTP responses can be forwarded or bypassed based on the HTTP header value and status code.

When configuring the ICAP profile, if `response` is enabled, the `respmod-default-action` option can be configured:

- If `respmod-default-action` is set to `forward`, FortiGate will treat every HTTP response and send ICAP requests to the ICAP server.
- If `respmod-default-action` is set to `bypass`, FortiGate will only send ICAP requests if the HTTP response matches the defined rules, and the rule's action is set to forward.

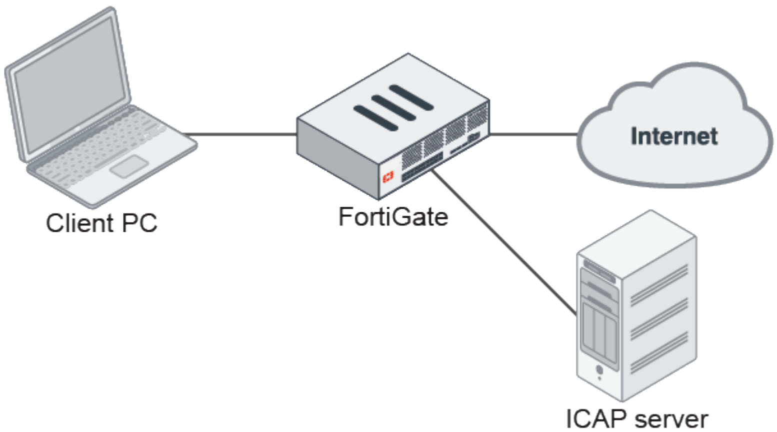
When configuring a response rule:

- The `http-resp-status-code` option is configured to specific HTTP response codes. If the HTTP response has any one of the configured values, then the rule takes effect.
- Multiple header value matching groups can be configured. If the header value matches one of the groups, then the rule takes effect.
- If both status codes and header values are specified in a rule, the response must match at least one of each.

The UTM ICAP log category is used for logging actions when FortiGate encounters errors with the ICAP server, such as no service, unreachable, error response code, or timeout. If an error occurs, a traffic log and an associated UTM ICAP log will be created.

Example

The FortiGate acts as a gateway for the client PC and connects to a reachable ICAP server. The ICAP server can be in NAT, transparent, or proxy mode.



In this example, client request HTTP responses will be forwarded to the ICAP server from all hosts if they have an HTTP status code of 200, 301, or 302, and have `content-type: image/jpeg` in the their header.

To configure an ICAP profile with HTTP response rules:

```
config icap profile
    edit "icap_profile2"
        set request disable
        set response enable
        set streaming-content-bypass disable
        set preview disable
        set response-server "icap_server1"
        set response-failure error
        set response-path ''
        set methods delete get head options post put trace other
        set response-req-hdr disable
        set respmod-default-action bypass
        config respmod-forward-rules
            edit "rule2"
                set host "all"
                set action forward
                set http-resp-status-code 200 301 302
                config header-group
                    edit 2
                        set header-name "content-type"
                        set header "image/jpeg"
                    next
                end
            next
        end
    next
end
```

To view the logs if an error occurs:

1. View the traffic log:

```
# execute log filter category 0
# execute log display
1 logs found.
1 logs returned.

 1: date=2019-10-25 time=17:43:47 logid="0000000013" type="traffic" subtype="forward"
level="notice" vd="vdom1" eventtime=1572050627037314464 tz="-0700" srcip=10.1.100.145
srcport=47968 srcintf="port1" srcintfrole="undefined" dstip=172.16.200.46 dstport=80
dstintf="port2" dstintfrole="undefined" poluuid="a4d5324e-f6c3-51e9-ce2d-f360994fb547"
sessionid=43549 proto=6 action="close" policyid=1 policytype="policy" service="HTTP"
dstcountry="Reserved" srccountry="Reserved" trandisp="snat" transip=172.16.200.1
transport=47968 duration=1 sentbyte=485 rcvdbyte=398 sentpkt=6 rcvdpkt=5
appcat="unscanned" wanin=478 wanout=165 lanin=165 lanout=165 utmaction="block"
counticap=1 crscore=5 craction=262144 crlevel="low" utmref=65532-0
```

2. View the UTM ICAP log:

```
# execute log filter category 20
# execute log display
1 logs found.
1 logs returned.

 1: date=2019-10-25 time=17:43:46 logid="2000060000" type="utm" subtype="icap"
eventtype="icap" level="warning" vd="vdom1" eventtime=1572050626010097145 tz="-0700"
msg="Request blocked due to ICAP server error" service="HTTP" srcip=10.1.100.145
dstip=172.16.200.46 srcport=47968 dstport=80 srcintf="port1" srcintfrole="undefined"
dstintf="port2" dstintfrole="undefined" policyid=1 sessionid=43549 proto=6
action="blocked" profile="icap_profile1" url="/icap_test/"
```

The logs show that the ICAP services stopped before the access. When the client tried to access HTTP and ICAP took effect, the FortiGate sent the ICAP request to the ICAP server and received an error. The client sees a *502 Bad Gateway* message, and FortiGate writes the two logs. In the GUI, the logged traffic is displayed as *Result: Deny: UTM Blocked*.

Secure ICAP clients

A secure SSL connection from the FortiGate to the ICAP server can be configured as follows:

```
config icap server
    edit <name>
        set secure {enable | disable}
        set ssl-cert <certificate>
    next
end
```

To configure a secure ICAP client:

1. Configure the ICAP server:

```
config icap server
    edit "icap_server1"
        set ip-version 4
        set ip-address 192.168.10.2
```

```
        set port 11344
        set max-connections 100
        set secure enable
        set ssl-cert "ACCVRAIZ1"
    next
end
```

Port 11344 is the standard port for secure ICAP. This must be configured manually if the secure connection is enabled.

2. Configure the ICAP profile:

```
config icap profile
    edit "icap_profile1"
        set request enable
        set response enable
        set streaming-content-bypass enable
        set request-server "icap_server1"
        set response-server "icap_server1"
    next
end
```

3. Configure the firewall policy:

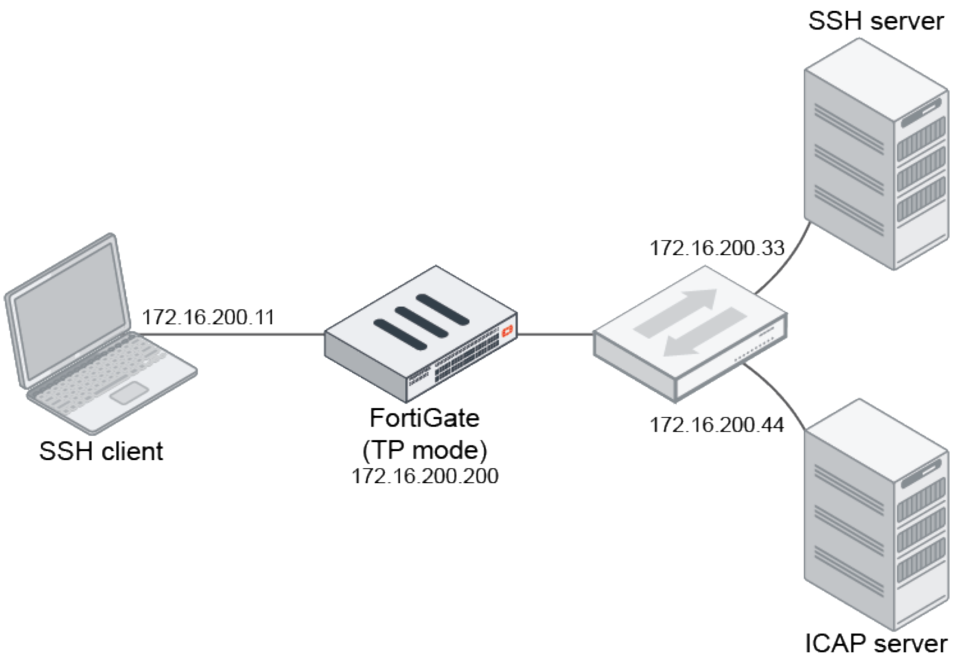
```
config firewall policy
    edit 1
        set utm-status enable
        set inspection-mode proxy
        set ssl-ssh-profile "protocols"
        set icap-profile "icap_profile1"
    next
end
```

ICAP scanning with SCP and FTP

A FortiGate can forward files transferred by SCP and FTP to an ICAP server for further scanning. Previously, only HTTP and HTTPS were supported for ICAP forwarding.

Example

The FortiGate used in this example is operating in transparent mode. The SSH client, 172.16.200.11, sends a file named `today` to the SSH server at 172.16.200.33 using SCP. Since SCP transfers are encrypted inside an SSH tunnel, for the FortiGate to scan the traffic, deep inspection must be enabled in the SSL SSH profile.



To configure ICAP scanning with SCP:

1. Configure the ICAP server settings:

```
config icap server
    edit "icap_server1"
        set ip-address 172.16.200.44
    next
end
```

2. Configure the ICAP profile for SSH:

```
config icap profile
    edit "icap_profile1"
        set file-transfer ssh
        set file-transfer-server "icap_server1"
        set file-transfer-path "ssh_test"
    next
end
```

If the file transfer is over FTP, configure the profile as follows:

```
config icap profile
    edit "icap_profile1"
        set file-transfer ftp
        set streaming-content-bypass enable
        set file-transfer-server "icap_server1"
        set file-transfer-path "ftp_test"
    next
end
```

3. Configure the SSL SSH profile:

```
config firewall ssl-ssh-profile
    edit "protocols"
        config ssh
            set ports 22
            set status deep-inspection
        end
    next
end
```

4. Configure the firewall policy:

```
config firewall policy
    edit 1
        set name "ICAP"
        set srcintf "lan"
        set dstintf "mgmt"
        set action accept
        set srcaddr "all"
        set dstaddr "all"
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set inspection-mode proxy
        set profile-protocol-options "protocol"
        set ssl-ssh-profile "protocols"
        set icap-profile "icap_profile1"
    next
end
```

To test the configuration:

1. On a Linux client, copy a filed named `today` to the SSH server using SCP:

```
scp today fosqa@172.16.200.33:/home/fosqa/ssh_depot/
```

2. Capture a sniffer trace between the FortiGate and ICAP server, then verify the output from the ICAP protocol session.
   a. The client request and the file to be inspected:

```
Icap_client REQMOD:
172.016.200.200.13185-172.016.200.044.01344: REQMOD icap://172.16.200.44:1344/ssh_
test ICAP/1.0
Host: 172.16.200.44:1344
X-Client-IP: 172.16.200.11
X-Server-IP: 172.16.200.33
X-Authenticated-User: TG9jYWw6Ly9hbm9ueW1vdXM=
X-Authenticated-Groups: TG9jYWw6Ly9sb2NhbGhvc3Qvbm8gYXV0aGVudGljYXRpb24=
User-Agent: FortiOS v7.2.0
Encapsulated: req-hdr=0, req-body=116

PUT /scp/today HTTP/1.1
Host: 172.16.200.11
Content-Type: application/octet-stream
Transfer-Encoding: chunked

1d
Tue Sep 20 04:01:50 UTC 2022
```

Where:

- `X-Client-IP` = the client sending the file
- `X-Server-IP` = the server receiving the file
- `Tue Sep 20 04:01:50 UTC 2022` = the content of the file, which is in clear text after the FortiGate performs deep inspection

b. The ICAP server response that the file is cleared and allowed to pass without modifications:

```
Icap-server reply:
172.016.200.044.01344-172.016.200.200.13185: ICAP/1.0 200 OK
ISTag: "GreasySpoon-1.0.7-b03"
Host: 0.0.0.0:1344
Encapsulated: req-hdr=0, req-body=136
Connection: keep-alive

PUT /scp/today HTTP/1.1
Host: 172.16.200.11
Content-Type: application/octet-stream
Transfer-Encoding: chunked
Content-Length: 29

1d
Tue Sep 20 04:01:50 UTC 2022
```

3. On a Linux client, copy the file from the server locally using SCP:

```
scp fosqa@172.16.200.33:/home/fosqa/ssh_depot/today2/
```

4. Similar outputs are observed. The ICAP client request indicates that the file is copied from the SSH server:

```
PUT /scp/today2 HTTP/1.1
Host: 172.16.200.33
```

Web application firewall

Web application firewall (WAF) profiles can detect and block known web application attacks. You can configure WAF profiles to use signatures and constraints to examine web traffic. You can also enforce an HTTP method policy, which controls the HTTP method that matches the specified pattern.

You can customize the default profile, or you can create your own profile to apply access rules and HTTP protocol constraints to traffic. You can apply WAF profiles to firewall policies when the inspection mode is set to proxy-based.

Web application firewall profiles cannot be used NGFW policy-based mode. See NGFW policy on page 772 for more information.

The following topic provides information about WAF profiles:

- Protecting a server running web applications on page 1292

Protecting a server running web applications

You can use a web application firewall profile to protect a server that is running a web application, such as webmail.

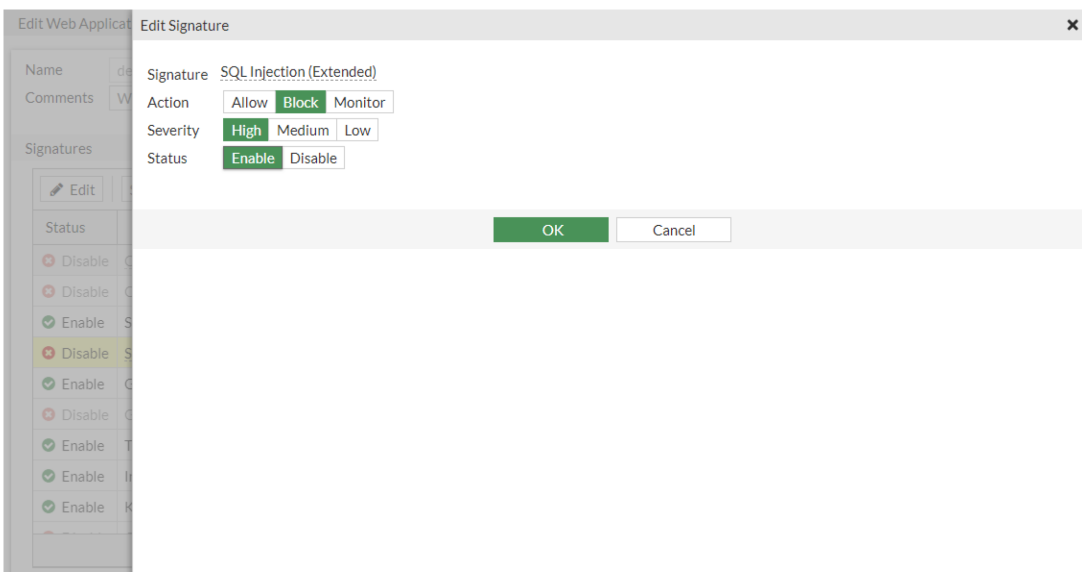
Web application firewall profiles are created with a variety of options called signatures and constraints. Once these options are enabled, the action can be set to allow, monitor, or block. The severity can be set to high, medium, or low.

In the following example, the default profile will be targeted to block SQL injection attempts and generic attacks.

The web application firewall feature is only available when the policy inspection mode is proxy-based.

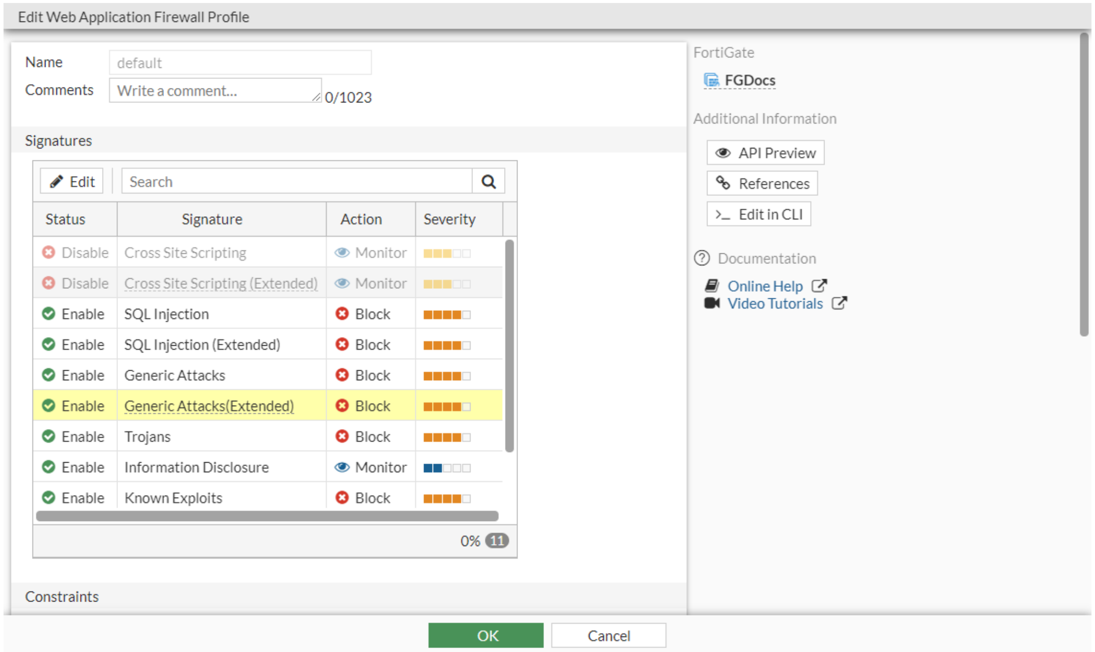
To protect a server running web applications:

1. Enable the web application firewall:
   - **a.** Go to *System > Feature Visibility*.
   - **b.** Under *Security Features*, enable *Web Application Firewall*.
   - **c.** Click *Apply*.
2. Edit the default web application firewall profile (*Trojans* and *Known Exploits* are blocked by default):
   - **a.** Go to *Security Profiles > Web Application Firewall* and edit the *default* profile signature.
   - **b.** Select *SQL Injection (Extended)* and edit it so that it is enabled, the *Action* is set to *Block*, and the *Severity* is set to *High*.
   - **c.** Click *OK*.



   - **d.** Enable *Generic Attacks (Extended)* and edit it so that it is enabled, the *Action* is set to *Block*, and the *Severity* is set to *High*.

e. Click *OK*.

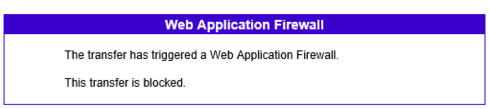


f. Click *OK*.

3. Apply the profile to a security policy:
   a. Go to *Policy & Objects* > *Firewall Policy* and edit the policy that allows access to the web server.
   b. For *Firewall / Network Options*, select the appropriate *Protocol Option*.
   c. For *Security Profiles*, enable *Web Application Firewall* and set it to use the *default* profile.
   d. Set the *SSL Inspection* to use the *deep-inspection* profile.
   e. Configure the other settings as needed.
   f. Click *OK*.
4. Verify that the web application firewall blocks traffic:
   a. Use the following URL to simulate an attack on your web server and substitute the IP address of your server:

```
http://<server IP>/index.php?username=1'%20or%20'1'%20=%20'1&password=1'%20or%20'1'%20=%20'1
```

   An error message appears, stating that the web application firewall has blocked the traffic:



Offloading to a FortiWeb

If you have a FortiWeb, you may be able to offload the functions of the web application control to your FortiWeb. To find out if this option is available, refer to the FortiOS or FortiWeb Release Notes for information about device compatibility.

To offload to a FortiWeb:

1. Go to *Security Fabric* > *Fabric Connectors*.
2. Click *Create New*, and click *Fabric Device*.

3. Enter the following for the device:
   a. Name (`FortiWeb`)
   b. FortiWeb IP address
   c. HTTPS service port
4. Click *Generate*.
5. Enter your credentials to generate the access token.
6. Click *OK*.

SSL & SSH Inspection

Secure Sockets Layer (SSL) content scanning and inspection allows you to apply antivirus scanning, web filtering, and email filtering to encrypted traffic. You can apply SSL inspection profiles to firewall policies.

FortiOS includes four preloaded SSL/SSH inspection profiles, three of which are read-only and can be cloned:

- *certificate-inspection*
- *deep-inspection*
- *no-inspection*

The *custom-deep-inspection* profile can be edited, or you can create your own SSL/SSH inspection profiles.

Deep inspection (also known as SSL/SSH inspection) is typically applied to outbound policies where destinations are unknown. Depending on your policy requirements, you can configure the following:

- Which CA certificate will be used to decrypt the SSL encrypted traffic
- Which SSL protocols will be inspected
- Which ports will be associated with which SSL protocols for inspection
- Whether or not to allow invalid SSL certificates
- Whether or not SSH traffic will be inspected
- Which addresses or web category allowlists can bypass SSL inspection

The following topics provide information about SSL & SSH Inspection:

- Certificate inspection on page 1294
- Deep inspection on page 1296
- Protecting an SSL server on page 1299
- Handling SSL offloaded traffic from an external decryption device on page 1300
- SSH traffic file scanning on page 1302
- Redirect to WAD after handshake completion on page 1303
- HTTP/2 support in proxy mode SSL inspection on page 1304
- Define multiple certificates in an SSL profile in replace mode on page 1305
- Disabling the FortiGuard IP address rating on page 1307

Certificate inspection

FortiGate supports certificate inspection. The default configuration has a built-in *certificate-inspection* profile which you can use directly. When you use certificate inspection, the FortiGate only inspects the headers up to the SSL/TLS layer.

If you do not want to deep scan for privacy reasons but you want to control web site access, you can use *certificate-inspection*.

SSL inspection options

The following options are available when configuring an SSL inspection profile (*Security Profiles > SSL/SSH Inspection*):

Enable SSL inspection of	Select *Multiple Clients Connecting to Multiple Servers*. This is normally used when inspecting outbound internet traffic
Inspection method	Select *SSL Certificate Inspection*.
CA certificate	Use the default *Fortinet_CA_SSL* certificate.
Blocked certificates	The FortiGate receives botnet C&C SSL connections from FortiGuard that contain SHA1 fingerprints of malicious certificates. By default, these certificates are blocked. Click *View Blocked Certificates* to see a detailed list.
Untrusted SSL certificates	Configure the action to take when a server certificate is not issued by a trusted CA. • *Allow*: Allow the untrusted server certificate. This is the default value. • *Block*: Block the session. • *Ignore*: This option is for *Full SSL inspection* only. It re-signs the server certificate as trusted. When configured in the GUI for certificate inspection it has no effect and the setting is not saved. Click *View Trusted CAs List* to see a list of the factory bundled and user imported CAs that are trusted by the FortiGate.
Server certificate SNI check	Check the SNI in the hello message with the CN or SAN field in the returned server certificate. • *Enable*: If mismatched, use the CN in the server certificate to do URL filtering. • *Strict*: If mismatched, close the connection. • *Disable*: Server certificate SNI check is disabled.

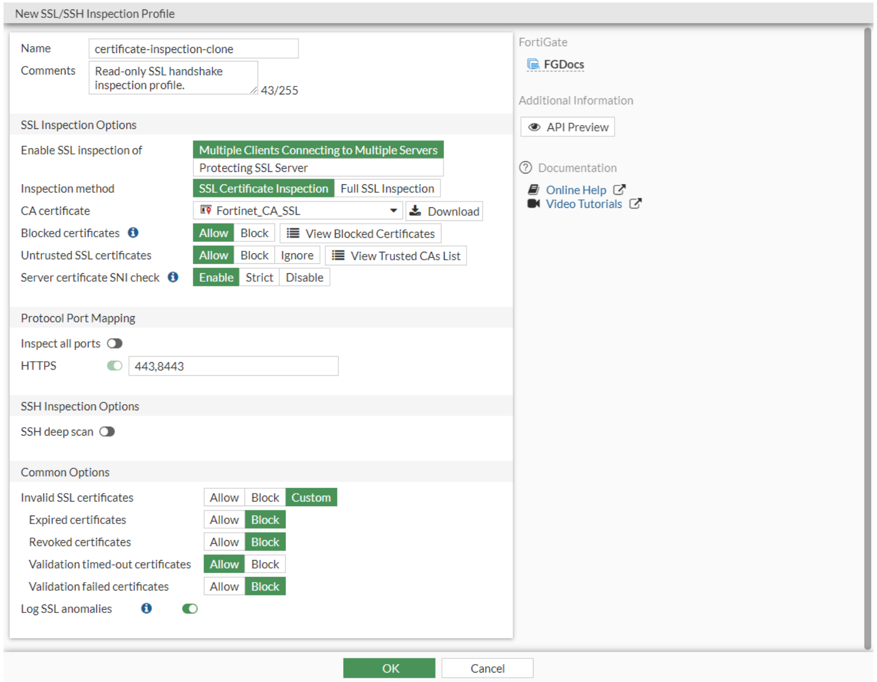
Inspect non-standard HTTPS ports

The built-in *certificate-inspection* profile is read-only and only listens on port 443. If you want to make changes, you must create a new certificate inspection profile.

If you know the non-standard port that the web server uses, such as port 8443, you can add this port to the *HTTPS* field.

To add a port to the inspection profile in the GUI:

1. Go to *Security Profiles > SSL/SSH Inspection*.
2. Create a new profile, or clone the default profile.
3. If you do no know what port is used in the HTTPS web server, under *Protocol Port Mapping*enable *Inspect All Ports*.
   If you know the port, such as port 8443, then set *HTTPS* to *443,8443*.



4. Configure the remaining setting as needed.
5. Click *OK*.

Common options

Invalid SSL certificates can be blocked, allowed, or a different actions can be configured for the different invalid certificates types:

Expired certificates	Action to take when the server certificate is expired. The default action is block.
Revoked certificates	Action to take when the server certificate is revoked. The default action is block.
Validation timed-out certificates	Action to take when the server certificate validation times out. The default action is allow.
Validation failed certificates	Action to take when the server certificate validation fails. The default action is block.

By default, SSL anomalies logging is enabled. Logs are generated in the UTM log type under the SSL subtype when invalid certificates are detected.

Deep inspection

You can configure address and web category allowlists to bypass SSL deep inspection.

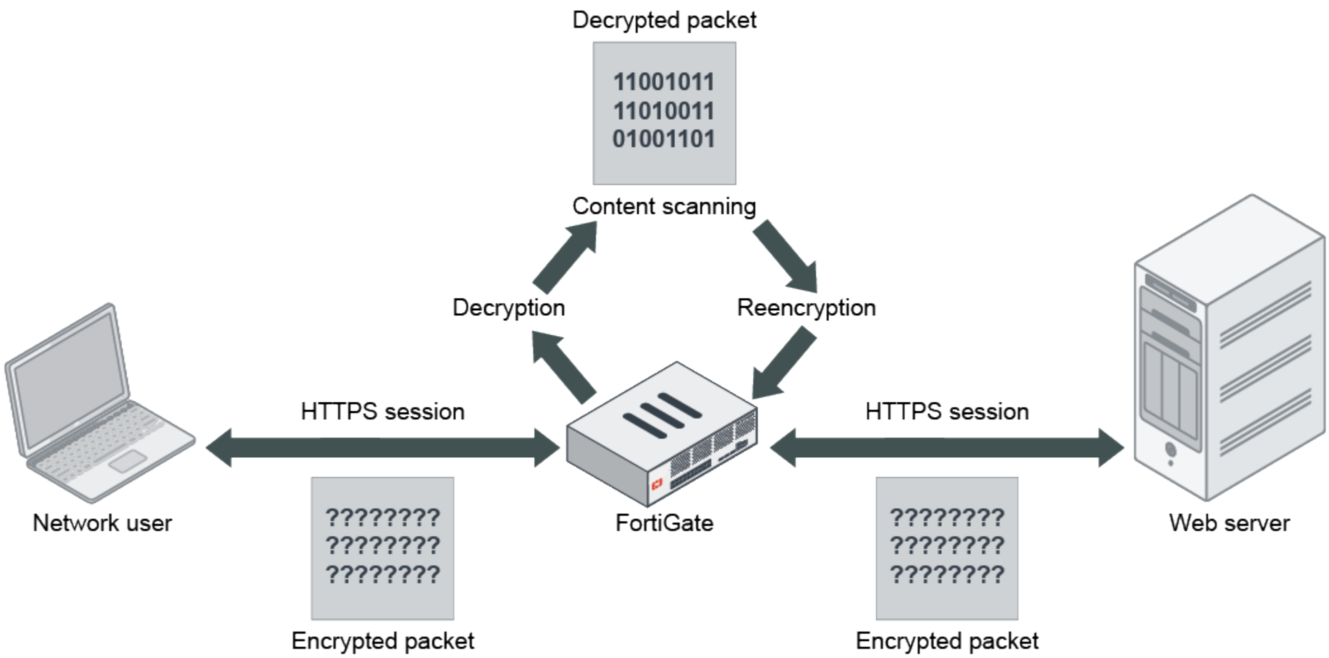
Reasons for using deep inspection

While Hypertext Transfer Protocol Secure (HTTPS) offers protection on the Internet by applying Secure Sockets Layer (SSL) encryption to web traffic, encrypted traffic can be used to get around your network's normal defenses.

For example, you might download a file containing a virus during an e-commerce session, or you might receive a phishing email containing a seemingly harmless download that, when launched, creates an encrypted session to a command and control (C&C) server and downloads malware onto your computer. Because the sessions in these attacks are encrypted, they might get past your network's security measures.

When you use deep inspection, the FortiGate impersonates the recipient of the originating SSL session, then decrypts and inspects the content to find threats and block them. It then re-encrypts the content and sends it to the real recipient.

Deep inspection not only protects you from attacks that use HTTPS, it also protects you from other commonly-used SSL-encrypted protocols such as SMTPS, POP3S, IMAPS, and FTPS.



Browser messages when using deep inspection

When the FortiGate re-encrypts the content, it uses a stored certificate, such as *Fortinet_CA_SSL*, *Fortinet_CA_Untrusted*, or your own CA certificate that you uploaded.

Because there is no *Fortinet_CA_SSL* in the browser trusted CA list, the browser displays an untrusted certificate warning when it receives a FortiGate re-signed server certificate. To stop the warning messages, trust the FortiGate-trusted CA *Fortinet_CA_SSL* and import it into your browser.

If you still get messages about untrusted certificates after importing *Fortinet_CA_SSL* into your browser, it is due to *Fortinet_CA_Untrusted*. Never import the *Fortinet_CA_Untrusted* certificate into your browser.

To import *Fortinet_CA_SSL* into your browser:

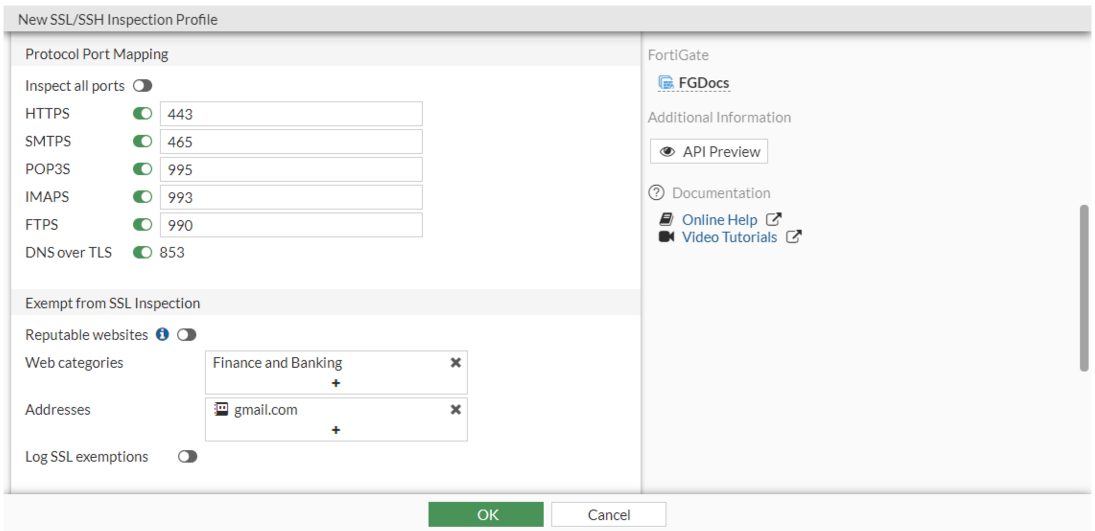
1. On the FortiGate, go to *Security Profiles > SSL/SSH Inspection* and edit the *deep-inspection* profile.
   The default *CA Certificate* is *Fortinet_CA_SSL*.
2. Click *Download* and save the certificate to the management computer.
3. On the client PC, use the *Certificate Import Wizard* to install the certificate into the *Trusted Root Certificate Authorities* store.
   If a security warning appears, select *Yes* to install the certificate.

Exempt web sites from deep inspection

If you do not want to apply deep inspection for privacy or other reasons, you can exempt the session by address, category, or allowlist.

If you know the address of the server you want to exempt, you can exempt that address. You can exempt specific address type including IP address, IP address range, IP subnet, FQDN, wildcard-FQDN, and geography.

If you want to exempt all bank web sites, an easy way is to exempt the *Finance and Banking* category, which includes all finance and bank web sites identified in FortiGuard. For information about creating and using custom local and remote categories, see Web rating override on page 1318 and Threat feeds on page 2504.



If you want to exempt commonly trusted web sites, you can bypass the SSL allowlist in the SSL/SSH profile by enabling *Reputable websites*. The allowlist includes common web sites trusted by FortiGuard.

SSL version support

There are two ways to limit which SSL versions deep inspection is applied to.

- In the global attributes:

```
config system global
    set strong-crypto enable
end
```

- In the protocol configuration of a deep inspection profile:

```
config firewall ssl-ssh-profile
    edit <name>
        config {ssl | https | ftps}
            set min-allowed-ssl-version {ssl-3.0 | tls-1.0 | tls-1.1 | tls-1.2 | tls-
1.3}
        end
    next
end
```

Enabling `strong-crypto` in the global attributes sets the `min-allowed-ssl-version` to `tls-1.1` by default.

When a session is attempted using an SSL version below the minimum allowed version, the session can be blocked (default) or allowed.

To configure the action based on the SSL version used being unsupported:

```
config firewall ssl-ssh-profile
    edit <name>
        config {ssl | https | ftps | imaps | pop3s | smtps | dot}
            set unsupported-ssl-version {allow | block}
        end
    next
end
```

Protecting an SSL server

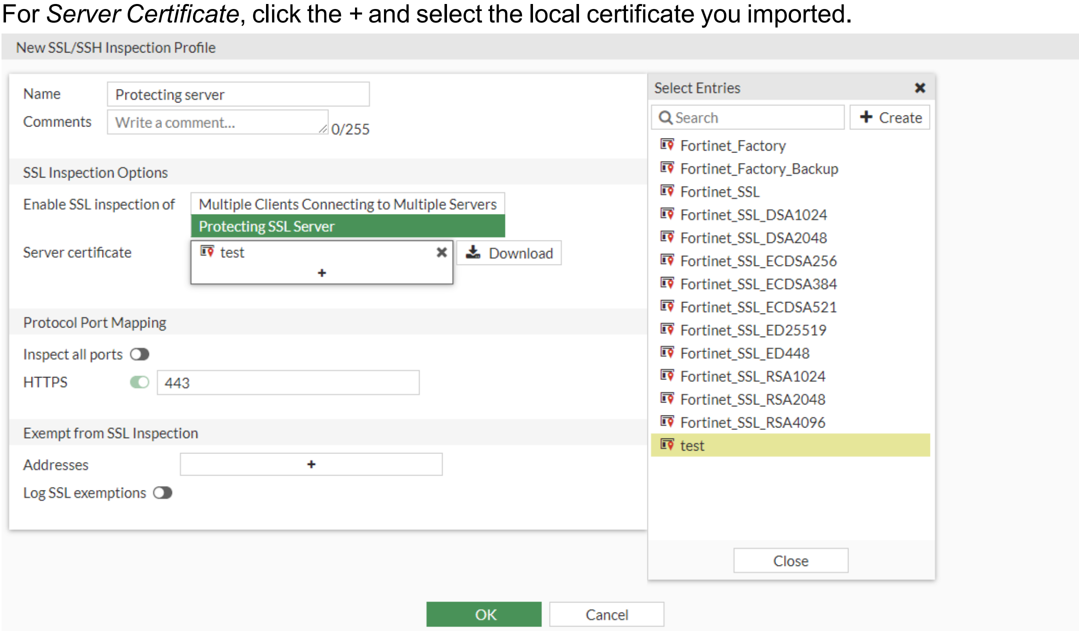
You typically use the FortiGate *Protecting SSL Server* profile as an inbound policy for clients on the internet that access the server through the internal side of the FortiGate.

Protecting SSL Server uses a server certificate to protect a single server.

You can use *Protecting SSL Server* if you do not want a client on the internet to directly access your internal server, and you want the FortiGate to simulate your real server.

To upload a server certificate into FortiGate and use that certificate in the SSL/SSH inspection profile:

1. Go to *System > Certificates*.
2. Select *Import > Local Certificate* and upload the certificate.
3. Go to *Security Profiles > SSL/SSH Inspection* and edit or create a new profile.
4. For *Enable SSL Inspection of*, select *Protecting SSL Server*.
5. For *Server Certificate*, click the + and select the local certificate you imported.



6. Click *OK*.

When you apply the *Protecting SSL Server* profile in a policy, the FortiGate will send the server certificate to the client as your server does.

Handling SSL offloaded traffic from an external decryption device

In scenarios where the FortiGate is sandwiched between load-balancers and SSL processing is offloaded on the external load-balancers, the FortiGate can perform scanning on the unencrypted traffic by specifying the `ssl-offloaded` option in firewall `profile-protocol-options`. This option is supported in proxy and flow mode (previous versions only supported proxy mode).

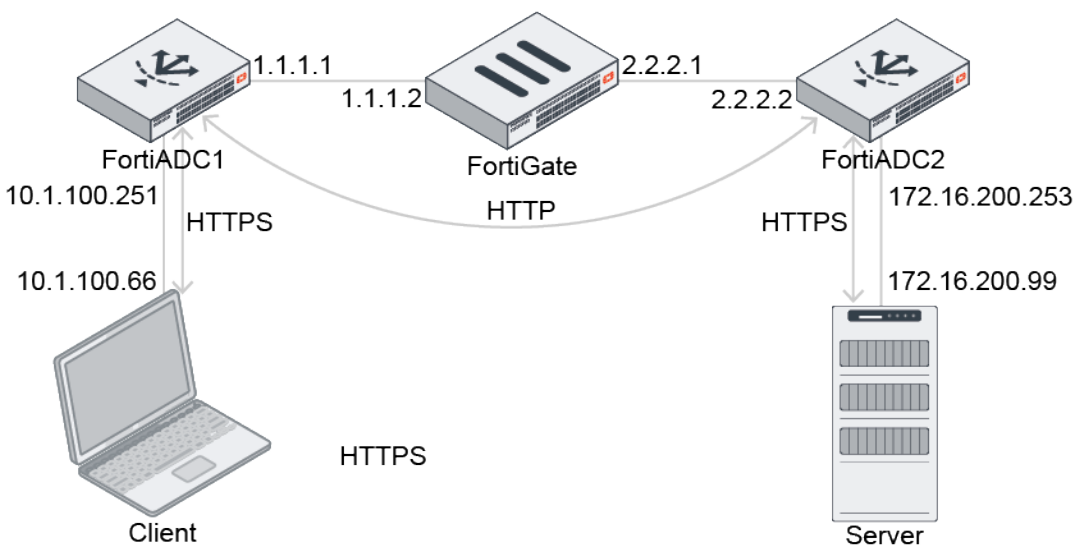
If the FortiGate receives an AUTH TLS, PBSZ, or PROT command before receiving plain text traffic from a decrypted device, by default, it will expect encrypted traffic, determine that the traffic belongs to an abnormal protocol, and bypass the traffic.

When the `ssl-offloaded` command is enabled, the AUTH TLS command is ignored, and the traffic is treated as plain text rather than encrypted data. SSL decryption and encryption are performed by the external device.

Sample topology

In this example, the FortiGate is between two FortiADCs and in SSL offload sandwich mode. The FortiGate receives plain text from ADC1 and forwards plain text to ADC2. There is no encrypted traffic passing through the FortiGate.

The client sends HTTPS traffic to ADC1, which then decrypts the traffic and sends HTTP to the FortiGate. The FortiGate forwards HTTP to ADC2, and the ADC2 re-encrypts the traffic to HTTPS.



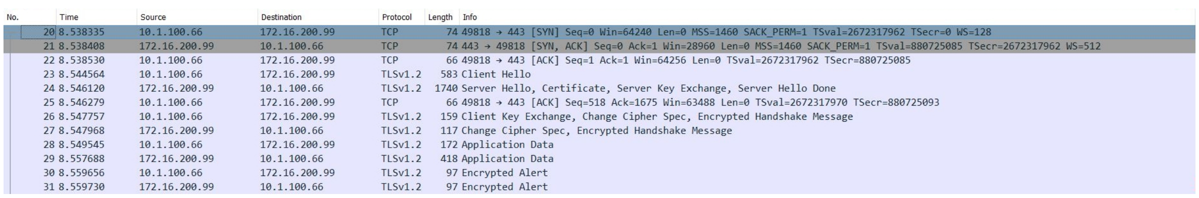
To configure SSL offloading:

```
config firewall profile-protocol-options
    edit "default-clone"
        config http
            set ports 80
            unset options
            unset post-lang
            set ssl-offloaded yes
        end
        config ftp
            set ports 21
            set options splice
```

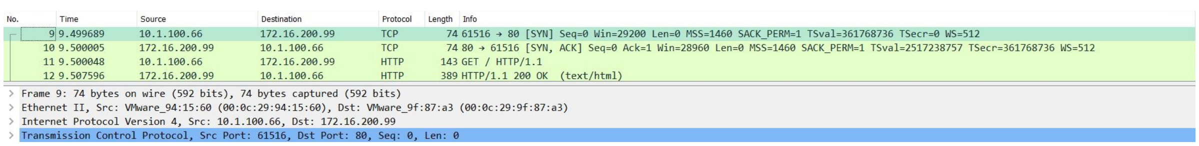
```
            set ssl-offloaded yes
        end
        config imap
            set ports 143
            set options fragmail
            set ssl-offloaded yes
        end
        config pop3
            set ports 110
            set options fragmail
            set ssl-offloaded yes
        end
        config smtp
            set ports 25
            set options fragmail splice
            set ssl-offloaded yes
        end
    next
end
```

Verifying the packet captures

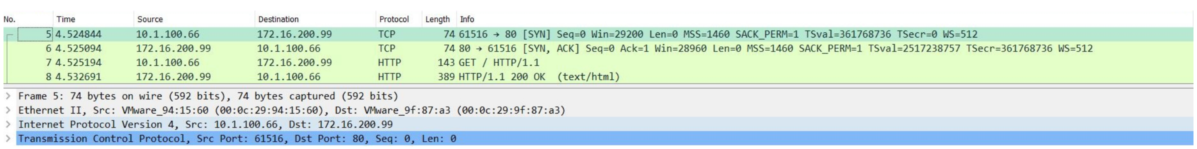
The ADC1 incoming port capture shows that ADC1 receives HTTPS traffic:

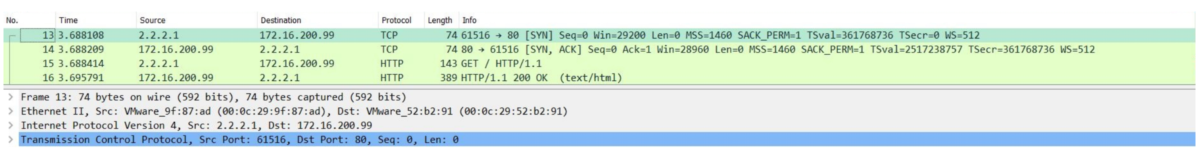


The ADC1 outgoing port capture shows that ADC1 decrypts traffic and forwards HTTP traffic to the FortiGate:

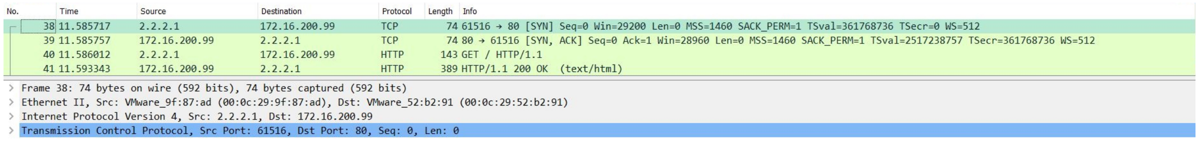


The FortiGate's incoming and outgoing port captures show that HTTP traffic passes through the FortiGate:

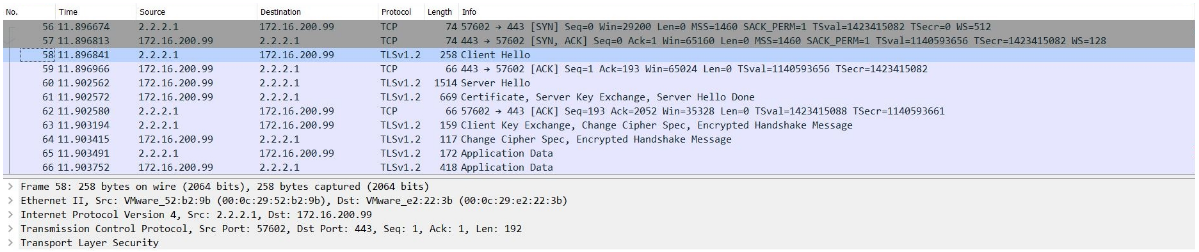




The ADC2 incoming port capture shows that the ADC2 receives HTTP traffic:



The ADC2 outgoing port capture shows that ADC2 forwards HTTPS traffic to the server:



SSH traffic file scanning

FortiGates can buffer, scan, log, or block files sent over SSH traffic (SCP and SFTP) depending on the file size, type, or contents (such as viruses or sensitive content).

This feature is supported in proxy-based inspection mode. It is currently not supported in flow-based inspection mode.

You can configure the following SSH traffic settings in the CLI:

- Protocol options
- DLP profile
- Antivirus (profile and quarantine options)

To configure SSH protocol options:

```
config firewall profile-protocol-options
    edit <name>
        config ssh
            set options {oversize clientcomfort servercomfort}
            set comfort-interval <1 - 900>
            set comfort-amount <1 - 65535>
            set oversize-limit <1 - 798>
            set uncompressed-oversize-limit <0 - 798>
            set uncompressed-nest-limit <2 - 100>
            set scan-bzip2 {enable | disable}
        end
    next
end
```

To configure SCP block and log options:

```
config ssh-filter profile
    edit <name>
        set block scp
```

```
        set log scp
    next
end
```

To configure the DLP profile:

```
config dlp profile
    edit <name>
        set full-archive-proto ssh
        set summary-proto ssh
        config filter
            edit 1
                set proto ssh
            next
        end
    next
end
```

To configure the antivirus profile options:

```
config antivirus profile
    edit <name>
        config ssh
            set av-scan {disable | block | monitor}
            set outbreak-prevention {disable | block | monitor}
            set external-blocklist {disable | block | monitor}
            set fortindr {disable | block | monitor}
            set quarantine {enable | disable}
            set archive-block {encrypted corrupted partiallycorrupted multipart nested
mailbomb timeout unhandled}
            set archive-log {encrypted corrupted partiallycorrupted multipart nested
mailbomb timeout unhandled}
            set emulator {enable | disable}
        end
    next
end
```

To configure the antivirus quarantine options:

```
config antivirus quarantine
    set drop-infected ssh
    set store-infected ssh
    set drop-blocked ssh
    set store-blocked ssh
    set drop-machine-learning ssh
    set store-machine-learning ssh
end
```

Redirect to WAD after handshake completion

In a proxy-based policy, the TCP connection is proxied by the FortiGate. A TCP three-way handshake can be established with the client even though the server did not complete the handshake.

This option uses IPS to handle the initial TCP three-way handshake. It rebuilds the sockets and redirects the session back to proxy only when the handshake with the server is established.

To enable proxy after a TCP handshake in an SSL/SSH profile:

```
config firewall ssl-ssh-profile
    edit "test"
        config https
            set ports 443
            set status certificate-inspection
            set proxy-after-tcp-handshake enable
        end
    next
end
```

To enable proxy after a TCP handshake in protocol options:

```
config firewall profile-protocol-options
    edit "test"
        config http
            set ports 80
            set proxy-after-tcp-handshake enable
            unset options
            unset post-lang
        end
    next
end
```

HTTP/2 support in proxy mode SSL inspection

Security profiles in proxy mode can perform SSL inspection on HTTP/2 traffic that is secured by TLS 1.2 or 1.3 using the Application-Layer Protocol Negotiation (ALPN) extension.

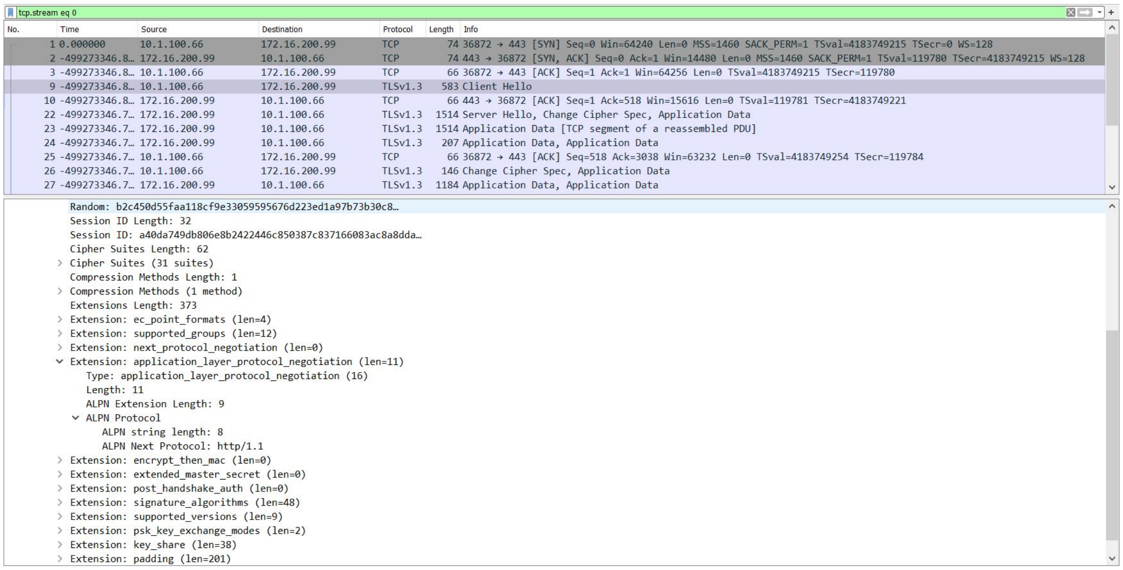
To set the ALPN support:

```
config firewall ssl-ssh-profile
    edit <profile>
        set supported-alpn {all | http1-1 | http2 | none}
    next
end
```

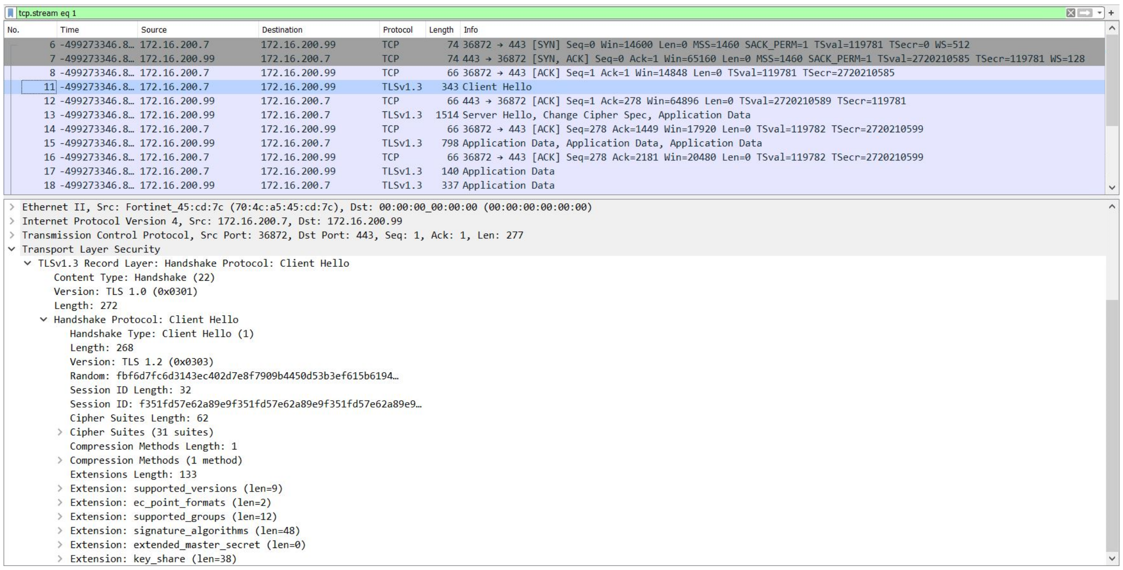
`all`	The FortiGate forwards ALPN extensions that use either HTTP/2 or HTTP/1.1. This is the default value.
`http1-1`	The FortiGate only forwards ALPN extensions that use HTTP/1.1. If the ALPN extension uses HTTP/2, then the FortiGate strips the ALPN header from the Client Hello.
`http2`	The FortiGate only forwards ALPN extensions that use HTTP/2. If the ALPN extension uses HTTP/1.1, then the FortiGate strips the ALPN header from the Client Hello.
`none`	The FortiGate always strips the ALPN header from the Client Hello when forwarding.

For example, if `supported-alpn` is set to `http2`, but the extension uses HTTP/1.1, the ALPN header is stripped from the Client Hello:

- Incoming packet capture:



- Outgoing packet capture:

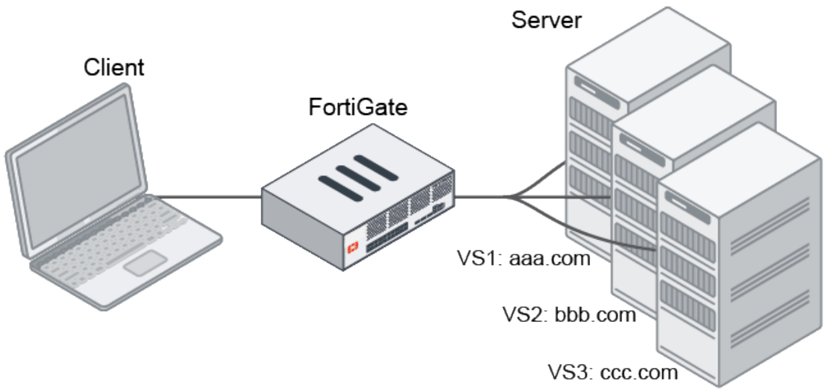


Define multiple certificates in an SSL profile in replace mode

Multiple certificates can be defined in an SSL inspection profile in replace mode (*Protecting SSL Server*). This allows multiple sites to be deployed on the same protected server IP address, and inspection based on matching the SNI in the certificate.

When the FortiGate receives the client and server hello messages, it will compare the server name identification (SNI) and the common name (CN) with the certificate list in the SSL profile, and use the matched certificate as a replacement. If there is no matched server certificate in the list, then the first server certificate in the list is used as a replacement.

Example



To configure an SSL profile in replace mode with multiple certificates:

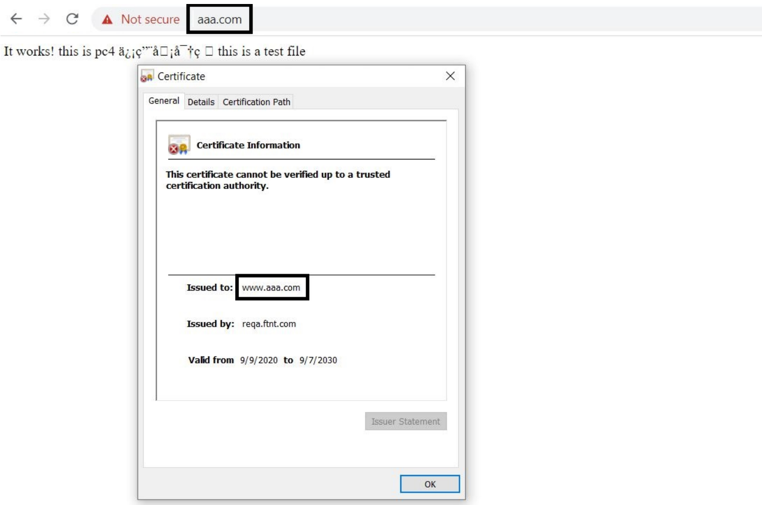
```
config firewall ssl-ssh-profile
    edit "multi-cert"
        set server-cert-mode replace
        set server-cert "bbb" "aaa"
    next
end
```

To configure a policy that uses the SSL profile:

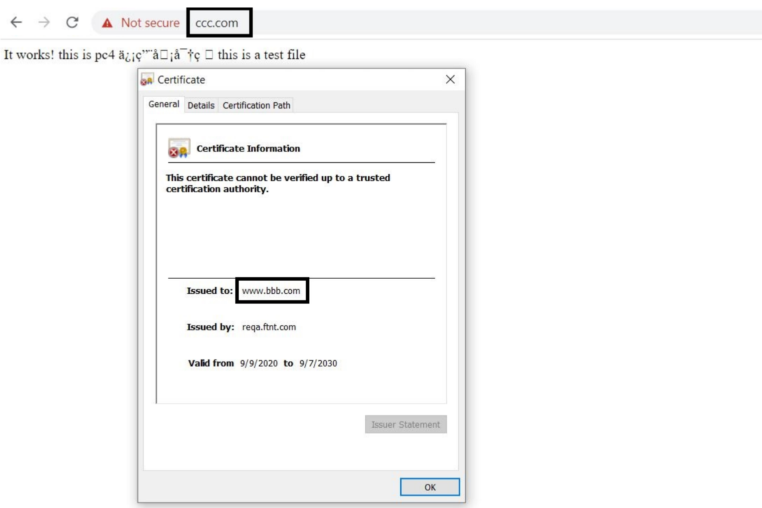
```
config firewall policy
    edit 1
        set name "multi-cert"
        set srcintf "port6"
        set dstintf "port11"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set ssl-ssh-profile "multi-cert"
        set av-profile "default"
        set webfilter-profile "default"
        set logtraffic all
        set nat enable
    next
end
```

Results

If the SNI matches the CN in the certificate list in the SSL profile, then the FortiGate uses the matched server certificate. In this example, when the client accesses *www.aaa.com*, the FortiGate will use the *aaa* certificate as a replacement.



If the SNI does not match the CN in the certificate list in the SSL profile, then the FortiGate uses the first server certificate in the list. In this example, when the client accesses *www.ccc.com*, because there is no certificate for *www.ccc.com*, the FortiGate will use the *bbb* certificate as a replacement.



Disabling the FortiGuard IP address rating

The FortiGuard IP address rating for SSL exemptions and proxy addresses can be disabled using the `ssl-exemption-ip-rating` and `address-ip-rating` options.

To disable using the FortiGuard IP address rating for SSL exemptions:

```
config firewall ssl-ssh-profile
    edit <name>
        set ssl-exemption-ip-rating {enable | disable}
    next
end
```

To disable using the FortiGuard IP address rating for proxy addresses:

```
config firewall profile-protocol-options
    edit <name>
        config http
            set address-ip-rating {enable | disable}
        end
    next
end
```

The `ssl-exemption-ip-rating` and `address-ip-rating` options are enabled by default, so when both a website domain and its IP address return different categories after being rated by FortiGuard, the IP address category takes precedence when evaluating SSL exemptions associated with the SSL inspection profile and proxy addresses associated with the proxy protocol options profile. SSL exemptions and the `ssl-exemption-ip-rating` option work in both inspection modes (proxy and flow).

When the categories associated with the website domain and IP address are different, disabling the FortiGuard IP rating ensures that the FortiGuard domain category takes precedence when evaluating the preceding objects. For most websites, the domain category is valid when its IP address is unrated by FortiGuard. Since being unrated is considered as not having a category, the FortiGate uses the domain category as the website category.

A website might have an IP category that differs from its domain category. If they are different, the FortiGate uses the rating weight of the IP address or domain name to determine the rating result and decision. The rating weight is hard-coded in the FortiGate and depending on the relative category weights, the FortiGate may use the IP category instead of the website category. If the `ssl-exemption-ip-rating` option is disabled in the SSL inspection profile, then the FortiGate uses the domain category as the website category, which ensures SSL exemption operation as intended.

The `address-ip-rating` option in a proxy protocol options profile functions the same way as the `ssl-exemption-ip-rating` option. If the `address-ip-rating` option is disabled in a profile that is used in an explicit proxy policy that also uses a web filter profile, for HTTP or HTTPS traffic to a website that has different IP and domain categories and that matches the policy, the FortiGate will use the domain category when it evaluates categories for the web filter.

Custom signatures

You can create the following custom signatures and apply them to firewall policies:

- IPS signature
- Application signature
- Application group

The following topic provides information about custom signatures:

- Application groups in traffic shaping policies on page 1308
- Blocking applications with custom signatures on page 1312
- Filters for application control groups on page 1314

Application groups in traffic shaping policies

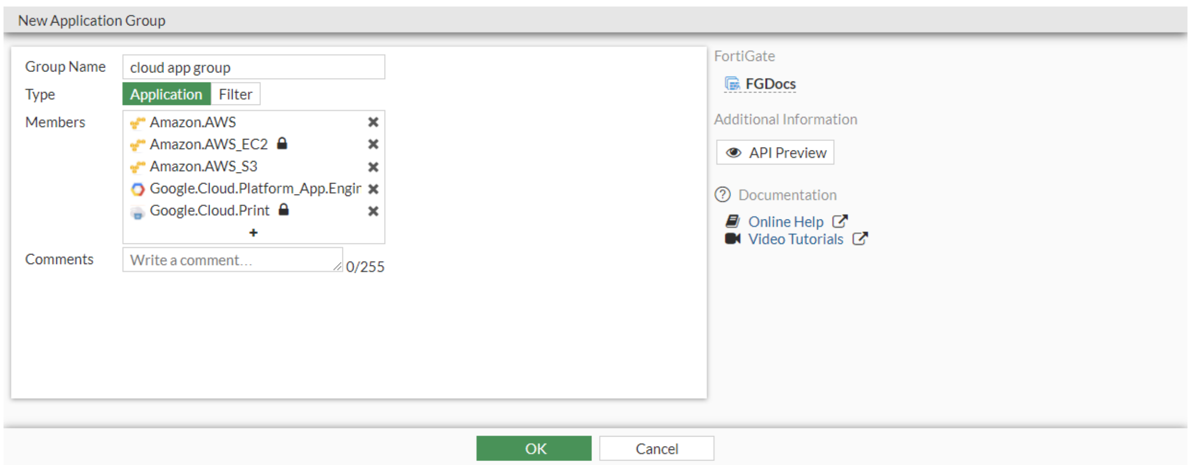
Application groups can be configured in traffic shaping policies. In this example, there are two traffic shaping policies:

- Policy 1 is for traffic related to cloud applications and has high priority.
- Policy 2 is for other traffic and has low priority.

At least one firewall policy must have application control enabled for the applications to match any policy traffic.

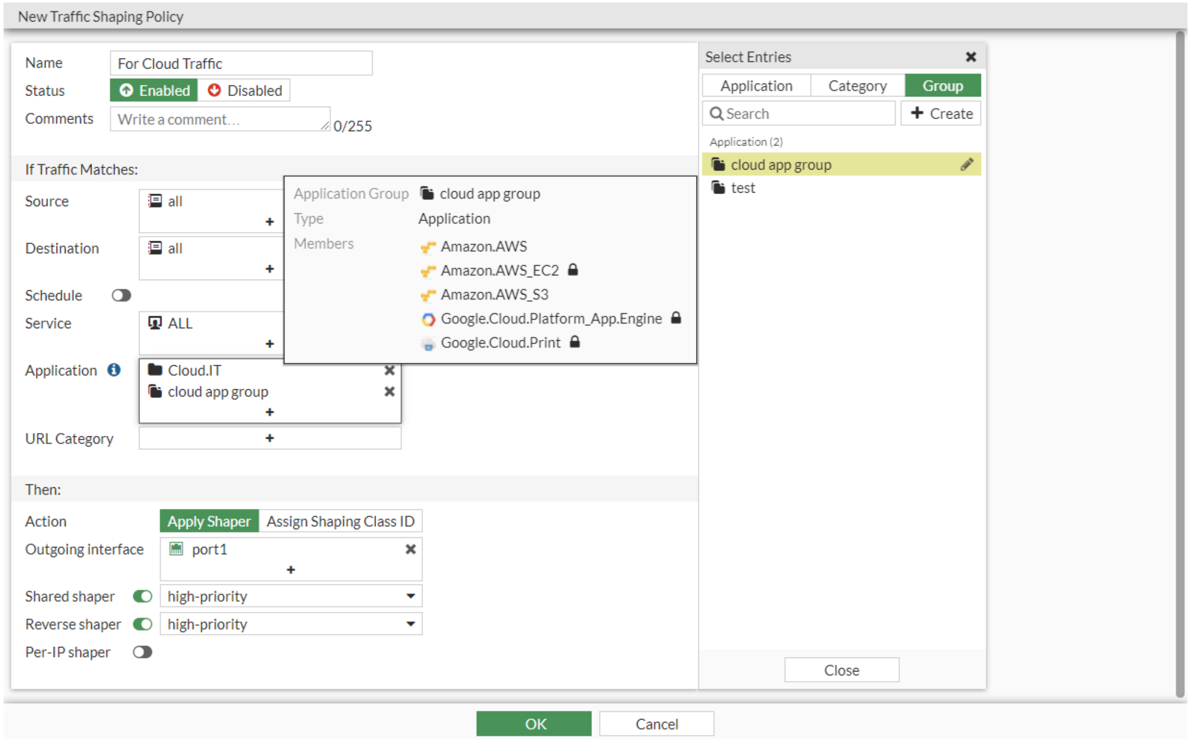
To configure a traffic shaping policy to use an application group in the GUI:

1. Configure an application group for cloud applications:
   a. Go to *Security Profiles > Application Signatures*.
   b. Click *Create New > Application Group*. The *New Application Group* page opens.
   c. Enter a name for the group, and for *Type*, select *Application*.
   d. Click the + to add the group the members.



   e. Click *OK*.
2. Create the shaping policy for the high priority cloud application traffic:
   a. Go to *Policy & Objects > Traffic Shaping*, select the *Traffic Shaping Policies* tab, and click *Create New*.
   b. Enter the following:

Name	For Cloud Traffic
Source	All
Destination	All
Service	All
Application	Add the *Cloud.IT* category and the *cloud app group* application group.
Outgoing interface	port1
Shared shaper	high-priority
Reverse shaper	high-priority

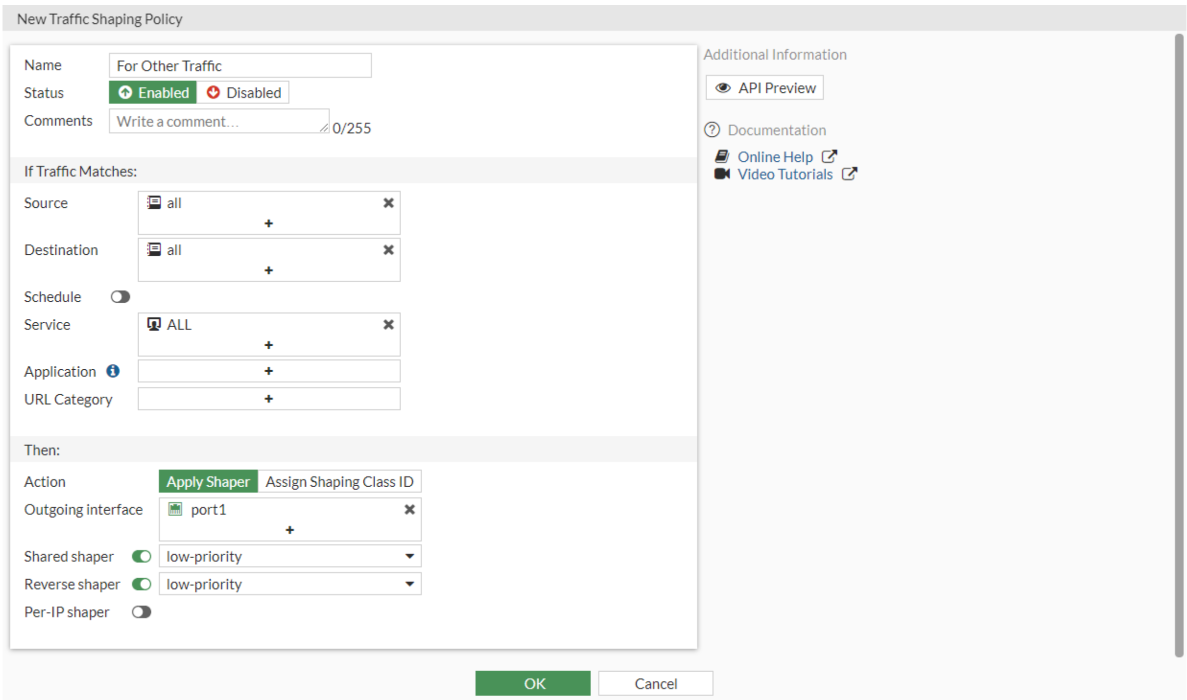


c. Click *OK*.

3. Create the shaping policy for the low priority other traffic:

a. Click *Create New* and enter the following:

Name	For Other Traffic
Source	All
Destination	All
Service	All
Outgoing interface	port1
Shared shaper	low-priority
Reverse shaper	low-priority



b. Click *OK*.

To configure a traffic shaping policy to use an application group in the CLI:

1. Configure an application group for cloud applications:

```
config application group
    edit "cloud app group"
        set application 27210 36740 35944 43296 33048
    next
end
```

2. Create the shaping policies for the high priority cloud application traffic and low priority other traffic:

```
config firewall shaping-policy
    edit 1
        set name "For Cloud Traffic"
        set service "ALL"
        set app-category 30
```

```
            set app-group "cloud app group"
            set dstintf "port1"
            set traffic-shaper "high-priority"
            set traffic-shaper-reverse "high-priority"
            set srcaddr "all"
            set dstaddr "all"
        next
        edit 2
            set name "For Other Traffic"
            set service "ALL"
            set dstintf "port1"
            set traffic-shaper "low-priority"
            set traffic-shaper-reverse "low-priority"
            set srcaddr "all"
            set dstaddr "all"
        next
    end
```

Blocking applications with custom signatures

Custom signatures can be used in application control profiles to block web traffic from specific applications, such as out of support operating systems.

In this example, a custom signature is created to detect PCs running Windows NT 6.1 operating systems, including Windows 7 and Windows Server 2008 R2. The signature is added to an application control profile and the action is set to block. The profile is then used in a firewall policy so that web traffic matching the signature is blocked. The logs generated by this example can be used to help identify other computers that need to be blocked.

To make the settings visible in the GUI:

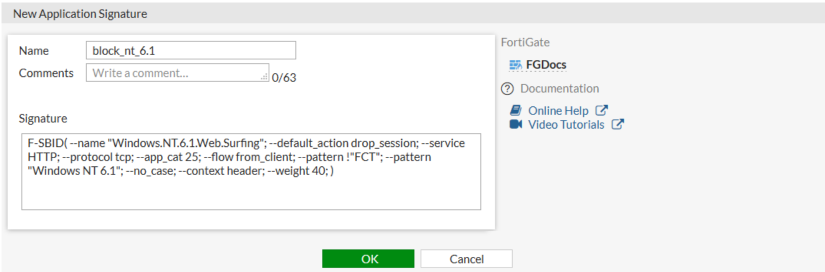
1. Go to *System* > *Feature Visibility*
2. In the *Security Features* section, enable *Application Control*.
3. Click *Apply*.

To create the custom application signature:

1. Go to *Security Profiles* > *Application Signatures* and click *Create New* > *Custom Application Signature*.
2. Enter a name for the custom signature, such as *block_nt_6.1*.
3. Enter the *Signature*. In this example:

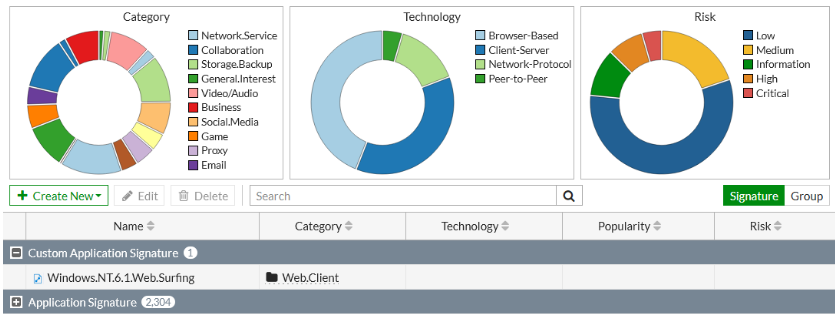
   ```
   F-SBID( --attack_id 6483; --name "Windows.NT.6.1.Web.Surfing"; --default_action drop_
   session; --service HTTP; --protocol tcp; --app_cat 25; --flow from_client; --pattern
   !"FCT"; --pattern "Windows NT 6.1"; --no_case; --context header; --weight 40; )
   ```

   This signature scans HTTP and HTTPS traffic that matches the pattern *Windows NT 6.1* in its header. For blocking older versions of Windows, such as Windows XP, you would use the pattern *Windows NT 5.1*. An attack ID is automatically generated when the signature is created.



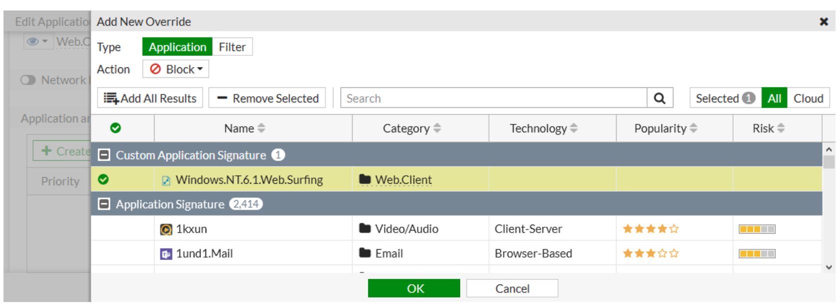
4. Click *OK*.

   The signature is included in the *Custom Application Signature* section of the signature list.



To use the signature in an application control profile:

1. Go to *Security Profiles > Application Control*.
2. Create a new profile, or edit an existing one.
3. In the *Application and Filter Overrides* table, click *Create New*.
4. Set *Type* to *Application* and *Action* to *Block*.
5. Select the custom signature from the list, using the search feature if required, then click *Add Selected*.



6. Click *OK*.

   The signature is added to the table.
7. Click *OK*.

To add the application control profile to a firewall policy:

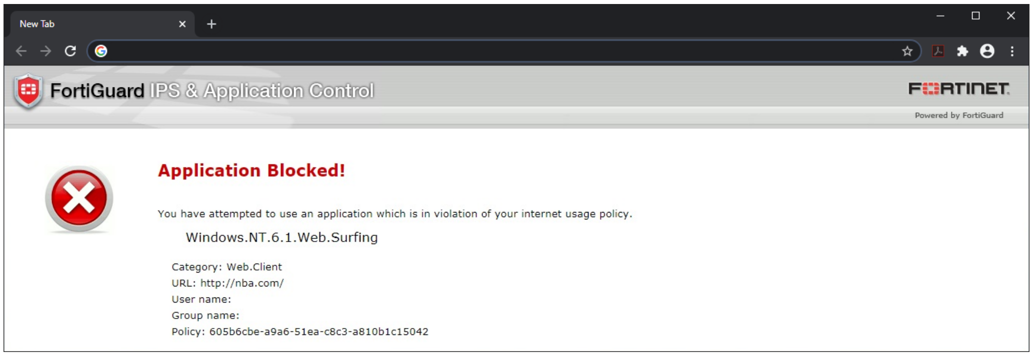
1. Go to *Policy & Objects > Firewall Policy*.
2. Edit the policy that currently allows a connection from the internal network to the internet.
3. In the *Security Profiles* section, enable *Application Control* and select the profile.

   If deep inspection is not enabled, then only HTTP traffic will be scanned. To scan HTTPS traffic, set SSL Inspection to a profile that includes deep inspection. See SSL & SSH Inspection on page 1294 for more information.

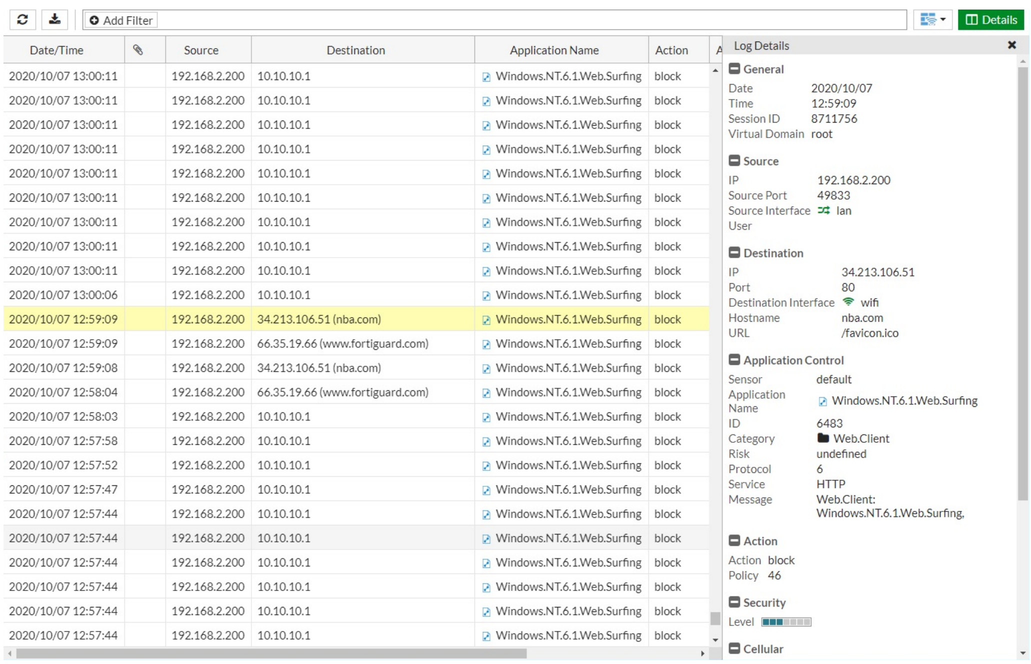
4. Click *OK*.

Results

When a PC running one of the affected operating systems tries to connect to the internet using a web browser, a replacement message is shown. For information on customizing replacement messages, see Replacement messages on page 2134.



Go to *Log & Report > Security Events* to view the web traffic that is logged for the PC that is blocked by the application signature in the *Application Control* card.



Filters for application control groups

When defining application groups in NGFW policy or profile mode, the following group filters are available: protocols, risk, vendor, technology, behavior, popularity, and category.

```
config application group
    edit <name>
        set type filter
        set protocols <integer>
        set risk <integer>
        set vendor <id>
        set technology <id>
        set behavior <id>
        set popularity <integer>
        set category <id>
    next
end
```

`protocols <integer>`	Application protocol filter (0 - 47, or all).
`risk <integer>`	Risk or impact of allowing traffic from this application to occur (1 - 5; low (1), elevated (2), medium (3), high (4), and critical (5)).
`vendor <id>`	Application vendor filter (0 - 25, or all).
`technology <id>`	Application technology filter: • all • 0 (network-protocol) • 1 (browser-based) • 2 (client-server) • 4 (peer-to-peer)
`behavior <id>`	Application behavior filter: • all • 2 (botnet) • 3 (evasive) • 5 (excessive bandwidth) • 6 (tunneling) • 9 (cloud)
`popularity <integer>`	Application popularity filter (1 - 5, from least to most popular).
`category <id>`	Application category filter: • 2 (P2P) • 3 (VoIP) • 5 (video/audio) • 6 (proxy) • 7 (remote access) • 8 (game) • 12 (general interest) • 15 (network service) • 17 (update) • 21 (email) • 22 (storage backup) • 23 (social media) • 25 (web client)

- 26 (industrial)
- 28 (collaboration)
- 29 (business)
- 30 (cloud IT)
- 31 (mobile)
- 32 (unknown applications)

Sample configurations

In this example, a single filter (risk level 1) is configured in the application group in NGFW policy mode, so only signatures matching this filter will match the security policy.

To configure the application group:

```
config application group
    edit "risk_1"
        set type filter
        set risk 1
    next
end
```

To configure the security policy:

```
config firewall security-policy
    edit 1
        set srcintf "port2"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set status enable
        set schedule "always"
        set enforce-default-app-port disable
        set service "ALL"
        set app-group risk_1
        set logtraffic all
    next
end
```

In this example, the application group is configured so that only signatures matching both filters, category 5 (video/audio) and technology 1 (browser-based), will match the security policy. The application group can also be configured in a traffic shaping policy.

To configure the application group:

```
config application group
    edit "two"
        set type filter
        set category 5
        set technology 1
    next
 end
```

To configure the security policy:

```
config firewall security-policy
    edit 1
        set srcintf "port2"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set status enable
        set schedule "always"
        set enforce-default-app-port disable
        set service "ALL"
        set app-group two
        set logtraffic all
    next
end
```

To configure the traffic shaping policy:

```
config firewall shaping-policy
    edit 1
        set ip-version 4
        set service "ALL"
        set app-group two
        set dstintf port1
        set traffic-shaper "max-100"
        set traffic-shaper-reverse "max-100"
        set srcaddr "all"
        set dstaddr "all"
    next
end
```

Overrides

Web filter configuration can be separated into profile configuration and profile overrides.

You can also override web filter behavior based on the FortiGuard website categorization:

- Use alternate categories (web rating overrides): this method manually assigns a specific website to a different Fortinet category or a locally-created category.
- Use alternate profiles: configured users or IP addresses can use an alternative web filter profile when attempting to access blocked websites.

Some features of this functionality require a subscription to FortiGuard Web Filtering.

The following topics provide information about web overrides:

- Web rating override on page 1318
- Web profile override on page 1323

Web rating override

Web rating overrides allow you to add specific URLs to both FortiGuard and custom web ratings categories.

In a web filter profile, the action for each category can be configured. See FortiGuard filter on page 1128 for details. A web rating override in a custom category will not impact any web filters until the category's action is changed to *Allow*, *Monitor* (default), *Block*, *Warning*, or *Authenticate* in the specific web filter profile's settings. If a URL is in multiple enabled categories, the order of precedence is local categories, then remote categories, and then FortiGuard categories.

In SSL/SSH inspection profiles, custom categories must be explicitly selected to be exempt from SSL inspection. In proxy addresses, custom categories must be explicitly selected as URL categories for them to apply. In both settings, if a URL is in multiple selected categories, the order of precedence is local categories, then remote categories, and then FortiGuard categories.

Web rating override requires a FortiGuard license.

Web filter profiles

In this example, *www.fortinet.com* is added to both a custom, or local, category (*Seriously*) and an external threat feed, or remote, category (*OnAworkComputer*). The local category action is set to *Monitor*, while the remote category action is set to *Block*. When a user browses to *www.fortinet.com*, the local category action takes precedence over both the remote category and the FortiGuard category (*Information Technology*), so the *Monitor* action is taken.

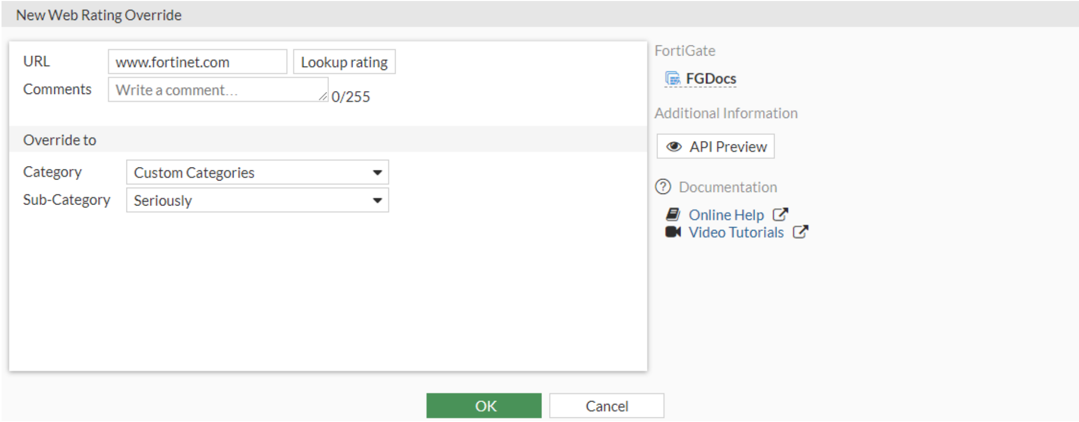
To create a custom category in the GUI:

1. Go to *Security Profiles > Web Rating Overrides*.
2. Click *Custom Categories*, then click *Create New*.
3. Enter a name for the category, and ensure the *Status* is set to *Enable*.
4. Click *OK*.

To create a web rating override in the GUI:

1. Go to *Security Profiles > Web Rating Overrides* and click *Create New*.
2. Enter the URL to override.
3. Optionally, click *Lookup rating* to see what its current rating is, if it has one.

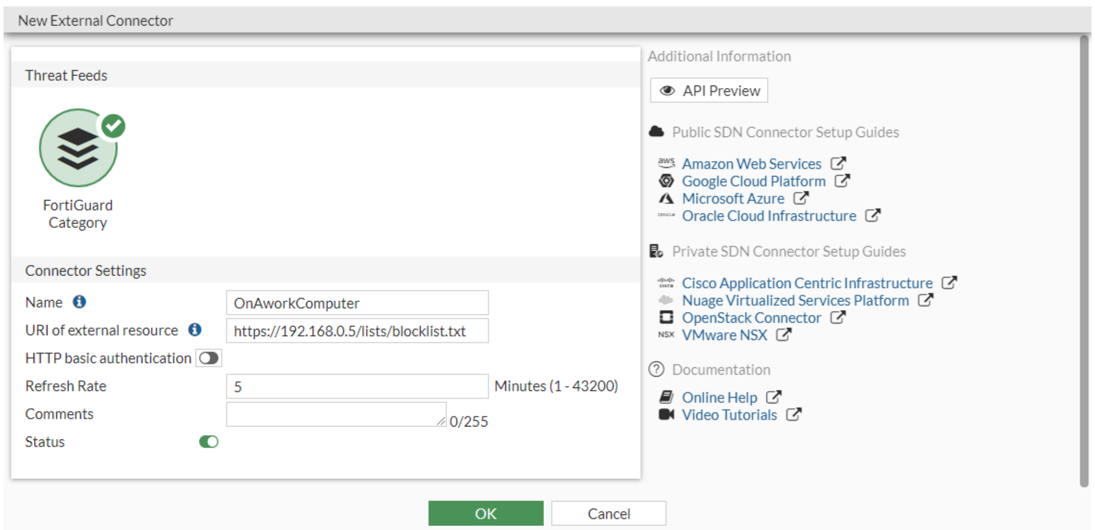
4. For *Category*, select *Custom Categories* and for *Sub-Category* select the category previously created.



5. Click *OK*.

To create a new FortiGuard category threat feed in the GUI:

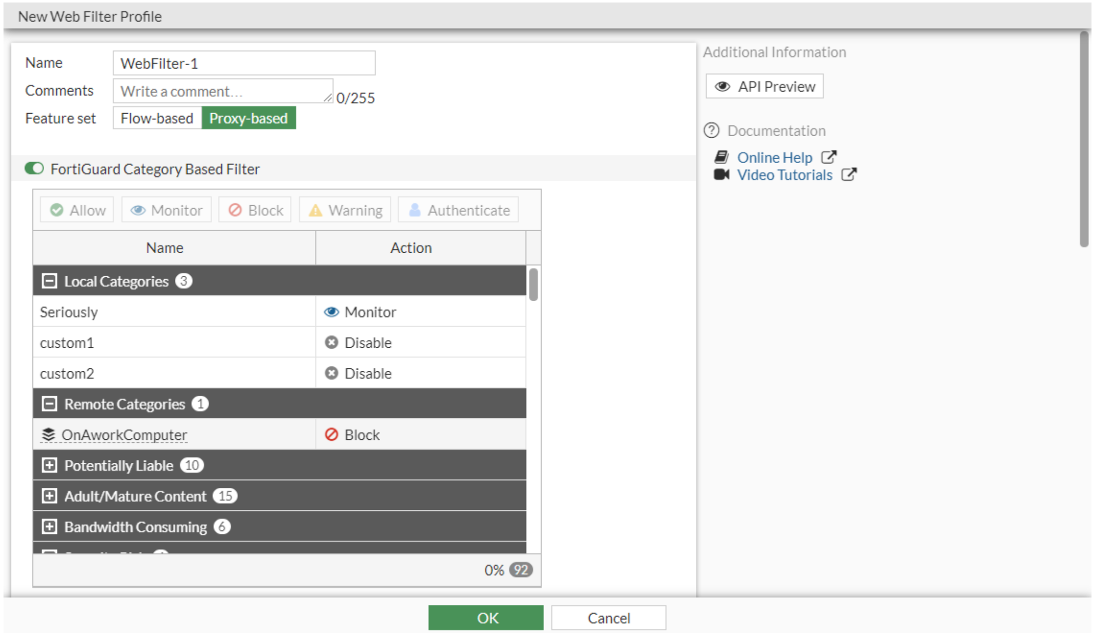
1. Go to *Security Fabric > External Connectors* and click *Create New*.
2. In the *Threat Feeds* section, click *FortiGuard Category*.
3. Enter a name for the threat feed, such as *OnAworkComputer*.
4. Enter the *URI of external resource*.



5. Configure the remaining settings as needed, then click *OK*.

To use the new categories in a web filter profile in the GUI:

1. Go to *Security Profiles > Web Filter* and create or edit a web filter profile. See FortiGuard filter on page 1128 for more information.
2. Enable *FortiGuard category based filter*
3. Set the action for the *Seriously* category in the *Local Categories* group to *Monitor*.
4. Set the action for the *OnAworkComputer* category in the *Remote Categories* group to *Block*.



Setting the custom category action to *Allow* is equivalent to setting the CLI `action` variable to `monitor` and `log` variable to `disable`.

5. Configure the remaining settings are required, then click *OK*.

To use local and remote categories in a web filter profile in the CLI:

1. Create the custom category and add a URL to it:

```
config vdom
    edit root
        config webfilter ftgd-local-cat
            edit "Seriously"
                set id 140
            next
        end
        config webfilter ftgd-local-rating
            edit "www.fortinet.com"
                set rating 140
            next
        end
    next
end
```

2. Create a *FortiGuard Category Threat Feed* external connector to import an external blocklist.

```
config global
    config system external-resource
        edit "OnAworkComputer"
            set category 192
            set resource "https://192.168.0.5/lists/blocklist.txt"
        next
```

```
    end
end
```

3. Enable the new category in a web filter profile. See FortiGuard filter on page 1128 for details.
   Custom local categories have an ID range of 140 to 191. Remote categories have an ID range of 192 to 221.

```
config vdom
    edit root
        config webfilter profile
            edit "WebFilter-1"
                set feature-set proxy
                config ftgd-wf
                    unset options
                    config filters
                        edit 12
                            set category 12
                            set action warning
                        next
                        ...
                        edit 23
                            set action warning
                        next
                        edit 140
                            set category 140
                        next
                        edit 192
                            set category 192
                            set action block
                        next
                    end
                end
            next
        end
    next
end
```

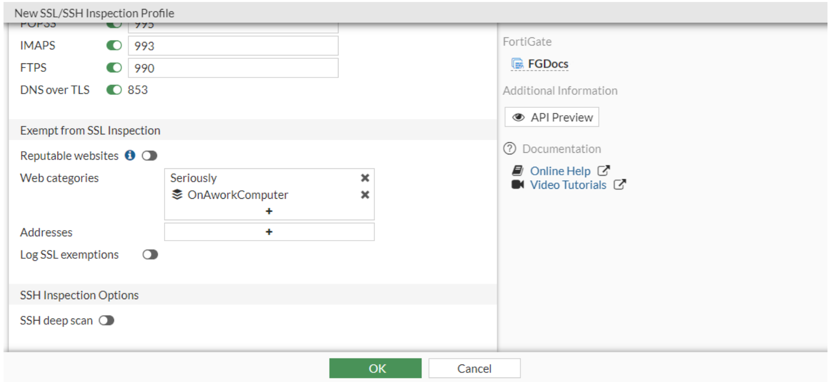
   When a filter is added for the local and remote categories (140 and 192 in this example), the default action is `monitor` with logging enabled.

SSL/SSH inspection profiles

To use local and remote categories in an SSL/SSH inspection profile to exempt them from SSL inspection in the GUI:

1. Go to *Security Profiles > SSL/SSH Inspection*.
2. Create a new profile or edit an existing one.
3. Ensure that *Inspection method* is *Full SSL Inspection*.

4. In the *Exempt from SSL Inspection* section, add the local and remote categories to the *Web categories* list .



5. Configure the remaining settings as required, then click *OK*.

To use local and remote categories in an SSL/SSH inspection profile to exempt them from SSL inspection in the CLI:

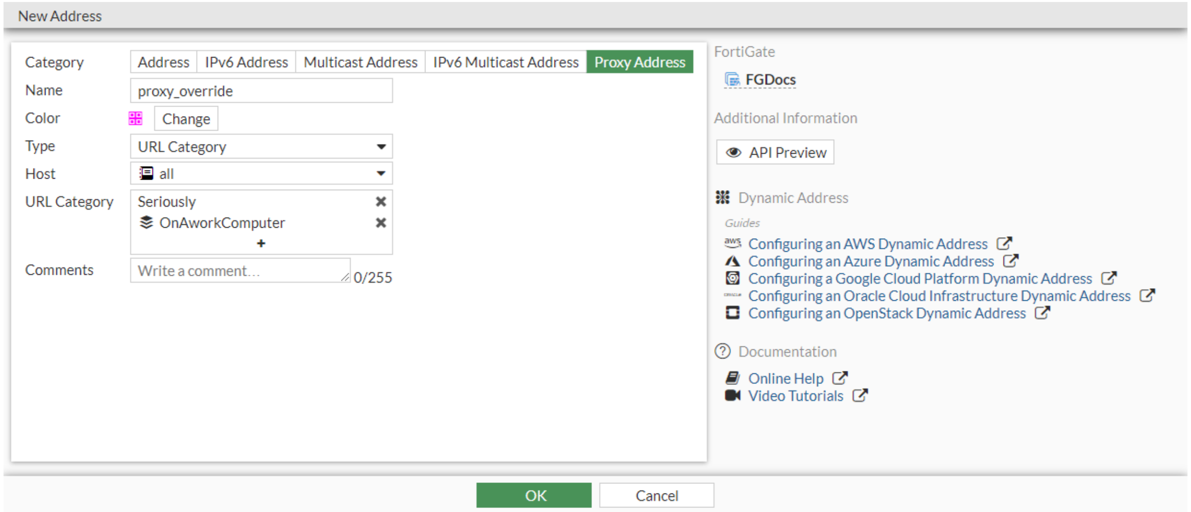
```
config vdom
    edit root
        config firewall ssl-ssh-profile
            edit "SSL_Inspection"
                config https
                    set ports 443
                    set status deep-inspection
                end
                ...
                config ssl-exempt
                    edit 1
                        set fortiguard-category 140
                    next
                    edit 2
                        set fortiguard-category 192
                    next
                end
            next
        end
    next
end
```

Proxy addresses

To use local and remote categories in a proxy address in the GUI:

1. Go to *Policy & Objects > Addresses* and click *Create New > Address*, or edit an existing proxy address.
2. Set *Category* to *Proxy Address*.
3. Set *Type* to *URL Category*.

4. In the *URL Category*, add the local and remote categories.



5. Configure the remaining settings as required, then click *OK*.

To use local and remote categories in a proxy address in the CLI:

```
config vdom
    edit root
        config firewall proxy-address
            edit "proxy_override"
                set type category
                set host "all"
                set category 140 192
                set color 23
            next
        end
    next
end
```

Web profile override

The following profile override methods are available:

- Administrative override
- Allow users to override blocked categories

Administrative override

Administrators can grant temporary access to sites that are otherwise blocked by a web filter profile. You can grant temporary access to a user, user group, or source IP address. You can set the time limit by selecting a date and time. The default is 15 minutes.

When the administrative web profile override is enabled, a blocked access page or replacement message does not appear, and authentication is not required.

Scope range

You can choose one of the following scope ranges:

- User: authentication for permission to override is based on whether or not the user is using a specific user account.
- User group: authentication for permission to override is based on whether or not the user account supplied as a credential is a member of the specified user group.
- Source IP: authentication for permission to override is based on the IP address of the computer that was used to authenticate. This would be used for computers that have multiple users. For example, if a user logs on to the computer, engages the override by using their credentials, and then logs off, anyone who logs on with an account on that computer would be using the alternate override web filter profile.

When you enter an IP address in the administrative override method, only individual IP addresses are allowed.

Differences between IP and identity-based scope

Using the IP scope does not require using an identity-based policy.

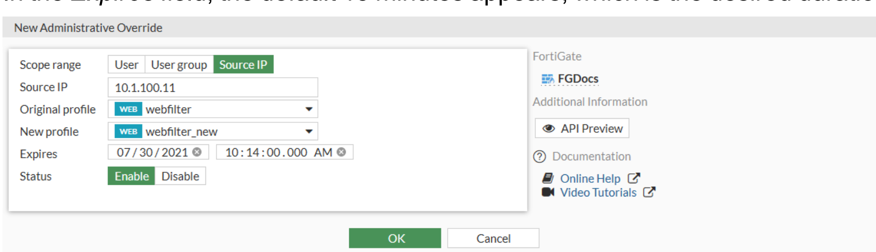
When using the administrative override method and IP scope, you might not see a warning message when you change from using the original web filter profile to using the alternate profile. There is no requirement for credentials from the user so, if allowed, the page will just appear in the browser.

Configuring a web profile administrative override

This example describes how to override the *webfilter* profile with the *webfilter_new* profile.

To configure web profile administrative override using the GUI:

1. Go to *Security Profiles > Web Profile Overrides* and click *Create New*.
2. Configure the administrative override:
   a. For *Scope Range*, click *Source IP*.
   b. In the *Source IP* field, enter the IP address for the client computer (*10.1.100.11* in this example).
   c. In the *Original profile* dropdown, select *webfilter*.
   d. In the *New profile* dropdown, select *webfilter_new*.
   In the *Expires* field, the default 15 minutes appears, which is the desired duration for this example.



3. Click *OK*.

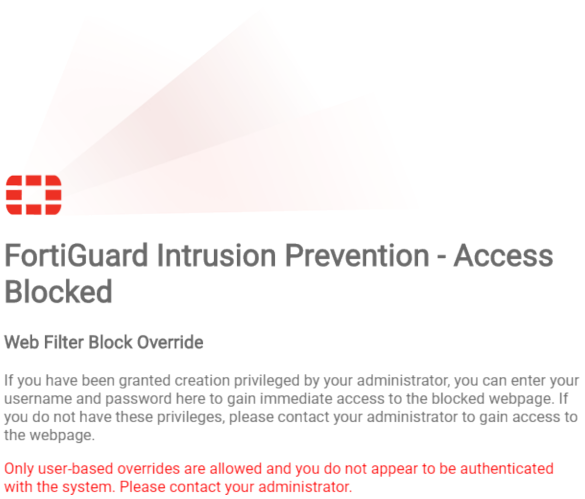
To configure web profile administrative override using the CLI:

```
config webfilter override
    edit 1
        set status enable
        set scope ip
        set old-profile "webfilter"
        set new-profile "webfilter_new"
        set expires 2021/07/30 10:14:00
        set initiator "admin"
        set ip 10.1.100.11
    next
end
```

Allow users to override blocked categories

For both override methods, the scope ranges (for specified users, user groups, or IP addresses) allow sites blocked by web filtering profiles to be overridden for a specified length of time.

But there is a difference between the override methods when the users or user group scope ranges are selected. In both cases, you would need to apply the user or user group as source in the firewall policy. With administrative override, if you do not apply the source in the firewall policy, the traffic will not match the override and will be blocked by the original profile. With the *Allow users to override blocked categories* setting, the traffic will also be blocked, but instead of displaying a blocking page, the following message appears:



When you choose the user group scope, once one user overrides, it will affect the other users in the group when they attempt to override. For example, user1 and user2 both belong to the local_user group. Once user1 successfully overrides, this will generate an override entry for the local_user group instead of one specific user. This means that if user2 logs in from another PC, they can override transparently.

Other features

Besides the scope, there are some other features in *Allow users to override blocked categories*.

Apply to user groups

Individual users can not be selected. You can select one or more of the user groups recognized by the FortiGate. They can be local to the system or from a third party authentication device, such as an AD server through FSSO.

Switch duration

Administrative override sets a specified time frame that is always used for that override. The available options are:

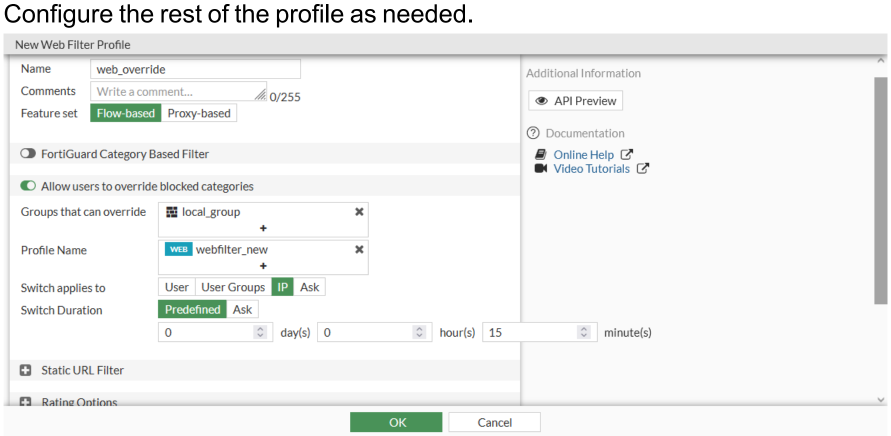
- *Predefined*: the value entered is the set duration (length of time in days, hours, or minutes) that the override will be in effect. If the duration variable is set to 15 minutes, the length of the override will always be 15 minutes. The option will be visible in the override message page, but the setting will be grayed out.
- *Ask*: the user has the option to set the override duration once it is engaged. The user can set the duration in terms of days, hours, or minutes.

Creating a web profile users override

This example describes how to allow users in the *local_group* to override the *webfilter_new* profile.

To allow users to override blocked categories using the GUI:

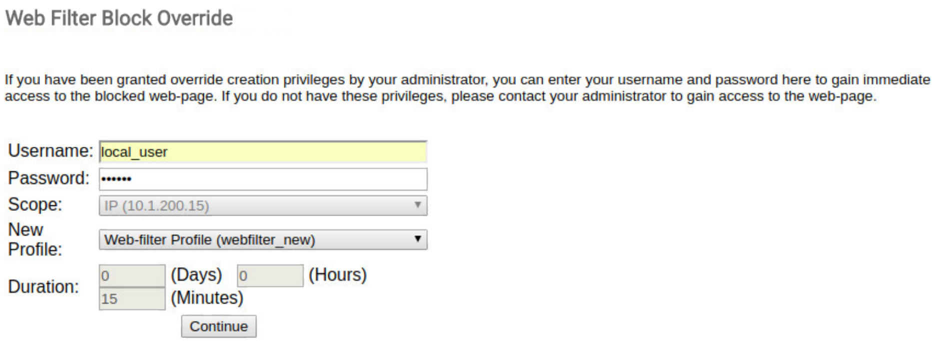
1. Go to *Security Profiles > Web Filter* and click *Create New*.
2. Enter a name for the profile.
3. Enable *Allow users to override blocked categories*.
4. Configure the web filter profile:
   a. Click the *Groups that can override* field, and select a group (*local_group* in this example).
   b. Click the *Profile Name* field, and select the *webfilter_new* profile.
   c. For the *Switch applies to* field, click *IP*.
   d. For the *Switch Duration* field, click *Predefined*. The default *15* minutes appears, which is the desired duration for this example.
   e. Configure the rest of the profile as needed.



5. Click *OK*.

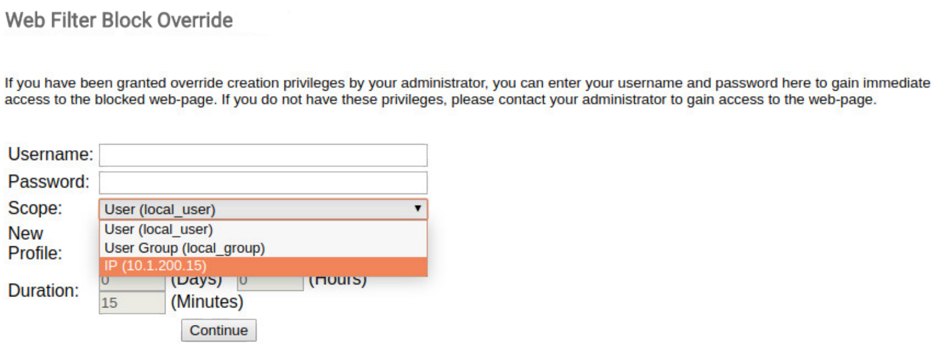
Using the ask feature

This option is only available in *Allow users to override blocked categories* is enabled. It configures the message page to have the user choose which scope they want to use. Normally on the message page, the scope options are grayed out and not editable. In the following example, the *Scope* is predefined with *IP*.



When the ask option is enabled (through the *Switch applies to* field in the GUI), the *Scope* dropdown is editable. Users can choose one of the following:

- User
- User group
- IP



User and *User Group* are only available when there is a user group in the firewall policy. You must specify a user group as a source in the firewall policy so the scope includes *User* and *User Group*; otherwise, only the IP option will be available.

VPN

Virtual Private Network (VPN) technology lets remote users connect to private computer networks to gain access to their resources in a secure way. For example, an employee traveling or working at home can use a VPN to securely access the office network through the Internet.

Instead of remotely logging into a private network using an unencrypted and unsecured Internet connection, using a VPN ensures that unauthorized parties cannot access the office network and cannot intercept information going between the employee and the office. Another common use of a VPN is to connect the private networks of multiple offices.

Fortinet offers VPN capabilities in the FortiGate Unified Threat Management (UTM) appliance and in the FortiClient Endpoint Security suite of applications. You can install a FortiGate unit on a private network and install FortiClient software on the user's computer. You can also use a FortiGate unit to connect to the private network instead of using FortiClient software.

The following sections provide information about VPN:

IPsec VPNs

The following sections provide instructions on configuring IPsec VPN connections in FortiOS 7.2.0.

General IPsec VPN configuration

The following sections provide instructions on general IPsec VPN configurations:

Network topologies

The topology of your network will determine how remote peers and clients connect to the VPN and how VPN traffic is routed.

Topology	Description
Site-to-Site	Standard one-to-one VPN between two FortiGates. See Site-to-site VPN on page 1358.
Hub and spoke/ADVPN	One central FortiGate (hub) has multiple VPNs to other remote FortiGates (spokes). In ADVPN, shortcuts can be created between spokes for direct communication. See ADVPN on page 1530.
OCVPN	Fortinet's cloud based solution for automating VPN setup between devices registered to the same account. See Overlay Controller VPN (OCVPN) on page 1499.
FortiClient dialup	Typically remote FortiClient dialup clients use dynamic IP addresses through NAT devices. The FortiGate acts as a dialup server allowing dialup VPN connections from multiple sources. See FortiClient as dialup client on page 1418.
FortiGate dialup	Similar to site-to-site except one end is a dialup server and the other end is a dialup client. This facilitates scenarios in which the remote dialup end has a dynamic address, or does not have a public IP, possibly because it is behind NAT. See FortiGate as dialup client on page 1412.
Aggregate VPN	Natively support aggregating multiple VPN tunnels to increase performance and provide redundancy over multiple links. See Packet distribution and redundancy for aggregate IPsec tunnels on page 1472.
Redundant VPN	Options for supporting redundant and partially redundant IPsec VPNs, using route-based approaches. See Redundant hub and spoke VPN on page 1493.
L2TP over IPsec	Configure VPN for Microsoft Windows dialup clients using the built in L2TP software. Users do not have to install any Fortinet software. See L2TP over IPsec on page 1436.
GRE over IPsec	Legacy support for routers requiring point-to-point GRE over IPsec for tunneling. See GRE over IPsec on page 1374.

Phase 1 configuration

Phase 1 configuration primarily defines the parameters used in IKE (Internet Key Exchange) negotiation between the ends of the IPsec tunnel. The local end is the FortiGate interface that initiates the IKE negotiations. The remote end is the remote gateway that responds and exchanges messages with the initiator. Hence, they are sometimes referred to as the initiator and responder. The purpose of phase 1 is to secure a tunnel with one bi-directional IKE SA (security association) for negotiating IKE phase 2 parameters.

The `auto-negotiate` and `negotiation-timeout` commands control how the IKE negotiation is processed when there is no traffic, and the length of time that the FortiGate waits for negotiations to occur.

IPsec tunnels can be configured in the GUI using the *VPN Creation Wizard*. Go to *VPN > IPsec Wizard*. The wizard includes several templates (site-to-site, hub and spoke, remote access), but a custom tunnel can be configured with the following settings.

The IPsec phase 1 interface type cannot be changed after it is configured. This is due to the tunnel ID parameter (`tun_id`), which is used to match routes to IPsec tunnels to forward traffic. If the IPsec phase 1 interface type needs to be changed, a new interface must be configured.

Name		Phase 1 definition name. The maximum length is 15 characters for an interface mode VPN and 35 characters for a policy-based VPN. For a policy-based VPN, the name normally reflects where the remote connection originates. For a route-based tunnel, the FortiGate also uses the name for the virtual IPsec interface that it creates automatically.
Network		
	IP Version	Protocol, either IPv4 or IPv6.
	Remote Gateway	Category of the remote connection: • *Static IP Address*: the remote peer has a static IP address. • *Dialup User*: one or more FortiClient or FortiGate dialup clients with dynamic IP addresses will connect to the FortiGate. • *Dynamic DNS*: a remote peer that has a domain name and subscribes to a dynamic DNS service will connect to the FortiGate.
	IP Address	The IP address of the remote peer. This option is only available when the *Remote Gateway* is *Static IP Address*.
	Dynamic DNS	The domain name of the remote peer. This option is only available when the *Remote Gateway* is *Dynamic DNS*.
	Interface	The interface through which remote peers or dialup clients connect to the FortiGate. This option is only available in NAT mode. By default, the local VPN gateway IP address is the IP address of the interface that was selected (*Primary IP* in the *Local Gateway* field).
	Local Gateway	IP address for the local end of the VPN tunnel (*Primary IP* is used by default): • *Secondary IP*: secondary address of the interface selected in the *Interface* field. • *Specify*: manually enter an address. Interface mode cannot be configured in a transparent mode VDOM.
	Mode Config	This option is only available when the *Remote Gateway* is *Dialup User*. Configure the client IP address range, subnet mask/prefix length, DNS server, and split tunnel capability to automate remote client addressing.
	NAT Traversal	This option is only available when the *Remote Gateway* is *Static IP Address* or *Dynamic DNS*.

	ESP (encapsulating security payload), the protocol for encrypting data in the VPN session, uses IP protocol 50 by default. However, it does not use any port numbers so when traversing a NAT device, the packets cannot be demultiplexed. Enabling NAT traversal encapsulates the ESP packet inside a UDP packet, thereby adding a unique source port to the packet. This allows the NAT device to map the packets to the correct session. • *Enable*: a NAT device exists between the local FortiGate and the VPN peer or client. Outbound encrypted packets are wrapped inside a UDP IP header that contains a port number. The local FortiGate and the VPN peer or client must have the same NAT traversal setting (both selected or both cleared) to connect reliably. When in doubt, enable NAT traversal. • *Disable*: disable the NAT traversal setting. • *Forced*: the FortiGate will use a port value of zero when constructing the NAT discovery hash for the peer. This causes the peer to think it is behind a NAT device, and it will use UDP encapsulation for IPsec, even if no NAT is present. This approach maintains interoperability with any IPsec implementation that supports the NAT-T RFC.
Keepalive Frequency	Keepalive frequency setting. This option is only available when *NAT Traversal* is set to *Enable* or *Forced*. The NAT device between the VPN peers may remove the session when the VPN connection remains idle for too long. The value represents an interval in seconds where the connection will be maintained with periodic keepalive packets. The keepalive interval must be smaller than the session lifetime value used by the NAT device. The keepalive packet is a 138-byte ISAKMP exchange.
Dead Peer Detection	Reestablishes VPN tunnels on idle connections and cleans up dead IKE peers if required. This feature minimizes the traffic required to check if a VPN peer is available or unavailable (dead). The available options are: • *Disable*: disable dead peer detection (DPD). • *On Idle*: triggers DPD when IPsec is idle. • *On Demand*: Passively sends DPD to reduce load on the firewall. Only triggers DPD when IPsec outbound packets are sent, but no reply is received from the peer. When there is no traffic and the last DPD-ACK has been received, IKE will not send DPDs periodically. Notifications are received whenever a tunnel goes up or down, or to keep the tunnel connection open when no traffic is being generated inside the tunnel. For example, in scenarios where a dialup client or dynamic DNS peer connects from an IP address that changes periodically, traffic may be suspended while the IP address changes. When *Dead Peer Detection* is selected, optionally specify a retry count and a retry interval using `dpd-retrycount` and `dpd-retryinterval`. See Dead peer detection on page 1336.
Forward Error Correction	Enable on both ends of the tunnel to correct errors in data transmission by sending redundant data across the VPN.

	Device creation	Advanced option. When enabled, a dynamic interface (network device) is created for each dialup tunnel.
	Aggregate member	Advanced option. When enabled, the tunnel can be used as an aggregate member candidate.
Authentication		
	Method	Either *Pre-shared Key* or *Signature*.
	Pre-shared Key	The pre-shared key that the FortiGate will use to authenticate itself to the remote peer or dialup client during phase 1 negotiations. The same key must be defined at the remote peer or client. See Pre-shared key.
	Certificate Name	The server certificate that the FortiGate will use to authenticate itself to the remote peer or dialup client during phase 1 negotiations. See Digital certificates.
	IKE Version	Either *1* or *2*. See Choosing IKE version 1 and 2 on page 1337.
	Mode	This option is only available when IKEv1 is selected. The two available options are: • *Aggressive*: the phase 1 parameters are exchanged in a single message with unencrypted authentication information. • *Main (ID protection)*: the phase 1 parameters are exchanged in multiple rounds with encrypted authentication information. When the remote VPN peer has a dynamic IP address and is authenticated by a pre-shared key, you must select *Aggressive* mode if there is more than one dialup phase 1 configuration for the interface IP address. When the remote VPN peer has a dynamic IP address and is authenticated by a certificate, you must select *Aggressive* mode if there is more than one phase 1 configuration for the interface IP address and these phase 1 configurations use different proposals.
	Peer Options	Options to authenticate VPN peers or clients depending on the *Remote Gateway* and *Authentication Method* settings.
	Any peer ID	Accepts the local ID of any remote VPN peer or client. The FortiGate does not check identifiers (local IDs). *Mode* can be set to *Aggressive* or *Main*. This option can be used with digital certificate authentication, but for higher security, use *Peer certificate*.
	Specific peer ID	This option is only available when *Aggressive Mode* is enabled. Enter the identifier that is used to authenticate the remote peer. The identifier must match the local ID configured by the remote peer's administrator. If the remote peer is a FortiGate, the identifier is specified in the *Local ID* field of the *Phase 1 Proposal* settings. If the remote peer is a FortiClient user, the identifier is specified in the *Local ID* field. In circumstances where multiple remote dialup VPN tunnels exist, each tunnel must have a peer ID set.

	Peer certificate	Define the CA certificate used to authenticate the remote peer when the authentication mode is *Signature*. If the FortiGate will act as a VPN client, and you are using security certificates for authentication, set the *Local ID* to the distinguished name (DN) of the local server certificate that the FortiGate unit will use for authentication purposes.
	Peer ID from dialup group	Authenticate multiple FortiGate or FortiClient dialup clients that use unique identifiers and unique pre-shared keys (or unique pre-shared keys only) through the same VPN tunnel. You must create a dialup user group for authentication purposes. Select the group from the list next to the *Peer ID from dialup group* option. You must set *Mode* to *Aggressive* when the dialup clients use unique identifiers and unique pre-shared keys. If the dialup clients use unique pre-shared keys only, you can set *Mode* to *Main* if there is only one dialup Phase 1 configuration for this interface IP address.
Phase 1 Proposal		The encryption and authentication algorithms used to generate keys for the IKE SA. There must be a minimum of one combination. The remote peer or client must be configured to use at least one of the proposals that you define.
	Encryption	The following symmetric-key encryption algorithms are available: • *DES*: Digital Encryption Standard, a 64-bit block algorithm that uses a 56-bit key. • *3DES*: triple-DES; plain text is encrypted three times by three keys. • *AES128*: Advanced Encryption Standard, a 128-bit block algorithm that uses a 128-bit key. • *AES128GCM*: AES in Galois/Counter Mode, a 128-bit block algorithm that uses a 128-bit key. Only available for IKEv2. • *AES192*: a 128-bit block algorithm that uses a 192-bit key. • *AES256*: a 128-bit block algorithm that uses a 256-bit key. • *AES256GCM*: AES in Galois/Counter Mode, a 128-bit block algorithm that uses a 256-bit key. Only available for IKEv2. • *CHACHA20POLY1305*: a 128-bit block algorithm that uses a 128-bit key and a symmetric cipher. Only available for IKEv2. See also HMAC settings.
	Authentication	The following message digests that check the message authenticity during an encrypted session are available: • *MD5*: message digest 5. • *SHA1*: secure hash algorithm 1; a 160-bit message digest. • *SHA256*: a 256-bit message digest. • *SHA384*: a 384-bit message digest. • *SHA512*: a 512-bit message digest.

		In IKEv2, encryption algorithms include authentication, but a PRF (pseudo random function) is still required (*PRFSHA1*, *PRFSHA256*, *PRFSHA384*, *PRFSHA512*). See also HMAC settings.
	Diffie-Hellman Groups	Asymmetric key algorithms used for public key cryptography. Select one or more from groups 1, 2, 5, and 14 through 32. At least one of the *Diffie-Hellman Groups* (DH) settings on the remote peer or client must match one the selections on the FortiGate. Failure to match one or more DH groups will result in failed negotiations.
	Key Lifetime	The time (in seconds) that must pass before the IKE encryption key expires. When the key expires, a new key is generated without interrupting service. The keylife can be from 120 to 172 800 seconds.
	Local ID	Optional setting. This value must match the peer ID value given for the remote VPN peer's *Peer Options*. • If the FortiGate will act as a VPN client and you are using peer IDs for authentication purposes, enter the identifier that the FortiGate will supply to the VPN server during the phase 1 exchange. • If the FortiGate will act as a VPN client and you are using security certificates for authentication, select the distinguished name (DN) of the local server certificate that the FortiGate will use for authentication purposes.
XAUTH		This option supports the authentication of dialup clients. It is only available for IKE version 1. • *Disable*: do not use XAuth. • *Client*: available only if the *Remote Gateway* is set to *Static IP Address* or *Dynamic DNS*. If the FortiGate is a dialup client, enter the user name and password for the FortiGate to authenticate itself to the remote XAuth server. • *PAP Server*, *CHAP Server*, *Auto Server*: available only if *Remote Gateway* is set to *Dialup User*. Dialup clients authenticate as members of a dialup user group. A user group must be created first for the dialup clients that need access to the network behind the FortiGate. The FortiGate must be configured to forward authentication requests to an external RADIUS or LDAP authentication server. Select the server type based on the encryption method used between the FortiGate, the XAuth client, and the external authentication server. Then select the user group (*Inherit from policy* or *Choose*). See Using XAuth authentication on page 1341.
	Username	User name used for authentication.
	Password	Password used for authentication.

Additional CLI configurations

The following phase 1 settings can be configured in the CLI:

VXLAN over IPsec	Packets with a VXLAN header are encapsulated within IPsec tunnel mode.
	
	To configure VXLAN over IPsec:
	
	`config vpn ipsec phase1-interface/phase1`
	`    edit ipsec`
	`        set interface <name>`
	`        set encapsulation vxlan/gre`
	`        set encapsulation-address ike/ipv4/ipv6`
	`        set encap-local-gw4 xxx.xxx.xxx.xxx`
	`        set encap-remote-gw xxx.xxx.xxx.xxx`
	`    next`
	`end`

IPsec tunnel idle timer	Define an idle timer for IPsec tunnels. When no traffic has passed through the tunnel for the configured `idle-timeout` value, the IPsec tunnel will be flushed.
	
	To configure IPsec tunnel idle timeout:
	
	`config vpn ipsec phase1-interface`
	`    edit p1`
	`        set idle-timeout [enable | disable]`
	`        set idle-timeoutinterval <integer>` *`IPsec tunnel idle timeout in minutes (10 - 43200).`*
	`    next`
	`end`

Monitor tunnel for failover	Monitor a site-to-site tunnel to guarantee operational continuity if the primary tunnel fails. Configure the secondary phase 1 interface to monitor the primary interface.
	
	To configure the monitor:
	
	`config vpn ipsec phase1-interface`
	`    edit <secondary phase1-interface>`
	`        set monitor <primary phase1-interface>`
	`    next`
	`end`

Passive mode	Passive mode turns one side of the tunnel to be a responder only. It does not initiate VPN tunnels either by auto-negotiation, rekey, or traffic initiated behind the FortiGate.
	
	To configure passive mode:
	
	`config vpn ipsec phase1-interface`
	`    edit <example>`
	`        set rekey {enable | disable}`
	`        set passive-mode {enable | disable}`
	`        set passive-tunnel-interface {enable | disable}`

<table>
<tr><td></td><td><pre>   next
end</pre></td></tr>
<tr><td>Network ID</td><td>
The network ID is a Fortinet-proprietary attribute that is used to select the correct phase 1 between IPsec peers, so that multiple IKEv2 tunnels can be established between the same local/remote gateway pairs.
In a dial-up VPN, <code>network-id</code> is in the first initiator message of an IKEv2 phase 1 negotiation. The responder (Hub) uses the <code>network-id</code> to match a phase 1 configuration with a matching <code>network-id</code>. The Hub can then differentiate multiple dial-up phase 1s that are bound to the same underlay interface and IP address. Without a <code>network-id</code>, the Hub cannot have multiple phase 1 dialup tunnels on the same interface.
In static phase 1 configurations, <code>network-id</code> is used with the pair of gateway IPs to negotiate the correct tunnel with a matching <code>network-id</code>. This allows IPsec peers to use the same pair of underlay IPs to establish multiple IPsec tunnels. Without it, only a single tunnel can be established over the same pair of underlay IPs.
To configure the network ID:
<pre>config vpn ipsec phase1-interface
    edit <example>
        set network-id <integer>
    next
end</pre>
</td></tr>
</table>

Dead peer detection

By default, dead peer detection (DPD) sends probe messages every five seconds. If you are experiencing high network traffic, you can experiment with increasing the ping interval. However, longer intervals will require more traffic to detect dead peers, which will result in more traffic.

In a dynamic (dialup) connection, the *On Idle* option encourages dialup server configurations to more proactively delete tunnels if the peer is unavailable.

In the GUI, the dead peer detection option can be configured when defining phase 1 options. The following CLI commands support additional options for specifying a retry count and a retry interval.

For example, enter the following to configure DPD on the existing IPsec phase 1 configuration to use 15-second intervals and to wait for three missed attempts before declaring the peer dead and taking action.

To configure DPD:

```
config vpn ipsec phase1-interface
    edit <value>
        set dpd [disable | on-idle | on-demand]
        set dpd-retryinveral 15
        set dpd-retrycount 3
```

```
    next
end
```

DPD scalability

On a dialup server, if many VPN connections are idle, the increased DPD exchange could negatively impact the performance/load of the daemon. The `on-demand` option in the CLI triggers DPD when IPsec traffic is sent, but no reply is received from the peer.

When there is no traffic and the last DPD-ACK had been received, IKE will not send DPDs periodically. IKE will only send out DPDs if there are outgoing packets to send, but no inbound packets have since been received.

HMAC settings

The FortiGate uses the HMAC based on the authentication proposal that is chosen in phase 1 or phase 2 of the IPsec configuration. Each proposal consists of the encryption-hash pair (such as `3des-sha256`). The FortiGate matches the most secure proposal to negotiate with the peer.

To view the chosen proposal and the HMAC hash used:

```
# diagnose vpn ike gateway list

vd: root/0
name: MPLS
version: 1
interface: port1 3
addr: 192.168.2.5:500 -> 10.10.10.1:500
tun_id: 10.10.10.1
virtual-interface-addr: 172.31.0.2 -> 172.31.0.1
created: 1015820s ago
IKE SA: created 1/13 established 1/13 time 10/1626/21010 ms
IPsec SA: created 1/24 established 1/24 time 0/11/30 ms

  id/spi: 124 43b087dae99f7733/6a8473e58cd8990a
  direction: responder
  status: established 68693-68693s ago = 10ms
  proposal: 3des-sha256
  key: e0fa6ab8dc509b33-aa2cc549999b1823-c3cb9c337432646e
  lifetime/rekey: 86400/17436
  DPD sent/recv: 000001e1/00000000
```

Choosing IKE version 1 and 2

If you create a route-based VPN, you have the option of selecting IKE version 2. Otherwise, IKE version 1 is used.

IKEv2, defined in RFC 4306, simplifies the negotiation process that creates the security association (SA).

If you select IKEv2:

- There is no choice in phase 1 of aggressive or main mode.
- Extended authentication (XAUTH) is not available.
- You can utilize EAP and MOBIKE.

Repeated authentication in IKEv2

This feature provides the option to control whether a device requires its peer to re-authenticate or whether re-key is sufficient. It does not influence the re-authentication or re-key behavior of the device itself, which is controlled by the peer (the default being to re-key). This solution is in response to RFC 4478. As described by the IETF, "the purpose of this is to limit the time that security associations (SAs) can be used by a third party who has gained control of the IPsec peer".

To configure IKE SA re-authentication:

```
config vpn ipsec phase1-interface
    edit p1
        set reauth [enable | disable]
    next
end
```

IKEv2 quick crash detection

There is support for IKEv2 quick crash detection (QCD) as described in RFC 6290.

RFC 6290 describes a method in which an IKE peer can quickly detect that the gateway peer it has and established an IKE session with has rebooted, crashed, or otherwise lost IKE state. When the gateway receives IKE messages or ESP packets with unknown IKE or IPsec SPIs, the IKEv2 protocol allows the gateway to send the peer an unprotected IKE message containing INVALID_IKE_SPI or INVALID_SPI notification payloads.

RFC 6290 introduces the concept of a QCD token, which is generated from the IKE SPIs and a private QCD secret, and exchanged between peers during the protected IKE AUTH exchange.

To configure QCD:

```
config system settings
    set ike-quick-crash-detect [enable | disable]
end
```

IKEv1 quick crash detection

Based on the IKEv2 QCD feature previously described, IKEv1 QCD is implemented using a new IKE vendor ID (Fortinet Quick Crash Detection) so both endpoints must be FortiGates. The QCD token is sent in the phase 1 exchange and must be encrypted, so this is only implemented for IKEv1 in main mode (aggressive mode is not supported as there is no available AUTH message to include the token). Otherwise, the feature works the same as in IKEv2 (RFC 6290).

IKEv1 fragmentation

UDP fragmentation can cause issues in IPsec when either the ISP or perimeter firewall(s) cannot pass or fragment the oversized UDP packets that occur when using a very large public security key (PSK). The result is that IPsec tunnels do not come up. The solution is IKE fragmentation.

For most configurations, enabling IKE fragmentation allows connections to automatically establish when they otherwise might have failed due to intermediate nodes dropping IKE messages containing large certificates, which typically push the packet size over 1500 bytes.

FortiOS will fragment a packet on sending if only all the following are true:

- Phase 1 contains `set fragmentation enable`.
- The packet is larger than the minimum MTU (576 for IPv4, 1280 for IPv6).
- The packet is being re-transmitted.

By default, IKE fragmentation is enabled.

To configure IKEv1 fragmentation:

```
config vpn ipsec phase1-interface
    edit 1
        set fragmentation [enable | disable]
    next
end
```

IKEv2 fragmentation

RFC 7383 requires each fragment to be individually encrypted and authenticated. With IKEv2, a copy of the unencrypted payloads around for each outgoing packet would need to be kept in case the original single packet was never answered and would retry with fragments. With the following implementation, if the IKE payloads are greater than a configured threshold, the IKE packets are preemptively fragmented and encrypted.

To configure IKEv2 fragmentation:

```
config vpn ipsec phase1-interface
    edit ike
        set ike-version 2
        set fragmentation [enable|disable]
        set fragmentation-mtu <500-16000>
    next
end
```

IPsec global IKE embryonic limit

When trying to establish thousands of tunnels simultaneously, a situation can arise where new negotiations starve other SAs from progressing to an established state in IKEv2. The IKE daemon can prioritize established SAs, offload groups 20 and 21 to CP9, and optimize the default embryonic limits for mid- and high-end platforms. The IKE embryonic limit can be configured in the CLI.

```
config system ike
    set embryonic-limit <integer>
end
```

`embryonic-limit <integer>`	Set the maximum number of IPsec tunnels to negotiate simultaneously (50 - 20000, default = 1000).

To configure an IKE embryonic limit of 50:

```
config system ike
    set embryonic-limit 50
end
```

Pre-shared key vs digital certificates

A FortiGate can authenticate itself to remote peers or dialup clients using either a pre-shared key or a digital certificate.

Pre-shared key

Using a pre-shared key is less secure than using certificates, especially if it is used alone, without requiring peer IDs or extended authentication (XAuth). There also needs to be a secure way to distribute the pre-shared key to the peers.

If you use pre-shared key authentication alone, all remote peers and dialup clients must be configured with the same pre-shared key. Optionally, you can configure remote peers and dialup clients with unique pre-shared keys. On the FortiGate, these are configured in user accounts, not in the phase 1 settings.

The pre-shared key must contain at least six printable characters and should be known by network administrators. For optimum protection against currently known attacks, the key must consist of a minimum of 16 randomly chosen alphanumeric characters. The limit is 128 characters.

If you authenticate the FortiGate using a pre-shared key, you can require remote peers or dialup clients to authenticate using peer IDs, but not client certificates.

To authenticate the FortiGate using a pre-shared key:

1. Go to *VPN > IPsec Tunnels* and create a new tunnel, or edit an existing one.
2. Configure or edit the *Network* section as needed.
3. Configure or edit the *Authentication* settings as follows:

Method	*Pre-shared Key*
Pre-shared Key	<string>
IKE Version	*1* or *2*
Mode	*Aggressive* or *Main*
Peer Options	Select an *Accept Type* and the corresponding peer. Options vary based on the *Remote Gateway* and *Authentication Method* settings in the *Network* section. *Peer Options* are only available in *Aggressive* mode.

4. For the *Phase 1 Proposal* section, keep the default settings unless changes are needed to meet your requirements.
5. Optionally, for authentication parameters for a dialup user group, define *XAUTH* parameters.
6. Click *OK*.

Digital certificates

To authenticate the FortiGate using digital certificates, you must have the required certificates installed on the remote peer and on the FortiGate. The signed server certificate on one peer is validated by the presence of the root certificate installed on the other peer. If you use certificates to authenticate the FortiGate, you can also require the remote peers or dialup clients to authenticate using certificates. See Site-to-site VPN with digital certificate on page 1363 for a detailed example.

To authenticate the FortiGate using a digital certificate:

1. Go to *VPN > IPsec Tunnels* and create a new tunnel, or edit an existing one.
2. Configure or edit the *Network* section as needed.

3. Configure or edit the *Authentication* settings as follows:

Method	*Signature*
Certificate Name	Select the certificate used to identify this FortiGate. If there are no imported certificates, use *Fortinet_Factory*.
IKE Version	*1* or *2*
Mode	*Aggressive* is recommended.
Peer Options	For *Accept Type*, select *Peer certificate* and select the peer and the CA certificate used to authenticate the peer. If the other end is using the Fortinet_ Factory certificate, then use the *Fortinet_CA* certificate here.

4. For the *Phase 1 Proposal* section, keep the default settings unless changes are needed to meet your requirements.
5. Optionally, for authentication parameters for a dialup user group, define *XAUTH* parameters.
6. Click *OK*.

Using XAuth authentication

Extended authentication (XAuth) increases security by requiring remote dialup client users to authenticate in a separate exchange at the end of phase 1. XAuth draws on existing FortiGate user group definitions and uses established authentication mechanisms such as PAP, CHAP, RADIUS, and LDAP to authenticate dialup clients. You can configure a FortiGate to function either as an XAuth server or client. If the server or client is attempting a connection using XAuth and the other end is not using XAuth, the failed connection attempts that are logged will not specify XAuth as the reason.

XAuth server

A FortiGate can act as an XAuth server for dialup clients. When the phase 1 negotiation completes, the FortiGate challenges the user for a user name and password. It then forwards the user's credentials to an external RADIUS or LDAP server for verification.

If the user records on the RADIUS server have suitably configured Framed-IP-Address fields, you can assign client virtual IP addresses by XAuth instead of from a DHCP address range.

The authentication protocol you use for XAuth depends on the capabilities of the authentication server and the XAuth client:

- Select *PAP Server* whenever possible.
- You must select *PAP Server* for all implementations of LDAP and some implementations of Microsoft RADIUS.
- Select *Auto Server* when the authentication server supports *CHAP Server* but the XAuth client does not. The FortiGate will use PAP to communicate with the XAuth client and CHAP to communicate with the authentication server. You can also use *Auto Server* to allow multiple source interfaces to be defined in an IPsec/IKE policy.

Before you begin, create user accounts and user groups to identify the dialup clients that need to access the network behind the FortiGate dialup server. If password protection will be provided through an external RADIUS or LDAP server, you must configure the FortiGate dialup server to forward authentication requests to the authentication server.

To configure XAuth to authenticate a dialup user group:

1. On the FortiGate dialup server, go to *VPN* > *IPsec Tunnels* and create a new tunnel, or edit an existing one.
2. Configure or edit the *Network*, *Authentication*, and *Phase 1 Proposal* sections as needed.

3. In the *XAUTH* section, select the encryption method *Type* to use between the XAuth client, the FortiGate, and the authentication server.
4. For *User Group*:
   a. Click *Inherit from policy* for multiple user groups defined in the IPsec/IKE policy, or
   b. Click *Choose* and in the dropdown, select the user group that needs to access the private network behind the FortiGate.

   Only one user group may be defined for *Auto Server*.

5. Click *OK*.
6. Create as many policies as needed, specifying the source user(s) and destination address.

XAuth client

If the FortiGate acts as a dialup client, the remote peer, acting as an XAuth server, might require a username and password. You can configure the FortiGate as an XAuth client with its own username and password, which it provides when challenged.

To configure the FortiGate dialup client as an XAuth client:

1. On the FortiGate dialup client, go to *VPN > IPsec Tunnels* and create a new tunnel, or edit an existing one.
2. Configure or edit the *Network*, *Authentication*, and *Phase 1 Proposal* sections as needed.
3. In the *XAUTH* section, for *Type*, select *Client*.
4. For *Username*, enter the FortiGate PAP, CHAP, RADIUS, or LDAP user name that the FortiGate XAuth server will compare to its records when the FortiGate XAuth client attempts to connect.
5. Enter the *Password* for the user name.
6. Click *OK*.

Dynamic IPsec route control

You can add a route to a peer destination selector by using the `add-route` option, which is available for all dynamic IPsec phases 1 and 2, for both policy-based and route-based IPsec VPNs.

The `add-route` option adds a route to the FortiGate routing information base when the dynamic tunnel is negotiated. You can use the distance and priority options to set the distance and priority of this route. If this results in a route with the lowest distance, it is added to the FortiGate forwarding information base.

You can also enable `add-route` in any policy-based or route-based phase 2 configuration that is associated with a dynamic (dialup) phase 1. In phase 2, `add-route` can be enabled, disabled, or set to use the same route as phase 1.

The `add-route` option is enabled by default.

To configure add-route in phase 1:

```
config vpn ipsec
    edit <name>
        set type dynamic
        set add-route {enable | disable}
```

```
    next
end
```

To configure add-route in phase 2:

```
config vpn ipsec {phase2 | phase2-interface}
    edit <name>
        set add-route {phase1 | enable | disable}
    next
end
```

Blocking IPsec SA negotiation

For interface-based IPsec, IPsec SA negotiation blocking can only be removed if the peer offers a wildcard selector. If a wildcard selector is offered, then the wildcard route will be added to the routing table with the distance/priority value configured in phase 1. If that is the route with the lowest distance, it will be installed into the forwarding information base.

In this scenario, it is important to ensure that the distance value configured for phase 1 is set appropriately.

Phase 2 configuration

After phase 1 negotiations end successfully, phase 2 begins. In Phase 2, the VPN peer or client and the FortiGate exchange keys again to establish a secure communication channel. The phase 2 proposal parameters select the encryption and authentication algorithms needed to generate keys for protecting the implementation details of security associations (SAs). The keys are generated automatically using a Diffie-Hellman algorithm.

The basic phase 2 settings associate IPsec phase 2 parameters with the phase 1 configuration that specifies the remote end point of the VPN tunnel. In most cases, you need to configure only basic Phase 2 settings.

Some settings can be configured in the CLI. The following options are available in the *VPN Creation Wizard* after the tunnel is created:

New Phase 2		
	Name	Phase 2 definition name.
	Local Address	A value of `0.0.0.0/0` means all IP addresses behind the local VPN peer. Add a specific address or range to allow traffic from and to only this local address. See Quick mode selectors on page 1345.
	Remote Address	Enter the destination IP address that corresponds to the recipients or network behind the remote VPN peer. A value of `0.0.0.0/0` means all IP addresses behind the remote VPN peer. See Quick mode selectors on page 1345.
Advanced		Select the encryption and authentication algorithms that will be proposed to the remote VPN peer. To establish a VPN connection, at least one of the proposals specified must match the configuration on the remote peer.
	Encryption	The following symmetric-key encryption algorithms are available: • *NULL*: do not use an encryption algorithm. • *DES*: Digital Encryption Standard, a 64-bit block algorithm that uses a

	56-bit key. • *3DES*: triple-DES; plain text is encrypted three times by three keys. • *AES128*: Advanced Encryption Standard, a 128-bit block algorithm that uses a 128-bit key. • *AES128GCM*: AES in Galois/Counter Mode, a 128-bit block algorithm that uses a 128-bit key. Only available for IKEv2. • *AES192*: a 128-bit block algorithm that uses a 192-bit key. • *AES256*: a 128-bit block algorithm that uses a 256-bit key. • *AES256GCM*: AES in Galois/Counter Mode, a 128-bit block algorithm that uses a 256-bit key. Only available for IKEv2. • *CHACHA20POLY1305*: a 128-bit block algorithm that uses a 128-bit key and a symmetric cipher. Only available for IKEv2. See ChaCha20 and Poly1305 AEAD cipher on page 1347, AES-GCM for IKEv2 phase 1 on page 1347, and HMAC settings.
Authentication	The following message digests that check the message authenticity during an encrypted session are available: • NULL: do not use a message digest. • *MD5*: message digest 5. • *SHA1*: secure hash algorithm 1; a 160-bit message digest. • *SHA256*: a 256-bit message digest. • *SHA384*: a 384-bit message digest. • *SHA512*: a 512-bit message digest. See also HMAC settings.
Enable Replay Detection	Replay attacks occur when an unauthorized party intercepts a series of IPsec packets and replays them back into the tunnel. Replay detection allows the FortiGate to check all IPsec packets to see if they have been received before. If any encrypted packets arrive out of order, the FortiGate discards them. Note that 64-bit extended sequence numbers (as described in RFC 4303, RFC 4304 as an addition to IKEv1, and RFC 5996 for IKEv2) are supported for IPsec when replay detection is enabled.
Enable Perfect Forward Secrecy (PFS)	Perfect forward secrecy (PFS) improves security by forcing a new Diffie-Hellman exchange whenever keylife expires.
Diffie-Hellman Group	Asymmetric key algorithms used for public key cryptography. Select one or more from groups 1, 2, 5, and 14 through 32. At least one of the *Diffie-Hellman Groups* (DH) settings on the remote peer or client must match one the selections on the FortiGate. Failure to match one or more DH groups will result in failed negotiations.
Local Port	Enter the port number that the local VPN peer uses to transport traffic related to the specified service (protocol number). The range is from 0 to 65535. To specify all ports, select *All*, or enter `0`.

Remote Port	Enter the port number that the remote VPN peer uses to transport traffic related to the specified service (protocol number). To specify all ports, select *All*, or enter `0`.
Protocol	Enter the IP protocol number of the service. To specify all services, select *All*, or enter `0`.
Auto-negotiate	Select this option for the tunnel to be automatically renegotiated when the it expires. See Auto-negotiate on page 1346.
Autokey Keep Alive	Select this option for the tunnel to remain active when no data is being processed.
Key Lifetime	Select the method for determining when the phase 2 key expires: • *Seconds* • *Kilobytes* • *Both* Enter a corresponding value for *Seconds* and/or *Kilobytes* in the text boxes. If *Both* is selected, the key expires when either the time has passed or the number of kilobytes have been processed.

Quick mode selectors

Quick mode selectors determine which IP addresses can perform IKE negotiations to establish a tunnel. By only allowing authorized IP addresses access to the VPN tunnel, the network is more secure.

The default settings are as broad as possible: any IP address or configured address object using any protocol on any port.

While the dropdown menus for specifying an address also show address groups, the use of address groups may not be supported on a remote endpoint device that is not a FortiGate.

When configuring a quick mode selector for *Local Address* and *Remote Address*, valid options include IPv4 and IPv6 single addresses, subnets, or ranges.

There are some configurations that require specific selectors:

- The VPN peer is a third-party device that uses specific phase2 selectors.
- The FortiGate connects as a dialup client to another FortiGate, in which case (usually) you must specify a local IP address, IP address range, or subnet. However, this is not required if you are using dynamic routing and `mode-cfg`.

With FortiOS VPNs, your network has multiple layers of security, with quick mode selectors being an important line of defense:

- Routes guide traffic from one IP address to another.
- Phase 1 and phase 2 connection settings ensure there is a valid remote end point for the VPN tunnel that agrees on the encryption and parameters.
- Quick mode selectors allow IKE negotiations only for allowed peers.
- Security policies control which IP addresses can connect to the VPN.
- Security policies also control what protocols are allowed over the VPN along with any bandwidth limiting.

If you are editing an existing phase 2 configuration, the local address and remote address fields are unavailable if the tunnel has been configured to use firewall addresses as selectors. This option exists only in the CLI.

Using the add-route option

Consider using the `add-route` option to add a route to a peer destination selector in phase 2 to automatically match the settings in phase 1.

To configure add-route:

```
config vpn ipsec {phase2 | phase2-interface}
    edit <name>
        set add-route {phase1 | enable | disable}
    next
end
```

Auto-negotiate

By default, the phase 2 security association (SA) is not negotiated until a peer attempts to send data. The triggering packet and some subsequent packets are dropped until the SA is established. Applications normally resend this data, so there is no loss, but there might be a noticeable delay in response to the user.

If the tunnel goes down, the auto-negotiate feature (when enabled) attempts to re-establish the tunnel. Auto-negotiate initiates the phase 2 SA negotiation automatically, repeating every five seconds until the SA is established.

Automatically establishing the SA can be important for a dialup peer. It ensures that the VPN tunnel is available for peers at the server end to initiate traffic to the dialup peer. Otherwise, the VPN tunnel does not exist until the dialup peer initiates traffic.

To configure auto-negotiate:

```
config vpn ipsec phase2
    edit <phase2_name>
        set auto-negotiate enable
    next
end
```

Installing dynamic selectors via auto-negotiate

The IPsec SA connect message generated is used to install dynamic selectors. These selectors can be installed via the auto-negotiate mechanism. When phase 2 has `auto-negotiate` enabled, and phase 1 has `mesh-selector-type` set to `subnet`, a new dynamic selector will be installed for each combination of source and destination subnets. Each dynamic selector will inherit the auto-negotiate option from the template selector and begin SA negotiation. Phase 2 selector sources from dialup clients will all establish SAs without traffic being initiated from the client subnets to the hub.

DHCP

The `dhcp-ipsec` option lets the FortiGate assign VIP addresses to FortiClient dialup clients through a DHCP server or relay. This option is only available if the remote gateway in the phase 1 configuration is set to dialup user, and it only works in policy-based VPNs.

With `dhcp-ipsec`, the FortiGate dialup server acts as a proxy for FortiClient dialup clients that have VIP addresses on the subnet of the private network behind the FortiGate. In this case, the FortiGate dialup server acts as a proxy on the local private network for the FortiClient dialup client. A host on the network behind the dialup server issues an ARP request, corresponding to the device MAC address of the FortiClient host (when a remote server sends an ARP to the local FortiClient dialup client). The FortiGate then answers the ARP request on behalf of the FortiClient host, and then forwards the associated traffic to the FortiClient host through the tunnel.

Acting as a proxy prevents the VIP address assigned to the FortiClient dialup client from causing possible ARP broadcast problems—the normal and VIP addresses can confuse some network switches when two addresses have the same MAC address.

ChaCha20 and Poly1305 AEAD cipher

In IKEv2 to support RFC 7634, the ChaCha20 and Poly1305 crypto algorithms can be used together as a combined mode AEAD cipher (like AES-GCM) in the `crypto_ftnt` cipher in `cipher_chacha20poly1305.c`:

```
config vpn ipsec phase2-interface
    edit <name>
        set phase1name <name>
        set proposal chacha20poly1305
    next
end
```

AES-GCM for IKEv2 phase 1

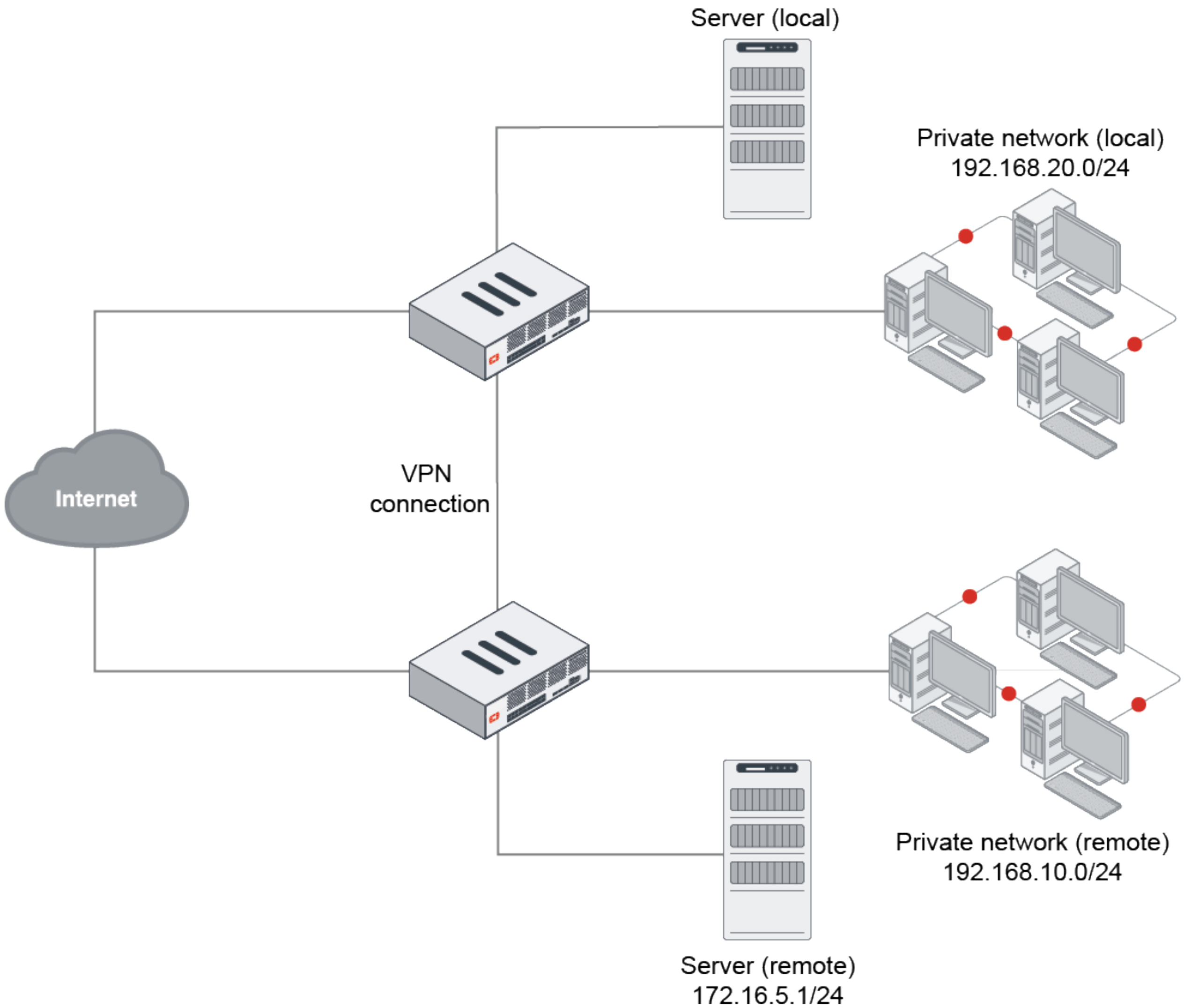
In IKEv2 to support RFC 5282, the AEAD algorithm AES-GCM supports 128- and 256-bit variants:

```
config vpn ipsec phase2-interface
    edit <name>
        set phase1name <name>
        set proposal [aes128gcm | aes256gcm]
    next
end
```

VPN security policies

This section explains how to specify the source and destination IP addresses of traffic transmitted through an IPsec VPN, and how to define appropriate security policies.

Topology



Defining policy addresses

In a gateway-to-gateway, hub-and-spoke, dynamic DNS, redundant tunnel, or transparent configuration, you need to define a policy address for the private IP address of the network behind the remote VPN peer (for example, `192.168.10.0/255.255.255.0` or `192.168.10.0/24`).

In a peer-to-peer configuration, you need to define a policy address for the private IP address of a server or host behind the remote VPN peer (for example, `172.16.5.1/255.255.255.255`, `172.16.5.1/32`, or `172.16.5.1`).

For a FortiGate dialup server in a dialup-client or internet-browsing configuration, the source IP should reflect the IP addresses of the dialup clients:

Defining security policies

Policy-based and route-based VPNs require different security policies.

- A policy-based VPN requires an IPsec policy. You specify the interface to the private network, the interface to the remote peer and the VPN tunnel. A single policy can enable traffic inbound, outbound, or in both directions.
- A route-based VPN requires an accept policy for each direction. For the source and destination interfaces, you specify the interface to the private network and the virtual IPsec interface (phase 1 configuration) of the VPN. The IPsec interface is the destination interface for the outbound policy and the source interface for the inbound policy. One security policy must be configured for each direction of each VPN interface.

If the policy that grants the VPN connection is limited to certain services, DHCP must be included, otherwise the client will not be able to retrieve a lease from the FortiGate's (IPsec) DHCP server because the DHCP request (coming out of the tunnel) will be blocked.

Policy-based VPN

An IPsec policy enables the transmission and reception of encrypted packets, specifies the permitted direction of VPN traffic, and selects the VPN tunnel. In most cases, a single policy is needed to control both inbound and outbound IP traffic through a VPN tunnel. For a detailed example, see Policy-based IPsec tunnel on page 1379. Be aware of the following before creating an IPsec policy.

Allow traffic to be initiated from the remote site

Policies specify which IP addresses can initiate a tunnel. By default, traffic from the local private network initiates the tunnel. When the *Allow traffic to be initiated from the remote site* option is selected, traffic from a dialup client, or a computer on a remote network, initiates the tunnel. Both can be enabled at the same time for bi-directional initiation of the tunnel.

Outbound and inbound NAT

When a FortiGate operates in NAT mode, you can enable inbound or outbound NAT. Outbound NAT may be performed on outbound encrypted packets or IP packets in order to change their source address before they are sent through the tunnel. Inbound NAT is performed to intercept and decrypt emerging IP packets from the tunnel.

By default, these options are not selected in security policies and can only be set through the CLI.

Defining multiple IPsec policies for the same tunnel

You must define at least one IPsec policy for each VPN tunnel. If the same remote server or client requires access to more than one network behind a local FortiGate, the FortiGate must be configured with an IPsec policy for each network. Multiple policies may be required to configure redundant connections to a remote destination or control access to different services at different times.

To ensure a secure connection, the FortiGate must evaluate policies with *Action* set to *IPsec* before *ACCEPT* and *DENY*. Because the FortiGate unit reads policies starting at the top of the list, you must move all IPsec policies to the top of the list, and be sure to reorder your multiple IPsec policies that apply to the tunnel so that specific constraints can be evaluated before general constraints. If you create two equivalent IPsec policies for two different tunnels, the system will select the correct policy based on the specified source and destination addresses.

Adding multiple IPsec policies for the same VPN tunnel can cause conflicts if the policies specify similar source and destination addresses, but have different settings for the same service. When policies overlap in this manner, the system may apply the wrong IPsec policy or the tunnel may fail.

Route-based VPN

When you define a route-based VPN, you create a virtual IPsec interface on the physical interface that connects to the remote peer. You create ordinary accept policies to enable traffic between the IPsec interface and the interface that connects to the private network. This makes configuration simpler than for policy-based VPNs.

To configure policies for a route-based VPN:

1. Go to *Policy & Objects > Firewall Policy*.
2. Click *Create New* and define an *ACCEPT* policy to permit communication between the local private network and the private network behind the remote peer and enter these settings in particular:

Name	Enter a name for the security policy.
Incoming Interface	Select the interface that connects to the private network behind this FortiGate.
Outgoing Interface	Select the IPsec interface you configured.
Source	Select the address name you defined for the private network behind this FortiGate.
Destination	Select the address name you defined for the private network behind the remote peer.
Action	Select *ACCEPT*.
NAT	Disable *NAT*.

3. Click *OK*.
   To permit the remote client to initiate communication, you need to define a security policy for communication in that direction.
4. Click *Create New* and enter these settings in particular:

Name	Enter a name for the security policy.
Incoming Interface	Select the IPsec interface you configured.
Outgoing Interface	Select the interface that connects to the private network behind this FortiGate.
Source	Select the address name you defined for the private network behind the remote peer.
Destination	Select the address name you defined for the private network behind this FortiGate.
Action	Select *ACCEPT*.
NAT	Disable *NAT*.

5. Click *OK*.

Blocking unwanted IKE negotiations and ESP packets with a local-in policy

It is not unusual to receive IPsec connection attempts or malicious IKE packets from all over the internet. Malicious parties use these probes to try to establish an IPsec tunnel in order to gain access to your private network. A good way to prevent this is to use local-in policies to deny such traffic.

Sometimes there are malicious attempts using crafted invalid ESP packets. These invalid attempts are automatically blocked by the FOS IPsec local-in handler when it checks the SPI value against the SAs of existing tunnels. The IPsec local-in handler processes the packet instead of the firewall's local-in handler. So when these attempts are blocked, you will notice an `unknown SPI` message in your VPN logs instead of being silently blocked by your local-in policy. These log messages are rate limited.

Sample log and alert email

```
Message meets Alert condition

date=2020-08-11 time=09:28:40 devname=toSite1 devid=FGT60Fxxxxxxxxxx logid="0101037131"
type="event" subtype="vpn" level="error" vd="root" eventtime=1597163320747963100 tz="-0700"
logdesc="IPsec ESP" msg="IPsec ESP" action="error" remip=131.62.25.102 locip=192.157.116.88
remport=40601 locport=500 outintf="wan1" cookies="N/A" user="N/A" group="N/A"
xauthuser="N/A" xauthgroup="N/A" assignip=N/A vpntunnel="N/A" status="esp_error" error_
num="Received ESP packet with unknown SPI." spi="f6c9e2x1" seq="02000400"
```

Note that invalid SPIs may not always indicate malicious activity. For example, the SPI may not match during rekey, or when one unit flushes its tunnel SAs. Administrators should collect as much information as possible before making a conclusion.

To block undesirable IPsec connection attempts and IKE packets using a local-in policy:

1. Configure an address group that excludes legitimate IPs:

```
config firewall addrgrp
    edit "All_exceptions"
        set member "all"
        set exclude enable
        set exclude-member "remote-vpn"
    next
end
```

2. Create a local-in policy that blocks IKE traffic from the address group:

```
config firewall local-in-policy
    edit 1
        set intf "wan1"
        set srcaddr "All_exceptions"
        set dstaddr "all"
        set service "IKE"
        set schedule "always"
    next
end
```

The default action is deny.

3. Verify the traffic blocked by the local-in policy:

```
# diagnose debug flow filter dport 500
# diagnose debug flow trace start 10
# diagnose debug enable

id=20085 trace_id=290 func=print_pkt_detail line=5588 msg="vd-root:0 received a packet
(proto=17, 10.10.10.13:500->10.10.10.1:500) from wan1. "
id=20085 trace_id=290 func=init_ip_session_common line=5760 msg="allocate a new session-
003442e7"
id=20085 trace_id=290 func=vf_ip_route_input_common line=2598 msg="find a route:
flag=84000000 gw-10.10.10.1 via root"
id=20085 trace_id=290 func=fw_local_in_handler line=430 msg="iprope_in_check() check
failed on policy 1, drop"
```

Configurable IKE port

Some ISPs block UDP port 500 or UDP port 4500, preventing an IPsec VPN from being negotiated and established. To accommodate this, the IKE port can be changed.

To set the IKE port:

```
config system settings
    set ike-port <integer>
end
```

`ike-port`	UDP port for IKE/IPsec traffic (1024 - 65535, default = 500).

Example 1: site-to-site VPN without NAT

In this example, the IKE port is set to 6000 on the two site-to-site VPN gateways. There is no NAT between the VPN gateways, but the ISP has blocked UDP port 500. A site-to-site VPN is established using the defined IKE port.

To set the IKE port:

```
config system settings
    set ike-port 6000
end
```

To configure and check the site-to-site VPN:

1. Configure the phase1 and phase2 interfaces:

```
config vpn ipsec phase1-interface
    edit "s2s"
        set interface "port27"
        set ike-version 2
        set peertype any
        set net-device disable
        set proposal aes128-sha256 aes256-sha256 aes128gcm-prfsha256 aes256gcm-prfsha384
chacha20poly1305-prfsha256
        set wizard-type static-fortigate
        set remote-gw 11.101.1.1
```

```
        set psksecret **********
    next
end
config vpn ipsec phase2-interface
    edit "s2s"
        set phase1name "s2s"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm 
aes256gcm chacha20poly1305
        set src-addr-type name
        set dst-addr-type name
        set src-name "s2s_local"
        set dst-name "s2s_remote"
    next
end
```

2. Check the IKE gateway list and confirm that the specified port is used:

```
# diagnose vpn ike gateway list

vd: root/0
name: s2s
version: 2
interface: port27 17
addr: 173.1.1.1:6000 -> 11.101.1.1:6000
tun_id: 11.101.1.1
remote_location: 0.0.0.0
created: 194s ago
PPK: no
IKE SA: created 1/2  established 1/2  time 0/4500/9000 ms
IPsec SA: created 1/2  established 1/2  time 0/4500/9000 ms
...
```

3. Check the VPN tunnel list:

```
# diagnose vpn tunnel list
list all ipsec tunnel in vd 0
------------------------------------------------------
name=s2s ver=2 serial=1 173.1.1.1:6000->11.101.1.1:6000 tun_id=11.101.1.1 dst_mtu=1500
dpd-link=on remote_location=0.0.0.0 weight=1
bound_if=17 lgwy=static/1 tun=tunnel/15 mode=auto/1 encap=none/520 options[0208]=npu
frag-rfc  run_state=0 accept_traffic=1 overlay_id=0
...
```

Example 2: dialup VPN with NAT

In this example, the IKE port is set to 5000 on the VPN gateway and the dialup peer. The dialup peer is behind NAT, so NAT traversal (NAT-T) is used. The ISP blocks both UDP port 500 and UDP port 4500. The VPN connection is initiated on UDP port 5000 from the dialup VPN client and remains on port 5000 since NAT-T floating to 4500 is only required when the IKE port is 500.

To set the IKE port:

```
config system settings
    set ike-port 5000
end
```

To configure and check the dialup VPN with NAT:

1. Configure the phase1 and phase2 interfaces:

```
config vpn ipsec phase1-interface
    edit "server"
        set type dynamic
        set interface "port27"
        set ike-version 2
        set peertype any
        set net-device disable
        set proposal aes128-sha256 aes256-sha256 aes128gcm-prfsha256 aes256gcm-prfsha384
chacha20poly1305-prfsha256
        set dpd on-idle
        set wizard-type static-fortigate
        set psksecret **********
        set dpd-retryinterval 60
    next
end
config vpn ipsec phase2-interface
    edit "server"
        set phase1name "server"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
aes256gcm chacha20poly1305
        set src-addr-type name
        set dst-addr-type name
        set src-name "server_local"
        set dst-name "server_remote"
    next
end
```

2. Check the IKE gateway list and confirm that the specified port is used:

```
# diagnose vpn ike gateway list

vd: root/0
name: server_0
version: 2
interface: port27 17
addr: 173.1.1.1:5000 -> 173.1.1.2:65416
tun_id: 173.1.1.2
remote_location: 0.0.0.0
created: 90s ago
nat: peer
PPK: no
IKE SA: created 1/1  established 1/1  time 0/0/0 ms
IPsec SA: created 1/1  established 1/1  time 0/0/0 ms
...
```

3. Check the VPN tunnel list:

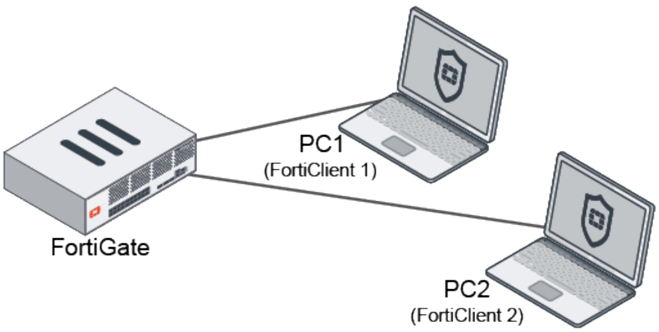
```
# diagnose vpn tunnel list
list all ipsec tunnel in vd 0
------------------------------------------------------
name=server_0 ver=2 serial=a 173.1.1.1:5000->173.1.1.2:65416 tun_id=173.1.1.2 dst_
mtu=1500 dpd-link=on remote_location=0.0.0.0 weight=1
bound_if=17 lgwy=static/1 tun=tunnel/15 mode=dial_inst/3 encap=none/904 options
```

```
[0388]=npu rgwy-chg rport-chg frag-rfc  run_state=0 accept_traffic=1 overlay_id=0
...
```

IPsec VPN IP address assignments

When a user disconnects from a VPN tunnel, it is not always desirable for the released IP address to be used immediately. In IPsec VPN, IP addresses can held for the specified delay interval before being released back into the pool for assignment. The first-available address assignment method is still used.

Example



In this example, two PCs connect to the VPN. The IP address reuse delay interval is used to prevent a released address from being reused for at least four minutes. After the interval elapses, the IP address becomes available to clients again. Dual stack address assignment (both IPv4 and IPv6) is used.

To configure IPsec VPN with an IP address reuse delay interval:

1. Configure the IPsec phase1 interface, setting the IP address reuse delay interval to 240 seconds:

```
config vpn ipsec phase1-interface
    edit "FCT"
        set type dynamic
        set interface "port27"
        set mode aggressive
        set peertype any
        set net-device disable
        set mode-cfg enable
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set wizard-type dialup-forticlient
        set xauthtype auto
        set authusrgrp "local-group"
        set ipv4-start-ip 10.20.1.1
        set ipv4-end-ip 10.20.1.100
        set dns-mode auto
        set ipv4-split-include "FCT_split"
        set ipv6-start-ip 2001::1
        set ipv6-end-ip 2001::2
        set ip-delay-interval 240
        set save-password enable
        set psksecret **********
    next
end
```

2. Configure the IPsec phase2 interface:

```
config vpn ipsec phase2-interface
    edit "FCT"
        set phase1name "FCT"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
aes256gcm chacha20poly1305
    next
    edit "FCT6"
        set phase1name "FCT"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
aes256gcm chacha20poly1305
        set src-addr-type subnet6
        set dst-addr-type subnet6
    next
end
```

To test the results:

1. Connect to the VPN with FortiClient 1 on PC1 then check the assigned IP address:

```
# diagnose vpn ike gateway list

vd: root/0
name: FCT_0
version: 1
interface: port27 17
addr: 173.1.1.1:4500 -> 173.1.1.2:60417
tun_id: 173.1.1.2
remote_location: 0.0.0.0
virtual-interface-addr: 169.254.1.1 -> 169.254.1.1
created: 14s ago
xauth-user: userc
2FA: no
FortiClient UID: 7C0897D80C8E4B6DAC775DD6B0F93BAA
assigned IPv4 address: 10.20.1.1/255.255.255.255
assigned IPv6 address: 2001::1/128
nat: peer
IKE SA: created 1/1  established 1/1  time 100/100/100 ms
IPsec SA: created 2/2  established 2/2  time 0/5/10 ms

  id/spi: 2 66140ba3e38b9b07/b64668f110ca4a48
  direction: responder
  status: established 14-14s ago = 100ms
  proposal: aes256-sha256
  key: 356637ee6e9a9cb5-fade432c09efb8aa-54be307fc1eeeab5-6e4b9ef19f98d5fa
  lifetime/rekey: 86400/86115
  DPD sent/recv: 00000000/00000394
```

2. Disconnect FortiClient 1 and connect with FortiClient 2. The IP address assigned to FortiClient 1 is not released to the pool, and a different IP address is assigned to FortiClient 2:

```
# diagnose vpn ike gateway list

vd: root/0
name: FCT_0
version: 1
```

```
interface: port27 17
addr: 173.1.1.1:4500 -> 173.1.1.2:64916
tun_id: 173.1.1.2
remote_location: 0.0.0.0
virtual-interface-addr: 169.254.1.1 -> 169.254.1.1
created: 6s ago
xauth-user: usera
2FA: no
FortiClient UID: EAF90E297393456AB546A041066C0720
assigned IPv4 address: 10.20.1.2/255.255.255.255
assigned IPv6 address: 2001::2/128
nat: peer
IKE SA: created 1/1  established 1/1  time 110/110/110 ms
IPsec SA: created 2/2  established 2/2  time 0/5/10 ms

  id/spi: 3 b25141d5a915e67e/b32decdb8cf98318
  direction: responder
  status: established 6-6s ago = 110ms
  proposal: aes256-sha256
  key: 374ab753f3207ea0-83496b5cb24b5a8d-c51da1fd505cf3a4-727884839897808a
  lifetime/rekey: 86400/86123
  DPD sent/recv: 00000000/00000453
```

3. Wait for 240 seconds, then disconnect and reconnect FortiClient 2. The IP address previously assigned to FortiClient 1 has been released back to the pool, and is assigned to FortiClient 2:

```
# diagnose vpn ike gateway list

vd: root/0
name: FCT_0
version: 1
interface: port27 17
addr: 173.1.1.1:4500 -> 173.1.1.2:64916
tun_id: 173.1.1.2
remote_location: 0.0.0.0
virtual-interface-addr: 169.254.1.1 -> 169.254.1.1
created: 20s ago
xauth-user: usera
2FA: no
FortiClient UID: EAF90E297393456AB546A041066C0720
assigned IPv4 address: 10.20.1.1/255.255.255.255
assigned IPv6 address: 2001::1/128
nat: peer
IKE SA: created 1/1  established 1/1  time 100/100/100 ms
IPsec SA: created 2/2  established 2/2  time 0/0/0 ms

  id/spi: 4 fb1fbad0c12f5476/aa06a2de76964f63
  direction: responder
  status: established 20-20s ago = 100ms
  proposal: aes256-sha256
  key: af43f1bb876dc79c-16448592fe608dc3-f251746d71b2c35d-c848e8c03bf738e9
  lifetime/rekey: 86400/86109
  DPD sent/recv: 00000000/000000a9
```

Instead of waiting for 240 seconds, you can instead use the `diagnose vpn ike gateway flush` command to release the previously used IP addresses back into the pool.

Site-to-site VPN

A site-to-site VPN connection lets branch offices use the Internet to access the main office's intranet. A site-to-site VPN allows offices in multiple, fixed locations to establish secure connections with each other over a public network such as the Internet.

The following sections provide instructions for configuring site-to-site VPNs:

- FortiGate-to-FortiGate on page 1358
- FortiGate-to-third-party on page 1386

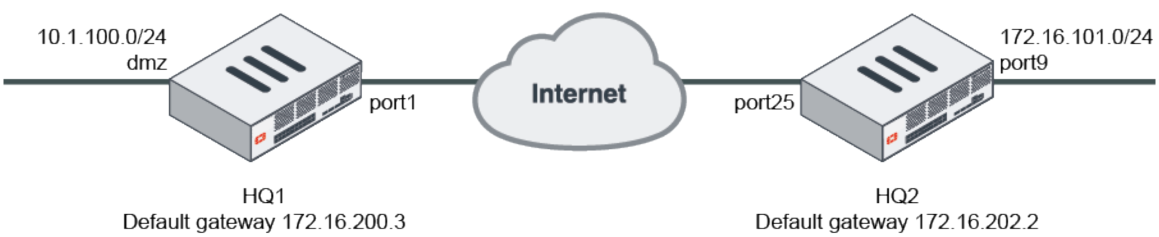
FortiGate-to-FortiGate

This section contains the following topics about FortiGate-to-FortiGate VPN configurations:

- Basic site-to-site VPN with pre-shared key on page 1358
- Site-to-site VPN with digital certificate on page 1363
- Site-to-site VPN with overlapping subnets on page 1370
- GRE over IPsec on page 1374
- Policy-based IPsec tunnel on page 1379

Basic site-to-site VPN with pre-shared key

This is a sample configuration of IPsec VPN authenticating a remote FortiGate peer with a pre-shared key.



To configure IPsec VPN authenticating a remote FortiGate peer with a pre-shared key in the GUI:

1. Configure the HQ1 FortiGate.
   a. Go to *VPN > IPsec Wizard* and configure the following settings for *VPN Setup*:
      i. Enter a VPN name.
      ii. For *Template Type*, select *Site to Site*.
      iii. For *Remote Device Type*, select *FortiGate*.
      iv. For *NAT Configuration*, select *No NAT Between Sites*.
      v. Click *Next*.

b. Configure the following settings for *Authentication*:
   i. For *Remote Device*, select *IP Address*.
   ii. For the IP address, enter *172.16.202.1*.
   iii. For *Outgoing interface*, enter *port1*.
   iv. For *Authentication Method*, select *Pre-shared Key*.
   v. In the *Pre-shared Key* field, enter *sample* as the key.
   vi. Click *Next*.
c. Configure the following settings for *Policy & Routing*:
   i. From the *Local Interface* dropdown menu, select the local interface.
   ii. Configure the *Local Subnets* as *10.1.100.0*.
   iii. Configure the *Remote Subnets* as *172.16.101.0*.
   iv. Click *Create*.

2. Configure the HQ2 FortiGate.
   a. Go to *VPN > IPsec Wizard* and configure the following settings for *VPN Setup*:
      i. Enter a VPN name.
      ii. For *Template Type*, select *Site to Site*.
      iii. For *Remote Device Type*, select *FortiGate*.
      iv. For *NAT Configuration*, select *No NAT Between Sites*.
      v. Click *Next*.
   b. Configure the following settings for *Authentication*:
      i. For *Remote Device*, select *IP Address*.
      ii. For the IP address, enter *172.16.2001*.
      iii. For *Outgoing interface*, enter *port25*.
      iv. For *Authentication Method*, select *Pre-shared Key*.
      v. In the *Pre-shared Key* field, enter *sample* as the key.
      vi. Click *Next*.
   c. Configure the following settings for *Policy & Routing*:
      i. From the *Local Interface* dropdown menu, select the local interface.
      ii. Configure *Local Subnets* as *172.16.101.0*.
      iii. Configure the *Remote Subnets* as *10.1.100.0*.
      iv. Click *Create*.

To configure IPsec VPN authenticating a remote FortiGate peer with a pre-shared key using the CLI:

1. Configure the WAN interface and default route. The WAN interface is the interface connected to the ISP. The IPsec tunnel is established over the WAN interface.
   a. Configure HQ1.

```
config system interface
    edit "port1"
        set vdom "root"
        set ip 172.16.200.1 255.255.255.0
    next
end
config router static
    edit 1
        set gateway 172.16.200.3
        set device "port1"
```

```
    next
end
```

b. Configure HQ2.

```
config system interface
    edit "port25"
        set vdom "root"
        set ip 172.16.202.1 255.255.255.0
    next
end
config router static
    edit 1
        set gateway 172.16.202.2
        set device "port25"
    next
end
```

2. Configure the internal (protected subnet) interface. The internal interface connects to the corporate internal network. Traffic from this interface routes out the IPsec VPN tunnel.
   a. Configure HQ1.

```
config system interface
    edit "dmz"
        set vdom "root"
        set ip 10.1.100.1 255.255.255.0
    next
end
```

   b. Configure HQ2.

```
config system interface
    edit "port9"
        set vdom "root"
        set ip 172.16.101.1 255.255.255.0
    next
end
```

3. Configure the IPsec phase1-interface.
   a. Configure HQ1.

```
config vpn ipsec phase1-interface
    edit "to_HQ2"
        set interface "port1"
        set peertype any
        set net-device enable
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set remote-gw 172.16.202.1
        set psksecret sample
    next
end
```

   b. Configure HQ2.

```
config vpn ipsec phase1-interface
    edit "to_HQ1"
        set interface "port25"
        set peertype any
        set net-device enable
```

```
            set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
            set remote-gw 172.16.200.1
            set psksecret  sample
        next
    end
```

4. **Configure the IPsec phase2-interface.**
   a. Configure HQ1.

```
config vpn ipsec phase2-interface
    edit "to_HQ2"
        set phase1name "to_HQ2"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
aes256gcm chacha20poly1305
        set auto-negotiate enable
    next
end
```

   b. Configure HQ2.

```
config vpn ipsec phase2-interface
    edit "to_HQ2"
        set phase1name "to_HQ1"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
aes256gcm chacha20poly1305
        set auto-negotiate enable
    next
end
```

5. Configure the static routes. Two static routes are added to reach the remote protected subnet. The blackhole route is important to ensure that IPsec traffic does not match the default route when the IPsec tunnel is down.
   a. Configure HQ1.

```
config router static
    edit 2
        set dst 172.16.101.0 255.255.255.0
        set device "to_HQ2"
    next
    edit 3
        set dst 172.16.101.0 255.255.255.0
        set blackhole enable
        set distance 254
    next
end
```

   b. Configure HQ2.

```
config router static
    edit 2
        set dst 10.1.100.0 255.255.255.0
        set device "to_HQ1"
    next
    edit 3
        set dst 10.1.100.0 255.255.255.0
        set blackhole enable
        set distance 254
     next
end
```

6. Configure two firewall policies to allow bidirectional IPsec traffic flow over the IPsec VPN tunnel.
   a. Configure HQ1.

```
config firewall policy
    edit 1
        set name "inbound"
        set srcintf "to_HQ2"
        set dstintf "dmz"
        set srcaddr "172.16.101.0"
        set dstaddr "10.1.100.0"
        set action accept
        set schedule "always"
        set service "ALL"
    next
    edit 2
        set name "outbound"
        set srcintf "dmz"
        set dstintf "to_HQ2"
        set srcaddr "10.1.100.0"
        set dstaddr "172.16.101.0"
        set action accept
        set schedule "always"
        set service "ALL"
    next
end
```

   b. Configure HQ2.

```
config firewall policy
    edit 1
        set name "inbound"
        set srcintf "to_HQ1"
        set dstintf "port9"
        set srcaddr "10.1.100.0"
        set dstaddr "172.16.101.0"
        set action accept
        set schedule "always"
        set service "ALL"
    next
    edit 2
        set name "outbound"
        set srcintf "port9"
        set dstintf "to_HQ1"
        set srcaddr "172.16.101.0"
        set dstaddr "10.1.100.0"
        set action accept
        set schedule "always"
        set service "ALL"
    next
end
```

7. Run `diagnose` commands. The `diagnose debug application ike -1` command is the key to troubleshoot why the IPsec tunnel failed to establish. If the PSK failed to match, the following error shows up in the debug output:

```
ike 0:to_HQ2:15037: parse error
ike 0:to_HQ2:15037: probable pre-shared secret mismatch'
```

   The following commands are useful to check IPsec phase1/phase2 interface status.

a. Run the `diagnose vpn ike gateway list` command on HQ1. The system should return the following:

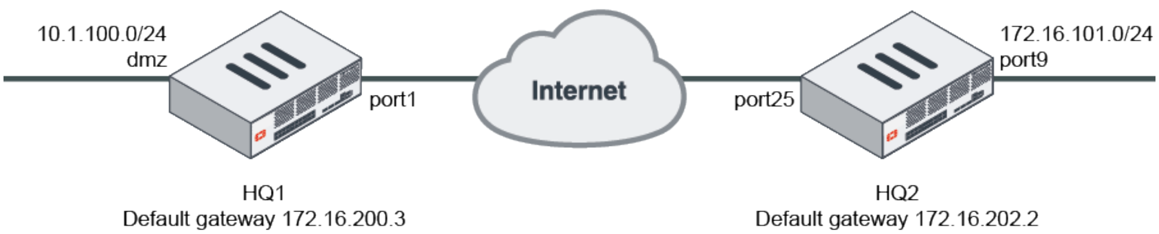
```
vd: root/0
name: to_HQ2
version: 1
interface: port1 11
addr: 172.16.200.1:500 -> 172.16.202.1:500
created: 5s ago
IKE SA: created 1/1 established 1/1 time 0/0/0 ms
IPsec SA: created 2/2 established 2/2 time 0/0/0 ms
id/spi: 12 6e8d0532e7fe8d84/3694ac323138a024
direction: responder
status: established 5-5s ago = 0ms
proposal: aes128-sha256
key: b3efb46d0d385aff-7bb9ee241362ee8d
lifetime/rekey: 86400/86124
DPD sent/recv: 00000000/00000000
```

b. Run the `diagnose vpn tunnel list` command on HQ1. The system should return the following:

```
list all ipsec tunnel in vd 0
name=to_HQ2 ver=1 serial=1 172.16.200.1:0->172.16.202.1:0 tun_id=172.16.202.1
bound_if=11 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/528 options[0210]=create_
dev frag-rfcaccept_traffic=1
proxyid_num=1 child_num=0 refcnt=11 ilast=7 olast=87 ad=/0
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-demand on=1 idle=20000ms retry=3 count=0 seqno=0
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=to_HQ2 proto=0 sa=1 ref=2 serial=1 auto-negotiate
src: 0:0.0.0.0/0.0.0.0:0
dst: 0:0.0.0.0/0.0.0.0:0
SA: ref=3 options=18227 type=00 soft=0 mtu=1438 expire=42927/0B replaywin=2048
seqno=1 esn=0 replaywin_lastseq=00000000 itn=0
life: type=01 bytes=0/0 timeout=42930/43200
dec: spi=ef9ca700 esp=aes key=16 a2c6584bf654d4f956497b3436f1cfc7
ah=sha1 key=20 82c5e734bce81e6f18418328e2a11aeb7baa021b
enc: spi=791e898e esp=aes key=16 0dbb4588ba2665c6962491e85a4a8d5a
ah=sha1 key=20 2054b318d2568a8b12119120f20ecac97ab730b3
dec:pkts/bytes=0/0, enc:pkts/bytes=0/0
```

Site-to-site VPN with digital certificate

This is a sample configuration of IPsec VPN authenticating a remote FortiGate peer with a certificate. The certificate on one peer is validated by the presence of the CA certificate installed on the other peer.



To configure IPsec VPN authenticating a remote FortiGate peer with a digital certificate in the GUI:

1. Import the certificate.
2. Configure user peers.
3. Configure the HQ1 FortiGate.
   a. Go to *VPN > IPsec Wizard* and configure the following settings for *VPN Setup*:
      i. Enter a VPN name.
      ii. For *Template Type*, select *Site to Site*.
      iii. For *Remote Device Type*, select *FortiGate*.
      iv. For *NAT Configuration*, select *No NAT Between Sites*.
      v. Click *Next*.
   b. Configure the following settings for *Authentication*:
      i. For *Remote Device*, select *IP Address*.
      ii. For the IP address, enter *172.16.202.1*.
      iii. For *Outgoing interface*, enter *port1*.
      iv. For *Authentication Method*, select *Signature*.
      v. In the *Certificate name* field, select the imported certificate.
      vi. From the *Peer Certificate CA* dropdown list, select the desired peer CA certificate.
      vii. Click *Next*.
   c. Configure the following settings for *Policy & Routing*:
      i. From the *Local Interface* dropdown menu, select the local interface.
      ii. Configure the *Local Subnets* as *10.1.100.0*.
      iii. Configure the *Remote Subnets* as *172.16.101.0*.
      iv. Click *Create*.
4. Configure the HQ2 FortiGate.
   a. Go to *VPN > IPsec Wizard* and configure the following settings for *VPN Setup*:
      i. Enter a VPN name.
      ii. For *Template Type*, select *Site to Site*.
      iii. For *Remote Device Type*, select *FortiGate*.
      iv. For *NAT Configuration*, select *No NAT Between Sites*.
      v. Click *Next*.
   b. Configure the following settings for *Authentication*:
      i. For *Remote Device*, select *IP Address*.
      ii. For the IP address, enter *172.16.2001*.
      iii. For *Outgoing interface*, enter *port25*.
      iv. For *Authentication Method*, select *Signature*.
      v. In the *Certificate name* field, select the imported certificate.
      vi. From the *Peer Certificate CA* dropdown list, select the peer CA certificate.
      vii. Click *Next*.
   c. Configure the following settings for *Policy & Routing*:
      i. From the *Local Interface* dropdown menu, select the local interface.
      ii. Configure *Local Subnets* as *172.16.101.0*.
      iii. Configure the *Remote Subnets* as *10.1.100.0*.
      iv. Click *Create*.

To configure IPsec VPN authenticating a remote FortiGate peer with a digital certificate using the CLI:

1. Configure the WAN interface and default route. The WAN interface is the interface connected to the ISP. The IPsec tunnel is established over the WAN interface.
   a. Configure HQ1.
   ```
   config system interface
       edit "port1"
           set vdom "root"
           set ip 172.16.200.1 255.255.255.0
       next
   end
   config router static
       edit 1
           set gateway 172.16.200.3
           set device "port1"
       next
   end
   ```
   b. Configure HQ2.
   ```
   config system interface
       edit "port25"
           set vdom "root"
           set ip 172.16.202.1 255.255.255.0
       next
   end
   config router static
       edit 1
           set gateway 172.16.202.2
           set device "port25"
       next
   end
   ```
2. Configure the internal (protected subnet) interface. The internal interface connects to the corporate internal network. Traffic from this interface routes out the IPsec VPN tunnel.
   a. Configure HQ1.
   ```
   config system interface
       edit "dmz"
           set vdom "root"
           set ip 10.1.100.1 255.255.255.0
       next
   end
   ```
   b. Configure HQ2.
   ```
   config system interface
       edit "port9"
           set vdom "root"
           set ip 172.16.101.1 255.255.255.0
       next
   end
   ```
3. Configure the import certificate and its CA certificate information. The certificate and its CA certificate must be imported on the remote peer FortiGate and on the primary FortiGate before configuring IPsec VPN tunnels. If the built-in Fortinet_Factory certificate and the Fortinet_CA CA certificate are used for authentication, you can skip this step.

a. Configure HQ1.

```
config vpn certificate local
    edit "test1"
        ...
        set range global
    next
end
config vpn certificate ca
    edit "CA_Cert_1"
        ...
        set range global
    next
end
```

b. Configure HQ2.

```
config vpn certificate local
    edit "test2"
        ...
        set range global
    next
end
config vpn certificate ca
    edit "CA_Cert_1"
        ...
        set range global
    next
end
```

4. Configure the peer user. The peer user is used in the IPsec VPN tunnel peer setting to authenticate the remote peer FortiGate.
   a. If not using the built-in Fortinet_Factory certificate and Fortinet_CA CA certificate, do the following:
      i. Configure HQ1.

```
config user peer
    edit "peer1"
        set ca "CA_Cert_1"
    next
end
```

      ii. Configure HQ2.

```
config user peer
    edit "peer2"
        set ca "CA_Cert_1"
    next
end
```

   b. If the built-in Fortinet_Factory certificate and Fortinet_CA CA certificate are used for authentication, the peer user must be configured based on Fortinet_CA.
      i. Configure HQ1.

```
config user peer
    edit "peer1"
        set ca "Fortinet_CA"
    next
end
```

ii. Configure HQ2.

```
config user peer
    edit "peer2"
        set ca "Fortinet_CA"
    next
end
```

5. Configure the IPsec phase1-interface.
   a. Configure HQ1.

```
config vpn ipsec phase1-interface
    edit "to_HQ2"
        set interface "port1"
        set authmethod signature
        set net-device enable
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set remote-gw 172.16.202.1
        set certificate "test1"
        set peer "peer1"
    next
end
```

   b. Configure HQ2.

```
config vpn ipsec phase1-interface
    edit "to_HQ1"
        set interface "port25"
        set authmethod signature
        set net-device enable
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set remote-gw 172.16.200.1
        set certificate "test2"
        set peer "peer2"
    next
end
```

6. Configure the IPsec phase2-interface.
   a. Configure HQ1.

```
config vpn ipsec phase2-interface
    edit "to_HQ2"
        set phase1name "to_HQ2"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
aes256gcm chacha20poly1305
        set auto-negotiate enable
    next
end
```

   b. Configure HQ2.

```
config vpn ipsec phase2-interface
    edit "to_HQ2"
        set phase1name "to_HQ1"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
aes256gcm chacha20poly1305
        set auto-negotiate enable
    next
end
```

7. Configure the static routes. Two static routes are added to reach the remote protected subnet. The blackhole route is important to ensure that IPsec traffic does not match the default route when the IPsec tunnel is down.
   a. Configure HQ1.

```
config router static
    edit 2
        set dst 172.16.101.0 255.255.255.0
        set device "to_HQ2"
    next
    edit 3
        set dst 172.16.101.0 255.255.255.0
        set blackhole enable
        set distance 254
    next
end
```

   b. Configure HQ2.

```
config router static
    edit 2
        set dst 10.1.100.0 255.255.255.0
        set device "to_HQ1"
    next
    edit 3
        set dst 10.1.100.0 255.255.255.0
        set blackhole enable
        set distance 254
     next
end
```

8. Configure two firewall policies to allow bidirectional IPsec traffic flow over the IPsec VPN tunnel.
   a. Configure HQ1.

```
config firewall policy
    edit 1
        set name "inbound"
        set srcintf "to_HQ2"
        set dstintf "dmz"
        set srcaddr "172.16.101.0"
        set dstaddr "10.1.100.0"
        set action accept
        set schedule "always"
        set service "ALL"
    next
    edit 2
        set name "outbound"
        set srcintf "dmz"
        set dstintf "to_HQ2"
        set srcaddr "10.1.100.0"
        set dstaddr "172.16.101.0"
        set action accept
        set schedule "always"
        set service "ALL"
    next
end
```

b. Configure HQ2.

```
config firewall policy
    edit 1
        set name "inbound"
        set srcintf "to_HQ1"
        set dstintf "port9"
        set srcaddr "10.1.100.0"
        set dstaddr "172.16.101.0"
        set action accept
        set schedule "always"
        set service "ALL"
    next
    edit 2
        set name "outbound"
        set srcintf "port9"
        set dstintf "to_HQ1"
        set srcaddr "172.16.101.0"
        set dstaddr "10.1.100.0"
        set action accept
        set schedule "always"
        set service "ALL"
    next
end
```

9. Run `diagnose` commands. The `diagnose debug application ike -1` command is the key to troubleshoot why the IPsec tunnel failed to establish. If the remote FortiGate certificate cannot be validated, the following error shows up in the debug output:

```
ike 0: to_HQ2:15314: certificate validation failed
```

The following commands are useful to check IPsec phase1/phase2 interface status.

a. Run the `diagnose vpn ike gateway list` command on HQ1. The system should return the following:

```
vd: root/0
name: to_HQ2
version: 1
interface: port1 11
addr: 172.16.200.1:500 -> 172.16.202.1:500
created: 7s ago
peer-id: C = CA, ST = BC, L = Burnaby, O = Fortinet, OU = QA, CN = test2
peer-id-auth: yes
IKE SA: created 1/1 established 1/1 time 70/70/70 ms
IPsec SA: created 1/1 established 1/1 time 80/80/80 ms
id/spi: 15326 295be407fbddfc13/7a5a52afa56adf14 direction: initiator status:
established 7-7s ago = 70ms proposal: aes128-sha256 key: 4aa06dbee359a4c7-
43570710864bcf7b lifetime/rekey: 86400/86092 DPD sent/recv: 00000000/00000000 peer-
id: C = CA, ST = BC, L = Burnaby, O = Fortinet, OU = QA, CN = test2
```

b. Run the `diagnose vpn tunnel list` command on HQ1. The system should return the following:

```
list all ipsec tunnel in vd 0
name=to_HQ2 ver=1 serial=1 172.16.200.1:0->172.16.202.1:0 tun_id=172.16.200.1
bound_if=11 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/528 options[0210]=create_
dev frag-rfcaccept_traffic=1
proxyid_num=1 child_num=0 refcnt=14 ilast=19 olast=179 ad=/0
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-demand on=1 idle=20000ms retry=3 count=0 seqno=0
```

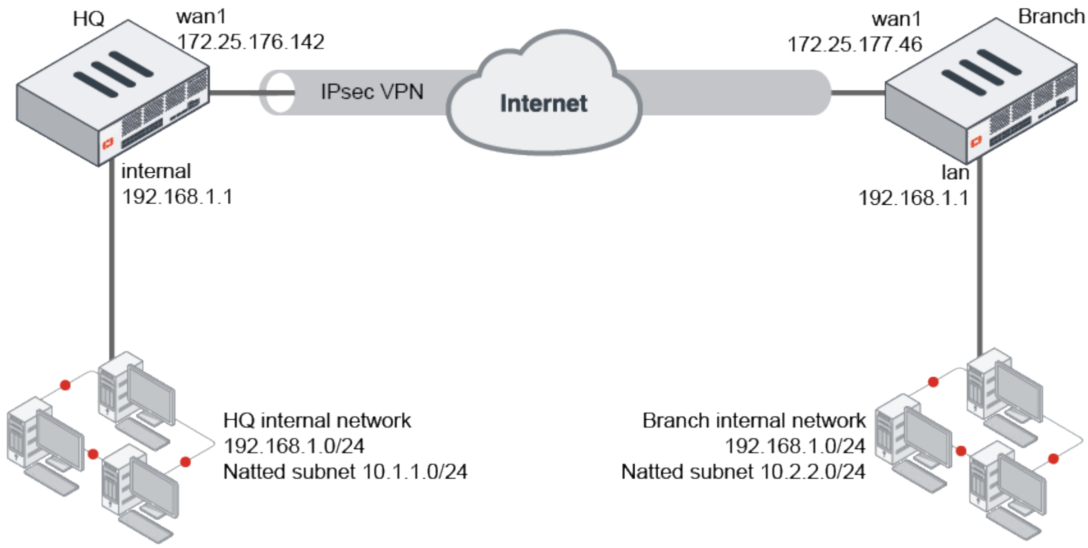
```
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=vpn-f proto=0 sa=1 ref=2 serial=1 auto-negotiate
src: 0:0.0.0.0/0.0.0.0:0
dst: 0:0.0.0.0/0.0.0.0:0
SA: ref=3 options=18227 type=00 soft=0 mtu=1438 expire=42717/0B replaywin=2048
seqno=1 esn=0 replaywin_lastseq=00000000 itn=0
life: type=01 bytes=0/0 timeout=42897/43200
dec: spi=72e87de7 esp=aes key=16 8b2b93e0c149d6f22b1c0b96ea450e6c
ah=sha1 key=20 facc655e5f33beb7c2b12e718a6d55413ce3efa2
enc: spi=5c52c865 esp=aes key=16 8d0c4e4adbf2338beed569b2b3205ece
ah=sha1 key=20 553331628612480ab6d7d563a00e2a967ebabcdd
dec:pkts/bytes=0/0, enc:pkts/bytes=0/0
```

Site-to-site VPN with overlapping subnets

This is a sample configuration of IPsec VPN to allow transparent communication between two overlapping networks that are located behind different FortiGates using a route-based tunnel with source and destination NAT.

In the following topology, both FortiGates (HQ and Branch) use 192.168.1.0/24 as their internal network, but both networks need to be able to communicate to each other through the IPsec tunnel.

New virtual subnets of equal size must be configured and used for all communication between the two overlapping subnets. The devices on both local networks do not need to change their IP addresses. However, the devices and users must use the new subnet range of the remote network to communicate across the tunnel.



Configuring the HQ FortiGate

To configure IPsec VPN:

1. Go to *VPN > IPsec Wizard* and select the *Custom* template.
2. Enter the name *VPN-to-Branch* and click *Next*.
3. For the *IP Address*, enter the Branch public IP address (*172.25.177.46*), and for *Interface*, select the HQ WAN interface (*wan1*).
4. For *Pre-shared Key*, enter a secure key. You will use the same key when configuring IPsec VPN on the Branch FortiGate.

5. In the *Phase 2 Selectors* section, enter the subnets for the *Local Address* (*10.1.1.0/24*) and *Remote Address* (*10.2.2.0/24*).
6. Optionally, expand *Advanced* and enable *Auto-negotiate*.
7. Click *OK*.

To configure the static routes:

1. Go to *Network > Static Routes* and click *Create New*.
2. In the *Destination* field, enter the remote address subnet (*10.2.2.0/24*).
3. For *Interface*, select the VPN tunnel you just created, *VPN-to-Branch*.
4. Click *OK*.
5. Create another route with the same *Destination*, but change the *Administrative Distance* to *200* and for *Interface*, select *Blackhole*. This is a best practice for route-based IPsec VPN tunnels because it ensures traffic for the remote FortiGate's subnet is not sent using the default route in the event that the IPsec tunnel goes down.

To configure the address objects:

1. Go to *Policy & Objects > Addresses* and click *Create New > Address*.
2. For *Name*, enter *HQ-original*.
3. For *IP/Netmask*, enter the original LAN subnet of HQ (*192.168.1.0/24*).
4. For *Interface*, select the LAN-side interface (*internal*).
5. Click *OK*
6. Create another address object named *Branch-new*, but for *IP/Netmask*, enter the new LAN subnet of Branch (*10.2.2.0/24*), and select the VPN interface (*VPN-to-Branch*).

To configure the IP pool:

1. Go to *Policy & Objects > IP Pools* and click *Create New*.
2. For *Name*, enter *HQ-new*.
3. For *Type*, select *Fixed Port Range*.
4. Enter the *External IP address/range* (*10.1.1.1 – 10.1.1.254*, the new HQ subnet) and *Internal IP Range* (*192.168.1.1 – 192.168.1.254*, the original HQ subnet).
5. Click *OK*.

To configure the VIP:

1. Go to *Policy & Objects > Virtual IPs* and click *Create New > Virtual IP*.
2. For *Name*, enter *HQ-new-to-original*.
3. For *Interface*, select the VPN interface (*VPN-to-Branch*).
4. Enter the *External IP address/range* (*10.1.1.1 – 10.1.1.254*, the new HQ subnet) and *Map to IPv4 address/range* (*192.168.1.1 – 192.168.1.254*, the original HQ subnet).
5. Click *OK*.

To configure the firewall policy for traffic from HQ to Branch:

1. Go to *Policy & Objects > Firewall Policy* and click *Create New*.
2. For *Name*, enter *From-HQ-to-Branch*.
3. For *Incoming Interface*, select the LAN-side interface (*internal*).
4. For *Outgoing Interface*, select the VPN tunnel interface (*VPN-to-Branch*).

5. For *Source*, select *HQ-original*.
6. For *Destination*, select *Branch-new*.
7. For *Service*, select *ALL*.
8. Enable *NAT*.
9. Select *Use Dynamic IP Pool* and select the *HQ-new* IP pool.
10. Click *OK*.

To configure the firewall policy for traffic from Branch to HQ:

1. Click *Create New* and for *Name*, enter *From-Branch-to HQ*.
2. For *Incoming Interface*, select the VPN tunnel interface (*VPN-to-Branch*).
3. For *Outgoing Interface*, select the LAN-side interface (*internal*).
4. For *Source*, select *Branch-new*.
5. For *Destination*, select the *HQ-new-to-original* VIP.
6. For *Service*, select *ALL*.
7. Disable *NAT*.
8. Click *OK*.

Configuring the Branch FortiGate

To configure IPsec VPN:

1. Go to *VPN > IPsec Wizard* and select the *Custom* template.
2. Enter the name *VPN-to-HQ* and click *Next*.
3. For the *IP Address*, enter the HQ public IP address (*172.25.176.142*), and for *Interface*, select the Branch WAN interface (*wan1*).
4. For *Pre-shared Key*, enter the matching secure key used in the *VPN-to-Branch* tunnel.
5. In the *Phase 2 Selectors* section, enter the subnets for the *Local Address* (*10.2.2.0/24*) and *Remote Address* (*10.1.1.0/24*).
6. Optionally, expand *Advanced* and enable *Auto-negotiate*.
7. Click *OK*.

To configure the static routes:

1. Go to *Network > Static Routes* and click *Create New*.
2. In the *Destination* field, enter the remote address subnet (*10.1.1.0/24*).
3. For *Interface*, select the VPN tunnel you just created, *VPN-to-HQ*.
4. Click *OK*.
5. Create another route with the same *Destination*, but change the *Administrative Distance* to *200* and for *Interface*, select *Blackhole*.

To configure the address objects:

1. Go to *Policy & Objects > Addresses* and click *Create New > Address*.
2. For *Name*, enter *Branch-original*.
3. For *IP/Netmask*, enter the original LAN subnet of Branch (*192.168.1.0/24*).
4. For *Interface*, select the LAN-side interface (*lan*).
5. Click *OK*

6. Create another address object named *HQ-new*, but for *IP/Netmask*, enter the new LAN subnet of HQ (*10.1.1.0/24*), and select the VPN interface (*VPN-to-HQ*).

To configure the IP pool:

1. Go to *Policy & Objects* > *IP Pools* and click *Create New*.
2. For *Name*, enter *Branch-new*.
3. For *Type*, select *Fixed Port Range*.
4. Enter the *External IP address/range* (*10.2.2.1 – 10.2.2.254*, the new Branch subnet) and *Internal IP Range* (*192.168.1.1 – 192.168.1.254*, the original Branch subnet).
5. Click *OK*.

To configure the VIP:

1. Go to *Policy & Objects* > *Virtual IPs* and click *Create New* > *Virtual IP*.
2. For *Name*, enter *Branch-new-to-original*.
3. For *Interface*, select the VPN interface (*VPN-to-HQ*).
4. Enter the *External IP address/range* (*10.2.2.1 – 10.2.2.254*, the new Branch subnet) and *Map to IPv4 address/range* (*192.168.1.1 – 192.168.1.254*, the original Branch subnet).
5. Click *OK*.

To configure the firewall policy for traffic from Branch to HQ:

1. Go to *Policy & Objects* > *Firewall Policy* and click *Create New*.
2. For *Name*, enter *From-Branch-to-HQ*.
3. For *Incoming Interface*, select the LAN-side interface (*lan*).
4. For *Outgoing Interface*, select the VPN tunnel interface (*VPN-to-HQ*).
5. For *Source*, select *Branch-original*.
6. For *Destination*, select *HQ-new*.
7. For *Service*, select *ALL*.
8. Enable *NAT*.
9. Select *Use Dynamic IP Pool* and select the *Branch-new* IP pool.
10. Click *OK*.

To configure the firewall policy for traffic from HQ to Branch:

1. Click *Create New* and for *Name*, enter *From-HQ-to-Branch*.
2. For *Incoming Interface*, select the VPN tunnel interface (*VPN-to-HQ*).
3. For *Outgoing Interface*, select the LAN-side interface (*lan*).
4. For *Source*, select *HQ-new*.
5. For *Destination*, select the *Branch-new-to-original* VIP.
6. For *Service*, select *ALL*.
7. Disable *NAT*.
8. Click *OK*.

To verify the communication across the tunnel:

1. Go to *Dashboard > Network* and click the *IPsec* widget to expand to full screen view. The tunnels should be up on both FortiGates. If you did not enable *Auto-negotiate* in the IPsec VPN settings, you may have to select the tunnel and click *Bring Up*.
2. From a PC on the HQ network, ping a PC on the Branch network using the new IP for the Branch PC. The ping should be successful.

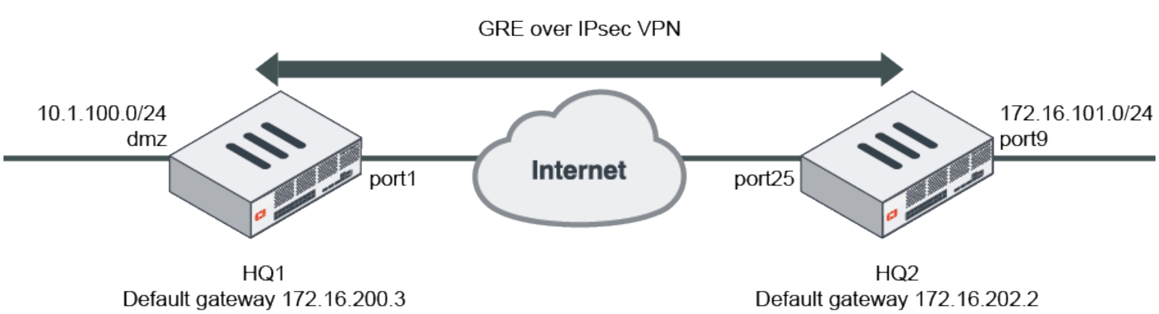
```
C:\Users\jheadley>ping 10.2.2.98

Pinging 10.2.2.98 with 32 bytes of data:
Reply from 10.2.2.98: bytes=32 time=7ms TTL=62
Reply from 10.2.2.98: bytes=32 time=1ms TTL=62
Reply from 10.2.2.98: bytes=32 time=1ms TTL=62
Reply from 10.2.2.98: bytes=32 time=1ms TTL=62

Ping statistics for 10.2.2.98:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 7ms, Average = 2ms
```

3. From a PC on the Branch network, ping a PC on the HQ network using the new IP for the HQ PC. The ping should be successful.

```
Johns-MacBook-Air:~ John$ ping 10.1.1.12
PING 10.1.1.12 (10.1.1.12): 56 data bytes
64 bytes from 10.1.1.12: icmp_seq=0 ttl=126 time=1.912 ms
64 bytes from 10.1.1.12: icmp_seq=1 ttl=126 time=1.743 ms
64 bytes from 10.1.1.12: icmp_seq=2 ttl=126 time=1.403 ms
64 bytes from 10.1.1.12: icmp_seq=3 ttl=126 time=1.425 ms
^C
--- 10.1.1.12 ping statistics ---
4 packets transmitted, 4 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 1.403/1.621/1.912/0.215 ms
```

GRE over IPsec

This is an example of GRE over an IPsec tunnel using a static route over GRE tunnel and `tunnel-mode` in the `phase2-interface` settings.



To configure GRE over an IPsec tunnel:

1. Enable subnet overlapping at both HQ1 and HQ2.

```
config system settings
    set allow-subnet-overlap enable
end
```

2. Configure the WAN interface and static route.
   a. HQ1.

```
config system interface
    edit "port1"
        set ip 172.16.200.1 255.255.255.0
    next
    edit "dmz"
        set ip 10.1.100.1 255.255.255.0
    next
end
config router static
    edit 1
        set gateway 172.16.200.3
        set device "port1"
    next
end
```

   b. HQ2.

```
config system interface
    edit "port25"
        set ip 172.16.202.1 255.255.255.0
    next
    edit "port9"
        set ip 172.16.101.1 255.255.255.0
    next
end
config router static
    edit 1
        set gateway 172.16.202.2
        set device "port25"
    next
end
```

3. Configure IPsec phase1-interface and phase2-interface.
   a. HQ1.

```
config vpn ipsec phase1-interface
    edit "greipsec"
        set interface "port1"
        set peertype any
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set remote-gw 172.16.202.1
        set psksecret sample
        next
end
config vpn ipsec phase2-interface
    edit "greipsec"
        set phase1name "greipsec"
```

```
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
aes256gcm chacha20poly1305
        set protocol 47
    next
end
```

b. HQ2.

```
config vpn ipsec phase1-interface
    edit "greipsec"
        set interface "port25"
        set peertype any
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set remote-gw 172.16.200.1
        set psksecret sample
        next
end
config vpn ipsec phase2-interface
    edit "greipsec"
        set phase1name "greipsec"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
aes256gcm chacha20poly1305
        set protocol 47
    next
end
```

4. Configure IPsec tunnel interface IP address.
   a. HQ1.

```
config system interface
    edit "greipsec"
        set ip 10.10.10.1 255.255.255.255
        set remote-ip 10.10.10.2 255.255.255.255
    next
end
```

   b. HQ2.

```
config system interface
    edit "greipsec"
        set ip 10.10.10.2 255.255.255.255
        set remote-ip 10.10.10.1 255.255.255.255
    next
end
```

5. Configure the GRE tunnel.
   a. HQ1.

```
config system gre-tunnel
    edit "gre_to_HQ2"
        set interface "greipsec"
        set remote-gw 10.10.10.2
        set local-gw 10.10.10.1
    next
end
```

b. HQ2.

```
config system gre-tunnel
    edit "gre_to_HQ1"
        set interface "greipsec"
        set remote-gw 10.10.10.1
        set local-gw 10.10.10.2
    next
end
```

6. Configure the firewall policy.
   a. HQ1.

```
config firewall policy
    edit 1
        set srcintf "dmz"
        set dstintf "gre_to_HQ2"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
    next
    edit 2
        set srcintf "gre_to_HQ2"
        set dstintf "dmz"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
    next
    edit 3
        set srcintf "greipsec"
        set dstintf "greipsec"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
    next
end
```

   b. HQ2.

```
config firewall policy
    edit 1
        set srcintf "port9"
        set dstintf "gre_to_HQ1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
    next
    edit 2
        set srcintf "gre_to_HQ1"
        set dstintf "port9"
```

```
            set srcaddr "all"
            set dstaddr "all"
            set action accept
            set schedule "always"
            set service "ALL"
        next
        edit 3
            set srcintf "greipsec"
            set dstintf "greipsec"
            set srcaddr "all"
            set dstaddr "all"
            set action accept
            set schedule "always"
            set service "ALL"
        next
    end
```

7. Configure the static route.
   a. HQ1.

```
config router static
    edit 2
        set dst 172.16.101.0 255.255.255.0
        set device "gre_to_HQ2"
    next
end
```

   b. HQ2.

```
config router static
    edit 2
        set dst 10.1.100.0 255.255.255.0
        set device "gre_to_HQ1"
    next
end
```

To view the VPN tunnel list on HQ1:

```
diagnose vpn tunnel list
list all ipsec tunnel in vd 0
----
name=greipsec ver=1 serial=1 172.16.200.1:0->172.16.202.1:0 tun_id=172.16.202.1
bound_if=5 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/16 options[0010]=create_dev
proxyid_num=1 child_num=0 refcnt=12 ilast=19 olast=861 ad=/0
stat: rxp=347 txp=476 rxb=58296 txb=51408
dpd: mode=on-demand on=1 idle=20000ms retry=3 count=0 seqno=8
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=greipsec proto=47 sa=1 ref=2 serial=2
  src: 47:0.0.0.0/0.0.0.0:0
  dst: 47:0.0.0.0/0.0.0.0:0
  SA:  ref=3 options=10226 type=00 soft=0 mtu=1438 expire=41689/0B replaywin=2048
       seqno=15c esn=0 replaywin_lastseq=0000015c itn=0
  life: type=01 bytes=0/0 timeout=42898/43200
  dec: spi=9897bd09 esp=aes key=16 5a60e67bf68379309715bd83931680bf
       ah=sha1 key=20 ff35a329056d0d506c0bfc17ef269978a4a57dd3
  enc: spi=e362f336 esp=aes key=16 5574acd8587c5751a88950e1bf8fbf57
```

```
      ah=sha1 key=20 d57ec76ac3c543ac89b2e4d0545518aa2d06669b
  dec:pkts/bytes=347/37476, enc:pkts/bytes=347/58296
```

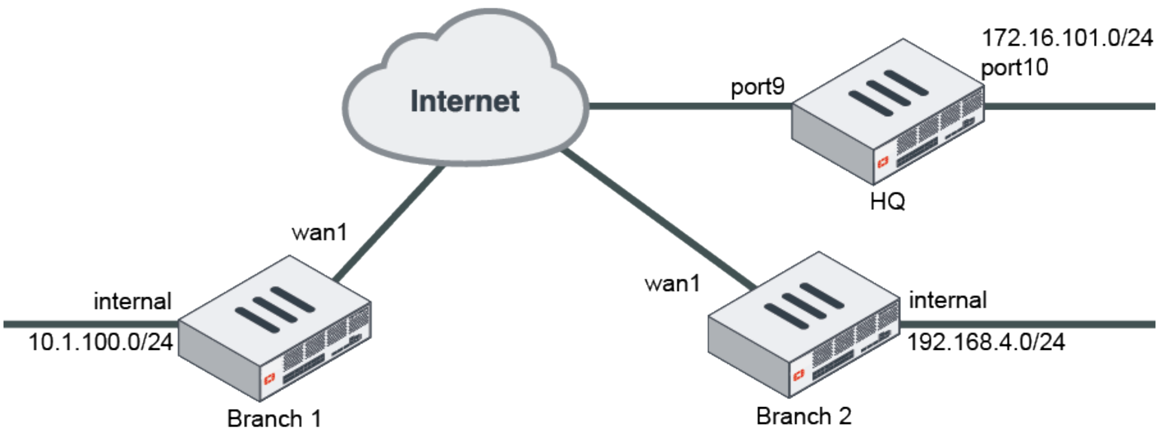
To view the static routing table on HQ1:

```
get router info routing-table static
Routing table for VRF=0
S*      0.0.0.0/0 [10/0] via 172.16.200.3, port1
S       172.16.101.0/24 [10/0] is directly connected, gre_to_HQ2
```

Policy-based IPsec tunnel

This is an example of policy-based IPsec tunnel using site-to-site VPN between branch and HQ. HQ is the IPsec concentrator.

Sample topology



Sample configuration

To configure a policy-based IPsec tunnel using the GUI:

- Configure the IPsec VPN at HQ.
- Configure the IPsec concentrator at HQ.
- Configure the firewall policy at HQ.
- Configure IPsec VPN at branch 1.
- Configure the firewall policy at branch 1.
- Configure IPsec VPN at branch 2.
- Configure the firewall policy at branch 2.

To configure the IPsec VPN at HQ:

1. Go to *VPN > IPsec Wizard* to set up branch 1.
   a. Enter a VPN *Name*. In this example, *to_branch1*.
   b. For *Template Type*, click *Custom*. Click *Next*.
   c. Uncheck *Enable IPsec Interface Mode*.
   d. For *Remote Gateway*, select *Static IP Address*.

   e. Enter IP address, in this example, *15.1.1.2*.
   f. For *Interface*, select *port9*.
   g. In the *Authentication* section, for *Method*, select *Pre-shared Key* and enter the *Pre-shared Key*.
   h. Click *OK*.
2. Go to *VPN > IPsec Wizard* to set up branch 2.
   a. Enter a VPN *Name*. In this example, *to_branch2*.
   b. For *Template Type*, click *Custom*. Click *Next*.
   c. Uncheck *Enable IPsec Interface Mode*.
   d. For *Remote Gateway*, select *Static IP Address*.
   e. Enter IP address, in this example, *13.1.1.2*.
   f. For *Interface*, select *port9*.
   g. In the *Authentication* section, for *Method*, select *Pre-shared Key* and enter the *Pre-shared Key*.
   h. Click *OK*.

To configure the IPsec concentrator at HQ:

1. Go to *VPN > IPsec Concentrator* and click *Create New*.
2. Enter a name. In this example, *branch*.
3. Add the *Members to_branch1* and *to_branch2*.
4. Click *OK*.

To configure the firewall policy at HQ:

1. Go to *Policy & Objects > Firewall Policy* and click *Create New*.
2. Enter a policy *Name*.
3. For *Incoming Interface*, select *port10*.
4. For *Outgoing Interface*, select *port9*.
5. Select the *Source*, *Destination*, *Schedule*, *Service*, and set *Action* to *IPsec*.
6. Select the *VPN Tunnel*, in this example, *Branch1/Branch2*.
7. In this example, enable *Allow traffic to be initiated from the remote site*.
8. Click *OK*.

To configure IPsec VPN at branch 1:

1. Go to *VPN > IPsec Wizard* to set up branch 1.
2. Enter a VPN name. In this example, *to_HQ*.
3. For *Template Type*, click *Custom*. Click *Next*.
4. Uncheck *Enable IPsec Interface Mode*.
5. For *Remote Gateway*, select *Static IP Address*.
6. Enter IP address, in this example, *22.1.1.1*.
7. For *Interface*, select *wan1*.
8. In the *Authentication* section, for *Method*, select *Pre-shared Key* and enter the *Pre-shared Key*.
9. Click *OK*.

To configure the firewall policy at branch 1:

1. Go to *Policy & Objects* > *Firewall Policy* and click *Create New*.
2. Enter a policy *Name*.
3. Choose the *Incoming Interface*, in this example, *internal*.
4. Choose the *Outgoing Interface*, in this example, *wan1*.
5. Select the *Source*, *Destination*, *Schedule*, *Service*, and set *Action* to *IPsec*.
6. Select the *VPN Tunnel*, in this example, *Branch1/Branch2*.
7. In this example, enable *Allow traffic to be initiated from the remote site*.
8. Click *OK*.

To configure IPsec VPN at branch 2:

1. Go to *VPN* > *IPsec Wizard* to set up branch 1.
2. Enter a VPN name. In this example, *to_HQ*.
3. For *Template Type*, click *Custom*. Click *Next*.
4. Uncheck *Enable IPsec Interface Mode*.
5. For *Remote Gateway*, select *Static IP Address*.
6. Enter IP address, in this example, *22.1.1.1*.
7. For *Interface*, select *wan1*.
8. In the *Authentication* section, for *Method*, select *Pre-shared Key* and enter the *Pre-shared Key*.
9. Click *OK*.

To configure the firewall policy at branch 2:

1. Go to *Policy & Objects* > *Firewall Policy* and click *Create New*.
2. Enter a policy *Name*.
3. Choose the *Incoming Interface*, in this example, *internal*.
4. Choose the *Outgoing Interface*, in this example, *wan1*.
5. Select the *Source*, *Destination*, *Schedule*, *Service*, and set *Action* to *IPsec*.
6. Select the *VPN Tunnel*, in this example, *to_HQ*.
7. In this example, enable *Allow traffic to be initiated from the remote site*.
8. Click *OK*.

To configure a policy-based IPsec tunnel using the CLI:

1. Configure the HQ WAN interface and static route.

```
config system interface
    edit "port9"
        set alias "WAN"
        set ip 22.1.1.1 255.255.255.0
    next
    edit "port10"
        set alias "Internal"
        set ip 172.16.101.1 255.255.255.0
    next
end
config router static
```

```
    edit 1
        set gateway 22.1.1.2
        set device "port9"
    next
end
```

2. Configure the HQ IPsec phase1 and phase2.

```
config vpn ipsec phase1
    edit "to_branch1"
        set interface "port9"
        set peertype any
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set remote-gw 15.1.1.2
        set psksecret sample
        next
    edit "to_branch2"
        set interface "port9"
        set peertype any
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set remote-gw 13.1.1.2
        set psksecret sample
        next
end
config vpn ipsec phase2
    edit "to_branch1"
        set phase1name "to_branch1"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
aes256gcm chacha20poly1305
    next
    edit "to_branch2"
        set phase1name "to_branch2"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
aes256gcm chacha20poly1305
    next
end
```

3. Configure the firewall policy at HQ.

```
config firewall policy
    edit 1
        set srcintf "port10"
        set dstintf "port9"
        set srcaddr "all"
        set dstaddr "10.1.100.0"
        set action ipsec
        set schedule "always"
        set service "ALL"
        set inbound enable
        set vpntunnel "to_branch1"
    next
    edit 2
        set srcintf "port10"
        set dstintf "port9"
        set srcaddr "all"
        set dstaddr "192.168.4.0"
        set action ipsec
```

```
        set schedule "always"
        set service "ALL"
        set inbound enable
        set vpntunnel "to_branch2"
    next
end
```

4. Configure the IPsec concentrator at HQ.

```
config vpn ipsec concentrator
    edit "branch"
        set member "to_branch1" "to_branch2"
    next
end
```

5. Configure the branch WAN interface and static route.
   a. For branch 1.

```
config system interface
    edit "wan1"
        set alias "primary_WAN"
        set ip 15.1.1.2 255.255.255.0
    next
    edit "internal"
        set ip 10.1.100.1 255.255.255.0
    next
end
config router static
    edit 1
        set gateway 15.1.1.1
        set device "wan1"
    next
end
```

   b. For branch 2.

```
config system interface
    edit "wan1"
        set alias "primary_WAN"
        set ip 13.1.1.2 255.255.255.0
    next
    edit "internal"
        set ip 192.168.4.1 255.255.255.0
    next
end
config router static
    edit 1
        set gateway 13.1.1.1
        set device "wan1"
    next
end
```

6. Configure the branch IPsec phase1 and phase2.
   a. For branch 1.

```
config vpn ipsec phase1
    edit "to_HQ"
        set interface "wan1"
```

```
            set peertype any
            set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
            set remote-gw 22.1.1.1
            set psksecret sample
            next
    end
    config vpn ipsec phase2
        edit "to_HQ"
            set phase1name "to_HQ"
            set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
    aes256gcm chacha20poly1305
        next
    end
```

b. For branch 2.

```
config vpn ipsec phase1
    edit "to_HQ"
        set interface "wan1"
        set peertype any
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set remote-gw 22.1.1.1
        set psksecret sample
        next
end
config vpn ipsec phase2
    edit "to_HQ"
        set phase1name "to_HQ"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
aes256gcm chacha20poly1305
    next
end
```

7. Configure the branch firewall policy.

a. For branch 1.

```
config firewall policy
    edit 1
        set srcintf "internal"
        set dstintf "wan1"
        set srcaddr "10.1.100.0"
        set dstaddr "all"
        set action ipsec
        set schedule "always"
        set service "ALL"
        set inbound enable
        set vpntunnel "to_HQ"
    next
end
```

b. For branch 2.

```
config firewall policy
    edit 1
        set srcintf "internal"
        set dstintf "wan1"
        set srcaddr "192.168.4.0"
        set dstaddr "all"
```

```
            set action ipsec
            set schedule "always"
            set service "ALL"
            set inbound enable
            set vpntunnel "to_HQ"
        next
    end
```

To view the IPsec VPN tunnel list at HQ:

```
# diagnose vpn tunnel list
```

```
list all ipsec tunnel in vd 0
----
name=to_branch1 ver=1 serial=4 22.1.1.1:0->15.1.1.2:0 tun_id=15.1.1.2
bound_if=42 lgwy=static/1 tun=tunnel/1 mode=auto/1 encap=none/8 options[0008]=npu
proxyid_num=1 child_num=0 refcnt=8 ilast=0 olast=0 ad=/0
stat: rxp=305409 txp=41985 rxb=47218630 txb=2130108
dpd: mode=on-demand on=1 idle=20000ms retry=3 count=0 seqno=0
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=to_branch1 proto=0 sa=1 ref=3 serial=1
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=6 options=10226 type=00 soft=0 mtu=1438 expire=42604/0B replaywin=2048
       seqno=1 esn=0 replaywin_lastseq=00000680 itn=0
  life: type=01 bytes=0/0 timeout=42932/43200
  dec: spi=ca646442 esp=aes key=16 58c91d4463968dddccc4fd97de90a4b8
       ah=sha1 key=20 c9176fe2fbc82ef7e726be9ad4af83eb1b55580a
  enc: spi=747c10c4 esp=aes key=16 7cf0f75b784f697bc7f6d8b4bb8a83c1
       ah=sha1 key=20 cdddc376a86f5ca0149346604a59af07a33b11c5
  dec:pkts/bytes=1664/16310, enc:pkts/bytes=0/16354
  npu_flag=03 npu_rgwy=15.1.1.2 npu_lgwy=22.1.1.1 npu_selid=3 dec_npuid=2 enc_npuid=2
----
name=to_branch2 ver=1 serial=5 22.1.1.1:0->13.1.1.2:0 tun_id=13.1.1.2
bound_if=42 lgwy=static/1 tun=tunnel/1 mode=auto/1 encap=none/8 options[0008]=npu
proxyid_num=1 child_num=0 refcnt=7 ilast=2 olast=43228 ad=/0
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-demand on=1 idle=20000ms retry=3 count=0 seqno=0
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=to_branch2 proto=0 sa=1 ref=2 serial=1
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=3 options=10226 type=00 soft=0 mtu=1280 expire=40489/0B replaywin=2048
       seqno=1 esn=0 replaywin_lastseq=00000000 itn=0
  life: type=01 bytes=0/0 timeout=42931/43200
  dec: spi=ca646441 esp=aes key=16 57ab680d29d4aad4e373579fb50e9909
       ah=sha1 key=20 12a2bc703d2615d917ff544eaff75a6d2c17f1fe
  enc: spi=f9cffb61 esp=aes key=16 3d64da9feb893874e007babce0229259
       ah=sha1 key=20 f92a3ad5e56cb8e89c47af4dac10bf4b4bebff16
  dec:pkts/bytes=0/0, enc:pkts/bytes=0/0
  npu_flag=00 npu_rgwy=13.1.1.2 npu_lgwy=22.1.1.1 npu_selid=4 dec_npuid=0 enc_npuid=0
```

To view the IPsec VPN concentrator at HQ:

```
# diagnose vpn concentrator list
```

```
list all ipsec concentrator in vd 0
name=branch              ref=3         tuns=2 flags=0
```

FortiGate-to-third-party

This section contains the following topics about FortiGate-to-third-party VPN configurations:

- IKEv2 IPsec site-to-site VPN to an AWS VPN gateway on page 1386
- IPsec VPN to Azure with virtual network gateway on page 1392
- IPsec VPN to an Azure with virtual WAN on page 1401
- IPSec VPN between a FortiGate and a Cisco ASA with multiple subnets on page 1405
- Cisco GRE-over-IPsec VPN on page 1405

IKEv2 IPsec site-to-site VPN to an AWS VPN gateway

This is a sample configuration of an IPsec site-to-site VPN connection between an on-premise FortiGate and an AWS virtual private cloud (VPC).

AWS uses unique identifiers to manipulate a VPN connection's configuration. Each VPN connection is assigned an identifier and is associated with two other identifiers: the customer gateway ID for the FortiGate and virtual private gateway ID.

This example includes the following IDs:

- VPN connection ID: vpn-07e988ccc1d46f749
- Customer gateway ID: cgw-0440c1aebed2f418a
- Virtual private gateway ID

This example assumes that you have configured VPC-related settings in the AWS management portal as described in Create a Secure Connection using AWS VPC.

This example includes creating and configuring two tunnels. You must configure both tunnels on your FortiGate.

To configure IKEv2 IPsec site-to-site VPN to an AWS VPN gateway:

1. Configure the first VPN tunnel:
    a. Configure Internet Key Exchange (IKE).
    b. Configure IPsec.
    c. Configure the tunnel interface.
    d. Configure border gateway protocol (BGP).
    e. Configure firewall policies.
2. Configure the second VPN tunnel:
    a. Configure Internet Key Exchange (IKE).
    b. Configure IPsec.
    c. Configure the tunnel interface.
    d. Configure BGP.
    e. Configure firewall policies.

To configure IKE for the first VPN tunnel:

A policy is established for the supported ISAKMP encryption, authentication, Diffie-Hellman (DH), lifetime, and key parameters. These sample configurations fulfill the minimum requirements for AES128, SHA1, and DH Group 2. Category VPN connections in the GovCloud AWS region have a minimum requirement of AES128, SHA2, and DH Group 14. To take advantage of AES256, SHA256, or other DH groups such as 14-18, 22, 23, and 24, you must modify these sample configuration files. Higher parameters are only available for VPNs of category "VPN", not for "VPN-Classic".

Your FortiGate's external interface's address must be static. Your FortiGate may reside behind a device performing NAT. To ensure NAT traversal can function, you must adjust your firewall rules to unblock UDP port 4500. If not behind NAT, it is recommended to disable NAT traversal.

Begin configuration in the root VDOM. The interface name must be shorter than 15 characters. It is best if the name is shorter than 12 characters. IPsec dead peer detection (DPD) causes periodic messages to be sent to ensure a security association remains operational.

```
config vpn ipsec phase1-interface
   edit vpn-07e988ccc1d46f749-0
      set interface "wan1"
      set dpd enable
      set local-gw 35.170.66.108
      set dhgrp 2
      set proposal aes128-sha1
      set keylife 28800
      set remote-gw 3.214.239.164
      set psksecret iCelks0UOob8z4SYMRM6zlx.rU2C3jth
      set dpd-retryinterval 10
   next
end
```

To configure IPsec for the first VPN tunnel:

The IPsec transform set defines the encryption, authentication, and IPsec mode parameters.

```
config vpn ipsec phase2-interface
   edit "vpn-07e988ccc1d46f749-0"
      set phase1name "vpn-07e988ccc1d46f749-0"
      set proposal aes128-sha1
      set dhgrp 2
      set pfs enable
      set keylifeseconds 3600
   next
end
```

To configure the tunnel interface for the first VPN tunnel:

You must configure a tunnel interface as the logical interface associated with the tunnel. All traffic routed to the tunnel interface must be encrypted and transmitted to the VPC. Similarly, traffic from the VPC will be logically received on this interface.

You must configure the interface's address with your FortiGate's address. If the address changes, you must recreate the FortiGate and VPN connection with Amazon VPC.

The `tcp-mss` option causes the router to reduce the TCP packets' maximum segment size to prevent packet fragmentation.

```
config system interface
   edit "vpn-07e988ccc1d46f749-0"
      set vdom "root"
      set ip 169.254.45.90 255.255.255.255
      set allowaccess ping
      set type tunnel
      set tcp-mss 1379
      set remote-ip 169.254.45.89
      set mtu 1427
      set interface "wan1"
   next
end
```

To configure BGP for the first VPN tunnel:

BGP is used within the tunnel to exchange prefixes between the virtual private gateway and your FortiGate. The virtual private gateway announces the prefix according to your VPC.

The local BGP autonomous system number (ASN) (65000) is configured as part of your FortiGate. If you must change the ASN, you must recreate the FortiGate and VPN connection with AWS.

Your FortiGate may announce a default route (0.0.0.0/0) to AWS. This is done using a prefix list and route map in FortiOS.

```
config router bgp
   set as 65000
   config neighbor
      edit 169.254.45.89
         set remote-as 64512
      end
   end
end
config router bgp
   config neighbor
      edit 169.254.45.89
         set capability-default-originate enable
      end
   end
end
config router prefix-list
   edit "default_route"
      config rule
         edit 1
            set prefix 0.0.0.0 0.0.0.0
         next
      end
   end
end
config router route-map
   edit "routemap1"
      config rule
         edit 1
            set match-ip-address "default_route"
         next
      end
   next
end
```

To advertise additional prefixes to the Amazon VPC, add these prefixes to the network statement and identify the prefix you want to advertise. Ensure that the prefix is present in the routing table of the device with a valid next-hop. If you want to advertise 192.168.0.0/16 to Amazon, you would do the following:

```
config router bgp
config network
   edit 1
      set prefix 192.168.0.0 255.255.0.0
   next
end
```

To configure firewall policies for the first VPN tunnel:

Create a firewall policy permitting traffic from your local subnet to the VPC subnet, and vice-versa.

This example policy permits all traffic from the local subnet to the VPC. First, view all existing policies using the `show firewall policy` command. Then, create a new firewall policy starting with the next available policy ID. In this example, running `show firewall policy` displayed policies 1, 2, 3, and 4, so you would proceed to create policy 5.

```
config firewall policy
   edit 5
      set srcintf "vpn-07e988ccc1d46f749-0"
      set dstintf internal
      set srcaddr all
      set dstaddr all
      set action accept
      set schedule always
      set service ANY
   next
end
config firewall policy
   edit 5
      set srcintf internal
      set dstintf "vpn-07e988ccc1d46f749-0"
      set srcaddr all
      set dstaddr all
      set action accept
      set schedule always
      set service ANY
   next
end
```

To configure IKE for the second VPN tunnel:

A policy is established for the supported ISAKMP encryption, authentication, DH, lifetime, and key parameters. These sample configurations fulfill the minimum requirements for AES128, SHA1, and DH Group 2. Category VPN connections in the GovCloud AWS region have a minimum requirement of AES128, SHA2, and DH Group 14. To take advantage of AES256, SHA256, or other DH groups such as 14-18, 22, 23, and 24, you must modify these sample configuration files. Higher parameters are only available for VPNs of category "VPN", not for "VPN-Classic".

Your FortiGate's external interface's address must be static. Your FortiGate may reside behind a device performing NAT. To ensure NAT traversal can function, you must adjust your firewall rules to unblock UDP port 4500. If not behind NAT, it is recommended to disable NAT traversal.

Begin configuration in the root VDOM. The interface name must be shorter than 15 characters. It is best if the name is shorter than 12 characters. IPsec DPD causes periodic messages to be sent to ensure a security association remains operational.

```
config vpn ipsec phase1-interface
   edit vpn-07e988ccc1d46f749-1
      set interface "wan1"
      set dpd enable
      set local-gw 35.170.66.108
      set dhgrp 2
      set proposal aes128-sha1
      set keylife 28800
      set remote-gw 100.25.187.58
      set psksecret IjFzyDneUtDdAT4RNmQ85apUG3y4Akre
      set dpd-retryinterval 10
   next
end
```

To configure IPsec for the second VPN tunnel:

The IPsec transform set defines the encryption, authentication, and IPsec mode parameters.

```
config vpn ipsec phase2-interface
   edit "vpn-07e988ccc1d46f749-1"
      set phase1name "vpn-07e988ccc1d46f749-1"
      set proposal aes128-sha1
      set dhgrp 2
      set pfs enable
      set keylifeseconds 3600
   next
end
```

To configure the tunnel interface for the second VPN tunnel:

You must configure a tunnel interface as the logical interface associated with the tunnel. All traffic routed to the tunnel interface must be encrypted and transmitted to the VPC. Similarly, traffic from the VPC will be logically received on this interface.

You must configure the interface's address with your FortiGate's address. If the address changes, you must recreate the FortiGate and VPN connection with Amazon VPC.

The `tcp-mss` option causes the router to reduce the TCP packets' maximum segment size to prevent packet fragmentation.

```
config system interface
   edit "vpn-07e988ccc1d46f749-1"
      set vdom "root"
      set ip 169.254.44.162 255.255.255.255
      set allowaccess ping
      set type tunnel
      set tcp-mss 1379
      set remote-ip 169.254.44.161
      set mtu 1427
      set interface "wan1"
   next
end
```

To configure BGP for the second VPN tunnel:

BGP is used within the tunnel to exchange prefixes between the virtual private gateway and your FortiGate. The virtual private gateway announces the prefix according to your VPC.

The local BGP ASN (65000) is configured as part of your FortiGate. If you must change the ASN, you must recreate the FortiGate and VPN connection with AWS.

Your FortiGate may announce a default route (0.0.0.0/0) to AWS. This is done using a prefix list and route map in FortiOS.

```
config router bgp
   set as 65000
   config neighbor
      edit 169.254.44.161
         set remote-as 64512
      end
config router bgp
   config neighbor
      edit 169.254.44.161
         set capability-default-originate enable
      end
   end
config router prefix-list
   edit "default_route"
      config rule
         edit 1
            set prefix 0.0.0.0 0.0.0.0
         next
      end
   end
end
config router route-map
   edit "routemap1"
      config rule
         edit 1
            set match-ip-address "default_route"
         next
      end
   next
end
```

To advertise additional prefixes to the Amazon VPC, add these prefixes to the network statement and identify the prefix you want to advertise. Ensure that the prefix is present in the routing table of the device with a valid next-hop. If you want to advertise 192.168.0.0/16 to Amazon, you would do the following:

```
config router bgp
config network
   edit 1
      set prefix 192.168.0.0 255.255.0.0
   next
end
```

To configure firewall policies for the second VPN tunnel:

Create a firewall policy permitting traffic from your local subnet to the VPC subnet, and vice-versa.

This example policy permits all traffic from the local subnet to the VPC. First, view all existing policies using the `show firewall policy` command. Then, create a new firewall policy starting with the next available policy ID. In this example, running `show firewall policy` displayed policies 1, 2, 3, 4, and 5, so you would proceed to create policy 6.

```
config firewall policy
   edit 6
      set srcintf "vpn-07e988ccc1d46f749-1"
      set dstintf internal
      set srcaddr all
      set dstaddr all
      set action accept
      set schedule always
      set service ANY
   next
end
config firewall policy
   edit 6
         set srcintf internal
         set dstintf "vpn-07e988ccc1d46f749-1"
         set srcaddr all
         set dstaddr all
         set action accept
         set schedule always
         set service ANY
   next
end
```

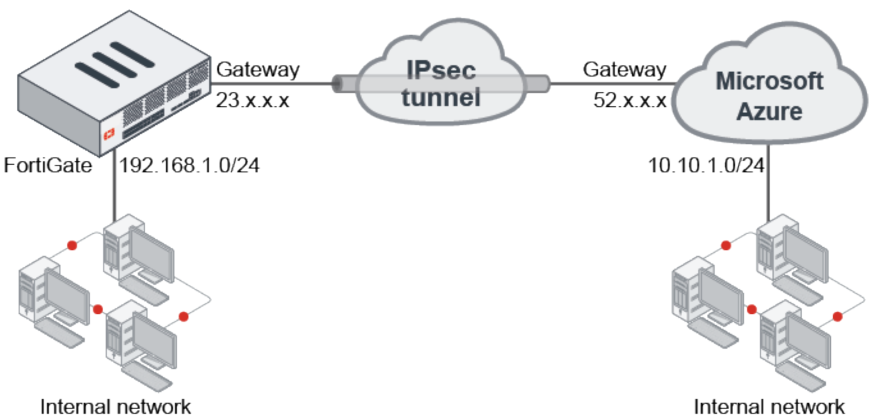
IPsec VPN to Azure with virtual network gateway

This example shows how to configure a site-to-site IPsec VPN tunnel to Microsoft Azure. It shows how to configure a tunnel between each site, avoiding overlapping subnets, so that a secure tunnel can be established.

Prerequisites

- A FortiGate with an Internet-facing IP address
- A valid Microsoft Azure account

Sample topology



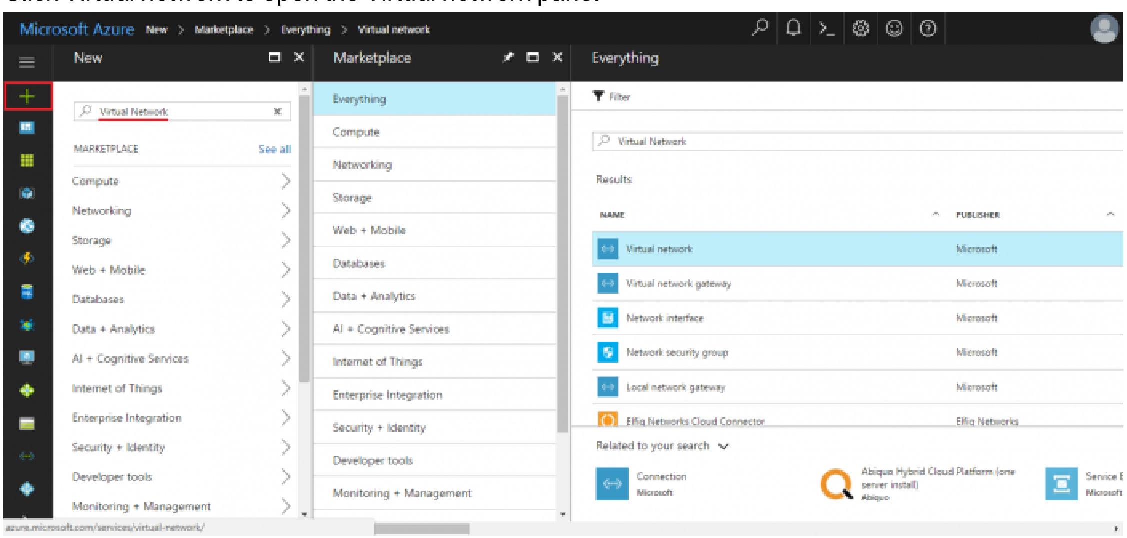
Sample configuration

This sample configuration shows how to:

1. Configure an Azure virtual network
2. Specify the Azure DNS server
3. Configure the Azure virtual network gateway
4. Configure the Azure local network gateway
5. Configure the FortiGate tunnel
6. Create the Azure firewall object
7. Create the FortiGate firewall policies
8. Create the FortiGate static route
9. Create the Azure site-to-site VPN connection
10. Check the results

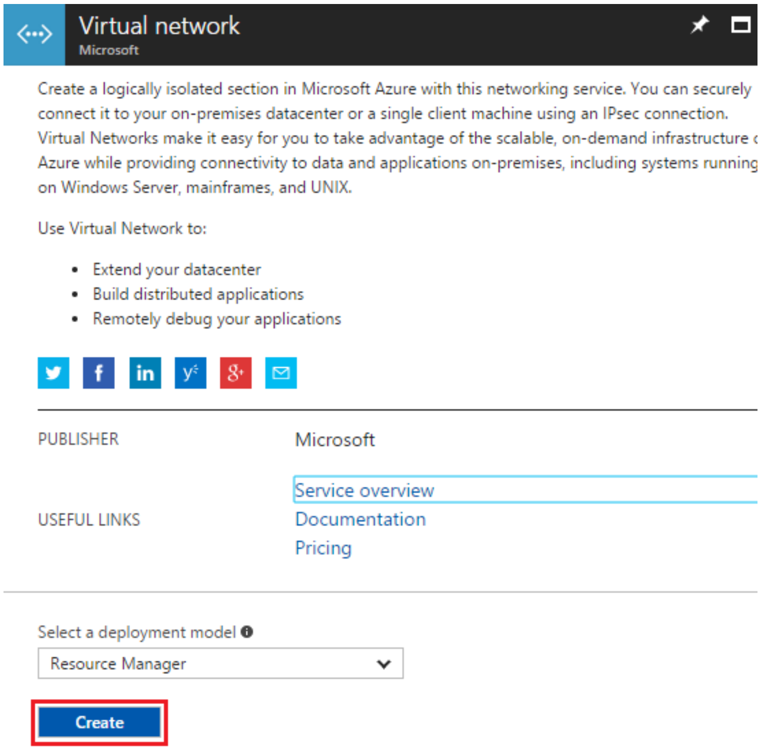
To configure an Azure virtual network:

1. Log in to Azure and click *New*.
2. In *Search the Marketplace*, type *Virtual network*.
3. Click *Virtual network* to open the *Virtual network* pane.

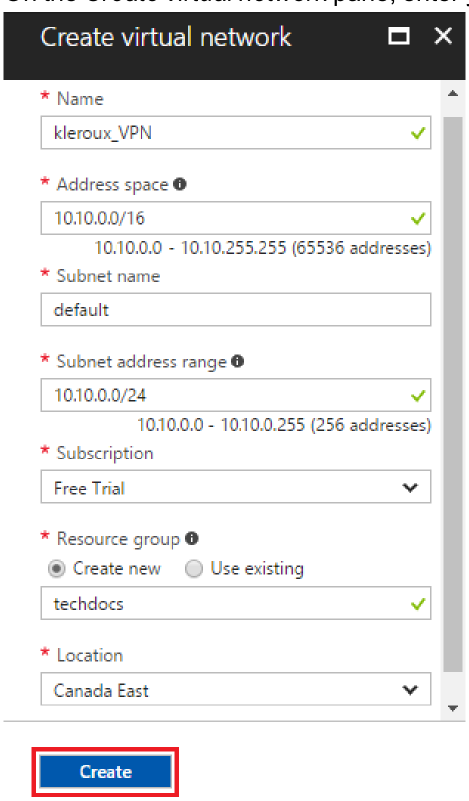


4. At the bottom of the *Virtual network* pane, click the *Select a deployment model* dropdown list and select *Resource Manager*.

5. Click *Create*.

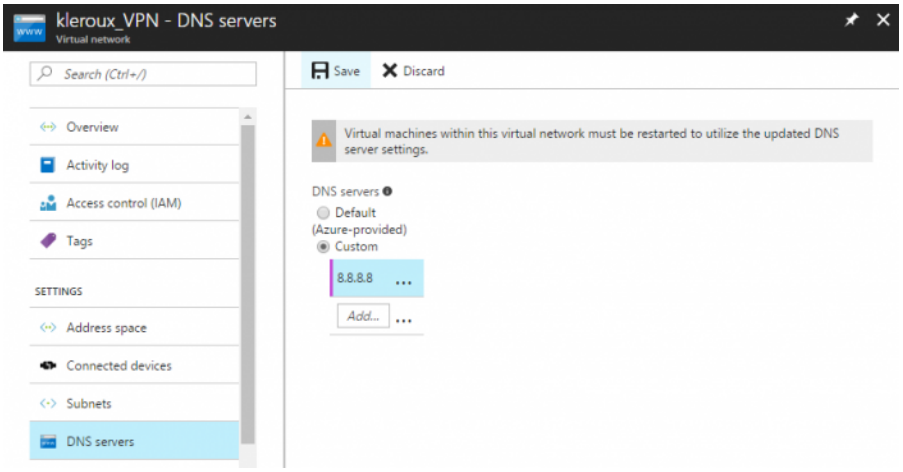


6. On the *Create virtual network* pane, enter you virtual network settings, and click *Create*.



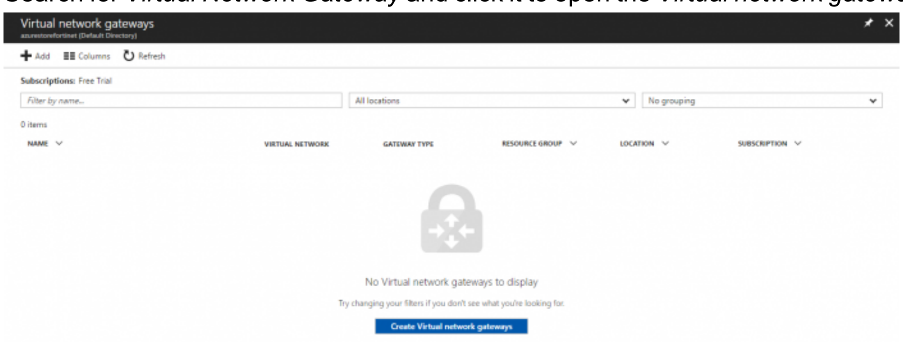
To specify the Azure DNS server:

1. Open the virtual network you just created.
2. Click *DNS servers* to open the *DNS servers* pane.
3. Enter the IP address of the DNS server and click *Save*.

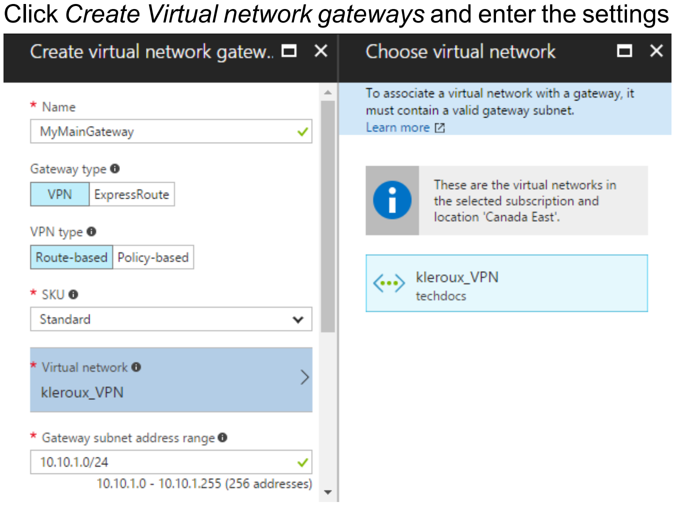


To configure the Azure virtual network gateway:

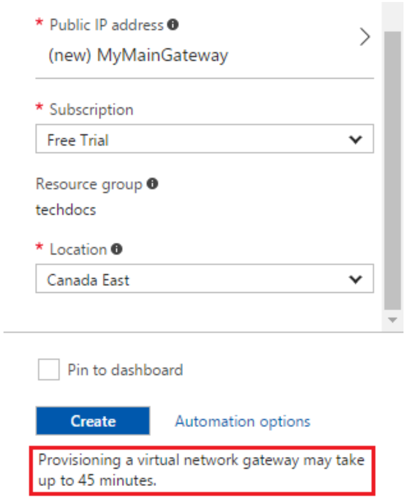
1. In the portal dashboard, go to *New*.
2. Search for *Virtual Network Gateway* and click it to open the *Virtual network gateway* pane.



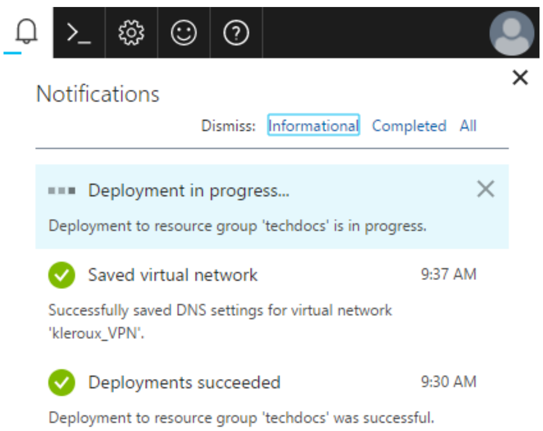
3. Click *Create Virtual network gateways* and enter the settings for your virtual network gateway.



4. If needed, create a Public IP address.

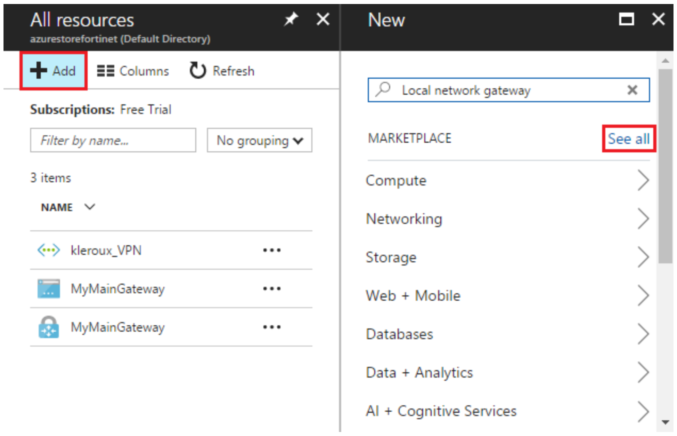


5. Click *Create*.
Creating the virtual network gateway might take some time. When the provisioning is done, you'll receive a notification.

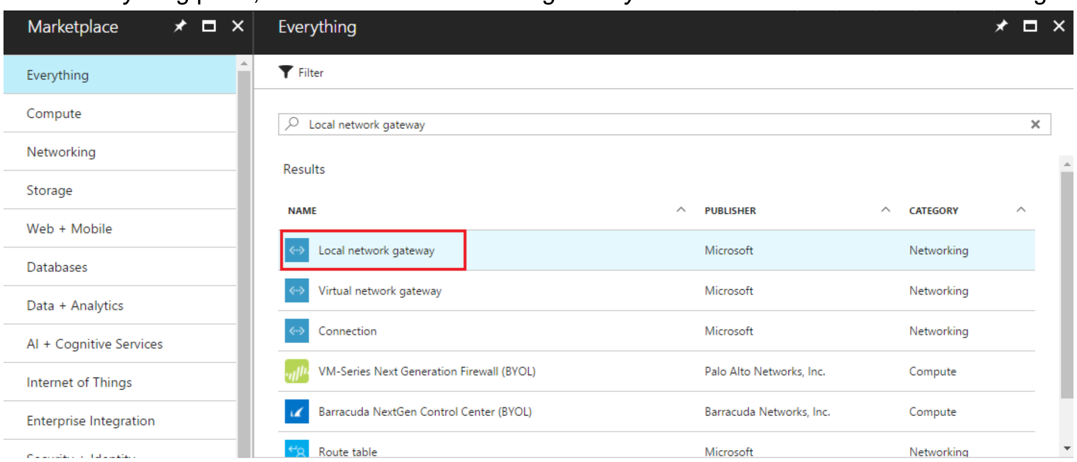


To configure the Azure local network gateway:

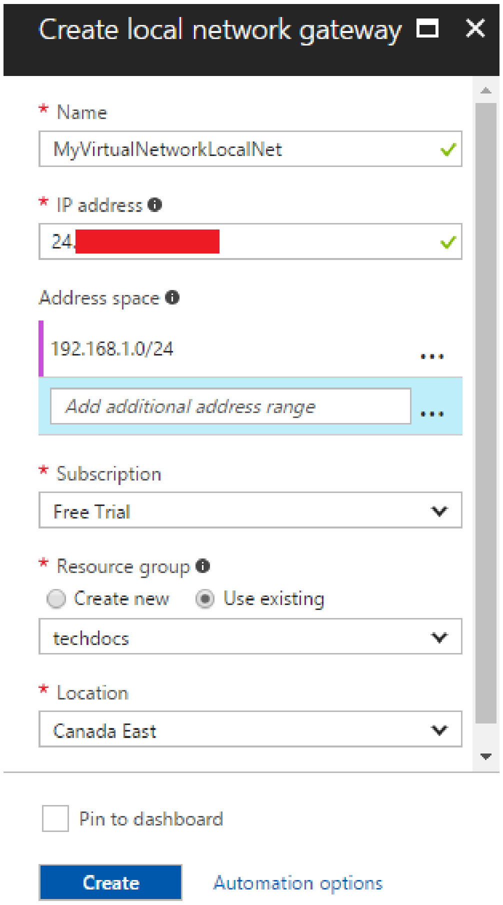
1. In the portal dashboard, click *All resources*.
2. Click *Add* and then click *See all*.



3. In the *Everything* pane, search for *Local network gateway* and then click *Create local network gateway*.



4. For the *IP address*, enter the local network gateway IP address, that is, the FortiGate's external IP address.



5. Set the remaining values for your local network gateway and click *Create*.

To configure the FortiGate tunnel:

1. In the FortiGate, go to *VPN > IP Wizard*.
2. Enter a *Name* for the tunnel, click *Custom*, and then click *Next*.
3. Configure the *Network* settings.
   a. For *Remote Gateway*, select *Static IP Address* and enter the IP address provided by Azure.
   b. For *Interface*, select *wan1*.
   c. For *NAT Traversal*, select *Disable*,
   d. For *Dead Peer Detection*, select *On Idle*.
   e. In the Authentication section, select
4. Configure the *Authentication* settings.
   a. For *Method*, select *Pre-shared Key* and enter the *Pre-shared Key*.
   b. For *IKE*, select 2.
5. Configure the *Phase 1 Proposal* settings.
   a. Set the Encryption and Authentication combination to the three supported encryption algorithm combinations accepted by Azure.
      - AES256 and SHA1
      - 3DES and SHA1

    - AES256 and SHA256
  b. For *Diffie-Hellman Groups*, select *2*.
  c. Set *Key Lifetime (seconds)* to *28800*.
6. In *Phase 2 Selectors*, expand the *Advanced* section to configure the *Phase 2 Proposal* settings.
    a. Set the Encryption and Authentication combinations:
        - AES256 and SHA1
        - 3DES and SHA1
        - AES256 and SHA256
    b. Uncheck *Enable Perfect Forward Secrecy (PFS)*.
    c. Set *Key Lifetime (seconds)* to *27000*.
7. Click *OK*.

To create the Azure firewall object:

1. In the FortiGate, go to *Policy & Objects > Addresses*.
2. Create a firewall object for the Azure VPN tunnel.

To create the FortiGate firewall policies:

1. In the FortiGate, go to *Policy & Objects > Firewall Policy*.
2. Create a policy for the site-to-site connection that allows outgoing traffic.
    a. Set the *Source* address and *Destination* address using the firewall objects you just created.
    b. Disable *NAT*.
3. Create another policy that allows incoming traffic.
    a. For this policy, reverse the *Source* address and *Destination* address.
4. We recommend limiting the TCP maximum segment size (MSS) being sent and received so as to avoid packet drops and fragmentation.
   To do this, use the following CLI commands on both policies.

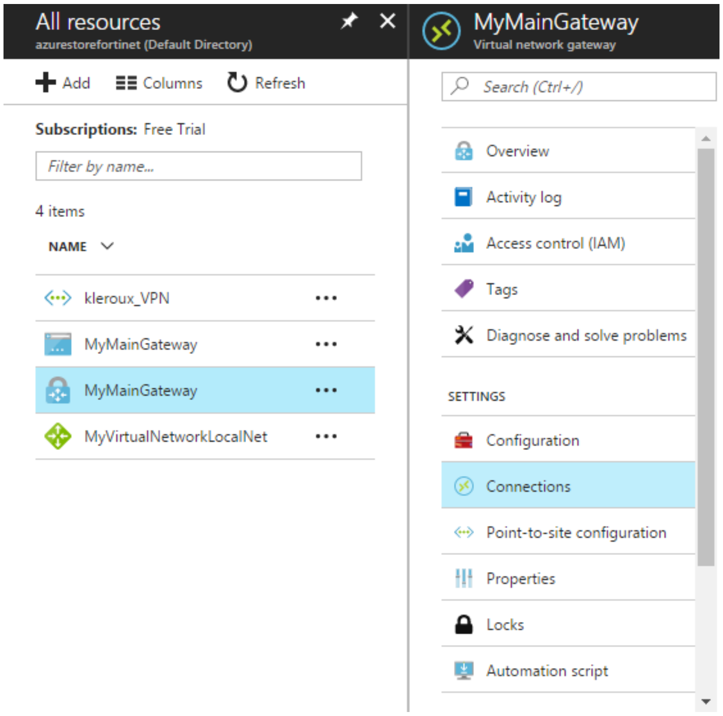
```
config firewall policy
   edit <policy-id>
      set tcp-mss-sender 1350
      set tcp-mss-receiver 1350
   next
end
```

To create the FortiGate static route:

1. In the FortiGate, go to *Network > Static Routes*.
2. Create an IPv4 Static Route that forces outgoing traffic going to Azure to go through the route-based tunnel.
3. Set the *Administrative Distance* to a value lower than the existing default route value.

To create the Azure site-to-site VPN connection:

1. In the Azure portal, locate and select your virtual network gateway.
2. In the *Settings* pane, click *Connections* and then click *Add*.

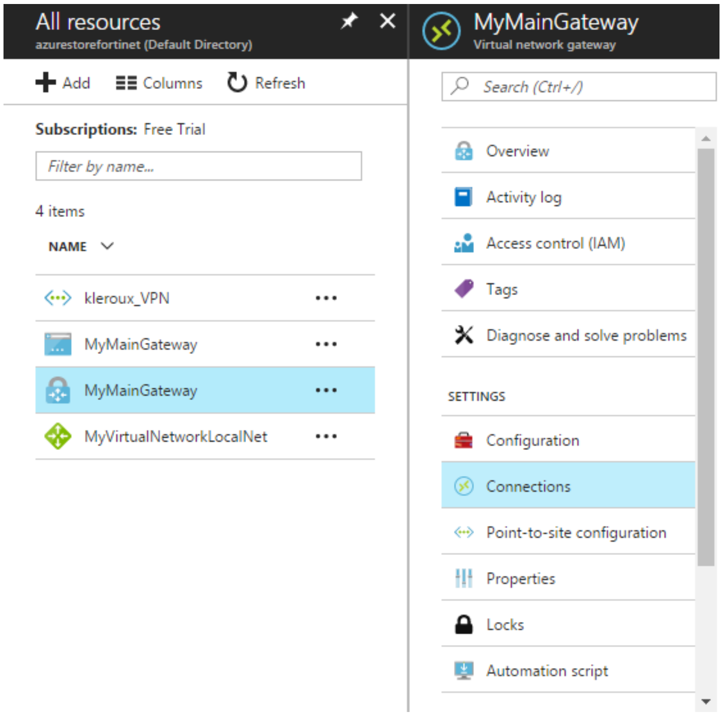


3. Enter the settings for your connection. Ensure the *Shared Key (PSK)* matches the *Pre-shared Key* for the FortiGate tunnel.

To check the results:

1. In the FortiGate, go to *Monitor > IPsec Monitor* and check that the tunnel is up. If the tunnel is down, right-click the tunnel and select *Bring Up*.
2. In the FortiGate, go to *Log & Report > System Events*.
   a. Select an event card to view more information and verify the connection.

3. In the Azure portal dashboard, click *All resources* and locate your virtual network gateway.
   a. In your virtual network gateway pane, click *Connections* to see the status of each connection.



   b. Click a connection to open the *Essentials* pane to view more information about that connection.
      - If the connection is successful, the *Status* shows *Connected*.
      - See the *ingress* and *egress* bytes to confirm traffic flowing through the tunnel.

IPsec VPN to an Azure with virtual WAN

This is a sample configuration of an IPsec site-to-site VPN connection between an on-premise FortiGate and an Azure virtual network (VNet). This example uses Azure virtual WAN (vWAN) to establish the VPN connection.

- Azure must use IPsec v2 for this configuration.
- Azure uses overlapped subnet IP addresses for the IPsec interfaces.

To configure IKEv2 IPsec site-to-site VPN to an Azure VPN gateway:

1. In the Azure management portal, configure vWAN-related settings as described in Tutorial: Create a Site-to-Site connection using Azure Virtual WAN.
   If a custom BGP IP address is configured on Azure's vWAN, such as 169.254.21.6 and 169.254.21.7, you must configure the FortiGate `remote-IP` to the corresponding *Custom BGP IP Address* value. If a custom BGP IP address is not configured, FortiGate `remote-IPs` should point to the *Default BGP IP Address* value.
2. Download the VPN configuration. The following shows an example VPN configuration:


```
[ {"configurationVersion":{"LastUpdatedTime":"2019-07-
     16T22:16:28.0409002Z","Version":"be5c5787-b903-43b1-a237-
     49eae1b373e4"},"vpnSiteConfiguration":
     {"Name":"toaws","IPAddress":"3.220.252.93","BgpSetting":
```

{"Asn":7225,"BgpPeeringAddress":"169.254.24.25","PeerWeight":32768},"LinkName":"toa
ws"},"vpnSiteConnections":[{"hubConfiguration":
{"AddressSpace":"10.1.0.0/16","Region":"West US","ConnectedSubnets":
["10.2.0.0/16"]},"gatewayConfiguration":{"IpAddresses":
{"Instance0":"52.180.90.47","Instance1":"52.180.89.94"},"BgpSetting":
{"Asn":65515,"BgpPeeringAddresses":
{"Instance0":"10.1.0.7","Instance1":"10.1.0.6"},"PeerWeight":0}},"connectionConfigu
ration":{"IsBgpEnabled":true,"PSK":"Fortinet123#","IPsecParameters":
{"SADataSizeInKilobytes":102400000,"SALifeTimeInSeconds":3600}}}]}]

3. Configure the following on the FortiGate. Note for `set proposal`, you can select from several proposals.

```
config vpn ipsec phase1-interface
    edit "toazure1"
        set interface "port1"
        set ike-version 2
        set keylife 28800
        set peertype any
        set proposal aes256-sha1
        set dhgrp 2
        set remote-gw 52.180.90.47
        set psksecret **********
    next
    edit "toazure2"
        set interface "port1"
        set ike-version 2
        set keylife 28800
        set peertype any
        set proposal aes256-sha1
        set dhgrp 2
        set remote-gw 52.180.89.94
        set psksecret **********
    next
end
config vpn ipsec phase2-interface
    edit "toazure1"
        set phase1name "toazure1"
        set proposal aes256-sha1
        set dhgrp 2
        set keylifeseconds 3600
    next
    edit "toazure2"
        set phase1name "toazure2"
        set proposal aes256-sha1
        set dhgrp 2
        set keylifeseconds 3600
    next
end
config system settings
    set allow-subnet-overlap enable
end
config system interface
    edit "toazure1"
        set vdom "root"
        set ip 169.254.24.25 255.255.255.255
        set type tunnel
        set remote-ip 10.1.0.7 255.255.255.255
        set snmp-index 4
        set interface "port1"
```

```
    next
    edit "toazure2"
        set vdom "root"
        set ip 169.254.24.25 255.255.255.255
        set type tunnel
        set remote-ip 10.1.0.6 255.255.255.255
        set snmp-index 5
        set interface "port1"
    next
end
config router bgp
    set as 7225
    set router-id 169.254.24.25
    config neighbor
        edit "10.1.0.7"
            set remote-as 65515
        next
        edit "10.1.0.6"
            set remote-as 65515
        next
    end
    config network
        edit 1
            set prefix 172.30.101.0 255.255.255.0
        next
    end
    config redistribute "connected"
        set status enable
    end
    config redistribute "rip"
    end
    config redistribute "ospf"
    end
    config redistribute "static"
    end
    config redistribute "isis"
    end
    config redistribute6 "connected"
    end
    config redistribute6 "rip"
    end
    config redistribute6 "ospf"
    end
    config redistribute6 "static"
    end
    config redistribute6 "isis"
    end
end
```

4. Run `diagnose vpn tunnel list`. If the configuration was successful, the output should resemble the following:

```
name=toazure1 ver=2 serial=3 172.30.1.83:4500->52.180.90.47:4500 tun_id=52.180.90.47
bound_if=3 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/0
proxyid_num=1 child_num=0 refcnt=15 ilast=16 olast=36 ad=/0
stat: rxp=41 txp=41 rxb=5104 txb=2209
dpd: mode=on-demand on=1 idle=20000ms retry=3 count=0 seqno=1
```

```
natt: mode=keepalive draft=0 interval=10 remote_port=4500
proxyid=toazure1 proto=0 sa=1 ref=2 serial=4
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=3 options=10226 type=00 soft=0 mtu=8926 expire=2463/0B replaywin=2048
       seqno=2a esn=0 replaywin_lastseq=00000029 itn=0
  life: type=01 bytes=0/0 timeout=3300/3600
  dec: spi=c13f7928 esp=aes key=32
009a86bb0d6f5fee66af7b8232c8c0f22e6ec5c61ba19c93569bd0cd115910a9
       ah=sha1 key=20 f05bfeb0060afa89d4afdfac35960a8a7a4d4856
  enc: spi=b40a6c70 esp=aes key=32
a1e361075267ba72b39924c5e6c766fd0b08e0548476de2792ee72057fe60d1d
       ah=sha1 key=20 b1d24bedb0eb8fbd26de3e7c0b0a3a799548f52f
  dec:pkts/bytes=41/2186, enc:pkts/bytes=41/5120
------------------------------------------------------
name=toazure2 ver=2 serial=4 172.30.1.83:4500->52.180.89.94:4500 tun_id=52.180.89.94
bound_if=3 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/0
proxyid_num=1 child_num=0 refcnt=16 ilast=16 olast=16 ad=/0
stat: rxp=40 txp=40 rxb=4928 txb=2135
dpd: mode=on-demand on=1 idle=20000ms retry=3 count=0 seqno=1
natt: mode=keepalive draft=0 interval=10 remote_port=4500
proxyid=toazure2 proto=0 sa=1 ref=2 serial=4
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=3 options=10626 type=00 soft=0 mtu=8926 expire=2427/0B replaywin=2048
       seqno=29 esn=0 replaywin_lastseq=00000028 itn=0
  life: type=01 bytes=0/0 timeout=3299/3600
  dec: spi=c13f791d esp=aes key=32
759898cbb7fafe448116b1fb0fb6d2f0eb99621ea6ed8dd4417ffdb901eb82be
       ah=sha1 key=20 533ec5dc8a1910221e7742b12f9de1b41205622c
  enc: spi=67934bfe esp=aes key=32
9b5710bfb4ba784722241ec371ba8066629febcd75da6f8471915bdeb874ca80
       ah=sha1 key=20 5099fed7edac2b960294094f1a8188ab42f34d7b
  dec:pkts/bytes=40/2087, enc:pkts/bytes=40/4976
```

```
Routing table for VRF=0
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default

S*      0.0.0.0/0 [5/0] via 172.30.1.1, port1
B       10.1.0.0/16 [20/0] via 10.1.0.6, toazure2, 00:15:01
C       10.1.0.6/32 is directly connected, toazure2
C       10.1.0.7/32 is directly connected, toazure1
B       10.2.0.0/16 [20/0] via 10.1.0.6, toazure2, 00:15:01
C       169.254.24.25/32 is directly connected, toazure1
                         is directly connected, toazure2
C       172.30.1.0/24 is directly connected, port1
C       172.30.101.0/24 is directly connected, port2
```

IPSec VPN between a FortiGate and a Cisco ASA with multiple subnets

When a Cisco ASA unit has multiple subnets configured, multiple phase 2 tunnels must be created on the FortiGate to allocate to each subnet (rather than having multiple subnets on one phase 2 tunnel).

The FortiGate uses the same SPI value to bring up the phase 2 negotiation for all of the subnets, while the Cisco ASA expects different SPI values for each of its configured subnets. Using multiple phase 2 tunnels on the FortiGate creates different SPI values for each subnet.

To configure multiple phase 2 interfaces in route-based mode:

```
config vpn ipsec phase2-interface
    edit "First subnet"
        set phase1name "VPN to Cisco"
        set src-subnet 192.168.227.253 255.255.255.255
        set dst-subnet 10.142.0.0 255.255.254.0
    next
    edit "Second subnet"
        set phase1name "VPN to Cisco"
        set src-subnet 192.168.227.253 255.255.255.255
        set dst-subnet 10.143.0.0 255.255.254.0
    next
end
```

To configure multiple phase 2 interfaces in policy-based mode:

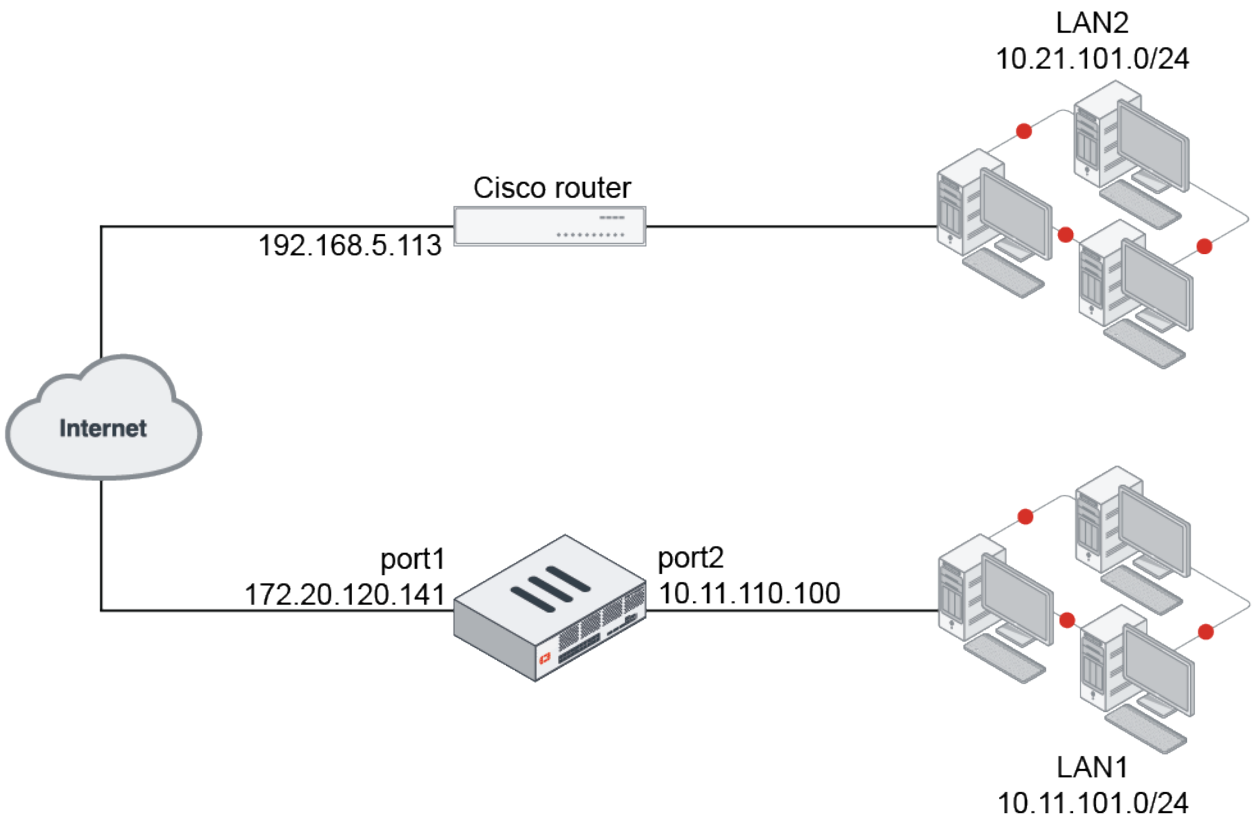
```
config vpn ipsec phase2
    edit "First subnet"
        set phase1name "VPN to Cisco"
        set src-subnet 192.168.227.253 255.255.255.255
        set dst-subnet 10.142.0.0 255.255.254.0
    next
    edit "Second subnet"
        set phase1name "VPN to Cisco"
        set src-subnet 192.168.227.253 255.255.255.255
        set dst-subnet 10.143.0.0 255.255.254.0
    next
end
```

Cisco GRE-over-IPsec VPN

This is a sample configuration of a FortiGate VPN that is compatible with Cisco-style VPNs that use GRE in an IPsec tunnel. Cisco products with VPN support often use the GRE protocol tunnel over IPsec encryption. Cisco VPNs can use either transport mode or tunnel mode IPsec.

Topology

In this example, LAN1 users are provided with access to LAN2.



Configuring the FortiGate

There are five steps to configure GRE-over-IPsec with a FortiGate and Cisco router:

1. Enable overlapping subnets.
2. Configure a route-based IPsec VPN on the external interface.
3. Configure a GRE tunnel on the virtual IPsec interface.
4. Configure security policies.
5. Configure the static route.

Enabling overlapping subnets

Overlapping subnets are required because the IPsec and GRE tunnels will use the same addresses. By default, each FortiGate network interface must be on a separate network. This configuration assigns an IPsec tunnel endpoint and the external interface to the same network.

To enable overlapping subnets:

```
config system settings
    set allow-subnet-overlap enable
    next
end
```

Configuring a route-based IPsec VPN

A route-based VPN that use encryption and authentication algorithms compatible with the Cisco router is required. Pre-shared key authentication is used in this configuration.

To configure route-based IPsec in the GUI:

1. Go to *VPN > IPsec Wizard* and select the *Custom* template.
2. Enter the tunnel name (*tocisco*) and click *Next*.
3. Enter the following:

Remote Gateway	Static IP Address
IP Address	Cisco router public interface (192.168.5.113)
Interface	FortiGate public interface (172.20.120.141)
Authentication Method	Pre-shared Key
Pre-shared Key	Entry must match the pre-shared key on the Cisco router
Mode	Main (ID Protection)
Phase 1 Proposal	3DES-SHA1, AES128-SHA1 (at least one proposal must match the settings on the Cisco router)
Local Address	GRE local tunnel endpoint IP address (172.20.120.141)
Remote Address	GRE remote tunnel endpoint IP address (192.168.5.113)
Phase 2 Proposal	3DES-MD5 (at least one proposal must match the settings on the Cisco router)
Local Port	0
Remote Port	0
Protocol	47

4. Click *OK*.
5. If the Cisco router is configured to use transport mode IPsec, configure transport mode on the FortiGate:

```
config vpn phase2-interface
    edit tocisco_p2
        set encapsulation transport-mode
    next
end
```

To configure route-based IPsec in the CLI:

```
config vpn ipsec phase1-interface
    edit tocisco
        set interface port1
        set proposal 3des-sha1 aes128-sha1
        set remote-gw 192.168.5.113
        set psksecret xxxxxxxxxxxxxxxx
    next
end
```

```
config vpn ipsec phase2-interface
    edit tocisco_p2
        set phase1name tocisco
        set proposal 3des-md5
        set encapsulation [tunnel-mode | transport-mode]
        set protocol 47
        set src-addr-type ip
        set dst-start-ip 192.168.5.113
        set src-start-ip 172.20.120.141
    next
end
```

To add the IPsec tunnel end addresses:

```
config system interface
    edit tocisco
        set ip 172.20.120.141 255.255.255.255
        set remote-ip 192.168.5.113
    next
end
```

Configuring the GRE tunnel

The local gateway and remote gateway addresses must match the local and remote gateways of the IPsec tunnel. The GRE tunnel runs between the virtual IPsec public interface on the FortiGate unit and the Cisco router.

To configure the GRE tunnel:

```
config system gre-tunnel
    edit gre1
        set interface tocisco
        set local-gw 172.20.120.141
        set remote-gw 192.168.5.113
        set keepalive-interval <integer>
        set keepalive-failtimes <integer>
    next
end
```

The Cisco router configuration requires an address for its end of the GRE tunnel, so you need to add the tunnel end addresses.

To add the tunnel end addresses:

```
config system interface
    edit gre1
        set ip 10.0.1.1 255.255.255.255
        set remote-ip 10.0.1.2
    next
end
```

Configuring the security policies

Two sets of security policies are required:

- Policies to allow traffic to pass in both directions between the GRE virtual interface and the IPsec virtual interface.
- Policies to allow traffic to pass in both directions between the protected network interface and the GRE virtual interface.

To configure security policies in the GUI:

1. Go to *Policy & Objects > Firewall Policy* and click *Create New*.
2. Enter the following to allow traffic between the protected network and the GRE tunnel:

Name	LANtoGRE
Incoming Interface	Interface that connects to the private network behind the FortiGate (port2)
Outgoing Interface	GRE tunnel virtual interface (gre1)
Source	All
Destination	All
Action	ACCEPT
NAT	Disable

3. Click *OK*.
4. Create a new policy and enter the following to allow traffic between the GRE tunnel and the protected network:

Name	GREtoLAN
Incoming Interface	GRE tunnel virtual interface (gre1)
Outgoing Interface	Interface that connects to the private network behind the FortiGate (port2)
Source	All
Destination	All
Action	ACCEPT
NAT	Disable

5. Click *OK*.
6. Create a new policy and enter the following to allow traffic between the GRE virtual interface and the IPsec virtual interface:

Name	GREtoIPsec
Incoming Interface	GRE tunnel virtual interface (gre1)
Outgoing Interface	Virtual IPsec interface (tocisco)
Source	All
Destination	All
Action	ACCEPT
NAT	Disable

7. Click *OK*.

8. Create a new policy and enter the following to allow traffic between the IPsec virtual interface and the GRE virtual interface:

Name	IPsectoGRE
Incoming Interface	Virtual IPsec interface (tocisco)
Outgoing Interface	GRE tunnel virtual interface (gre1)
Source	All
Destination	All
Action	ACCEPT
NAT	Disable

9. Click *OK*.

To configure security policies in the CLI:

```
config firewall policy
    edit 1
        set name LANtoGRE
        set srcintf port2
        set dstintf gre1
        set srcaddr all
        set dstaddr all
        set action accept
        set schedule always
        set service ALL
    next
    edit 2
        set name GREtoLAN
        set srcintf gre1
        set dstintf port2
        set srcaddr all
        set dstaddr all
        set action accept
        set schedule always
        set service ALL
    next
    edit 3
        set name GREtoIPsec
        set srcintf gre1
        set dstintf tocisco
        set srcaddr all
        set dstaddr all
        set action accept
        set schedule always
        set service ALL
    next
    edit 4
        set name IPsectoGRE
        set srcintf tocisco
        set dstintf gre1
        set srcaddr all
        set dstaddr all
```

```
        set action accept
        set schedule always
        set service ALL
    next
end
```

Configuring routing

to direct traffic destined for the network behind the Cisco router into the GRE-over-IPsec tunnelTraffic destined for the network behind the Cisco router must be routed to the GRE tunnel. To do this, create a static route

To create the static route in the GUI:

1. Go to *Network > Static Routes* and click *Create New*.
2. Enter the following:

Destination	IP and netmask for the network behind the Cisco router (10.21.101.0 255.255.255.0)
Interface	GRE tunnel virtual interface (gre1)
Administrative Distance	Leave the default setting

3. Click *OK*.

To create the static route in the CLI:

```
config router static
    edit 0
        set device gre1
        set dst 10.21.101.0 255.255.255.0
    next
end
```

Configuring the Cisco router

For more information, refer to Configuring and verifying a GRE over IPsec tunnel in the Fortinet Knowledge Base.

Remote access

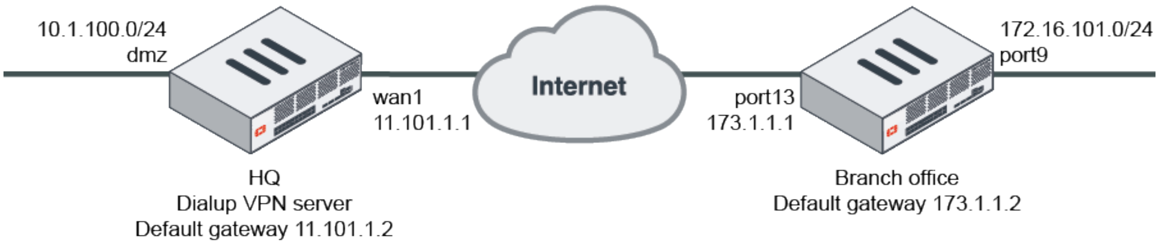
Remote access lets users connect to the Internet using a dialup connection over traditional POTS or ISDN telephone lines. Virtual private network (VPN) protocols are used to secure these private connections.

The following topics provide instructions on configuring remote access:

- FortiGate as dialup client on page 1412
- FortiClient as dialup client on page 1418
- Add FortiToken multi-factor authentication on page 1422
- Add LDAP user authentication on page 1423
- iOS device as dialup client on page 1424
- IKE Mode Config clients on page 1428
- IPsec VPN with external DHCP service on page 1433

FortiGate as dialup client

This is a sample configuration of dialup IPsec VPN and the dialup client. In this example, a branch office FortiGate connects via dialup IPsec VPN to the HQ FortiGate.



You can configure dialup IPsec VPN with FortiGate as the dialup client using the GUI or CLI.

To configure IPsec VPN with FortiGate as the dialup client in the GUI:

1. Configure the dialup VPN server FortiGate:
   a. Go to *VPN > IPsec Wizard* and configure the following settings for *VPN Setup*:
      i. Enter a VPN name.
      ii. For *Template Type*, select *Site to Site*.
      iii. For *Remote Device Type*, select *FortiGate*.
      iv. For *NAT Configuration*, select *The remote site is behind NAT*.
      v. Click *Next*.
   b. Configure the following settings for *Authentication*:
      i. For *Incoming Interface*, select the incoming interface.
      ii. For *Authentication Method*, select *Pre-shared Key*.
      iii. In the *Pre-shared Key* field, enter *your-psk* as the key.
      iv. Click *Next*.
   c. Configure the following settings for *Policy & Routing*:
      i. From the *Local Interface* dropdown menu, select the local interface.
      ii. Configure the *Local Subnets* as *10.1.100.0/24*.
      iii. Configure the *Remote Subnets* as *172.16.101.0/24*.
      iv. Click *Create*.
2. Configure the dialup VPN client FortiGate:
   a. Go to *VPN > IPsec Wizard* and configure the following settings for *VPN Setup*:
      i. Enter a VPN name.
      ii. For *Template Type*, select *Site to Site*.
      iii. For *Remote Device Type*, select *FortiGate*.
      iv. For *NAT Configuration*, select *This site is behind NAT*.
      v. Click *Next*.

b. Configure the following settings for *Authentication*:
   i. For *IP Address*, enter *11.101.1.1*.
   ii. For *Outgoing Interface*, select *port13*.
   iii. For *Authentication Method*, select *Pre-shared Key*.
   iv. In the *Pre-shared Key* field, enter *your-psk* as the key.
   v. Click *Next*.
c. Configure the following settings for *Policy & Routing*:
   i. From the *Local Interface* dropdown menu, select the local interface. In this example, it is *port9*.
   ii. Configure the *Local Subnets* as *172.16.101.0*.
   iii. Configure the *Remote Subnets* as *10.1.100.0*.
   iv. Click *Create*.

To configure IPsec VPN with FortiGate as the dialup client in the CLI:

1. In the CLI, configure the user, user group, and firewall address. Only the HQ dialup server FortiGate needs this configuration. The address is an IP pool to assign an IP address for the dialup client FortiGate.

```
config user local
    edit "vpnuser1"
        set type password
        set passwd your-password
    next
end
config user group
    edit "vpngroup"
        set member "vpnuser1"
    next
end
config firewall address
    edit "client_range"
        set type iprange
        set start-ip 10.10.10.1
        set end-ip 10.10.10.200
    next
end
```

2. Configure the WAN interface and default route. The WAN interface is the interface connected to the ISP. It can work in static mode (as shown in this example), DHCP, or PPPoE mode. The IPsec tunnel is established over the WAN interface.
   a. Configure the HQ FortiGate.

```
config system interface
    edit "wan1"
        set vdom "root"
        set ip 11.101.1 255.255.255.0
    next
end
config router static
    edit 1
        set gateway 11.101.1.2
        set device "wan1"
    next
end
```

b. Configure the branch office FortiGate.

```
config system interface
    edit "port13"
        set vdom "root"
        set ip 173.1.1.1 255.255.255.0
    next
end
config router static
    edit 1
        set gateway 173.1.1.2
        set device "port13"
    next
end
```

3. Configure the internal interface and protected subnet. The internal interface connects to the internal network. Traffic from this interface will route out the IPsec VPN tunnel.
   a. Configure the HQ FortiGate.

```
config system interface
    edit "dmz"
        set vdom "root"
        set ip 10.1.100.1 255.255.255.0
    next
end
config firewall address
    edit "10.1.100.0"
        set subnet 10.1.100.0 255.255.255.0
    next
end
```

   b. Configure the branch office FortiGate.

```
config system interface
    edit "port9"
        set vdom "root"
        set ip 172.16.101.1 255.255.255.0
    next
end
config firewall address
    edit "172.16.101.0"
        set subnet 172.16.101.0 255.255.255.0
    next
end
```

4. Configure the IPsec phase1-interface. In this example, PSK is used as the authentication method. Signature authentication is also an option.
   a. Configure the HQ FortiGate.

```
config vpn ipsec phase1-interface
    edit "for_Branch"
        set type dynamic
        set interface "wan1"
        set mode aggressive
        set peertype any
        set mode-cfg enable
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set add-route disable
```

```
            set dpd on-idle
            set xauthtype auto
            set authusrgrp "vpngroup"
            set net-device enable
            set assign-ip-from name
            set dns-mode auto
            set ipv4-split-include "10.1.100.0"
            set ipv4-name "client_range"
            set save-password enable
            set psksecret sample
            set dpd-retryinterval 60
        next
    end
```

b. Configure the branch office FortiGate.

```
config vpn ipsec phase1-interface
    edit "to_HQ"
        set interface "port13"
        set mode aggressive
        set peertype any
        set mode-cfg enable
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set add-route disable
        set xauthtype client
        set authusr "vpnuser1"
        set authpasswd vpnuser1-password
        set remote-gw 11.101.1.1
        set psksecret sample
    next
end
```

5. Configure the IPsec phase2-interface.
   a. Configure the HQ FortiGate:

```
config vpn ipsec phase2-interface
    edit "for_Branch_p2"
        set phase1 name "for_Branch"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
aes256gcm chacha20poly1305
    next
end
```

   b. Configure the branch office FortiGate.

```
config vpn ipsec phase2-interface
    edit "to_HQ_p2"
        set phase1name "to_HQ"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
aes256gcm chacha20poly1305
    next
end
```

6. Configure the static routes on the branch office FortiGate. The blackhole route is important to ensure that IPsec traffic does not match the default route when the IPsec tunnel is down.

```
config router static
    edit 2
```

```
        set dst 10.1.100.0 255.255.255.0
        set device "to_HQ"
    next
    edit 3
        set dst 10.1.100.0 255.255.255.0
        set blackhole enable
        set distance 254
    next
end
```

7. Configure the firewall policy to allow the branch office to HQ network flow over the IPsec tunnel. This configuration only supports traffic from the branch office FortiGate to the HQ FortiGate. Traffic is dropped from the HQ FortiGate to the branch office FortiGate.
   a. Configure the HQ FortiGate.

```
config firewall policy
    edit 1
        set name "inbound"
        set srcintf "for_Branch"
        set dstintf "dmz"
        set srcaddr "172.16.101.0"
        set dstaddr "10.1.100.0"
        set action accept
        set schedule "always"
        set service "ALL"
    next
end
```

   b. Configure the branch office FortiGate.

```
config firewall policy
    edit 1
        set name "outbound"
        set srcintf "port9"
        set dstintf "to_HQ"
        set srcaddr "172.16.101.0"
        set dstaddr "10.1.100.0"
        set action accept
        set schedule "always"
        set service "ALL"
    next
end
```

8. Run `diagnose` commands to check the IPsec phase1/phase2 interface status. The `diagnose debug application ike -1` command is the key to troubleshoot why the IPsec tunnel failed to establish.
   a. Run the `diagnose vpn ike gateway list` command on the HQ FortiGate. The system should return the following:

```
vd: root/0
name: for_Branch_0
version: 1
interface: wan1 5
addr: 11.101.1.1:500 -> 173.1.1.1:500
created: 1972s ago
xauth-user: vpnuser1
assigned IPv4 address: 10.10.10.1/255.255.255.252
IKE SA: created 1/1 established 1/1 time 10/10/10 ms
```

```
IPsec SA: created 1/1 established 1/1 time 0/0/0 ms
id/spi: 184 5b1c59fab2029e43/bf517e686d3943d2
direction: responder
status: established 1972-1972s ago = 10ms
proposal: aes128-sha256
key: 8046488e92499247-fbbb4f6dfa4952d0
lifetime/rekey: 86400/84157
DPD sent/recv: 00000020/00000000
```

b. Run the `diagnose vpn tunnel list` command on the HQ FortiGate. The system should return the following:

```
list all ipsec tunnel in vd 0
name=for_Branch_0 ver=1 serial=9 11.101.1.1:0->173.1.1.1:0 tun_id=173.1.1.1
bound_if=5 lgwy=static/1 tun=intf/0 mode=dial_inst/3 encap=none/208 options
[00d0]=create_dev no-sysctlrgwy-chg
parent=for_Branch index=0
proxyid_num=1 child_num=0 refcnt=12 ilast=8 olast=8 ad=/0
stat: rxp=8 txp=8 rxb=1216 txb=672
dpd: mode=on-idle on=1 idle=60000ms retry=3 count=0 seqno=31
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=for_Branch_p2 proto=0 sa=1 ref=2 serial=1
src: 0:0.0.0.0-255.255.255.255:0
dst: 0:0.0.0.0-255.255.255.255:0
SA: ref=3 options=226 type=00 soft=0 mtu=1438 expire=41297/0B replaywin=2048 seqno=9
esn=0 replaywin_lastseq=00000009 itn=0
life: type=01 bytes=0/0 timeout=43190/43200
dec: spi=747c10c6 esp=aes key=16 278c2430e09e74f1e229108f906603b0
ah=sha1 key=20 21dad76b008d1e8b8e53148a2fcbd013a277974a
enc: spi=ca646448 esp=aes key=16 b7801d125804e3610a556da7caefd765
ah=sha1 key=20 a70164c3094327058bd84c1a0c954ca439709206
dec:pkts/bytes=8/672, enc:pkts/bytes=8/1216

name=for_Branchver=1 serial=6 11.101.1.1:0->0.0.0.0:0 tun_id=1.0.0.0
bound_if=5 lgwy=static/1 tun=intf/0 mode=dialup/2 encap=none/16 options[0010]=create_
dev
proxyid_num=0 child_num=1 refcnt=14 ilast=8523 olast=8523 ad=/0
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-idle on=0 idle=60000ms retry=3 count=0 seqno=0
natt: mode=none draft=0 interval=0 remote_port=0
run_tally=0
```

c. Run the `diagnose vpn ike gateway list` command on the branch office FortiGate. The system should return the following:

```
vd: root/0
name: to_HQ
version: 1
interface: port13 42
addr: 173.1.1.1:500 -> 11.101.1.1:500
created: 2016s ago
assigned IPv4 address: 10.10.10.1/255.255.255.252
IKE SA: created 1/1 established 1/1 time 0/0/0 ms
IPsec SA: created 1/1 established 1/1 time 0/0/0 ms
id/spi: 93 5b1c59fab2029e43/bf517e686d3943d2
direction: initiator
status: established 2016-2016s ago = 0ms
proposal: aes128-sha256
```

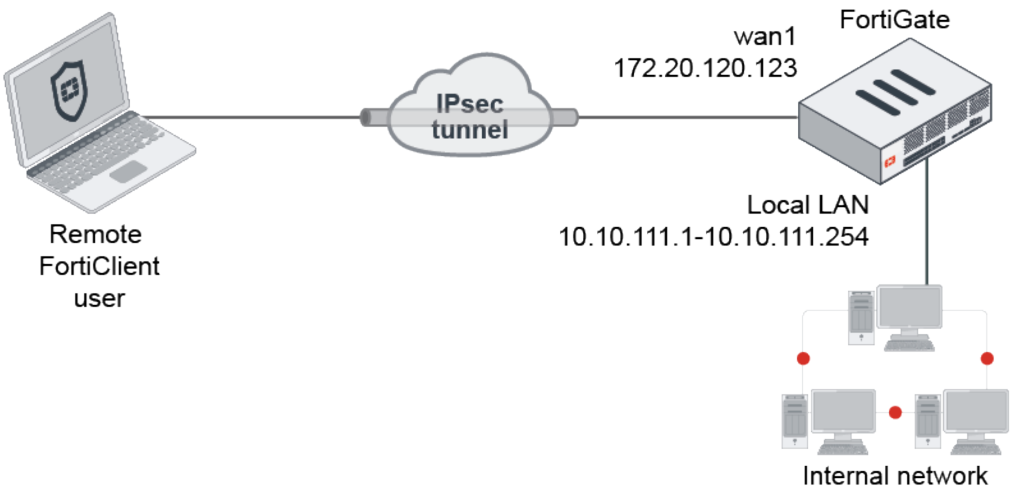
```
key: 8046488e92499247-fbbb4f6dfa4952d0
lifetime/rekey: 86400/84083
DPD sent/recv: 00000000/00000020
```

d. Run the `diagnose vpn tunnel list` command on the branch office FortiGate. The system should return the following:

```
list all ipsec tunnel in vd 0
name=to_HQver=1 serial=7 173.1.1.1:0->11.101.1.1:0 tun_id=11.101.1.1
bound_if=42 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/8 options[0008]=npu
proxyid_num=1 child_num=0 refcnt=13 ilast=18 olast=58 ad=/0
stat: rxp=1 txp=2 rxb=152 txb=168
dpd: mode=on-demand on=1 idle=20000ms retry=3 count=0 seqno=0
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=to_HQ proto=0 sa=1 ref=2 serial=1
src: 0:0.0.0.0/0.0.0.0:0
dst: 0:0.0.0.0/0.0.0.0:0
SA: ref=6 options=10226 type=00 soft=0 mtu=1438 expire=41015/0B replaywin=2048
seqno=3 esn=0 replaywin_lastseq=00000002 itn=0
life: type=01 bytes=0/0 timeout=42898/43200
dec: spi=ca646448 esp=aes key=16 b7801d125804e3610a556da7caefd765
ah=sha1 key=20 a70164c3094327058bd84c1a0c954ca439709206
enc: spi=747c10c6 esp=aes key=16 278c2430e09e74f1e229108f906603b0
ah=sha1 key=20 21dad76b008d1e8b8e53148a2fcbd013a277974a
dec:pkts/bytes=1/84, enc:pkts/bytes=2/304
npu_flag=03 npu_rgwy=11.101.1.1 npu_lgwy=173.1.1.1 npu_selid=5 dec_npuid=2 enc_
npuid=2
```

FortiClient as dialup client

This is a sample configuration of dialup IPsec VPN with FortiClient as the dialup client.



You can configure dialup IPsec VPN with FortiClient as the dialup client using the GUI or CLI.

If multiple dialup IPsec VPNs are defined for the same dialup server interface, each phase1 configuration must define a unique peer ID to distinguish the tunnel that the remote client is connecting to. When a client connects, the first IKE message that is in aggressive mode contains the client's local ID. FortiGate matches the local ID to the dialup tunnel referencing the same Peer ID, and the connection continues with that tunnel.

To configure IPsec VPN with FortiClient as the dialup client on the GUI:

1. Configure a user and user group.
   a. Go to *User & Authentication > User Definition* to create a local user *vpnuser1*.
   b. Go to *User & Authentication > User Groups* to create a group *vpngroup* with the member *vpnuser1*.
2. Go to *VPN > IPsec Wizard* and configure the following settings for *VPN Setup*:
   a. Enter a VPN name.
   b. For *Template Type*, select *Remote Access*.
   c. For *Remote Device Type*, select *Client-based > FortiClient*.
   d. Click *Next*.
3. Configure the following settings for *Authentication*:
   a. For *Incoming Interface*, select *wan1*.
   b. For *Authentication Method*, select *Pre-shared Key*.
   c. In the *Pre-shared Key* field, enter *your-psk* as the key.
   d. From the *User Group* dropdown list, select *vpngroup*.
   e. Click *Next*.
4. Configure the following settings for *Policy & Routing*:
   a. From the *Local Interface* dropdown menu, select *lan*.
   b. Configure the *Local Address* as *local_network*.
   c. Configure the *Client Address Range* as *10.10.2.1-10.10.2.200*.
   d. Keep the default values for the *Subnet Mask*, *DNS Server*, *Enable IPv4 Split tunnel*, and *Allow Endpoint Registration*.
   e. Click *Next*.
5. Adjust the *Client Options* as needed, then click *Create*.
6. Optionally, define a unique Peer ID in the phase1 configuration:
   a. Go to *VPN > IPsec Tunnels* and edit the just created tunnel.
   b. Click *Convert To Custom Tunnel*.
   c. In the *Authentication* section, click *Edit*.
   d. Under *Peer Options*, set *Accept Types* to *Specific peer ID*.
   e. In the *Peer ID* field, enter a unique ID, such as *dialup1*.
   f. Click *OK*.

To configure IPsec VPN with FortiClient as the dialup client using the CLI:

1. In the CLI, configure the user and group.

```
config user local
    edit "vpnuser1"
        set type password
        set passwd your-password
    next
end
config user group
    edit "vpngroup"
        set member "vpnuser1"
    next
end
```

2. Configure the internal interface. The LAN interface connects to the corporate internal network. Traffic from this interface routes out the IPsec VPN tunnel. Creating an address group for the protected network behind this

FortiGate causes traffic to this network group to go through the IPsec tunnel.

```
config system interface
    edit "lan"
        set vdom "root"
        set ip 10.10.111.1 255.255.255.0
    next
end
config firewall address
    edit "local_subnet_1"
        set subnet 10.10.111.0 255.255.255.0
    next
    edit "local_subnet_2"
        set subnet 10.10.112.0 255.255.255.0
    next
end
config firewall addrgrp
    edit "local_network"
        set member "local_subnet_1" "local_subnet_2"
    next
end
```

3. Configure the WAN interface. The WAN interface is the interface connected to the ISP. It can work in static mode (as shown in this example), DHCP, or PPPoE mode. The IPsec tunnel is established over the WAN interface.

```
config system interface
    edit "wan1"
        set vdom "root"
        set ip 172.20.120.123 255.255.255.0
    next
end
```

4. Configure the client address pool. You must create a firewall address to assign an IP address to a client from the address pool.

```
config firewall address
    edit "client_range"
        set type iprange
        set comment "VPN client range"
        set start-ip 10.10.2.1
        set end-ip 10.10.2.200
    next
end
```

5. Configure the IPsec phase1-interface. In this example, PSK is used as the authentication method. Signature authentication is also an option.

```
config vpn ipsec phase1-interface
    edit "for_client"
        set type dynamic
        set interface "wan1"
        set mode aggressive
        set peertype one
        set peerid "dialup1"
        set net-device enable
        set mode-cfg enable
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set dpd on-idle
        set xauthtype auto
```

```
            set authusrgrp "vpngroup"
            set assign-ip-from name
            set ipv4-name "client_range"
            set dns-mode auto
            set ipv4-split-include "local_network"
            set save-password enable
            set psksecret your-psk
            set dpd-retryinterval 60
        next
    end
```

6. Configure the IPsec phase2-interface.

```
config vpn ipsec phase2-interface
    edit "for_client"
        set phase1name "for_client"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
aes256gcm chacha20poly1305
    next
end
```

7. Configure the firewall policy to allow client traffic flow over the IPsec VPN tunnel.

```
config firewall policy
    edit 1
        set name "inbound"
        set srcintf "for_client"
        set dstintf "lan"
        set srcaddr "client_range"
        set dstaddr "local_network"
        set action accept
        set schedule "always"
        set service "ALL"
    next
end
```

To configure FortiClient:

1. In FortiClient, go to *Remote Access* and click *Add a new connection*.
2. Set the *VPN* to *IPsec VPN* and the *Remote Gateway* to the FortiGate IP address.
3. Set the *Authentication Method* to *Pre-Shared Key* and enter the key.
4. Expand *Advanced Settings* > *Phase 1* and in the *Local ID* field, enter *dialup1*.
5. Configure remaining settings as needed, then click *Save*.
6. Select the VPN, enter the username and password, then select *Connect*.

Diagnose the connection

Run `diagnose` commands to check the IPsec phase1/phase2 interface status. The `diagnose debug application ike -1` command is the key to troubleshoot why the IPsec tunnel failed to establish.

1. Run the `diagnose vpn ike gateway list` command. The system should return the following:

```
vd: root/0
name: for_client_0
version: 1
interface: port1 15
```

```
addr: 172.20.120.123:4500 ->172.20.120.254:64916
created: 37s ago
xauth-user: vpnuser1
assigned IPv4 address: 10.10.1.1/255.255.255.255
nat: me peer
IKE SA: created 1/1 established 1/1 time 10/10/10 ms
IPsec SA: created 1/1 established 1/1 time 0/0/0 ms
id/spi: 1 b40a32d878d5e262/8bba553563a498f4
direction: responder
status: established 37-37s ago = 10ms
proposal: aes256-sha256
key: f4ad7ec3a4fcfd09-787e2e9b7bceb9a7-0dfa183240d838ba-41539863e5378381
lifetime/rekey: 86400/86092
DPD sent/recv: 00000000/00000a0e
```

2. Run the `diagnose vpn tunnel list` command. The system should return the following:

```
list all ipsec tunnel in vd 0
=
=
name=for_client_0 ver=1 serial=3 172.20.120.123:4500->172.20.120.254:64916 tun_
id=172.20.120.254
bound_if=15 lgwy=static/1 tun=intf/0 mode=dial_inst/3 encap=none/984 options
[03d8]=npucreate_dev no-sysctlrgwy-chgrport-chg frag-rfcaccept_traffic=1
parent=for_client index=0
proxyid_num=1 child_num=0 refcnt=12 ilast=3 olast=3 ad=/0
stat: rxp=1 txp=0 rxb=16402 txb=0
dpd: mode=on-idle on=1 idle=20000ms retry=3 count=0 seqno=0
natt: mode=keepalive draft=32 interval=10 remote_port=64916
proxyid=for_client proto=0 sa=1 ref=2 serial=1 add-route
src: 0:0.0.0.0-255.255.255.255:0
dst: 0:10.10.1.1-10.10.1.1:0
SA: ref=4 options=2a6 type=00 soft=0 mtu=1422 expire=42867/0B replaywin=2048
seqno=1 esn=0 replaywin_lastseq=00000001 itn=0
life: type=01 bytes=0/0 timeout=43189/43200
dec: spi=36274d14 esp=aes key=16 e518b84b3c3b667b79f2e61c64a225a6
ah=sha1 key=20 9cceaa544ed042fda800c4fe5d3fd9d8b811984a
enc: spi=8b154deb esp=aes key=16 9d50f004b45c122e4e9fb7af085c457c
ah=sha1 key=20 f1d90b2a311049e23be34967008239637b50a328
dec:pkts/bytes=1/16330, enc:pkts/bytes=0/0
npu_flag=02 npu_rgwy=172.20.120.254 npu_lgwy=172.20.120.123npu_selid=0 dec_npuid=2 enc_
npuid=0
name=for_clientver=1 serial=2 172.20.120.123:0->0.0.0.0:0
bound_if=15 lgwy=static/1 tun=intf/0 mode=dialup/2 encap=none/536 options
[0218]=npucreate_dev frag-rfcaccept_traffic=1
proxyid_num=0 child_num=1 refcnt=11 ilast=350 olast=350 ad=/0
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-idle on=0 idle=20000ms retry=3 count=0 seqno=0
natt: mode=none draft=0 interval=0 remote_port=0
```

Add FortiToken multi-factor authentication

This configuration adds multi-factor authentication (MFA) to the FortiClient dialup VPN configuration (FortiClient as dialup client on page 1418). It uses one of the two free mobile FortiTokens that is already installed on the FortiGate.

To configure MFA using the GUI:

1. Edit the user:
   a. Go to *User & Authentication > User Definition* and edit local user *vpnuser1*.
   b. Enable *Two-factor Authentication*.
   c. For *Authentication Type*, click *FortiToken* and select one mobile *Token* from the list.
   d. Enter the user's *Email Address*.
   e. Enable *Send Activation Code* and select *Email*.
   f. Click *Next* and click *Submit*.
2. Activate the mobile token.
   a. When a FortiToken is added to user *vpnuser1*, an email is sent to the user's email address. Follow the instructions to install your FortiToken mobile application on your device and activate your token.

To configure MFA using the CLI:

1. Edit the user and user group:

```
config user local
    edit "vpnuser1"
        set type password
        set two-factor fortitoken
        set fortitoken <select mobile token for the option list>
        set email-to <user's email address>
        set passwd <user's password>
    next
end
```

2. Activate the mobile token.
   a. When a FortiToken is added to user *vpnuser1*, an email is sent to the user's email address. Follow the instructions to install your FortiToken mobile application on your device and activate your token.

Add LDAP user authentication

This configuration adds LDAP user authentication to the FortiClient dialup VPN configuration (FortiClient as dialup client on page 1418). You must have already generated and exported a CA certificate from your AD server.

To configure LDAP user authentication using the GUI:

1. Import the CA certificate into FortiGate:
   a. Go to *System > Certificates*.
      If the *Certificates* option is not visible, enable it in *Feature Visibility*. See Feature visibility on page 2163 for details.
   b. Click *Import > CA Certificate*.
   c. Set *Type* to *File*.
   d. Click *Upload* then find and select the certificate file.
   e. Click *OK*.
      The CA certificate now appears in the list of *External CA Certificates*. In this example, it is called *CA_Cert_1*.

f. Optionally, rename the system generated *CA_Cert_1* to something more descriptive:

```
config vpn certificate ca
    rename CA_Cert_1 to LDAPS-CA
end
```

2. Configure the LDAP user:
   a. Go to *User & Authentication > LDAP Servers* and click *Create New*.
   b. Set *Name* to *ldaps-server* and specify *Server IP/Name*.
   c. Specify *Common Name Identifier* and *Distinguished Name*.
   d. Set *Bind Type* to *Regular*.
   e. Specify *Username* and *Password*.
   f. Enable *Secure Connection* and set *Protocol* to *LDAPS*.
   g. For *Certificate*, select *LDAP server CA LDAPS-CA* from the list.
   h. Click *OK*.
3. Add the LDAP user to the user group:
   a. Go to *User & Authentication > User Groups* and edit the *vpngroup* group.
   b. In *Remote Groups*, click *Add* to add the *ldaps-server* remote server.
   c. Click *OK*.

To configure LDAP user authentication using the CLI:

1. Import the CA certificate using the GUI.
2. Configure the LDAP user:

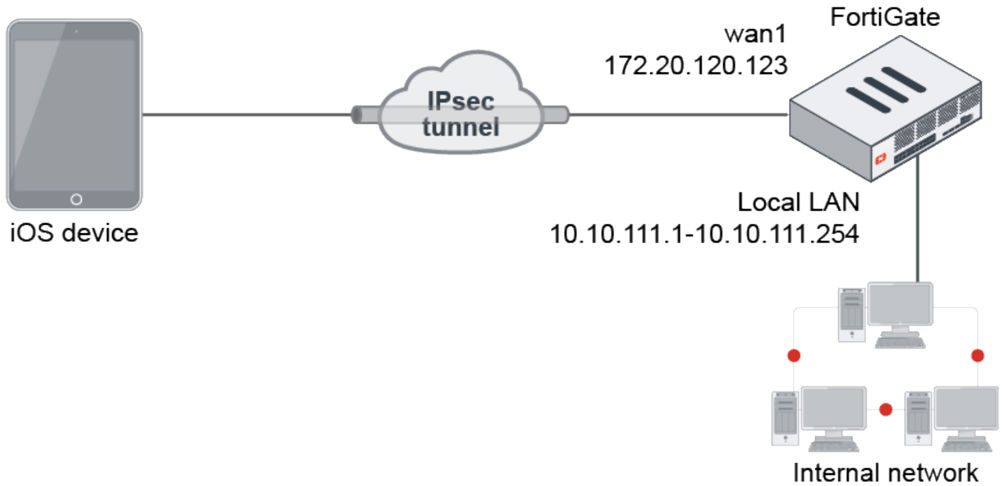
```
config user ldap
    edit "ldaps-server"
        set server "172.20.120.161"
        set cnid "cn"
        set dn "cn=Users,dc=qa,dc=fortinet,dc=com"
        set type regular
        set username "CN=Administrator,cn=users,DC=qa,DC=fortinet,DC=com"
        set password **********
        set group-member-check group-object
        set secure ldaps
        set ca-cert "LDAPS-CA"
        set port 636
    next
end
```

3. Add the LDAP user to the user group:

```
config user group
    edit "vpngroup"
        append member "ldaps-server"
    next
end
```

iOS device as dialup client

This is a sample configuration of dialup IPsec VPN with an iPhone or iPad as the dialup client.



You can configure dialup IPsec VPN with an iOS device as the dialup client using the GUI or CLI.

To configure IPsec VPN with an iOS device as the dialup client on the GUI:

1. Go to *VPN > IPsec Wizard* and configure the following settings for *VPN Setup*:
   a. Enter a VPN name.
   b. For *Template Type*, select *Remote Access*.
   c. For *Remote Device Type*, select *Native > iOS Native*.
   d. For *NAT Configuration*, set *No NAT Between Sites*.
   e. Click *Next*.
2. Configure the following settings for *Authentication*:
   a. For *Incoming Interface*, select *wan1*.
   b. For *Authentication Method*, select *Pre-shared Key*.
   c. In the *Pre-shared Key* field, enter *your-psk* as the key.
   d. From the *User Group* dropdown list, select *vpngroup*.
   e. Deselect *Require 'Group Name' on VPN client*.
   f. Click *Next*.
3. Configure the following settings for *Policy & Routing*:
   a. From the *Local Interface* dropdown menu, select *lan*.
   b. Configure the *Local Address* as *local_network*.
   c. Configure the *Client Address Range* as *10.10.2.1-10.10.2.200*.
   d. Keep the default values for the *Subnet Mask*, *DNS Server*, and *Enable IPv4 Split tunnel*.
   e. Click *Create*.

To configure IPsec VPN with an iOS device as the dialup client using the CLI:

1. In the CLI, configure the user and group.

```
config user local
    edit "vpnuser1"
        set type password
        set passwd your-password
    next
end
```

```
config user group
    edit "vpngroup"
        set member "vpnuser1"
    next
end
```

2. Configure the internal interface. The LAN interface connects to the corporate internal network. Traffic from this interface routes out the IPsec VPN tunnel. Creating an address group for the protected network behind this FortiGate causes traffic to this network group to go through the IPsec tunnel.

```
config system interface
    edit "lan"
        set vdom "root"
        set ip 10.10.111.1 255.255.255.0
    next
end

config firewall address
    edit "local_subnet_1"
        set ip 10.10.111.0 255.255.255.0
    next
end

config firewall address
    edit "local_subnet_2"
        set ip 10.10.112.0 255.255.255.0
    next
end

config firewall addrgrp
    edit "local_network"
        set member "local_subnet_1" "local_subnet_2"
    next
end
```

3. Configure the WAN interface. The WAN interface is the interface connected to the ISP. It can work in static mode (as shown in this example), DHCP, or PPPoE mode. The IPsec tunnel is established over the WAN interface.

```
config system interface
    edit "wan1"
        set vdom "root"
        set ip 172.20.120.123 255.255.255.0
    next
end
```

4. Configure the client address pool. You must create a firewall address to assign an IP address to a client from the address pool.

```
config firewall address
    edit "client_range"
        set type iprange
        set comment "VPN client range"
        set start-ip 10.10.2.1
        set end-ip 10.10.2.200
    next
end
```

5. Configure the IPsec phase1-interface. In this example, PSK is used as the authentication method. Signature authentication is also an option.

```
config vpn ipsec phase1-interface
    edit "for_ios_p1"
        set type dynamic
        set interface "wan1"
        set peertype any
        set net-device enable
        set mode-cfg enable
        set proposal aes256-sha256 aes256-md5 aes256-sha1
        set dpd on-idle
        set dhgrp 14 5 2
        set xauthtype auto
        set authusrgrp "vpngroup"
        set assign-ip-from name
        set ipv4-name "client_range"
        set dns-mode auto
        set ipv4-split-include "local_network"
        set psksecret your-psk
        set dpd-retryinterval 60
    next
end
```

6. Configure the IPsec phase2-interface.

```
config vpn ipsec phase2-interface
    edit "for_ios_p2"
        set phase1name "for_ios_p1"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
aes256gcm chacha20poly1305
        set pfs disable
        set keepalive enable
    next
end
```

7. Configure the firewall policy to allow client traffic flow over the IPsec VPN tunnel.

```
config firewall policy
    edit 1
        set name "ios_vpn"
        set srcintf "for_ios_p1"
        set dstintf "lan"
        set srcaddr "ios_range"
        set dstaddr "local_network"
        set action accept
        set schedule "always"
        set service "ALL"
    next
end
```

8. Configure the iOS device.
   a. In the iOS device, go to *Settings* > *General* > *VPN* and select *Add VPN Configuration*.
   b. Set the *Type* to *IPsec* and enter a *Description*. Set the *Server* to the FortiGate's Internet-facing interface, and enter the username in *Account*. Enter the user password, the preshared IPsec VPN secret, then select *Done*.
   c. Ensure that the IPsec VPN configuration is highlighted (indicated by a checkmark), and select the *Not Connected* button. The IPsec VPN connects with the user's credentials and secret. The status changes to *Connected*, and a VPN icon appears at the top of the screen.
9. Run `diagnose` commands to check the IPsec phase1/phase2 interface status. The `diagnose debug application ike -1` command is the key to troubleshoot why the IPsec tunnel failed to establish.

a. Run the `diagnose vpn ike gateway list` command. The system should return the following:

```
vd: root/0
name: for_ios_p1_0
version: 1
interface: port1 15
addr: 172.20.120.123:4500 -> 172.20.120.254:64916
created: 17s ago
xauth-user: u1
assigned IPv4 address: 10.10.2.1/255.255.255.255
nat: me peer
IKE SA: created 1/1 established 1/1 time 150/150/150 ms
IPsec SA: created 1/1 established 1/1 time 10/10/10 ms
id/spi: 2 3c844e13c75591bf/80c2db92c8d3f602 direction: responder status: established
17-17s ago = 150ms proposal: aes256-sha256 key: 0032ea5ee160d775-51f3bf1f9909101b-
b89c7b5a77a07784-2c92cf9c921801ac lifetime/rekey: 3600/3312 DPD sent/recv:
00000000/00000000
```

b. Run the `diagnose vpn tunnel list` command. The system should return the following:

```
list all ipsec tunnel in vd 0
=
=
name=for_ios_p1_0 ver=1 serial=172.20.120.123:4500->172.20.120.254:64916 tun_
id=172.20.120.254
bound_if=15 lgwy=static/1 tun=intf/0 mode=dial_inst/3 encap=none/984 options
[03d8]=npu create_dev no-sysctl rgwy-chg rport-chg frag-rfc accept_traffic=1
parent=for_ios_p1 index=0
proxyid_num=1 child_num=0 refcnt=12 ilast=23 olast=23 ad=/0
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-idle on=1 idle=60000ms retry=3 count=0 seqno=0
natt: mode=keepalive draft=32 interval=10 remote_port=64916
proxyid=for_ios_p1 proto=0 sa=1 ref=2 serial=1 add-route
src: 0:10.10.111.0-10.10.111.255:0 dst: 0:10.10.2.1-10.10.2.1:0 SA: ref=3 options=a7
type=00 soft=0 mtu=1422 expire=3564/0B replaywin=2048
seqno=1 esn=0 replaywin_lastseq=00000000 itn=0
life: type=01 bytes=0/0 timeout=3587/3600 dec: spi=36274d15 esp=aes key=32
5a599d796f8114c83d6589284f036fc33bdf4456541e2154b4ac2217b6aec869
ah=sha1 key=20 f1efdeb77d6f856a8dd3a30cbc23cb0f8a3e0340
enc: spi=00b0d9ab esp=aes key=32
e9232d7a1c4f390fd09f8409c2d85f80362d940c08c73f245908ab1ac3af322f
ah=sha1 key=20 a3890d6c5320756291cad85026d3a78fd42a1b42
dec:pkts/bytes=0/0, enc:pkts/bytes=0/0 npu_flag=00 npu_rgwy=172.20.120.254 npu_
lgwy=172.20.120.123 npu_selid=1 dec_npuid=0 enc_npuid=0
```

IKE Mode Config clients

IKE Mode Config is an alternative to DHCP over IPsec. It allows dialup VPN clients to obtain virtual IP address, network, and DNS configurations amongst others from the VPN server. A FortiGate can be configured as either an IKE Mode Config server or client.

IKE Mode Config can configure the host IP address, domain, DNS addresses ,and WINS addresses. IPsec parameters such as gateway address, encryption, and authentication algorithms must be configured. Several network equipment vendors support IKE Mode Config.

An IKE Mode Config server or client is configured using `config vpn ipsec phase1-interface` and involves the following parameters:

Parameter	Description
ike-version {1 | 2}	IKE v1 is the default for FortiGate IPsec VPNs. IKE Mode Config is also compatible with IKE v2.
mode-cfg {enable | disable}	Enable/disable IKE Mode Config.
type {static | dynamic | ddns}	If you set `type` to `dynamic`, an IKE Mode Config server is created. The other settings create an IKE Mode Config client.
assign-ip {enable | disable}	Enable to request an IP address from the server. This configuration is for IKE Mode Config clients only.
interface <interface_name>	Specify the physical, aggregate, or VLAN interface to which the IPsec tunnel will be bound.
proposal <encryption_ combination>	The encryption and authentication settings that the client will accept.
ip-version {4 | 6}	By default, IPsec VPNs use IPv4 addressing.
ipv4-split-include <string> ipv6-split-include <string>	Mode Config server configuration. Applicable to IKEv1 and IKEv2. Specify the firewall address or address group that represents the subnets that the clients will have access to. This information is sent to the clients so that default traffic should not flow over the IPsec tunnel except for the specified subnets.
split-include-service <string>	Mode Config server configuration. Applicable to IKEv1 and IKEv2. Specify the service or service group that represents the services that the clients will have access to. This information is sent to the clients so that default traffic should not flow over the IPsec tunnel except for the specified services.
ipv4-split-exclude <string> ipv6-split-exclude <string>	Specify the subnets that should not be accessed over the IPsec tunnel. This information is sent to the clients so that all default traffic should flow over the IPsec tunnel except for the specified subnets. See Split-exclude in IKEv1.

Creating an IKE Mode Config client

In this example, the FortiGate connects to a VPN gateway with a static IP address that can be reached through port 1. Only the port, gateway, and proposal information needs to be configured. All other configuration information will come from the IKE Mode Config server.

To configure an IKE Mode Config client:

```
config vpn ipsec phase1-interface
    edit vpn1
        set ip-version 4
        set type static
        set remote-gw <gw_address>
        set interface port1
        set proposal 3des-sha1 aes128-sha1
        set mode-cfg enable
```

```
        set assign-ip enable
    next
end
```

Split-exclude in IKEv1

The `split-exclude` option specifies that default traffic flows over the IPsec tunnel except for specified subnets. This is the opposite of `split-include`, which specifies that default traffic should not flow over the IPsec tunnel except for specified subnets. The `split-include` and `split-exclude` options can be specified at the same time.

To configure split-exclude:

```
config vpn ipsec phase1-interface
    edit <name>
        set ike-version 1
        set type dynamic
        set mode-cfg enable
        set ipv4-split-exclude <string>
        set ipv6-split-exclude <string>
    next
end
```

Creating an IKE Mode Config server

To configure IKE Mode config settings, the following must be configured first :

```
config vpn ipsec phase1-interface
    edit "vpn-p1"
        set type dynamic
        set interface <interface_name>
        set ike-version < 1 | 2 >
        set mode-cfg enable
        set proposal <encryption_combination>
        set ip-version < 4 | 6 >
    next
end
```

In this example, the FortiGate assigns IKE Mode Config clients addresses in the range of 10.11.101.160 - 10.11.101.180. DNS and WINS server addresses are also provided. The public interface of the FortiGate unit is port1.

When IKE Mode-Configuration is enabled, multiple server IPs can be defined in IPsec phase 1.

The `ipv4-split-include` parameter specifies a firewall address (`OfficeLAN`), which represents the networks that the clients will have access to. This destination IP address information is sent to the clients.

To configure an IKE Mode Config server:

```
config vpn ipsec phase1-interface
    edit "vpn-p1"
        set type dynamic
        set interface "wan1"
        set xauthtype auto
        set mode aggressive
        set mode-cfg enable
```

```
        set proposal 3des-sha1 aes128-sha1
        set dpd disable
        set dhgrp 2
        set authusrgrp "FG-Group1"
        set ipv4-start-ip 10.10.10.10
        set ipv4-end-ip 10.10.10.20
        set ipv4-dns-server1 1.1.1.1
        set ipv4-dns-server2 2.2.2.2
        set ipv4-dns-server3 3.3.3.3
        set ipv4-wins-server1 4.4.4.4
        set ipv4-wins-server2 5.5.5.5
        set domain "fgt1c-domain"
        set banner "fgt111C-banner"
        set backup-gateway "100.100.100.1" "host1.com" "host2"
        set ipv4-split-include OfficeLAN
    next
end
```

Assigning IP addresses

Once the basic configuration is enabled, you can configure IP address assignment for clients, as well as DNS and WINS server assignments. Usually you will want to assign IP addresses to clients. The easiest way is to assign addresses from a specific range, similar to a DHCP server.

To assign an IP from an address range:

```
config vpn ipsec phase1-interface
    edit vpn1
        set ip-version 4
        set assign-ip enable
        set assign-ip-from range
        set ipv4-start-ip <range_start>
        set ipv4-end-ip <range_end>
        set ipv4-netmask <netmask>
    next
end
```

To assign an IP from a named firewall address or group:

```
config vpn ipsec phase1-interface
    edit vpn1
        set type dynamic
        set assign-ip-from name
        set ipv4-name <name>
        set ipv6-name <name>
    next
end
```

RADIUS server

If the client is authenticated by a RADIUS server, you can obtain the user's IP address assignment from the Framed-IP-Address attribute. The user must be authenticated using XAuth.

The users must be authenticated by a RADIUS server and assigned to the FortiGate user group `<grp_name>`. Since the IP address is not static, type is set to dynamic and `mode-cfg` is enabled. With IKE Mode Config, compatible clients can configure themselves with settings provided by the FortiGate.

To assign an IP from a RADIUS server:

```
config vpn ipsec phase1-interface
    edit vpn1
        set type dynamic
        set mode-cfg enable
        set assign-ip enable
        set assign-ip-from usrgrp
        set xauthtype auto
        set authusrgrp <grp_name>
    next
end
```

DHCP server

IKE Mode Config can use a remote DHCP server to assign the client IP addresses. Up to eight server addresses can be selected for either IPv4 or IPv6. The DHCP proxy must be enabled first.

To assign an IP from a DHCP server:

```
config system settings
    set dhcp-proxy enable
    set dhcp-server-ip <address>
    set dhcp6-server-ip <address>
end
```

```
config vpn ipsec phase1-interface
    edit vpn1
        set mode-cfg enable
        set assign-ip-from dhcp
    next
end
```

Certificate groups

IKE certificate groups consisting of up to four RSA certificates can be used in IKE phase 1. Since CA and local certificates are global, the IKE daemon loads them once for all VDOMs and indexes them into trees based on subject and public key hash (for CA certificates), or certificate name (for local certificates). Certificates are linked together based on the issuer, and certificate chains are built by traversing these links. This reduces the need to keep multiple copies of certificates that could exist in multiple chains.

To configure the IKE local ID:

```
config vpn certificate local
    edit <name>
        set ike-localid <string>
        set ike-localid-type {asnldn | fqdn}
    next
end
```

Split-exclude in IKEv1

The `split-exclude` setting specifies that default traffic flows over the IPsec tunnel except for specified subnets. This is the opposite of `split-include`, which specifies that default traffic should not flow over the IPsec tunnel except for specified subnets. The `split-include` and `split-exclude` settings can be specified at the same time.

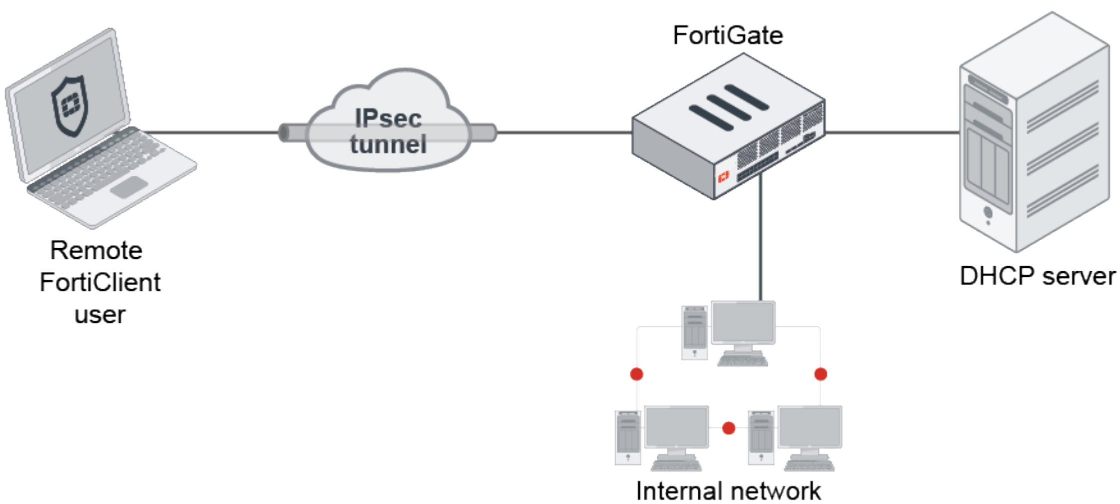
To configure split-exclude:

```
config vpn ipsec phase1-interface
    edit <name>
        set ike-version 1
        set type dynamic
        set mode-cfg enable
        set ipv4-split-exclude <string>
        set ipv6-split-exclude <string>
    next
end
```

IPsec VPN with external DHCP service

You can use an external DHCP server to assign IP addresses to your IPsec VPN clients. This is a common scenario found in enterprises where all DHCP leases need to be managed centrally.

In this example, the DHCP server assigns IP addresses in the range of 172.16.6.100 to 172.16.6.120. The server is attached to internal2 on the FortiGate and has an IP address of 192.168.3.70.



To configure a DHCP server to assign IP addresses to IPsec VPN clients:

1. Create a user group for remote users:
   a. Go to *User & Authentication > User Definition* and click *Create New*.
   b. For *User Type*, select *Local User*.
   c. Complete the wizard, and click *Submit*.
   d. Go to *User & Authentication > User Groups* and click *Create New*..
   e. Create a *Firewall* user group for your remote users.

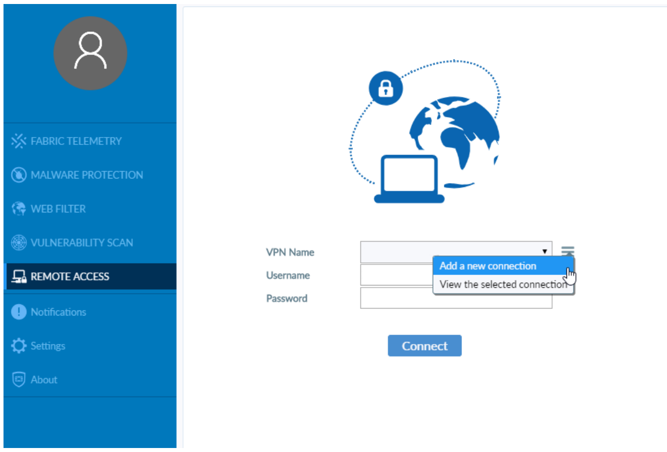
   f. For *Members*, add the user you just created.
   g. Click *OK*.
2. Add a firewall address for the local network and IPsec VPN client range:
   a. Go to *Policy & Objects* > *Addresses*.
   b. Create a new *Subnet* address for the LAN, including the IP mask and local interface (*internal2*).
   c. Click *OK*.
   d. Create a new *IP Range* address for the IPsec VPN client range (`172.16.6.100-172.16.6.120`).
   e. Click *OK*.
3. Configure the IPsec VPN using a VPN tunnel in the CLI:

```
config vpn ipsec phase1-interface
    edit "dhcp_vpn"
        set type dynamic
        set interface "wan1"
        set mode aggressive
        set peertype any
        set net-device disable
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set dpd on-idle
        set dhgrp 5
        set xauthtype auto
        set authusrgrp "ipsecvpn"
        set psksecret **********
        set dpd-retryinterval 60
    next
end

config vpn ipsec phase2-interface
    edit "toclient"
        set phase1name "dhcp_vpn"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
aes256gcm chacha20poly1305
        set dhgrp 5
        set dhcp-ipsec enable
    next
end
```

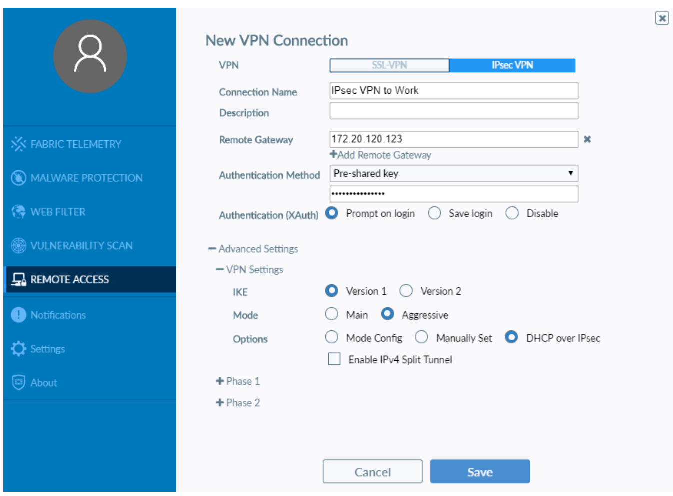
4. Configure the IPsec VPN interface:
   a. Go to *Network* > *Interfaces* and edit the newly created IPsec VPN interface.
   b. Enable the *DHCP Server*.
   c. Expand *Advanced* and change the *Mode* to *Relay*.
   d. Enter the external DHCP server IP address (`192.168.3.70`).
   e. Change the *Type* to *IPsec*.
   f. Click *OK*.
5. Create a security policy for access to the local network:
   a. Go to *Policy & Objects* > *Firewall Policy* and click *Create New*.
   b. Configure the following parameters:
      i. Set the *Incoming Interface* to the tunnel interface created in step 3 (*dhcp_vpn*).
      ii. Set the *Outgoing Interface* (*internal2*).
      iii. Set the *Source* to the IPsec VPN client range defined in step 2 (*ipsecvpn_range*).
      iv. Set the *Destination* to the subnet address defined in step 2 (*Local LAN*).
      v. Set the *Service* to *ALL*.

c. Click *OK*.

6. Configure FortiClient:
   a. In FortiClient, go to *REMOTE ACCESS > Add a new connection*.

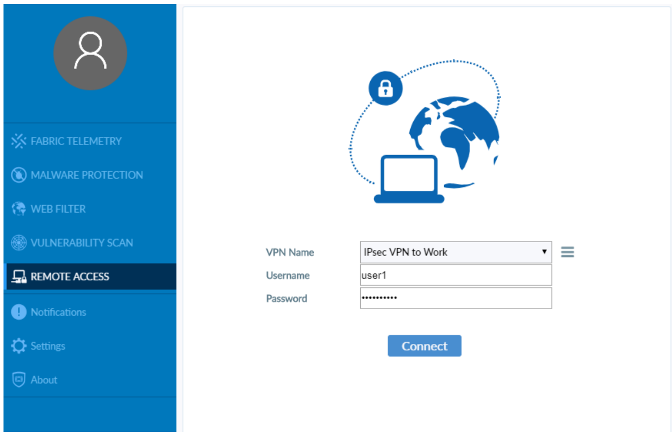


   b. Configure the following parameters:
      i. Set the *VPN* type to *IPsec VPN*.
      ii. Enter a connection name.
      iii. Set the *Remote Gateway* to the FortiGate external IP address.
      iv. Set the *Authentication Method* to *Pre-shared key* and enter the key below.
      v. Expand the *Advanced Settings > VPN Settings* and for *Options*, select *DHCP over IPsec*.
      vi. Click *Save*.



   c. Select the new connection, and enter the user name and password.

d. Click *Connect*.



Once the connection is established, the external DHCP server assigns the user an IP address and FortiClient displays the connection status, including the IP address, connection duration, and bytes sent and received.

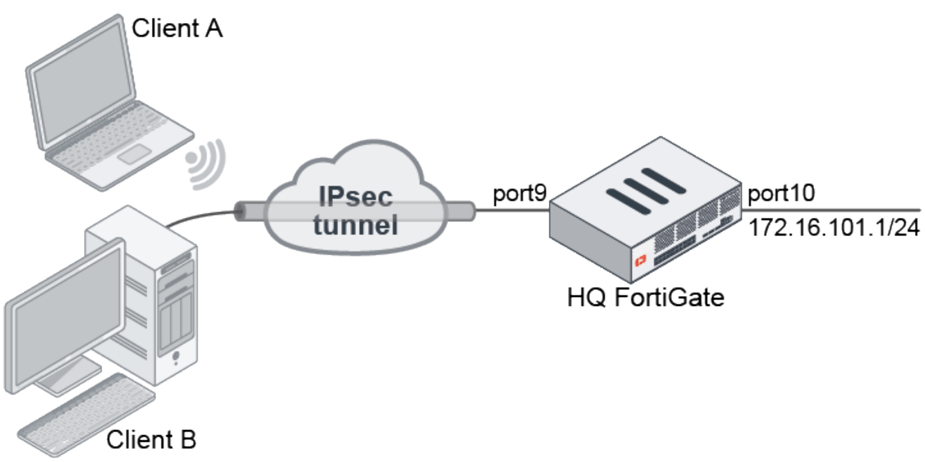
Verification

1. In FortiOS, go to *Monitor > IPsec Monitor* and verify that the tunnel *Status* is *Up*.
2. Go to *Log & Report > Forward Traffic* and verify the *Sent / Received* column displays the traffic flow through the tunnel.

L2TP over IPsec

This is an example of L2TP over IPsec.

This example uses a locally defined user for authentication, a Windows PC or Android tablet as the client, and `net-device` is set to `enable` in the `phase1-interface` settings. If `net-device` is set to `disable`, only one device can establish an L2TP over IPsec tunnel behind the same NAT device.



To configure L2TP over an IPsec tunnel using the GUI:

1. Go to *VPN > IPsec Wizard*.
2. Enter a VPN *Name*. In this example, *L2tpoIPsec*.

3. Configure the following settings for *VPN Setup*:
   a. For *Template Type*, select *Remote Access*.
   b. For *Remote Device Type*, select *Native* and *Windows Native*.
   c. Click *Next*.
4. Configure the following settings for *Authentication*:
   a. For *Incoming Interface*, select *port9*.
   b. For *Authentication Method*, select *Pre-shared Key*.
   c. In the *Pre-shared Key* field, enter *your-psk* as the key.
   d. For *User Group*, select *L2tpusergroup*
   e. Click *Next*.
5. Configure the following settings for *Policy & Routing*:
   a. From the *Local Interface* dropdown menu, select *port10*.
   b. Configure the *Local Address* as *172.16.101.0*.
   c. Configure the *Client Address Range* as *10.10.10.1-10.10.10.100*.
   d. Leave the *Subnet Mask* at its default value.
   e. Click *Create*.

To configure L2TP over an IPsec tunnel using the CLI:

1. Configure the WAN interface and static route on HQ.

```
config system interface
    edit "port9"
        set alias "WAN"
        set ip 22.1.1.1 255.255.255.0
    next
    edit "port10"
        set alias "Internal"
        set ip 172.16.101.1 255.255.255.0
    next
end
config router static
    edit 1
        set gateway 22.1.1.2
        set device "port9"
    next
end
```

2. Configure IPsec phase1-interface and phase2-interface on HQ.

```
config vpn ipsec phase1-interface
    edit "L2tpoIPsec"
        set type dynamic
        set interface "port9"
        set peertype any
        set proposal aes256-md5 3des-sha1 aes192-sha1
        set dpd on-idle
        set dhgrp 2
        set net-device enable
        set psksecret sample
        set dpd-retryinterval 60
    next
end
```

```
config vpn ipsec phase2-interface
    edit "L2tpoIPsec"
        set phase1name "L2tpoIPsec"
        set proposal aes256-md5 3des-sha1 aes192-sha1
        set pfs disable
        set encapsulation transport-mode
        set l2tp enable
    next
end
```

3. Configure a user and user group on HQ.

```
config user local
    edit "usera"
        set type password
        set passwd usera
        next
end
config user group
    edit "L2tpusergroup"
        set member "usera"
    next
end
```

4. Configure L2TP on HQ.

```
config vpn l2tp
    set status enable
    set eip 10.10.10.100
    set sip 10.10.10.1
    set usrgrp "L2tpusergroup"
end
```

5. Configure a firewall address that is applied in L2TP settings to assign IP addresses to clients once the L2TP tunnel is established.

```
config firewall address
    edit "L2TPclients"
        set type iprange
        set start-ip 10.10.10.1
        set end-ip 10.10.10.100
    next
end
```

6. Configure a firewall policy.

```
config firewall policy
    edit 1
        set name "Bridge_IPsec_port9_for_l2tp negotiation"
        set srcintf "L2tpoIPsec"
        set dstintf "port9"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "L2TP"
    next
    edit 2
        set srcintf "L2tpoIPsec"
```

```
            set dstintf "port10"
            set srcaddr "L2TPclients"
            set dstaddr "172.16.101.0"
            set action accept
            set schedule "always"
            set service "ALL"
            set nat enable
        next
    end
```

To view the VPN tunnel list on HQ:

```
# diagnose vpn tunnel list

list all ipsec tunnel in vd 0
----
name=L2tpoIPsec_0 ver=1 serial=8 22.1.1.1:0->10.1.100.15:0 tun_id=10.10.100.15
bound_if=4 lgwy=static/1 tun=intf/0 mode=dial_inst/3 encap=none/216 options[00d8]=npu
create_dev no-sysctl rgwy-chg
 parent=L2tpoIPsec index=0
proxyid_num=1 child_num=0 refcnt=13 ilast=0 olast=0 ad=/0
stat: rxp=470 txp=267 rxb=57192 txb=12679
dpd: mode=on-idle on=1 idle=60000ms retry=3 count=0 seqno=0
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=L2tpoIPsec proto=17 sa=1 ref=3 serial=1 transport-mode add-route
  src: 17:22.1.1.1-22.1.1.1:1701
  dst: 17:10.1.100.15-10.1.100.15:0
  SA:  ref=3 options=1a6 type=00 soft=0 mtu=1470 expire=2339/0B replaywin=2048
       seqno=10c esn=0 replaywin_lastseq=000001d6 itn=0
  life: type=01 bytes=0/0 timeout=3585/3600
  dec: spi=ca646443 esp=3des key=24 af62a0fffe85d3d534b5bfba29307aafc8bfda5c3f4650dc
       ah=sha1 key=20 89b4b67688bed9be49fb86449bb83f8c8d8d7432
  enc: spi=700d28a0 esp=3des key=24 5f68906eca8d37d853814188b9e29ac4913420a9c87362c9
       ah=sha1 key=20 d37f901ffd0e6ee1e4fdccebc7fdcc7ad44f0a0a
  dec:pkts/bytes=470/31698, enc:pkts/bytes=267/21744
  npu_flag=00 npu_rgwy=10.1.100.15 npu_lgwy=22.1.1.1 npu_selid=6 dec_npuid=0 enc_npuid=0
----
name=L2tpoIPsec_1 ver=1 serial=a 22.1.1.1:4500->22.1.1.2:64916 tun_id=22.1.1.2
bound_if=4 lgwy=static/1 tun=intf/0 mode=dial_inst/3 encap=none/472 options[01d8]=npu
create_dev no-sysctl rgwy-chg rport-chg
 parent=L2tpoIPsec index=1
proxyid_num=1 child_num=0 refcnt=17 ilast=2 olast=2 ad=/0
stat: rxp=5 txp=4 rxb=592 txb=249
dpd: mode=on-idle on=1 idle=60000ms retry=3 count=0 seqno=0
natt: mode=keepalive draft=32 interval=10 remote_port=64916
proxyid=L2tpoIPsec proto=17 sa=1 ref=3 serial=1 transport-mode add-route
  src: 17:22.1.1.1-22.1.1.1:1701
  dst: 17:22.1.1.2-22.1.1.2:0
  SA:  ref=3 options=1a6 type=00 soft=0 mtu=1454 expire=28786/0B replaywin=2048
       seqno=5 esn=0 replaywin_lastseq=00000005 itn=0
  life: type=01 bytes=0/0 timeout=28790/28800
  dec: spi=ca646446 esp=aes key=32
ea60dfbad709b3c63917c3b7299520ff7606756ca15d2eb7cbff349b6562172e
       ah=md5 key=16 2f2acfff0b556935d0aab8fc5725c8ec
  enc: spi=0b514df2 esp=aes key=32
a8a92c2ed0e1fd7b6e405d8a6b9eb3be5eff573d80be3f830ce694917d634196
```

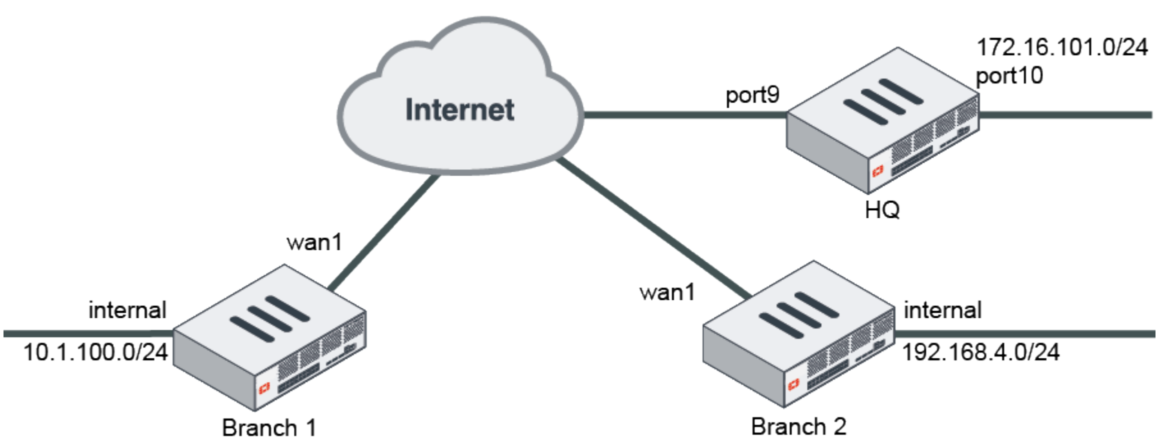
```
      ah=md5 key=16 e426c33a7fe9041bdc5ce802760e8a3d
  dec:pkts/bytes=5/245, enc:pkts/bytes=4/464
  npu_flag=00 npu_rgwy=22.1.1.2 npu_lgwy=22.1.1.1 npu_selid=8 dec_npuid=0 enc_npuid=0
```

To view the L2TP VPN status:

```
# diagnose debug enable
# diagnose vpn l2tp status
----
----

HQ # Num of tunnels: 2
----
Tunnel ID = 1 (local id), 42 (remote id)  to 10.1.100.15:1701
   control_seq_num = 2, control_rec_seq_num = 4,
   last recv pkt = 2
Call ID = 1 (local id), 1 (remote id), serno = 0, dev=ppp1,
      assigned ip = 10.10.10.2
      data_seq_num = 0,
      tx = 152 bytes (2), rx= 21179 bytes (205)
Tunnel ID = 3 (local id), 34183 (remote id)  to 22.1.1.2:58825
   control_seq_num = 2, control_rec_seq_num = 4,
   last recv pkt = 2
Call ID = 3 (local id), 18820 (remote id), serno = 2032472593, dev=ppp2,
      assigned ip = 10.10.10.3
      data_seq_num = 0,
      tx = 152 bytes (2), rx= 0 bytes (0)
----
--VD 0: Startip = 10.10.10.1, Endip = 10.10.10.100
        enforece-ipsec = false
----
```

Tunneled Internet browsing

This is a sample configuration of tunneled internet browsing using a dialup VPN. To centralize network management and control, all branch office traffic is tunneled to HQ, including Internet browsing.



To configure a dialup VPN to tunnel Internet browsing using the GUI:

1. Configure the dialup VPN server FortiGate at HQ:
   a. Go to *VPN > IPsec Wizard* and configure the following settings for *VPN Setup*:
      i. Enter a VPN name, in this example, *HQ*.
      ii. For *Template Type*, select *Site to Site*.
      iii. For *Remote Device Type*, select *FortiGate*.
      iv. For *NAT Configuration*, select *The remote site is behind NAT*.
      v. Click *Next*.
   b. Configure the following settings for *Authentication*:
      i. For *Incoming Interface*, select *port9*.
      ii. For *Authentication Method*, select *Pre-shared Key*.
      iii. In the *Pre-shared Key* field, enter *sample* as the key.
      iv. Click *Next*.
   c. Configure the following settings for *Policy & Routing*:
      i. From the *Local Interface* dropdown menu, select *port10*.
      ii. Configure the *Local Subnets* as *172.16.101.0*.
      iii. Configure the *Remote Subnets* as *0.0.0.0/0*.
      iv. For *Internet Access*, select *Share Local*.
      v. For *Shared WAN*, select *port9*.
      vi. Click *Create*.
2. Configure the dialup VPN client FortiGate at a branch:
   a. Go to *VPN > IPsec Wizard* and configure the following settings for *VPN Setup*:
      i. Enter a VPN name, in this example, *Branch1* or *Branch2*.
      ii. For *Template Type*, select *Site to Site*.
      iii. For *Remote Device Type*, select *FortiGate*.
      iv. For *NAT Configuration*, select *The remote site is behind NAT*.
      v. Click *Next*.
   b. Configure the following settings for *Authentication*:
      i. For *IP Address*, select *Remote Device* and enter *22.1.1.1*.
      ii. For *Outgoing Interface*, select *wan1*.
      iii. For *Authentication Method*, select *Pre-shared Key*.
      iv. In the *Pre-shared Key* field, enter *sample* as the key.
      v. Click *Next*.
   c. Configure the following settings for *Policy & Routing*:
      i. From the *Local Interface* dropdown menu, select *internal*.
      ii. Configure the *Local Subnets* as *10.1.100.0/192.1684.0*.
      iii. Configure the *Remote Subnets* as *0.0.0.0/0*.
      iv. For *Internet Access*, select *Use Remote*.
      v. Configure the *Local Gateway* to *15.1.1.1/13.1.1.1*.
      vi. Click *Create*.

To configure a dialup VPN to tunnel Internet browsing using the CLI:

1. Configure the WAN interface and static route on the FortiGate at HQ.

```
config system interface
    edit "port9"
        set alias "WAN"
        set ip 22.1.1.1 255.255.255.0
    next
    edit "port10"
        set alias "Internal"
        set ip 172.16.101.1 255.255.255.0
    next
end
config router static
    edit 1
        set gateway 22.1.1.2
        set device "port9"
    next
end
```

2. Configure IPsec phase1-interface and phase2-interface configuration at HQ.

```
config vpn ipsec phase1-interface
    edit "HQ"
        set type dynamic
        set interface "port9"
        set peertype any
        set net-device enable
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set dpd on-idle
        set psksecret sample
        set dpd-retryinterval 60
    next
end
config vpn ipsec phase2-interface
    edit "HQ"
        set phase1name "HQ"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
aes256gcm chacha20poly1305
    next
end
```

3. Configure the firewall policy at HQ.

```
config firewall policy
    edit 1
        set srcintf "HQ"
        set dstintf "port9" "port10"
        set srcaddr "10.1.100.0" "192.168.4.0"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set nat enable
    next
end
```

4. Configure the WAN interface and static route on the FortiGate at the branches.
   a. Branch1.

```
config system interface
    edit "wan1"
        set ip 15.1.1.2 255.255.255.0
    next
    edit "internal"
        set ip 10.1.100.1 255.255.255.0
    next
end
config router static
    edit 1
        set gateway 15.1.1.1
        set device "wan1"
    next
end
```

   b. Branch2.

```
config system interface
    edit "wan1"
        set ip 13.1.1.2 255.255.255.0
    next
    edit "internal"
        set ip 192.168.4.1 255.255.255.0
    next
end
config router static
    edit 1
        set gateway 13.1.1.1
        set device "wan1"
    next
end
```

5. Configure IPsec phase1-interface and phase2-interface configuration at the branches.
   a. Branch1.

```
config vpn ipsec phase1-interface
    edit "branch1"
        set interface "wan1"
        set peertype any
        set net-device enable
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set dpd on-idle
        set remote-gw 22.1.1.1
        set psksecret sample
        set dpd-retryinterval 5
    next
end
config vpn ipsec phase2-interface
    edit "branch1"
        set phase1name "branch1"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
aes256gcm chacha20poly1305
        set auto-negotiate enable
        set src-subnet 10.1.100.0 255.255.255.0
```

```
    next
end
```

b. Branch2.

```
config vpn ipsec phase1-interface
    edit "branch2"
        set interface "wan1"
        set peertype any
        set net-device enable
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set dpd on-idle
        set remote-gw 22.1.1.1
        set psksecret sample
        set dpd-retryinterval 5
    next
end
config vpn ipsec phase2-interface
    edit "branch2"
        set phase1name "branch2"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
aes256gcm chacha20poly1305
        set auto-negotiate enable
        set src-subnet 192.168.4.0 255.255.255.0
    next
end
```

6. Configure the firewall policy at the branches.

a. Branch1.

```
config firewall policy
    edit 1
        set name "outbound"
        set srcintf "internal"
        set dstintf "branch1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
    next
    edit 2
        set name "inbound"
        set srcintf "branch1"
        set dstintf "internal"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
    next
end
```

b. Branch2.

```
config firewall policy
    edit 1
        set name "outbound"
```

```
        set srcintf "internal"
        set dstintf "branch2"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
    next
    edit 2
        set name "inbound"
        set srcintf "branch2"
        set dstintf "internal"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
    next
end
```

7. Configure the static routes at the branches.
   a. Branch1.

```
config router static
    edit 2
        set dst 22.1.1.1/32
        set gateway 15.1.1.1
        set device "wan1"
        set distance 1
    next
    edit 3
        set device "branch1"
        set distance 5
    next
end
```

   b. Branch2.

```
config router static
    edit 2
        set dst 22.1.1.1/32
        set gateway 13.1.1.1
        set device "wan1"
        set distance 1
    next
    edit 3
        set device "branch2"
        set distance 5
    next
end
```

8. Optionally, view the VPN tunnel list on a branch with the `diagnose vpn tunnel list` command:

```
list all ipsec tunnel in vd 0
----
name=branch1 ver=1 serial=2 15.1.1.2:0->22.1.1.1:0 tun_id=22.1.1.1.1
bound_if=7 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/536 options[0218]=npu create_
dev frag-rfc  accept_traffic=1
```

```
proxyid_num=1 child_num=1 refcnt=19 ilast=0 olast=0 ad=r/2
stat: rxp=1 txp=1661 rxb=65470 txb=167314
dpd: mode=on-idle on=1 idle=5000ms retry=3 count=0 seqno=2986
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=branch1 proto=0 sa=1 ref=5 serial=1 auto-negotiate adr
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=6 options=1a227 type=00 soft=0 mtu=1438 expire=697/0B replaywin=1024
       seqno=13a esn=0 replaywin_lastseq=00000000 itn=0
  life: type=01 bytes=0/0 timeout=2368/2400
  dec: spi=c53a8f7e esp=aes key=16 ecee0cd48664d903d3d6822b1f902fd2
       ah=sha1 key=20 2440a189126c222093ca9acd8b37127285f1f8a7
  enc: spi=6e3636fe esp=aes key=16 fdaa20bcc96f74ae9885e824d3efa29d
       ah=sha1 key=20 70c0891c769ad8007ea1f31a39978ffbc73242d0
  dec:pkts/bytes=0/16348, enc:pkts/bytes=313/55962
  npu_flag=03 npu_rgwy=22.1.1.1 npu_lgwy=15.1.1.2 npu_selid=1 dec_npuid=1 enc_npuid=1
```

9. Optionally, view static routing table on a branch with the `get router info routing-table static` command:

```
Routing table for VRF=0
S*      0.0.0.0/0 [5/0] is directly connected, branch1
S*      22.1.1.1/32 [1/0] via 15.1.1.1, wan1
```

Dialup IPsec VPN with certificate authentication

In a dialup IPsec VPN setup, a company may choose to use X.509 certificates as their authentication solution for remote users. This method includes the option to verify the remote user using a user certificate, instead of a username and password. This method can be simpler for end users.

Administrators need to issue unique user certificates to each user for remote access management. The user certificate can be verified by the subject field, common name, or the principal name in the Subject Alternative Name (SAN) field.

Subject field verification

This is the basic method that verifies the subject string defined in the PKI user setting matches a substring in the subject field of the user certificate. For example:

```
config user peer
    edit "tgerber"
        set ca "CA_Cert_2"
        set subject "CN=tgerber"
    next
end
```

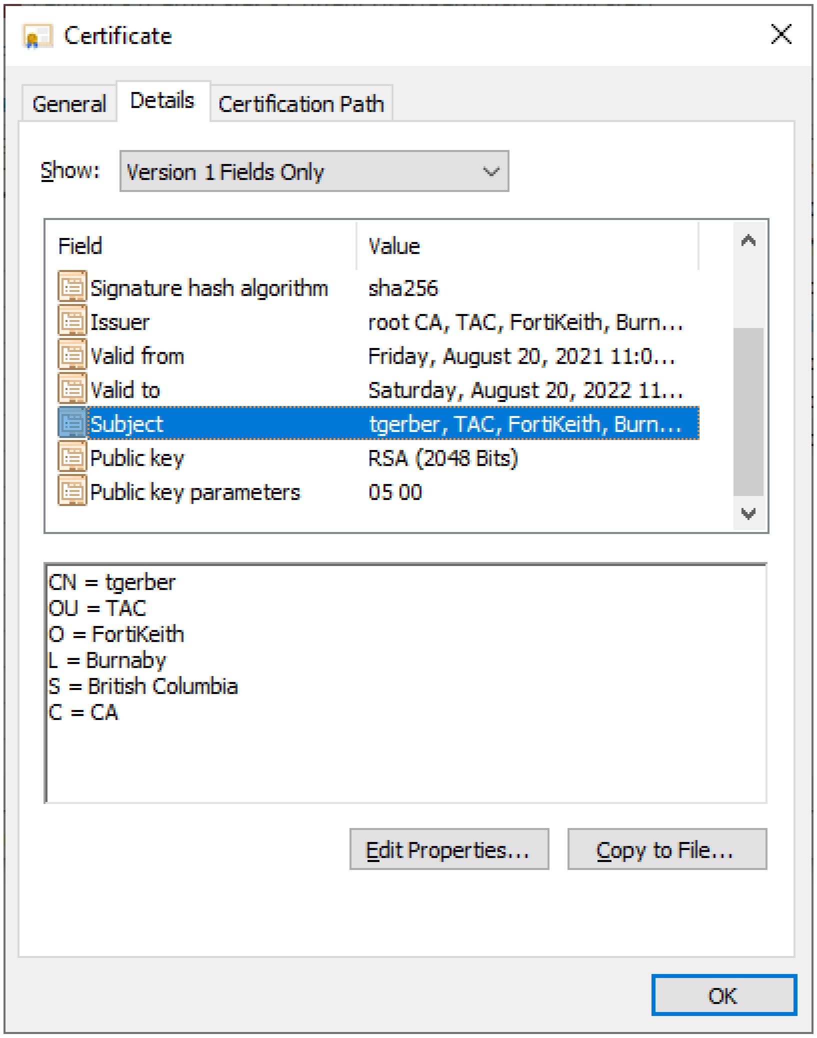
Common name verification

In this method, administrators can define the CN string to match the common name (CN) in the subject field of the certificate. For example:

```
config user peer
    edit "tgerber"
        set ca "CA_Cert_2"
        set cn "tgerber"
```

```
    next
end
```

The matching certificate looks like the following:



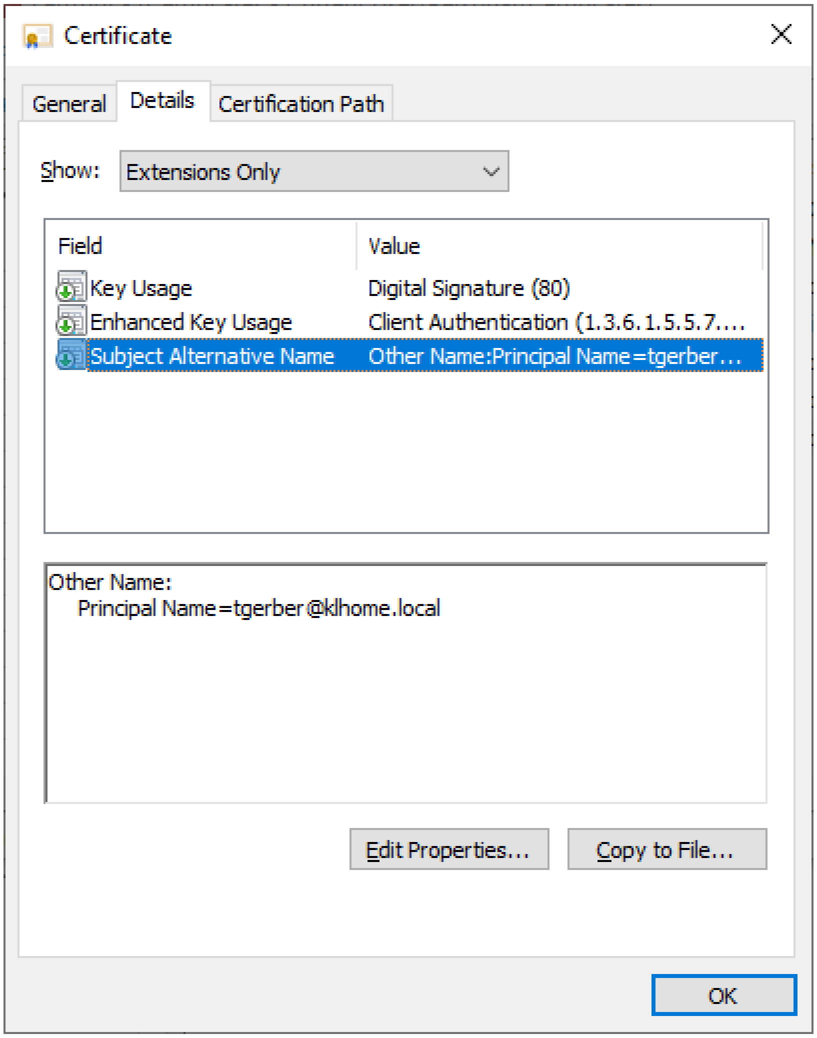
A PKI user must be created on the FortiGate for each remote user that connects to the VPN with a unique user certificate.

Principal name with LDAP integration

In this method, the PKI user setting references an LDAP server. When `ldap-mode` is set to `principal-name`, the UPN in the user certificate's SAN field is used to look up the user in the LDAP directory. If a match is found, then authentication succeeds. For example:

```
config user peer
    edit "ldap-peer"
        set ca "CA_Cert_2"
        set ldap-server "WIN2K16-KLHOME-LDAPS"
        set ldap-mode principal-name
    next
end
```

The matching certificate looks like the following:



This method is more scalable because only one PKI user needs to be created on the FortiGate. Remote users connect with their unique user certificate that are matched against users in the LDAP server.

Certificate management

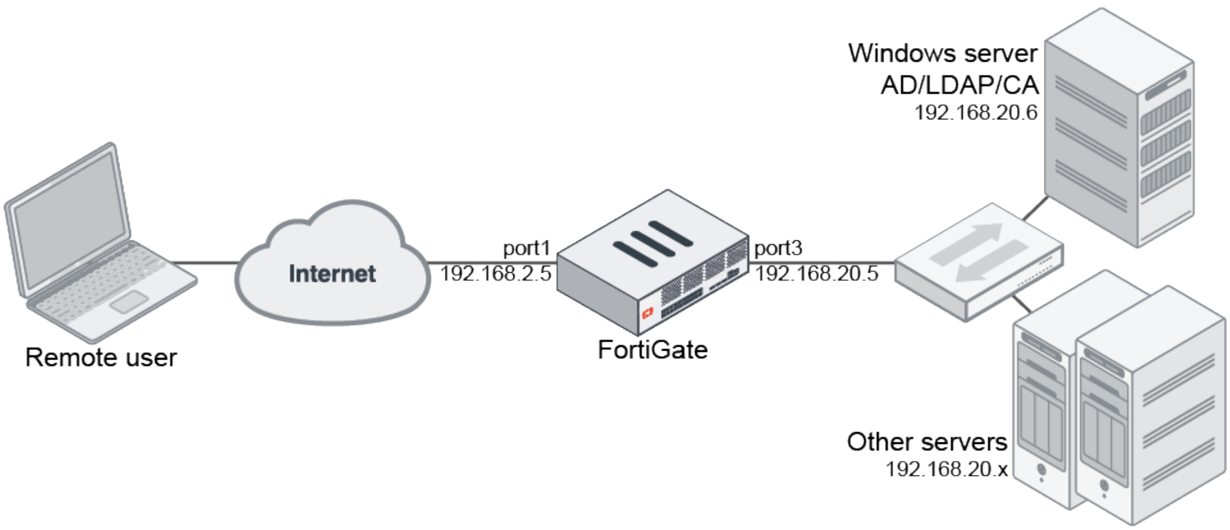
Dialup IPsec VPN with certificate authentication requires careful certificate management planning. Assuming that a company's private certificate authority (CA) is used to generate and sign all the certificates, the following certificates are needed:

Certificate type	Description
Server certificate	The server certificate is used to identify the FortiGate IPsec dialup gateway. A CSR can be generated on the FortiGate and signed by the CA, or the CA can generate the private and public keys and export the certificate package to the FortiGate.

Certificate type	Description
User certificate	The user certificate is generated and signed by the CA with unique CNs in the subject field and/or unique Principal Names in the SAN field. They are used to identify the user that is connecting to the VPN. User certificates must be installed on client machines.
CA certificate	The root CA certificate, and any subordinate CA that signed the actual user and server certificates, must be imported into the FortiGate and client machines. The CA certificate is used to verify the certificate chain of the server and user certificates.

Example

In this example, a dialup IPsec VPN tunnel is configured with certificate authentication using the subject field verification method and the LDAP integration method.



The company CA, named root CA, signs all the server and user certificates. The user, tgerber@klhome.local, has a user certificate signed by root CA installed on their endpoint. The corresponding user account is also present under the company's Active Directory.

There are five major steps to configure this example:

1. Importing the certificates
2. Configuring user authentication
3. Configuring the VPN
4. Configuring FortiClient and the endpoints
5. Testing and verifying the certificate authentication

Importing the certificates

The server certificate and CA certificate need to be imported into the FortiGate.

To import the server certificate:

1. Go to *System* > *Certificates* and select *Import* > *Local Certificate*.
2. For *Type*, select *PKCS #12 Certificate*.
3. Upload the key file exported from the CA and enter the password.
4. Click *OK*. The certificate now appears in the *Local Certificate* section.

To import the CA certificate:

1. Go to *System* > *Certificates* and select *Import* > *CA Certificate*.
2. For *Type*, select *File*.
3. Upload the CA certificate (usually a .CRT file). This certificate only contains the public key.
4. Click *OK*. The certificate now appears in the *Remote CA Certificate* section.

If any subordinate CA is involved in signing the certificates, you need to import its certificate.

Configuring user authentication

FortiGate PKI users do not appear in the GUI until at least one PKI user has been created in the CLI. The following instructions create the PKI users in the CLI.

To configure PKI users for subject field verification:

1. Create the PKI user and choose the CA certificate that was imported (if the certificate was signed by a subordinate CA, choose the subordinate CA's certificate):

   ```
   config user peer
       edit "tgerber"
           set ca "CA_Cert_2"
           set subject "CN=tgerber"
       next
   end
   ```

   For an example of CN field matching, see Common name verification.
2. Create additional users as needed.
3. Place the users into a peer group:

   ```
   config user peergrp
       edit "pki-users"
           set member "tgerber" <user> ... <user>
       next
   end
   ```

To configure PKI users for LDAP integration:

1. Configure the LDAP server that users connect to for authentication:

   ```
   config user ldap
       edit "WIN2K16-KLHOME-LDAPS"
           set server "192.168.20.6"
   ```

```
        set cnid "sAMAccountName"
        set dn "dc=KLHOME,dc=local"
        set type regular
        set username "KLHOME\\Administrator"
        set password ************
        set secure ldaps
        set ca-cert "CA_Cert_1"
        set port 636
    next
end
```

2. Configure the PKI user to reference the LDAP server using the CA certificate that was imported:

```
config user peer
    edit "ldap-peer"
        set ca "CA_Cert_2"
        set ldap-server "WIN2K16-KLHOME-LDAPS"
        set ldap-mode principal-name
    next
end
```

3. Place the user into a peer group:

```
config user peergrp
    edit "pki-ldap"
        set member "ldap-peer"
    next
end
```

Configuring the VPN

To configure the VPN, the address objects must be defined first so they can be used in the VPN and policy configurations. In this example, the VPN is configured in custom mode to define the authentication settings.

To configure the address objects:

1. Create the address range for the dialup clients:
   a. Go to *Policy & Objects > Addresses* and click *Create New > Address*.
   b. For *Name*, enter *remote-user-range*.
   c. For *Type*, select *IP Range* and enter *172.18.200.10-172.18.200.99* in the *IP Range* field.
   d. Click *OK*.
2. Create the address subnet for the destination 192.168.20.0/24:
   a. Click *Create New > Address*.
   b. For *Name*, enter *192.168.20.0*.
   c. For *Type*, select *Subnet* and enter *192.168.20.0/24* in the *IP/Netmask* field.
   d. Click *OK*.

To configure the IPsec dialup tunnel:

1. Go to *VPN > IPsec Tunnels* and click *Create New > IPsec Tunnel*.
2. Enter a name for the tunnel, *Dialup-cert_0*.
3. For *Template type*, select *Custom* then click *Next*.

4. In the *Network* section, enter the following:

Remote Gateway	*Dialup User*
Interface	*port1*
Mode Config	Enable
Assign IP From	*Range*
IPv4 mode config > Client Address Range	*172.18.200.10-172.18.200.99*
Enable IPv4 Split Tunnel	Enable
Accessible Networks	*192.168.20.0*

5. In the *Authentication* section, enter the following:

Method	*Signature*
Certificate Name	Select the server certificate that was imported.
Mode	*Aggressive*
Peer Options > Accept Types	*Peer certificate group*
Peer Options > Peer certificate group	Select the group based on the preferred method: • For subject verification, select *pki-users*. • For LDAP integration, select *pki-ldap*.

When IKEv1 is used, aggressive mode should be selected so that the connecting endpoint will provide its peer ID in the first message of the IKE exchange. The peer identifier allows the FortiGate to match the correct tunnel when multiple dialup tunnels are defined.

6. For *Phase 2 Selectors*, leave the local and remote selectors as *0.0.0.0/0.0.0.0*.
7. Click *OK*.

To configure the firewall policy:

1. Go to *Policy & Objects* > *Firewall Policy* and click *Create New*.
2. Configure the following:

Name	Enter a policy name.
Incoming interface	*Dialup-cert_0*
Outgoing Interface	*port3*
Source	*remote-user-range*
Destination	*192.168.20.0*
Schedule	*always*
Service	*ALL*
Action	*ACCEPT*

3. Configure the other settings as needed.
4. Click *OK*.

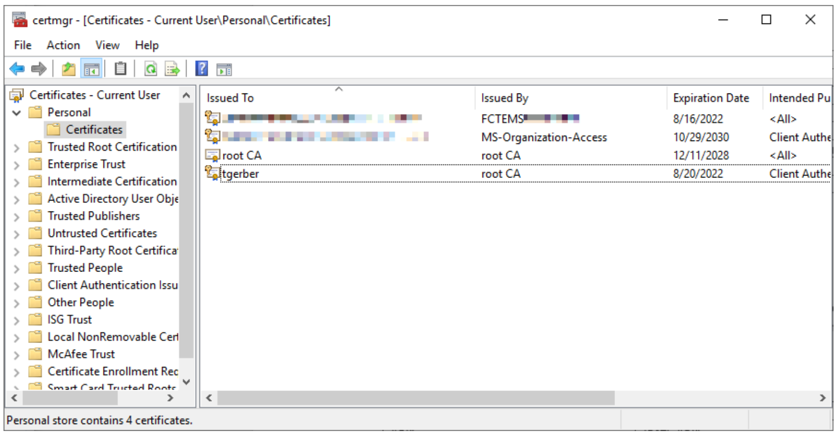
Configuring FortiClient and the endpoints

The following example is configured on a Windows PC with FortiClient 7.0.0. Other configurations may differ slightly.

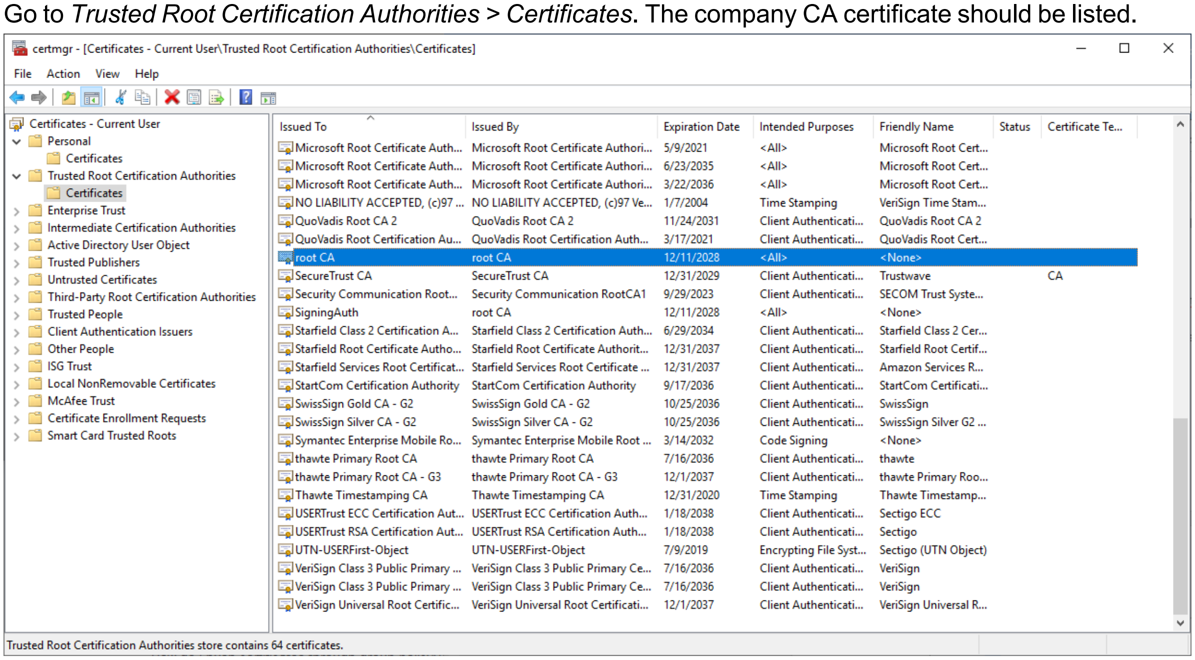
The user certificate and CA certificate must be installed on the endpoint device. They may be pushed by the administrator through group policies or another method. This example assumes that the user certificate and CA certificate are already installed on the endpoint.

To verify the user and CA certificates:

1. Open the Windows certificate manager (certmgr):
   a. In the Control Panel, type *Manage user certificate* in the search box.
   b. Click the result, *Manage user certificates*.
2. Go to *Personal* > *Certificate*. The user certificate should be listed.



3. Go to *Trusted Root Certification Authorities* > *Certificates*. The company CA certificate should be listed.



To configure the FortiClient endpoint settings:

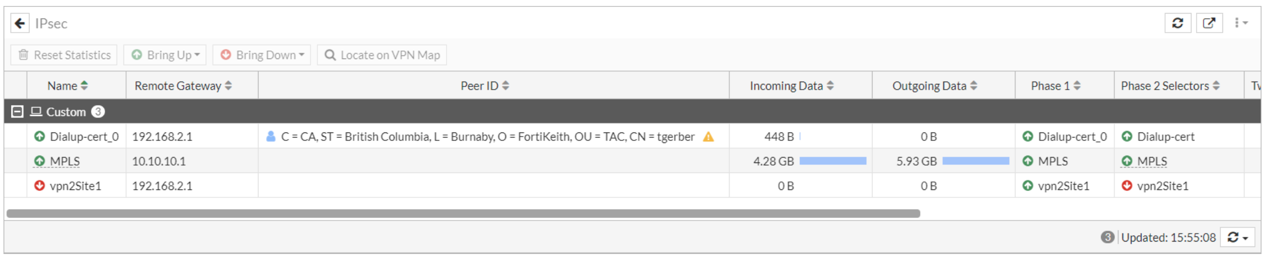
1. In FortiClient, click the *Remote Access* tab and add a new connection:
   a. If there are no existing connections, click *Configure VPN*.
   b. If there are existing connections, click the menu icon and select *Add a new connection*.
2. Configure the following:

VPN	*IPsec VPN*
Connection Name	*Dialup-cert_0*
Remote Gateway	*192.168.2.5*
Authentication Method	*X.509 Certificate* Select the user certificate, *tgerber/root CA*, from the dropdown.
Authentication (XAuth)	*Disable*

3. Click *Save*.

Testing and verifying the certificate authentication

1. On the client PC, open FortiClient and click the *Remote Access* tab.
2. Select the VPN tunnel, *Dialup-cert_0*, and click *Connect*.
   If the connection is successful, a FortiClient pop-up will appear briefly indicating that the IKE negotiation succeeded. The *Remote Access* window now displays *VPN Connected* and the associated VPN tunnel details.
3. On the FortiGate, go to *Dashboard > Network* and locate the *IPsec* widget to view the VPN tunnel monitor. Click the widget to expand to full view.
   The widget displays tunnel information, including the *Peer ID* containing the subject field of the user certificate.



4. Go to *Log & Report > System Events* and select the *VPN Events* card. Several tunnel related logs are recorded.
5. The same logs can be viewed in the CLI:

```
# execute log filter category 1
# execute log filter field subtype vpn
# execute log display
7: date=2021-08-23 time=15:53:08 eventtime=1629759188862005740 tz="-0700"
logid="0101037138" type="event" subtype="vpn" level="notice" vd="root" logdesc="IPsec
connection status changed" msg="IPsec connection status change" action="tunnel-up"
remip=192.168.2.1 locip=192.168.2.5 remport=64916 locport=4500 outintf="port1"
cookies="19f05ebc8c2f7a0d/7716190005538db5" user="C = CA, ST = British Columbia, L =
Burnaby, O = FortiKeith, OU = TAC, CN = tgerber" group="pki-ldap" useralt="C = CA, ST =
British Columbia, L = Burnaby, O = FortiKeith, OU = TAC, CN = tgerber" xauthuser="N/A"
xauthgroup="N/A" assignip=172.18.200.10 vpntunnel="Dialup-cert_0" tunnelip=172.18.200.10
tunnelid=3418215253 tunneltype="ipsec" duration=0 sentbyte=0 rcvdbyte=0 nextstat=0
```

6. If any issues arise during the connection, run the following debug commands to troubleshoot the issue:

```
# diagnose debug application ike -1
# diagnose debug application fnbamd -1
# diagnose debug enable
```

Aggregate and redundant VPN

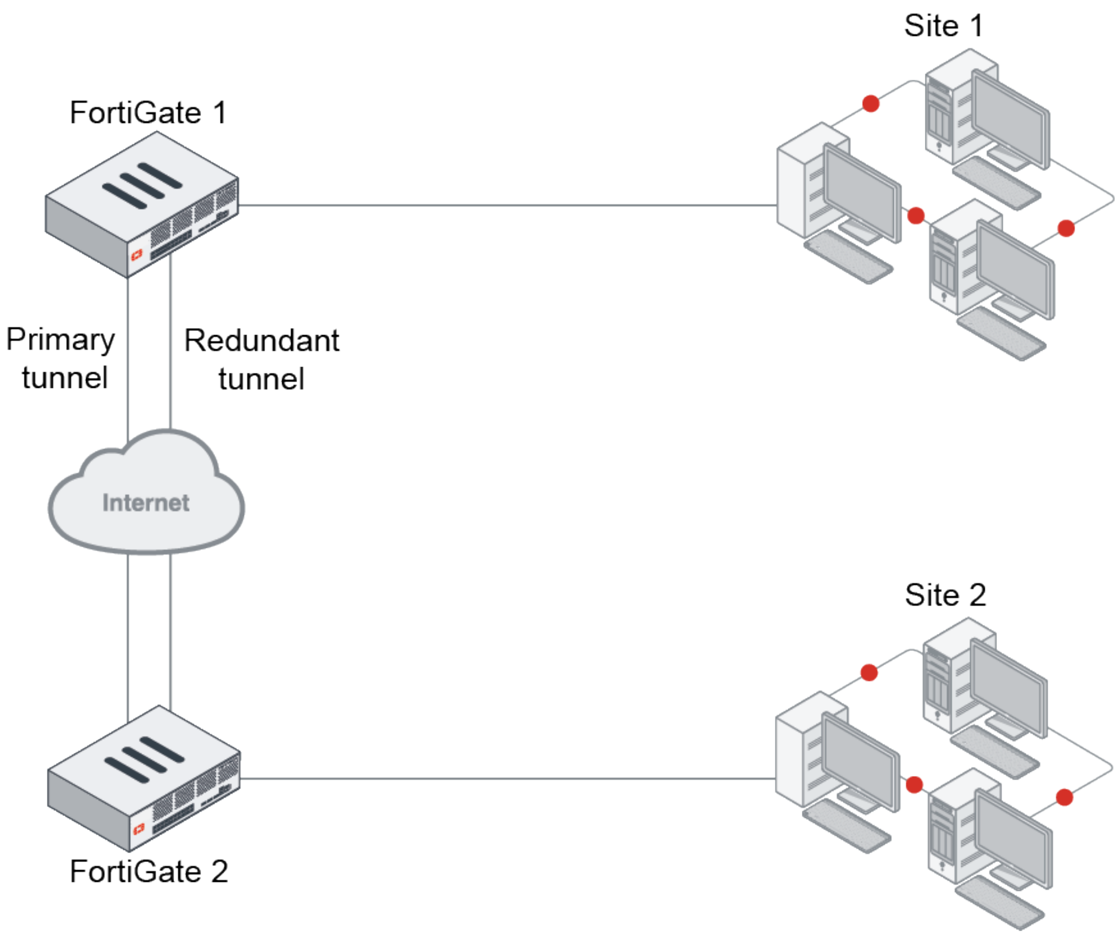
The following topics provide instructions on configuring aggregate and redundant VPNs:

- Manual redundant VPN configuration on page 1455
- OSPF with IPsec VPN for network redundancy on page 1459
- IPsec VPN in an HA environment on page 1466
- Packet distribution and redundancy for aggregate IPsec tunnels on page 1472
- Packet distribution for aggregate dial-up IPsec tunnels using location ID on page 1483
- Packet distribution for aggregate static IPsec tunnels in SD-WAN on page 1487
- Packet distribution for aggregate IPsec tunnels using weighted round robin on page 1492
- Redundant hub and spoke VPN on page 1493

Manual redundant VPN configuration

A FortiGate with two interfaces connected to the internet can be configured to support redundant VPNs to the same remote peer. Four distinct paths are possible for VPN traffic from end to end. If the primary connection fails, the FortiGate can establish a VPN using the other connection.

Topology



The redundant configuration in this example uses route-based VPNs. The FortiGates must operate in NAT mode and use auto-keying.

This example assumes the redundant VPNs are essentially equal in cost and capability. When the original VPN returns to service, traffic continues to use the replacement VPN until the replacement VPN fails. If the redundant VPN uses more expensive facilities, only use it as a backup while the main VPN is down.

A redundant configuration for each VPN peer includes:

- One phase 1 configuration for each path between the two peers with dead peer detection enabled
- One phase 2 definition for each phase 1 configuration
- One static route for each IPsec interface with different distance values to prioritize the routes
- Two firewall policies per IPsec interface, one for each direction of traffic

To configure the phase 1 and phase 2 VPN settings:

1. Go to *VPN > IPsec Wizard* and select the *Custom* template.
2. Enter the tunnel name and click *Next*.

3. Enter the following phase 1 settings for path 1:

Remote Gateway	Static IP Address
IP Address	Enter the IP address of the primary interface of the remote peer.
Interface	Select the primary public interface of this peer.
Dead Peer Detection	On-Demand

4. Configure the remaining phase 1 and phase 2 settings as needed.
5. Click *OK*.
6. Repeat these steps for the remaining paths.
   a. Path 2:

Remote Gateway	Static IP Address
IP Address	Enter the IP address of the secondary interface of the remote peer.
Interface	Select the primary public interface of this peer.
Dead Peer Detection	On-Demand

   b. Path 3:

Remote Gateway	Static IP Address
IP Address	Enter the IP address of the primary interface of the remote peer.
Interface	Select the secondary public interface of this peer.
Dead Peer Detection	On-Demand

   c. Path 4:

Remote Gateway	Static IP Address
IP Address	Enter the IP address of the secondary interface of the remote peer.
Interface	Select the secondary public interface of this peer.
Dead Peer Detection	On-Demand

To configure the static routes:

1. Go to *Network > Static Routes* and click *Create New*.
2. In the *Destination* field, enter the subnet of the private network.
3. For *Interface*, select one of the IPsec interfaces on the local peer.
4. Enter a value for *Administrative Distance*.
5. Click *OK*.
6. Repeat these steps for the three remaining paths, and enter different values for *Administrative Distance* to prioritize the paths.

To configure the firewall policies:

1. Create the policies for the local primary interface:
   a. Go to *Policy & Objects > Firewall Policy* and click *Create New*.
   b. Enter the following:

Name	Enter a name for the policy.
Incoming Interface	Select the local interface to the internal (private) network.
Outgoing Interface	Select one of the virtual IPsec interfaces.
Source	All
Destination	All
Schedule	Always
Service	All
Action	ACCEPT

   c. Click *OK*.
   d. Click *Create New* and configure the policy for the other direction of traffic:

Name	Enter a name for the policy.
Incoming Interface	Select one of the virtual IPsec interfaces.
Outgoing Interface	Select the local interface to the internal (private) network.
Source	All
Destination	All
Schedule	Always
Service	All
Action	ACCEPT

   e. In the policy list, drag the VPN policies above any other policies with similar source and destination addresses.
2. Repeat these steps to create the policies for the three remaining paths.

Creating a backup IPsec interface

A route-based VPN can be configured to act as a backup IPsec interface when the main VPN is out of service. This can only be configured in the CLI.

The backup feature works on interfaces with static addresses that have dead peer detection enabled. The `monitor` option creates a backup VPN for the specified phase 1 configuration.

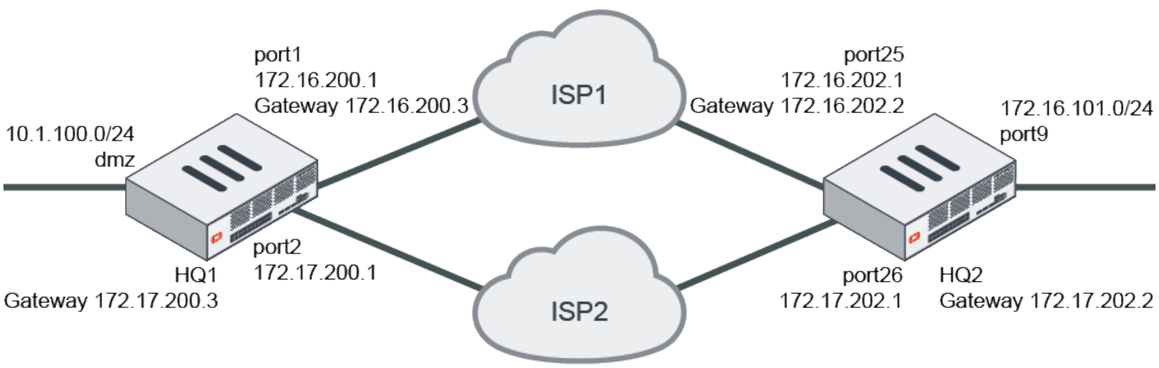
To create a backup IPsec interface:

```
config vpn ipsec phase1-interface
    edit main_vpn
        set dpd on-demand
        set interface port1
```

```
        set nattraversal enable
        set psksecret ********
        set remote-gw 192.168.10.8
        set type static
    next
    edit backup_vpn
        set dpd on-demand
        set interface port2
        set monitor main_vpn
        set nattraversal enable
        set psksecret ********
        set remote-gw 192.168.10.8
        set type static
    next
end
```

OSPF with IPsec VPN for network redundancy

This is a sample configuration of using OSPF with IPsec VPN to set up network redundancy. Route selection is based on OSPF cost calculation. You can configure ECMP or primary/secondary routes by adjusting OSPF path cost.



Because the GUI can only complete part of the configuration, we recommend using the CLI.

To configure OSPF with IPsec VPN to achieve network redundancy using the CLI:

1. Configure the WAN interface and static route.
   Each FortiGate has two WAN interfaces connected to different ISPs. The ISP1 link is for the primary FortiGate and the IPS2 link is for the secondary FortiGate.
   a. Configure HQ1.

```
config system interface
   edit "port1"
      set alias to_ISP1
      set ip 172.16.200.1 255.255.255.0
   next
   edit "port2"
      set alias to_ISP2
      set ip 172.17.200.1 255.255.255.0
   next
end
config router static
   edit 1
```

```
        set gateway 172.16.200.3
        set device "port1"
    next
    edit 2
        set gateway 172.17.200.3
        set device "port2"
        set priority 100
    next
end
```

b. Configure HQ2.

```
config system interface
    edit "port25"
        set alias to_ISP1
        set ip 172.16.202.1 255.255.255.0
    next
    edit "port26"
        set alias to_ISP2
        set ip 172.17.202.1 255.255.255.0
    next
end
config router static
    edit 1
        set gateway 172.16.202.2
        set device "port25"
    next
    edit 2
        set gateway 172.17.202.2
        set device "port26"
        set priority 100
    next
end
```

2. Configure the internal (protected subnet) interface.
   a. Configure HQ1.

```
config system interface
    edit "dmz"
        set ip 10.1.100.1 255.255.255.0
    next
end
```

   b. Configure HQ2.

```
config system interface
    edit "port9"
        set ip 172.16.101.1 255.255.255.0
    next
end
```

3. Configure IPsec phase1-interface and phase-2 interface. On each FortiGate, configure two IPsec tunnels: a primary and a secondary.
   a. Configure HQ1.

```
config vpn ipsec phase1-interface
    edit "pri_HQ2"
        set interface "port1"
        set peertype any
        set net-device enable
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set remote-gw 172.16.202.1
        set psksecret sample1
```

```
      next
      edit "sec_HQ2"
         set interface "port2"
         set peertype any
         set net-device enable
         set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
         set remote-gw 172.17.202.1
         set psksecret sample2
      next
      end
   config vpn ipsec phase2-interface
      edit "pri_HQ2"
         set phase1name "pri_HQ2"
         set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
               aes256gcm chacha20poly1305
         set auto-negotiate enable
      next
      edit "sec_HQ2"
         set phase1name "sec_HQ2"
         set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
               aes256gcm chacha20poly1305
         set auto-negotiate enable
      next
   end
```

b. Configure HQ2.

```
   config vpn ipsec phase1-interface
      edit "pri_HQ1"
         set interface "port25"
         set peertype any
         set net-device enable
         set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
         set remote-gw 172.16.200.1
         set psksecret sample1
      next
      edit "sec_HQ1"
         set interface "port26"
         set peertype any
         set net-device enable
         set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
         set remote-gw 172.17.200.1
         set psksecret sample2
      next
   end
   config vpn ipsec phase2-interface
      edit "pri_HQ1"
         set phase1name "pri_HQ1"
         set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
               aes256gcm chacha20poly1305
         set auto-negotiate enable
      next
      edit "sec_HQ1"
         set phase1name "sec_HQ1"
         set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
               aes256gcm chacha20poly1305
         set auto-negotiate enable
      next
   end
```

4. Configure an inbound and outbound firewall policy for each IPsec tunnel.
   a. Configure HQ1.
```
config firewall policy
   edit 1
      set name "pri_inbound"
      set srcintf "pri_HQ2"
      set dstintf "dmz"
      set srcaddr "172.16.101.0"
      set dstaddr "10.1.100.0"
      set action accept
      set schedule "always"
      set service "ALL"
   next
   edit 2
      set name "pri_outbound"
      set srcintf "dmz"
      set dstintf "pri_HQ2"
      set srcaddr "10.1.100.0"
      set dstaddr "172.16.101.0"
      set action accept
      set schedule "always"
      set service "ALL"
   next
   edit 3
      set name "sec_inbound"
      set srcintf "sec_HQ2"
      set dstintf "dmz"
      set srcaddr "172.16.101.0"
      set dstaddr "10.1.100.0"
      set action accept
      set schedule "always"
      set service "ALL"
   next
   edit 4
      set name "sec_outbound"
      set srcintf "dmz"
      set dstintf "sec_HQ2"
      set srcaddr "10.1.100.0"
      set dstaddr "172.16.101.0"
      set action accept
      set schedule "always"
      set service "ALL"
   next
end
```
   b. Configure HQ2.
```
config firewall policy
   edit 1
      set name "pri_inbound"
      set srcintf "pri_HQ1"
      set dstintf "port9"
      set srcaddr "10.1.100.0"
      set dstaddr "172.16.101.0"
      set action accept
      set schedule "always"
      set service "ALL"
   next
```

```
    edit 2
        set name "pri_outbound"
        set srcintf "port9"
        set dstintf "pri_HQ1"
        set srcaddr "10.1.100.0"
        set dstaddr "172.16.101.0"
        set action accept
        set schedule "always"
        set service "ALL"
    next
    edit 3
        set name "sec_inbound"
        set srcintf "sec_HQ1"
        set dstintf "port9"
        set srcaddr "10.1.100.0"
        set dstaddr "172.16.101.0"
        set action accept
        set schedule "always"
        set service "ALL"
    next
    edit 4
        set name "sec_outbound"
        set srcintf "port9"
        set dstintf "sec_HQ1"
        set srcaddr "172.16.101.0"
        set dstaddr "10.1.100.0"
        set action accept
        set schedule "always"
        set service "ALL"
    next
end
```

5. Assign an IP address to the IPsec tunnel interface.
   a. Configure HQ1.

```
config system interface
    edit "pri_HQ2"
        set ip 10.10.10.1 255.255.255.255
        set remote-ip 10.10.10.2 255.255.255.255
    next
    edit "sec_HQ2"
        set ip 10.10.11.1 255.255.255.255
        set remote-ip 10.10.11.2 255.255.255.255
    next
end
```

   b. Configure HQ2.

```
config system interface
    edit "pri_HQ1"
        set ip 10.10.10.2 255.255.255.255
        set remote-ip 10.10.10.1 255.255.255.255
    next
    edit "sec_HQ1"
        set ip 10.10.11.2 255.255.255.255
        set remote-ip 10.10.11.1 255.255.255.255
    next
end
```

6. Configure OSPF.
   a. Configure HQ1.
```
config router ospf
   set router-id 1.1.1.1
   config area
      edit 0.0.0.0
      next
   end
   config ospf-interface
      edit "pri_HQ2"
         set interface "pri_HQ2"
         set cost 10
         set network-type point-to-point
      next
      edit "sec_HQ2"
         set interface "sec_HQ2"
         set cost 20
         set network-type point-to-point
      next
   end
   config network
      edit 1
         set prefix 10.10.10.0 255.255.255.0
      next
      edit 2
         set prefix 10.10.11.0 255.255.255.0
      next
      edit 3
         set prefix 10.1.100.0 255.255.255.0
      next
   end
end
```
   b. Configure HQ2.
```
config router ospf
   set router-id 2.2.2.2
   config area
      edit 0.0.0.0
      next
   end
   config ospf-interface
      edit "pri_HQ1"
         set interface "pri_HQ1"
         set cost 10
         set network-type point-to-point
      next
      edit "sec_HQ1"
         set interface "sec_HQ1"
         set cost 20
         set network-type point-to-point
      next
   end
   config network
      edit 1
         set prefix 10.10.10.0 255.255.255.0
      next
      edit 2
```

```
            set prefix 10.10.11.0 255.255.255.0
        next
        edit 3
            set prefix 172.16.101.0 255.255.255.0
        next
    end
end
```

To check VPN and OSPF states using diagnose and get commands:

1. Run the `HQ1 # diagnose vpn ike gateway list` command. The system should return the following:
```
vd: root/0
name: pri_HQ2
version: 1
interface: port1 11
addr: 172.16.200.1:500 -> 172.16.202.1:500
virtual-interface-addr: 10.10.10.1 -> 10.10.10.2
created: 1024s ago
IKE SA: created 1/1 established 1/1 time 0/0/0 ms
IPsec SA: created 1/3 established 1/2 time 0/5/10 ms
  id/spi: 45 d184777257b4e692/e2432f834aaf5658 direction: responder status: established
      1024-1024s ago = 0ms proposal: aes128-sha256 key: 9ed41fb06c983344-
      189538046f5ad204 lifetime/rekey: 86400/85105 DPD sent/recv: 00000003/00000000
      vd: root/0
name: sec_HQ2
version: 1
interface: port2 12
addr: 172.17.200.1:500 -> 172.17.202.1:500
virtual-interface-addr: 10.10.11.1 -> 10.10.11.2
created: 346s ago
IKE SA: created 1/1 established 1/1 time 0/0/0 ms
IPsec SA: created 1/1 established 1/1 time 0/10/15 ms
  id/spi: 48 d909ed68636b1ea5/163015e73ea050b8 direction: initiator status: established
      0-0s ago = 0ms proposal: aes128-sha256 key: b9e93c156bdf4562-29db9fbafa256152
      lifetime/rekey: 86400/86099 DPD sent/recv: 00000000/00000000
```
2. Run the `HQ1 # diagnose vpn tunnel list` command. The system should return the following:
```
list all ipsec tunnel in vd 0
name=pri_HQ2 ver=1 serial=1 172.16.200.1:0->172.16.202.1:0 tun_id=172.16.202.1
bound_if=11 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/528 options[0210]=create_dev
    frag-rfc accept_traffic=1
proxyid_num=1 child_num=0 refcnt=14 ilast=2 olast=2 ad=/0
stat: rxp=102 txp=105 rxb=14064 txb=7816
dpd: mode=on-demand on=1 idle=20000ms retry=3 count=0 seqno=3
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=pri_HQ2 proto=0 sa=1 ref=2 serial=1 auto-negotiate
  src: 0:0.0.0.0/0.0.0.0:0 dst: 0:0.0.0.0/0.0.0.0:0 SA: ref=3 options=18227 type=00
      soft=0 mtu=1438 expire=42254/0B replaywin=2048
    seqno=6a esn=0 replaywin_lastseq=00000067 itn=0
  life: type=01 bytes=0/0 timeout=42932/43200 dec: spi=1071b4ee esp=aes key=16
      032036b24a4ec88da63896b86f3a01db
    ah=sha1 key=20 3962933e24c8da21c65c13bc2c6345d643199cdf
  enc: spi=ec89b7e3 esp=aes key=16 92b1d85ef91faf695fca05843dd91626
    ah=sha1 key=20 2de99d1376506313d9f32df6873902cf6c08e454
  dec:pkts/bytes=102/7164, enc:pkts/bytes=105/14936
name=sec_HQ2 ver=1 serial=2 172.17.200.1:0->172.17.202.1:0 tun_id=172.17.202.1
bound_if=12 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/528 options[0210]=create_dev
    frag-rfc accept_traffic=1
```

```
proxyid_num=1 child_num=0 refcnt=14 ilast=3 olast=0 ad=/0
stat: rxp=110 txp=114 rxb=15152 txb=8428
dpd: mode=on-demand on=1 idle=20000ms retry=3 count=0 seqno=3
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=sec_HQ2 proto=0 sa=1 ref=2 serial=1 auto-negotiate
  src: 0:0.0.0.0/0.0.0.0:0 dst: 0:0.0.0.0/0.0.0.0:0 SA: ref=3 options=18227 type=00
      soft=0 mtu=1438 expire=42927/0B replaywin=2048
    seqno=2 esn=0 replaywin_lastseq=00000002 itn=0
  life: type=01 bytes=0/0 timeout=42931/43200 dec: spi=1071b4ef esp=aes key=16
      bcdcabdb7d1c7c695d1f2e0f5441700a
    ah=sha1 key=20 e7a0034589f82eb1af41efd59d0b2565fef8d5da
  enc: spi=ec89b7e4 esp=aes key=16 234240b69e61f6bdee2b4cdec0f33bea
    ah=sha1 key=20 f9d4744a84d91e5ce05f5984737c2a691a3627e8
  dec:pkts/bytes=1/68, enc:pkts/bytes=1/136
```

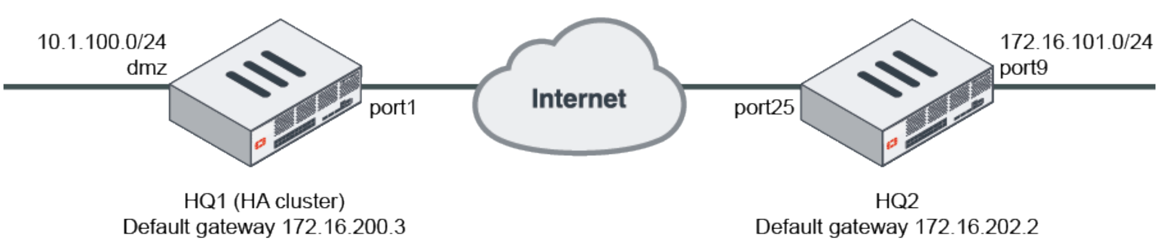
3. Run the `HQ1 # get router info ospf neighbor` command. The system should return the following:
```
OSPF process 0, VRF 0:
Neighbor ID Pri State Dead Time Address Interface
2.2.2.2 1. Full/ - 00:00:37 10.10.10.2 pri_HQ2
2.2.2.2 1. Full/ - 00:00:32 10.10.11.2 sec_HQ2
```
4. Run the `HQ1 # get router info routing-table ospf` command. The system should return the following:
```
Routing table for VRF=0
O 172.16.101.0/24 [110/20] via 10.10.10.2, pri_HQ2 , 00:03:21
```
In case the primary tunnel is down after route convergence.
5. Run the `HQ1 # get router info routing-table ospf` command. The system should return the following:
```
Routing table for VRF=0
O 172.16.101.0/24 [110/110] via 10.10.11.2, sec_HQ2 , 00:00:01
```

IPsec VPN in an HA environment

This is a sample configuration of site-to-site IPsec VPN in an HA environment.

For this example, set up HA as described in the HA topics. When setting up HA, enable the following options to ensure IPsec VPN traffic is not interrupted during an HA failover:

- `session-pickup` under HA settings.
- `ha-sync-esp-seqno` under IPsec phase1-interface settings.



You can configure IPsec VPN in an HA environment using the GUI or CLI.

In this example, the VPN name for HQ1 is "to_HQ2", and the VPN name for HQ2 is "to_HQ1".

To configure IPsec VPN in an HA environment in the GUI:

1. Set up IPsec VPN on HQ1 (the HA cluster):
   a. Go to *VPN > IPsec Wizard* and configure the following settings for *VPN Setup*:
      i. Enter a VPN name.
      ii. For *Template Type*, select *Site to Site*.
      iii. For *Remote Device Type*, select *FortiGate*.
      iv. For *NAT Configuration*, set *No NAT between sites*.
      v. Click *Next*.
   b. Configure the following settings for *Authentication*:
      i. For *Remote Device*, select *IP Address*.
      ii. In the *IP address* field, enter *172.16.202.1*.
      iii. For *Outgoing Interface*, select port1.
      iv. For *Authentication Method*, select *Pre-shared Key*.
      v. In the *Pre-shared Key* field, enter an example key.
      vi. Click *Next*.
   c. Configure the following settings for *Policy & Routing*:
      i. From the *Local Interface* dropdown menu, select the local interface.
      ii. Configure the *Local Subnets* as *10.1.100.0/24*.
      iii. Configure the *Remote Subnets* as *172.16.101.0/24*.
      iv. Click *Create*.
2. Set up IPsec VPN on HQ2:
   a. Go to *VPN > IPsec Wizard* and configure the following settings for *VPN Setup*:
      i. Enter a VPN name.
      ii. For *Template Type*, select *Site to Site*.
      iii. For *Remote Device Type*, select *FortiGate*.
      iv. For *NAT Configuration*, set *No NAT between sites*.
      v. Click *Next*.
   b. Configure the following settings for *Authentication*:
      i. For *Remote Device*, select *IP Address*.
      ii. In the *IP address* field, enter 172.16.200.1.
      iii. For *Outgoing Interface*, select port13.
      iv. For *Authentication Method*, select *Pre-shared Key*.
      v. In the *Pre-shared Key* field, enter an example key.
      vi. Click *Next*.
   c. Configure the following settings for *Policy & Routing*:
      i. From the *Local Interface* dropdown menu, select the desired local interface. In this example, it is port9.
      ii. Configure the *Local Subnets* as 172.16.101.0.
      iii. Configure the *Remote Subnets* as 10.1.100.0
      iv. Click *Create*.

To configure IPsec VPN in an HA environment using the CLI:

1. Configure HA. In this example, two FortiGates work in active-passive mode. The HA heartbeat interfaces are WAN1 and WAN2:
   ```
   config system ha
   ```

```
    set group-name "FGT-HA"
    set mode a-p
    set password sample
    set hbdev "wan1" 50 "wan2" 50
    set session-pickup enable
    set priority 200
    set override-wait-time 10
end
```

2. Configure the WAN interface and default route. The WAN interface is the interface connected to the ISP. It can work in static mode (as shown in this example), DHCP, or PPPoE mode. The IPsec tunnel is established over the WAN interface.
   a. Configure HQ1:
   ```
   config system interface
       edit "port1"
           set vdom "root"
           set ip 172.16.200.1 255.255.255.0
       next
   end
   config router static
       edit 1
           set gateway 172.16.200.3
           set device "port1"
       next
   end
   ```
   b. Configure HQ2:
   ```
   config system interface
       edit "port25"
           set vdom "root"
           set ip 172.16.202.1 255.255.255.0
       next
   end
   config router static
       edit 1
           set gateway 172.16.202.2
           set device "port25"
       next
   end
   ```
3. Configure the internal (protected subnet) interface. The internal interface connects to the corporate internal network. Traffic from this interface routes out the IPsec VPN tunnel.
   a. Configure HQ1:
   ```
   config system interface
       edit "dmz"
           set vdom "root"
           set ip 10.1.100.1 255.255.255.0
       next
   end
   ```
   b. Configure HQ2:
   ```
   config system interface
       edit "port9"
           set vdom "root"
           set ip 172.16.101.1 255.255.255.0
       next
   end
   ```
4. Configure the IPsec phase1-interface. This example uses PSK as the authentication method. You can also use signature authentication.

a. Configure HQ1:

```
config vpn ipsec phase1-interface
   edit "to_HQ2"
      set interface "port1"
      set peertype any
      set net-device enable
      set ha-sync-esp-seqno enable
      set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
      set remote-gw 172.16.202.1
      set psksecret sample
   next
end
```

b. Configure HQ2:

```
config vpn ipsec phase1-interface
   edit "to_HQ1"
      set interface "port25"
      set peertype any
      set net-device enable
      set ha-sync-esp-seqno enable
      set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
      set remote-gw 172.16.200.1
      set psksecret sample
   next
```

5. Configure the IPsec phase2-interface:

a. Configure HQ1:

```
config vpn ipsec phase2-interface
   edit "to_HQ2"
      set phase1name "to_HQ2"
      set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
           aes256gcm chacha20poly1305
      set auto-negotiate enable
   next
end
```

b. Configure HQ2:

```
config vpn ipsec phase2-interface
   edit "to_HQ1"
      set phase1name "to_HQ1"
      set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
           aes256gcm chacha20poly1305
      set auto-negotiate enable
   next
end
```

6. Configure static routes. Two static routes are added to reach the remote protected subnet. The blackhole route is important to ensure IPsec traffic does not match the default route when the IPsec tunnel is down.

a. Configure HQ1:

```
config router static
   edit 2
      set dst 172.16.101.0 255.255.255.0
      set device "to_HQ2"
   next
   edit 3
      set dst 172.16.101.0 255.255.255.0
      set blackhole enable
      set distance 254
   next
```

```
end
```

b. Configure HQ2:

```
config router static
  edit 2
    set dst 10.1.100.0 255.255.255.0
    set device "to_HQ1"
  next
  edit 3
    set dst 10.1.100.0 255.255.255.0
    set blackhole enable
    set distance 254
  next
end
```

7. Configure two firewall policies to allow bi-directional IPsec traffic flow over the IPsec tunnel:
   a. Configure HQ1:

```
config firewall policy
  edit 1
    set name "inbound"
    set srcintf "to_HQ2"
    set dstintf "dmz"
    set srcaddr "172.16.101.0"
    set dstaddr "10.1.100.0"
    set action accept
    set schedule "always"
    set service "ALL"
  next
  edit 2
    set name "outbound"
    set srcintf "dmz"
    set dstintf "to_HQ2"
    set srcaddr "10.1.100.0"
    set dstaddr "172.16.101.0"
    set action accept
    set schedule "always"
    set service "ALL"
  next
end
```

   b. Configure HQ2:

```
config firewall policy
  edit 1
    set name "inbound"
    set srcintf "to_HQ1"
    set dstintf "port9"
    set srcaddr "10.1.100.0"
    set dstaddr "172.16.101.0"
    set action accept
    set schedule "always"
    set service "ALL"
  next
  edit 2
    set name "outbound"
    set srcintf "port9"
    set dstintf "to_HQ1"
    set srcaddr "172.16.101.0"
    set dstaddr "10.1.100.0"
    set action accept
```

```
        set schedule "always"
        set service "ALL"
    next
end
```

8. Use the following `diagnose` commands to check IPsec phase1/phase2 interface status including the sequence number on the secondary FortiGate. The `diagnose debug application ike -1` command is the key to troubleshoot why the IPsec tunnel failed to establish.
   a. Run the `HQ1 # diagnose vpn ike gateway list` command. The system should return the following:
   ```
   vd: root/0
   name: to_HQ2
   version: 1
   interface: port1 11
   addr: 172.16.200.1:500 -> 172.16.202.1:500
   created: 5s ago
   IKE SA: created 1/1 established 1/1 time 0/0/0 ms
   IPsec SA: created 2/2 established 2/2 time 0/0/0 ms
     id/spi: 12 6e8d0532e7fe8d84/3694ac323138a024 direction: responder status:
           established 5-5s ago = 0ms proposal: aes128-sha256 key: b3efb46d0d385aff-
           7bb9ee241362ee8d lifetime/rekey: 86400/86124 DPD sent/recv: 00000000/00000000
   ```
   b. Run the `HQ1 # diagnose vpn tunnel list` command. The system should return the following:
   ```
   list all ipsec tunnel in vd 0

   name=to_HQ2 ver=1 serial=1 172.16.200.1:0->172.16.202.1:0 tun_id=172.16.202.1
   bound_if=11 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/528 options[0210]=create_
         dev frag-rfc accept_traffic=1
   proxyid_num=1 child_num=0 refcnt=11 ilast=7 olast=87 ad=/0
   stat: rxp=0 txp=0 rxb=0 txb=0
   dpd: mode=on-demand on=1 idle=20000ms retry=3 count=0 seqno=0
   natt: mode=none draft=0 interval=0 remote_port=0
   proxyid=to_HQ2 proto=0 sa=1 ref=2 serial=1 auto-negotiate
     src: 0:0.0.0.0/0.0.0.0:0 dst: 0:0.0.0.0/0.0.0.0:0 SA: ref=3 options=18227 type=00
           soft=0 mtu=1438 expire=42927/0B replaywin=2048
       seqno=1 esn=0 replaywin_lastseq=00000000 itn=0
     life: type=01 bytes=0/0 timeout=42930/43200 dec: spi=ef9ca700 esp=aes key=16
           a2c6584bf654d4f956497b3436f1cfc7
       ah=sha1 key=20 82c5e734bce81e6f18418328e2a11aeb7baa021b
     enc: spi=791e898e esp=aes key=16 0dbb4588ba2665c6962491e85a4a8d5a
       ah=sha1 key=20 2054b318d2568a8b12119120f20ecac97ab730b3
     dec:pkts/bytes=0/0, enc:pkts/bytes=0/0
   ```
   ESP seqno synced to primary FortiGate every five minutes, and big gap between primary and secondary to ensure that no packet is dropped after HA failover caused by tcp-replay. Check ESP sequence number synced on secondary FortiGate.
   c. Run the `HQ1 # execute ha manage 0 admin` command.
   d. Run the `HQ1-Sec # diagnose vpn tunnel list` command. The system should return the following:
   ```
   list all ipsec tunnel in vd 0

   name=to_HQ2 ver=1 serial=1 172.16.200.1:0->172.16.202.1:0 tun_id=172.16.202.1
   bound_if=11 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/528 options[0210]=create_
         dev frag-rfc accept_traffic=1
   proxyid_num=1 child_num=0 refcnt=11 ilast=13 olast=274 ad=/0
   stat: rxp=0 txp=0 rxb=0 txb=0
   dpd: mode=on-demand on=1 idle=20000ms retry=3 count=0 seqno=0
   natt: mode=none draft=0 interval=0 remote_port=0
   proxyid=to_HQ2 proto=0 sa=1 ref=2 serial=1 auto-negotiate
   ```

```
src: 0:0.0.0.0/0.0.0.0:0 dst: 0:0.0.0.0/0.0.0.0:0 SA: ref=3 options=27 type=00
      soft=0 mtu=1280 expire=42740/0B replaywin=2048
   seqno=47868c01 esn=0 replaywin_lastseq=00000000 itn=0
life: type=01 bytes=0/0 timeout=42930/43200 dec: spi=ef9ca700 esp=aes key=16
      a2c6584bf654d4f956497b3436f1cfc7
   ah=sha1 key=20 82c5e734bce81e6f18418328e2a11aeb7baa021b
enc: spi=791e898e esp=aes key=16 0dbb4588ba2665c6962491e85a4a8d5a
   ah=sha1 key=20 2054b318d2568a8b12119120f20ecac97ab730b3
dec:pkts/bytes=0/0, enc:pkts/bytes=0/0
```

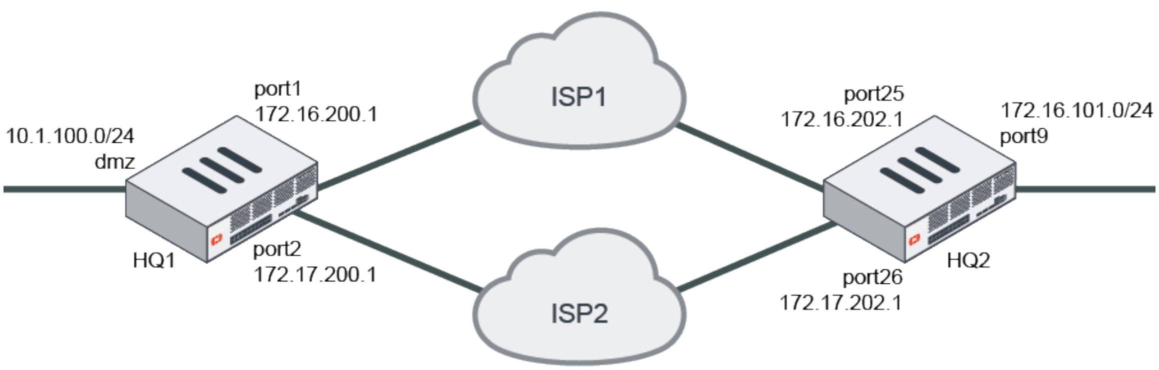
Packet distribution and redundancy for aggregate IPsec tunnels

This is a sample configuration of a multiple site-to-site IPsec VPN that uses an IPsec aggregate interface to set up redundancy and traffic load-balancing. The VPN tunnel interfaces must have `net-device` disabled in order to be members of the IPsec aggregate.

Each FortiGate has two WAN interfaces connected to different ISPs. OSPF runs over the IPsec aggregate in this configuration.

The supported load balancing algorithms are: L3, L4, round-robin (default), weighted round-robin, and redundant. The first four options allow traffic to be load-balanced, while the last option (redundant) uses the first tunnel that is up for all traffic.

Dynamic routing can run on the aggregate interface, and it can be a member interface in SD-WAN (not shown in this configuration).



Configuring the HQ1 FortiGate in the GUI

There are five steps to configure the FortiGate:

1. Create the IPsec tunnels.
2. Create the IPsec aggregate.
3. Configure the firewall policies.
4. Configure the aggregate VPN interface IPs.
5. Configure OSPF.

To create the IPsec tunnels:

1. Go to *VPN > IPsec Wizard* and select the *Custom* template.
2. For *Name*, enter pri_HQ2 and click *Next*.
3. Enter the following:

Phase 1	
IP Address	172.16.202.1
Interface	port1
Device creation	Disabled
Aggregate member	Enabled
Authentication Method	Pre-shared Key
Pre-shared Key	Enter the secure key
IKE Mode	Aggressive
Peer Options Accept Types	Any peer ID
Phase 2	
Auto-negotiate	Enable

4. Configure the other settings as needed.
5. Click *OK*.
6. Create another tunnel named sec_HQ2 with the following settings:

Phase 1	
IP Address	172.17.202.1
Interface	port2
Device creation	Disabled
Aggregate member	Enabled
Authentication Method	Pre-shared Key
Pre-shared Key	Enter the secure key
IKE Mode	Aggressive
Peer Options Accept Types	Any peer ID
Phase 2	
Auto-negotiate	Enable

To create the IPsec aggregate:

1. Go to *VPN > IPsec Tunnels* and click *Create New > IPsec Aggregate*.
2. For *Name*, enter agg_HQ2.
3. Select a load balancing algorithm.
4. From the *Tunnel* dropdown, select the tunnels that you created previously (*pri_HQ2* and *sec_HQ2*). If required, enter weights for each tunnel.
5. Click *OK*.

To configure the firewall policies:

1. Go to *Policy & Objects* > *Firewall Policy*.
2. Create an inbound traffic policy with the following settings:

Name	inbound
Incoming Interface	agg_HQ2
Outgoing Interface	dmz
Source	172.16.101.0
Destination	10.1.100.0
Schedule	always
Action	ACCEPT
Service	ALL

3. Click *OK*.
4. Create an outbound traffic policy with the following settings:

Name	outbound
Incoming Interface	dmz
Outgoing Interface	agg_HQ2
Source	10.1.100.0
Destination	172.16.101.0
Schedule	always
Action	ACCEPT
Service	ALL

To configure the aggregate VPN interface IPs:

1. Go to *Network* > *Interfaces* and edit agg_HQ2.
2. For *IP*, enter 10.10.10.1.
3. For *Remote IP/Netmask*, enter 10.10.10.2 255.255.255.255.
4. Click *OK*.

To configure OSPF:

1. Go to *Network* > *OSPF*.
2. For *Router ID*, enter 1.1.1.1.
3. In the *Areas* table, click *Create New*.
   a. For *Area ID*, enter 0.0.0.0.
   b. Click *OK*.

4. In the *Networks* table, click *Create New*.
   a. Set the *Area* to *0.0.0.0*.
   b. For *IP/Netmask*, enter 10.1.100.0/24.
   c. Click *OK*.
   d. Click *Create New*.
   e. For *IP/Netmask*, enter 10.10.10.0/24.
   f. Click *OK*.
5. Click *Apply*.

Configuring the HQ2 FortiGate in the GUI

There are five steps to configure the FortiGate:

1. Create the IPsec tunnels.
2. Create the IPsec aggregate.
3. Configure the firewall policies.
4. Configure the aggregate VPN interface IPs.
5. Configure OSPF.

To create the IPsec tunnels:

1. Go to *VPN > IPsec Wizard* and select the *Custom* template.
2. For *Name*, enter pri_HQ1 and click *Next*.
3. Enter the following:

Phase 1	
IP Address	172.16.200.1
Interface	port25
Device creation	Disabled
Aggregate member	Enabled
Authentication Method	Pre-shared Key
Pre-shared Key	Enter the secure key
IKE Mode	Aggressive
Peer Options Accept Types	Any peer ID
Phase 2	
Auto-negotiate	Enable

4. Configure the other settings as needed.
5. Click *OK*.
6. Create another tunnel named sec_HQ1 with the following settings:

Phase 1

IP Address	172.17.200.1
Interface	port26
Device creation	Disabled
Aggregate member	Enabled
Authentication Method	Pre-shared Key
Pre-shared Key	Enter the secure key
IKE Mode	Aggressive
Peer Options Accept Types	Any peer ID
Phase 2	
Auto-negotiate	Enable

To create the IPsec aggregate:

1. Go to *VPN > IPsec Tunnels* and click *Create New > IPsec Aggregate*.
2. For *Name*, enter agg_HQ1.
3. Select a load balancing algorithm.
4. From the *Tunnel* dropdown, select the tunnels that you created previously (*pri_HQ1* and *sec_HQ1*). If required, enter weights for each tunnel.
5. Click *OK*.

To configure the firewall policies:

1. Go to *Policy & Objects > Firewall Policy*.
2. Create an inbound traffic policy with the following settings:

Name	inbound
Incoming Interface	agg_HQ1
Outgoing Interface	port9
Source	10.1.100.0
Destination	172.16.101.0
Schedule	always
Action	ACCEPT
Service	ALL

3. Click *OK*.
4. Create an outbound traffic policy with the following settings:

Name	outbound
Incoming Interface	port9

Outgoing Interface	agg_HQ1
Source	172.16.101.0
Destination	10.1.100.0
Schedule	always
Action	ACCEPT
Service	ALL

To configure the aggregate VPN interface IPs:

1. Go to *Network > Interfaces* and edit agg_HQ1.
2. For *IP*, enter 10.10.10.2.
3. For *Remote IP/Netmask*, enter 10.10.10.1 255.255.255.255.
4. Click *OK*.

To configure OSPF:

1. Go to *Network > OSPF*.
2. For *Router ID*, enter 2.2.2.2.
3. In the *Areas* table, click *Create New*.
   a. For *Area ID*, enter 0.0.0.0.
   b. Click *OK*.
4. In the *Networks* table, click *Create New*.
   a. Set the *Area* to *0.0.0.0*.
   b. For *IP/Netmask*, enter 172.16.101.0/24.
   c. Click *OK*.
   d. Click *Create New*.
   e. For *IP/Netmask*, enter 10.10.10.0/24.
   f. Click *OK*.
5. Click *Apply*.

Monitoring the traffic in the GUI

To monitor the traffic:

1. Go to *Dashboard > Network* , hover over the *IPsec* widget, then click *Expand to Full Screen*.
2. Expand the aggregate tunnel in the table to view statistics for each aggregate member.

Configuring the HQ1 FortiGate in the CLI

There are six steps to configure the FortiGate:

1. Configure the interfaces.
2. Configure two IPsec phase 1 and phase 2 interfaces.
3. Configure the IPsec aggregate.
4. Configure the firewall policies.

5. Configure the aggregate VPN interface IPs.
6. Configure OSPF.

To configure the interfaces:

1. Configure port1, port2, and dmz as shown in the topology diagram.

To configure two IPsec phase 1 and phase 2 interfaces:

```
config vpn ipsec phase1-interface
   edit "pri_HQ2"
      set interface "port1"
      set peertype any
      set net-device disable
      set aggregate-member enable
      set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
      set remote-gw 172.16.202.1
      set psksecret sample1
   next
   edit "sec_HQ2"
      set interface "port2"
      set peertype any
      set net-device disable
      set aggregate-member enable
      set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
      set remote-gw 172.17.202.1
      set psksecret sample2
   next
   end
config vpn ipsec phase2-interface
   edit "pri_HQ2"
      set phase1name "pri_HQ2"
      set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm aes256gcm
          chacha20poly1305
      set auto-negotiate enable
   next
   edit "sec_HQ2"
      set phase1name "sec_HQ2"
      set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm aes256gcm
          chacha20poly1305
      set auto-negotiate enable
   next
end
```

To configure the IPsec aggregate:

```
config system ipsec-aggregate
   edit "agg_HQ2"
      set member "pri_HQ2" "sec_HQ2"
   next
end
```

To configure the firewall policies:

```
config firewall policy
   edit 1
```

```
      set name "inbound"
      set srcintf "agg_HQ2"
      set dstintf "dmz"
      set srcaddr "172.16.101.0"
      set dstaddr "10.1.100.0"
      set action accept
      set schedule "always"
      set service "ALL"
   next
   edit 2
      set name "outbound"
      set srcintf "dmz"
      set dstintf "agg_HQ2"
      set srcaddr "10.1.100.0"
      set dstaddr "172.16.101.0"
      set action accept
      set schedule "always"
      set service "ALL"
   next
end
```

To configure the aggregate VPN interface IPs:

```
config system interface
   edit "agg_HQ2"
      set ip 10.10.10.1 255.255.255.255
      set remote-ip 10.10.10.2 255.255.255.255
   next
end
```

To configure OSPF:

```
config router ospf
   set router-id 1.1.1.1
   config area
      edit 0.0.0.0
      next
   end
   config network
      edit 1
         set prefix 10.1.100.0 255.255.255.0
      next
      edit 2
         set prefix 10.10.10.0 255.255.255.0
      next
   end
end
```

Configuring the HQ2 FortiGate in the CLI

There are six steps to configure the FortiGate:

1. Configure the interfaces.
2. Configure two IPsec phase 1 and phase 2 interfaces.
3. Configure the IPsec aggregate.
4. Configure the firewall policies.

5. Configure the aggregate VPN interface IPs.
6. Configure OSPF.

To configure the interfaces:

1. Configure port25, port26, and port9 as shown in the topology diagram.

To configure two IPsec phase 1 and phase 2 interfaces:

```
config vpn ipsec phase1-interface
   edit "pri_HQ1"
      set interface "port25"
      set peertype any
      set net-device disable
      set aggregate-member enable
      set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
      set remote-gw 172.16.200.1
      set psksecret sample1
   next
   edit "sec_HQ1"
      set interface "port26"
      set peertype any
      set net-device disable
      set aggregate-member enable
      set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
      set remote-gw 172.17.200.1
      set psksecret sample2
   next
end
config vpn ipsec phase2-interface
   edit "pri_HQ1"
      set phase1name "pri_HQ1"
      set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm aes256gcm
          chacha20poly1305
      set auto-negotiate enable
   next
   edit "sec_HQ1"
      set phase1name "sec_HQ1"
      set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm aes256gcm
          chacha20poly1305
      set auto-negotiate enable
   next
end
```

To configure the IPsec aggregate:

```
config system ipsec-aggregate
   edit "agg_HQ1"
      set member "pri_HQ1" "sec_HQ1"
   next
end
```

To configure the firewall policies:

```
config firewall policy
   edit 1
```

```
      set name "inbound"
      set srcintf "agg_HQ1"
      set dstintf "port9"
      set srcaddr "10.1.100.0"
      set dstaddr "172.16.101.0"
      set action accept
      set schedule "always"
      set service "ALL"
   next
   edit 2
      set name "outbound"
      set srcintf "port9"
      set dstintf "agg_HQ1"
      set srcaddr "172.16.101.0"
      set dstaddr "10.1.100.0"
      set action accept
      set schedule "always"
      set service "ALL"
   next
end
```

To configure the aggregate VPN interface IPs:

```
config system interface
   edit "agg_HQ1"
      set ip 10.10.10.2 255.255.255.255
      set remote-ip 10.10.10.1 255.255.255.255
   next
end
```

To configure OSPF:

```
config router ospf
   set router-id 2.2.2.2
   config area
      edit 0.0.0.0
      next
   end
   config network
      edit 1
         set prefix 172.16.101.0 255.255.255.0
      next
      edit 2
         set prefix 10.10.10.0 255.255.255.0
      next
   end
end
```

Monitoring the traffic in the CLI

To view debugging information:

1. Verify the status of the phase 1 IKE SAs:
   ```
   # diagnose vpn ike gateway list
   vd: root/0
   ```

```
name: pri_HQ2
version: 1
interface: port1 11
addr: 172.16.200.1:500 -> 172.16.202.1:500
tun_id: 172.16.202.1
created: 1520s ago
IKE SA: created 1/2 established 1/1 time 10/10/10 ms
IPsec SA: created 2/2 established 1/1 time 0/0/0 ms
  id/spi: 173 dcdede154681579b/e32f4c48c4349fc0 direction: responder status: established
      1498-1498s ago = 10ms proposal: aes128-sha256 key: d7230a68d7b83def-
      588b94495cfa9d38 lifetime/rekey: 86400/84631 DPD sent/recv: 0000000d/00000006
vd: root/0
name: sec_HQ2
version: 1
interface: port2 12
addr: 172.17.200.1:500 -> 172.17.202.1:500
created: 1520s ago
IKE SA: created 1/2 established 1/1 time 10/10/10 ms
IPsec SA: created 2/2 established 1/1 time 0/0/0 ms
  id/spi: 174 a567bd7bf02a04b5/4251b6254660aee2 direction: responder status: established
      1498-1498s ago = 10ms proposal: aes128-sha256 key: 9f44f500c28d8de6-
      febaae9d1e6a164c lifetime/rekey: 86400/84631 DPD sent/recv: 00000008/0000000c
```

2. Verify the phase 2 IPsec tunnel SAs:

```
# diagnose vpn tunnel list
list all ipsec tunnel in vd 0
name=sec_HQ2 ver=1 serial=2 172.17.200.1:0->172.17.202.1:0 tun_id=172.17.202.1
bound_if=5 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/512 options[0200]=frag-rfc
    run_state=1 accept_traffic=1
proxyid_num=1 child_num=0 refcnt=7 ilast=5 olast=5 ad=/0
stat: rxp=39 txp=40 rxb=5448 txb=2732
dpd: mode=on-demand on=1 idle=20000ms retry=3 count=0 seqno=15
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=sec_HQ2 proto=0 sa=1 ref=2 serial=2 auto-negotiate
  src: 0:0.0.0.0/0.0.0.0:0 dst: 0:0.0.0.0/0.0.0.0:0 SA: ref=3 options=18227 type=00
      soft=0 mtu=1438 expire=41230/0B replaywin=2048
    seqno=29 esn=0 replaywin_lastseq=00000028 itn=0
  life: type=01 bytes=0/0 timeout=42899/43200 dec: spi=1071b4f9 esp=aes key=16
      1f4dbb78bea8e97650b52d8170b5ece7
    ah=sha1 key=20 cd9bf2de0f49296cf489dd915d7baf6d78bc8f12
  enc: spi=ec89b7ee esp=aes key=16 0546efecd0d1b9ba5944f635896e4404
    ah=sha1 key=20 34599bc7dc25e1ce63ac9615bd50928ce0667dc8
  dec:pkts/bytes=39/2796, enc:pkts/bytes=40/5456
name=pri_HQ2 ver=1 serial=1 172.16.200.1:0->172.16.202.1:0 tun_id=172.16.202.1
bound_if=11 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/512 options[0200]=frag-rfc
    run_state=1 accept_traffic=1
proxyid_num=1 child_num=0 refcnt=5 ilast=15 olast=15 ad=/0
stat: rxp=38 txp=39 rxb=5152 txb=2768
dpd: mode=on-demand on=1 idle=20000ms retry=3 count=0 seqno=20
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=pri_HQ2 proto=0 sa=1 ref=2 serial=2 auto-negotiate
  src: 0:0.0.0.0/0.0.0.0:0 dst: 0:0.0.0.0/0.0.0.0:0 SA: ref=3 options=18227 type=00
      soft=0 mtu=1438 expire=41231/0B replaywin=2048
    seqno=28 esn=0 replaywin_lastseq=00000027 itn=0
  life: type=01 bytes=0/0 timeout=42900/43200 dec: spi=1071b4f8 esp=aes key=16
      142cce377b3432ba41e64128ade6848c
    ah=sha1 key=20 20e64947e2397123f561584321adc0e7aa0c342d
  enc: spi=ec89b7ed esp=aes key=16 2ec13622fd60dacce3d28ebe5fe7ab14
```

```
    ah=sha1 key=20 c1787497508a87f40c73c0db0e835c70b3c3f42d
  dec:pkts/bytes=38/2568, enc:pkts/bytes=39/5432
```

3. Debug the IPsec aggregation list:

```
# diagnose sys ipsec-aggregate list
agg_HQ2 algo=RR member=2 run_tally=2
members:
  pri_HQ2
  sec_HQ2
```

4. Verify the OSPF neighbor information:

```
# get router info ospf neighbor
OSPF process 0, VRF 0:
Neighbor ID Pri State Dead Time Address Interface
2.2.2.2 1. Full/ - 00:00:34 10.10.10.2 agg1_HQ2
```

5. Verify the OSPF routing table:

```
# get router info routing-table ospf
Routing table for VRF=0
O 172.16.101.0/24 [110/20] via 10.10.10.2, agg1_HQ2 , 00:18:43
```

Packet distribution for aggregate dial-up IPsec tunnels using location ID

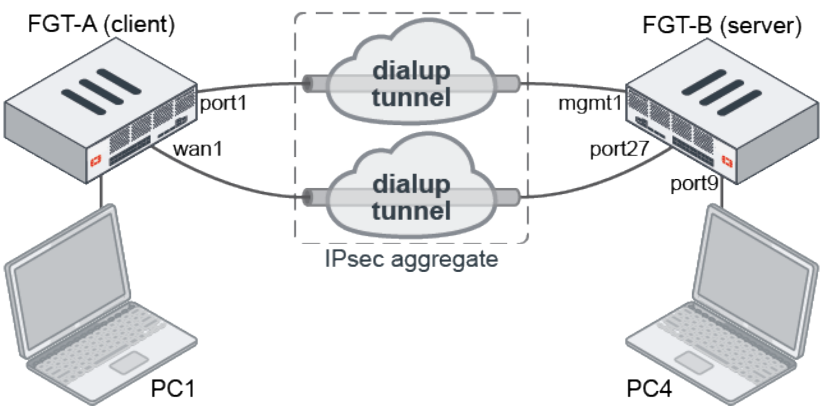
To support per-packet load balancing on aggregate dial-up IPsec tunnels between sites, each spoke must be configured with a location ID. On the hub, per-packet load balancing is performed on the tunnels in the IPsec aggregate that have the same location ID.

Multiple dial-up VPN tunnels from the same location can be aggregated on the VPN hub and load balanced based on the configured load balance algorithm.

IPsec traffic cannot be offloaded to the NPU.

Example

In this example, an IPsec aggregate tunnel is formed between two dial-up IPsec tunnels in order to support per-packet load balancing.



To configure the client FortiGate (FGT-A):

1. Configure the IPsec tunnels:

```
config vpn ipsec phase1-interface
    edit "client1"
        set interface "port1"
        set peertype any
        set net-device disable
        set aggregate-member enable
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set remote-gw 172.16.200.4
        set psksecret **********
    next
    edit "client2"
        set interface "wan1"
        set peertype any
        set net-device disable
        set aggregate-member enable
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set remote-gw 173.1.1.1
        set psksecret **********
    next
end
```

2. Configure an aggregate of the IPsec tunnels:

```
config system ipsec-aggregate
    edit "agg1"
        set member "client1" "client2"
    next
end
```

3. Configure the location ID:

```
config system settings
    set location-id 1.1.1.1
end
```

To configure the server FortiGate (FGT-B):

1. Configure the IPsec tunnels:

```
config vpn ipsec phase1-interface
    edit "server1"
        set type dynamic
        set interface "mgmt1"
        set peertype any
        set net-device disable
        set aggregate-member enable
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set dpd on-idle
        set psksecret ***********
        set dpd-retryinterval 60
    next
    edit "server2"
        set type dynamic
        set interface "port27"
```

```
        set peertype any
        set net-device disable
        set aggregate-member enable
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set dpd on-idle
        set psksecret **********
        set dpd-retryinterval 60
    next
end
config vpn ipsec phase2-interface
    edit "server1"
        set phase1name "server1"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
aes256gcm chacha20poly1305
    next
    edit "server2"
        set phase1name "server2"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
aes256gcm chacha20poly1305
    next
end
```

2. Configure an aggregate of the IPsec tunnels:

```
config system ipsec-aggregate
    edit "server"
        set member "server1" "server2"
    next
end
```

3. Configure a firewall policy:

```
config firewall policy
    edit 1
        set srcintf "server"
        set dstintf "port9"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
    next
end
```

To check the IPsec tunnel and aggregate state:

1. List all of the VPN tunnels:

```
FGDocs # diagnose vpn tunnel list
list all ipsec tunnel in vd 0
------------------------------------------------------
name=server1 ver=1 serial=1 172.16.200.4:500->0.0.0.0:500 tun_id=1.0.0.0 dst_mtu=0 dpd-
link=on remote_location=0.0.0.0 weight=1
bound_if=4 lgwy=static/1 tun=tunnel/15 mode=dialup/2 encap=none/4616 options[1208]=npu
frag-rfc  accept_traffic=1 overlay_id=0

proxyid_num=0 child_num=2 refcnt=4 ilast=14210 olast=14210 ad=/0
stat: rxp=798921 txp=819074 rxb=121435992 txb=68802216
```

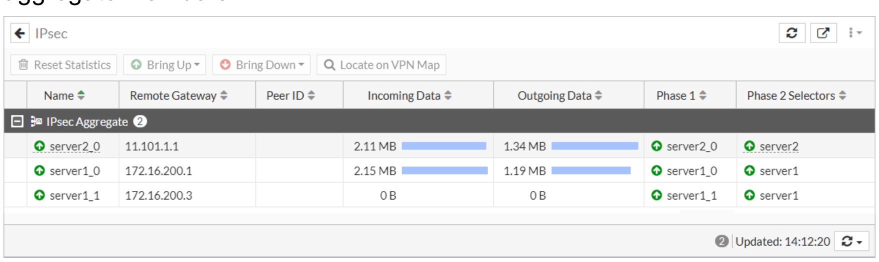
```
dpd: mode=on-idle on=0 idle=60000ms retry=3 count=0 seqno=0
natt: mode=none draft=0 interval=0 remote_port=0
run_tally=0
------------------------------------------------------
name=server2 ver=1 serial=2 173.1.1.1:500->0.0.0.0:500 tun_id=2.0.0.0 dst_mtu=0 dpd-
link=on remote_location=0.0.0.0 weight=1
bound_if=17 lgwy=static/1 tun=tunnel/15 mode=dialup/2 encap=none/4616 options[1208]=npu
frag-rfc  accept_traffic=1 overlay_id=0

proxyid_num=0 child_num=1 refcnt=3 ilast=14177 olast=14177 ad=/0
stat: rxp=836484 txp=819111 rxb=137429352 txb=80046050
dpd: mode=on-idle on=0 idle=60000ms retry=3 count=0 seqno=0
natt: mode=none draft=0 interval=0 remote_port=0
run_tally=0
------------------------------------------------------
name=server1_0 ver=1 serial=8 172.16.200.4:500->172.16.200.1:500 tun_id=172.16.200.1
dst_mtu=1500 dpd-link=on remote_location=1.1.1.1 weight=1
bound_if=4 lgwy=static/1 tun=tunnel/15 mode=dial_inst/3 encap=none/4744 options
[1288]=npu rgwy-chg frag-rfc  run_state=0 accept_traffic=1 overlay_id=0

parent=server1 index=0
proxyid_num=1 child_num=0 refcnt=5 ilast=45 olast=45 ad=/0
stat: rxp=17176 txp=17176 rxb=2610752 txb=1442784
dpd: mode=on-idle on=1 idle=60000ms retry=3 count=0 seqno=12
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=server1 proto=0 sa=1 ref=2 serial=1 add-route
  src: 0:0.0.0.0-255.255.255.255:0
  dst: 0:10.1.100.0-10.1.100.255:0
  SA:  ref=3 options=2a6 type=00 soft=0 mtu=1438 expire=42342/0B replaywin=2048
       seqno=4319 esn=0 replaywin_lastseq=00004319 itn=0 qat=0 hash_search_len=1
  life: type=01 bytes=0/0 timeout=43186/43200
  dec: spi=0aef2a07 esp=aes key=16 12738c8a1db02c23bfed73eb3615a5a1
       ah=sha1 key=20 0f3edd28e3165d184292b4cd397a6edeef9d20dc
  enc: spi=2cb75665 esp=aes key=16 982b418e40f0bb18b89916d8c92270c0
       ah=sha1 key=20 08cbf9bf78a968af5cd7647dfa2a0db066389929
  dec:pkts/bytes=17176/1442784, enc:pkts/bytes=17176/2610752
  npu_flag=00 npu_rgwy=172.16.200.1 npu_lgwy=172.16.200.4 npu_selid=6 dec_npuid=0 enc_
npuid=0
------------------------------------------------------
name=server1_1 ver=1 serial=a 172.16.200.4:500->172.16.200.3:500 tun_id=172.16.200.3
dst_mtu=0 dpd-link=on remote_location=2.2.2.2 weight=1
bound_if=4 lgwy=static/1 tun=tunnel/15 mode=dial_inst/3 encap=none/4744 options
[1288]=npu rgwy-chg frag-rfc  run_state=0 accept_traffic=1 overlay_id=0

parent=server1 index=1
proxyid_num=1 child_num=0 refcnt=5 ilast=27 olast=27 ad=/0
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-idle on=1 idle=60000ms retry=3 count=0 seqno=0
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=server1 proto=0 sa=1 ref=2 serial=1 add-route
  src: 0:0.0.0.0-255.255.255.255:0
  dst: 0:0.0.0.0-255.255.255.255:0
  SA:  ref=3 options=2a6 type=00 soft=0 mtu=1280 expire=43167/0B replaywin=2048
       seqno=1 esn=0 replaywin_lastseq=00000000 itn=0 qat=0 hash_search_len=1
  life: type=01 bytes=0/0 timeout=43187/43200
  dec: spi=0aef2a0a esp=aes key=16 4b7a17ba9d239e4ae5fe95ec100fca8b
```

```
        ah=sha1 key=20 7d3e058088f21e0c4f1c13c297293f06c8b592e7
  enc: spi=7e961809 esp=aes key=16 ecd1aa8657c5a509662aed45002d3990
        ah=sha1 key=20 d159e06c1cf0ded18a4e4ac86cbe5aa0315c21c9
  dec:pkts/bytes=0/0, enc:pkts/bytes=0/0
  npu_flag=00 npu_rgwy=172.16.200.3 npu_lgwy=172.16.200.4 npu_selid=9 dec_npuid=0 enc_
npuid=0
------------------------------------------------------
name=server2_0 ver=1 serial=7 173.1.1.1:500->11.101.1.1:500 tun_id=11.101.1.1 dst_
mtu=1500 dpd-link=on remote_location=1.1.1.1 weight=1
bound_if=17 lgwy=static/1 tun=tunnel/15 mode=dial_inst/3 encap=none/4744 options
[1288]=npu rgwy-chg frag-rfc  run_state=0 accept_traffic=1 overlay_id=0

parent=server2 index=0
proxyid_num=1 child_num=0 refcnt=5 ilast=45 olast=45 ad=/0
stat: rxp=16001 txp=17179 rxb=2113664 txb=1594824
dpd: mode=on-idle on=1 idle=60000ms retry=3 count=0 seqno=12
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=server2 proto=0 sa=1 ref=2 serial=1 add-route
  src: 0:0.0.0.0-255.255.255.255:0
  dst: 0:10.1.100.0-10.1.100.255:0
  SA:  ref=6 options=2a6 type=00 soft=0 mtu=1438 expire=42342/0B replaywin=2048
        seqno=431a esn=0 replaywin_lastseq=00003e80 itn=0 qat=0 hash_search_len=1
  life: type=01 bytes=0/0 timeout=43185/43200
  dec: spi=0aef2a08 esp=aes key=16 394d4e444e90ccb5184e744d49aabe3c
        ah=sha1 key=20 faabea35c2b9b847461cbd263c4856cfb679f342
  enc: spi=2cb75666 esp=aes key=16 0b3a2fbac4d5610670843fa1925d1207
        ah=sha1 key=20 97e99beff3d8f61a8638f6ef887006a9c323acd4
  dec:pkts/bytes=16001/2113596, enc:pkts/bytes=17179/2762792
  npu_flag=03 npu_rgwy=11.101.1.1 npu_lgwy=173.1.1.1 npu_selid=7 dec_npuid=1 enc_npuid=1
```

2. List the IPsec aggregate members:

```
# diagnose sys ipsec-aggregate list
server
members(3):
        server1_1
        server1_0
        server2_0
```

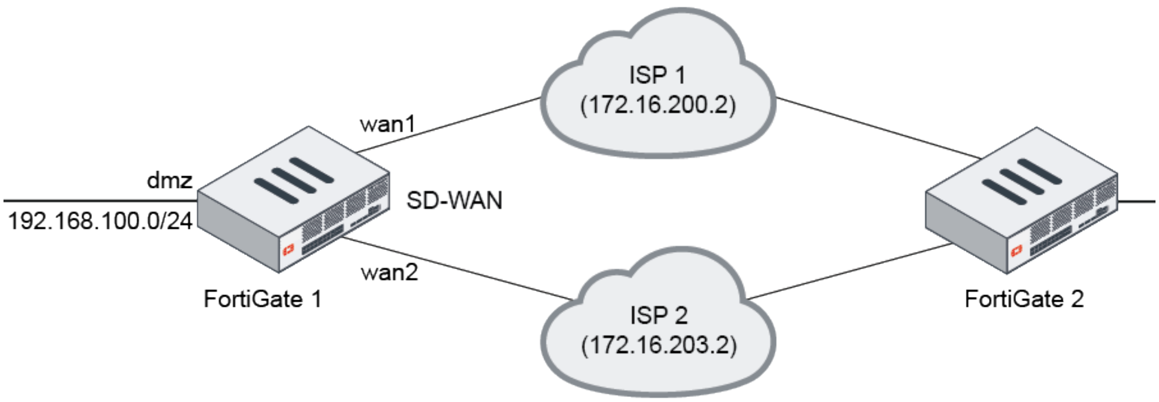
3. In the GUI, go to *Dashboard* > *Network* and expand the *IPsec* widget to review the traffic distributed over the aggregate members:



Packet distribution for aggregate static IPsec tunnels in SD-WAN

This is a sample configuration of aggregating IPsec tunnels by using per-packet load-balancing.

For example, a customer has two ISP connections, wan1 and wan2. On each FortiGate, two IPsec VPN interfaces are created. Next, an `ipsec-aggregate` interface is created and added as an SD-WAN member.



Configuring FortiGate 1

To create two IPsec VPN interfaces:

```
config vpn ipsec phase1-interface
    edit "vd1-p1"
        set interface "wan1"
        set peertype any
        set net-device disable
        set aggregate-member enable
        set proposal aes256-sha256
        set dhgrp 14
        set remote-gw 172.16.201.2
        set psksecret ftnt1234
    next
    edit "vd1-p2"
        set interface "wan2"
        set peertype any
        set net-device disable
        set aggregate-member enable
        set proposal aes256-sha256
        set dhgrp 14
        set remote-gw 172.16.202.2
        set psksecret ftnt1234
    next
end
```

```
config vpn ipsec phase2-interface
    edit "vd1-p1"
        set phase1name "vd1-p1"
    next
    edit "vd1-p2"
        set phase1name "vd1-p2"
    next
end
```

To create an IPsec aggregate interface:

```
config system ipsec-aggregate
    edit "agg1"
        set member "vd1-p1" "vd1-p2"
```

```
        set algorithm L3
    next
end
```

```
config system interface
    edit "agg1"
        set vdom "root"
        set ip 172.16.11.1 255.255.255.255
        set allowaccess ping
        set remote-ip 172.16.11.2 255.255.255.255
end
```

To configure the firewall policy:

```
config firewall policy
    edit 1
        set name "1"
        set srcintf "dmz"
        set dstintf ""virtual-wan-link""
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set nat enable
    next
end
```

To configure SD-WAN:

```
config system sdwan
    set status enable
    config members
        edit 1
            set interface "agg1"
            set gateway 172.16.11.2
        next
    end
end
```

Configuring FortiGate 2

To create two IPsec VPN interfaces:

```
config vpn ipsec phase1-interface
    edit "vd2-p1"
        set interface "wan1"
        set peertype any
        set net-device disable
        set proposal aes256-sha256
        set dhgrp 14
        set remote-gw 172.16.200.1
        set psksecret ftnt1234
    next
    edit "vd2-p2"
```

```
        set interface "wan2"
        set peertype any
        set net-device disable
        set proposal aes256-sha256
        set dhgrp 14
        set remote-gw 172.16.203.1
        set psksecret ftnt1234
    next
end
```

```
config vpn ipsec phase2-interface
    edit "vd2-p1"
        set phase1name "vd2-p1"
    next
    edit "vd2-p2"
        set phase1name "vd2-p2"
    next
end
```

To create an IPsec aggregate interface:

```
config system ipsec-aggregate
    edit "agg2"
        set member "vd2-p1" "vd2-p2"
        set algorithm L3
    next
end
```

```
config system interface
    edit "agg2"
        set vdom "root"
        set ip 172.16.11.2 255.255.255.255
        set allowaccess ping
        set remote-ip 172.16.11.1 255.255.255.255
    next
end
```

To configure the firewall policy:

```
config firewall policy
    edit 1
        set name "1"
        set srcintf "dmz"
        set dstintf ""virtual-wan-link""
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set nat enable
    next
end
```

To configure SD-WAN:

```
config system sdwan
    set status enable
    config members
        edit 1
            set interface "agg2"
            set gateway 172.16.11.1
        next
    end
end
```

Related diagnose commands

To display aggregate IPsec members:

```
# diagnose sys ipsec-aggregate list
agg1 algo=L3 member=2 run_tally=2
members:
        vd1-p1
        vd1-p2
```

To check the VPN status:

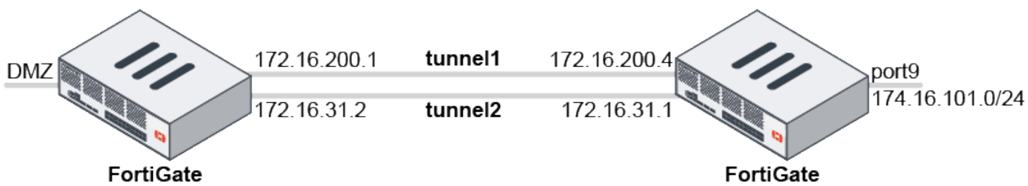
```
# diagnose vpn tunnel list
list all ipsec tunnel in vd 0
------------------------------------------------------
name=vd1-p1 ver=1 serial=2 172.16.200.1:0->172.16.201.2:0 tun_id=172.16.201.2 dst_mtu=0
bound_if=10 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/520 options[0208]=npu frag-rfc
run_state=1 accept_traffic=0

proxyid_num=1 child_num=0 refcnt=5 ilast=15 olast=676 ad=/0
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-demand on=1 idle=20000ms retry=3 count=0 seqno=0
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=vd1-p1 proto=0 sa=0 ref=1 serial=1
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
------------------------------------------------------
name=vd1-p2 ver=1 serial=3 172.16.203.1:0->172.16.202.2:0 tun_id=172.16.202.2 dst_mtu=1500
bound_if=28 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/520 options[0208]=npu frag-rfc
run_state=1 accept_traffic=1

proxyid_num=1 child_num=0 refcnt=12 ilast=1 olast=1 ad=/0
stat: rxp=1 txp=1686 rxb=16602 txb=111717
dpd: mode=on-demand on=1 idle=20000ms retry=3 count=0 seqno=0
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=vd1-p2 proto=0 sa=1 ref=9 serial=1
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=4 options=10226 type=00 soft=0 mtu=1438 expire=42164/0B replaywin=2048
       seqno=697 esn=0 replaywin_lastseq=00000002 itn=0 qat=0
  life: type=01 bytes=0/0 timeout=42902/43200
  dec: spi=f6ae9f83 esp=aes key=16 f6855c72295e3c5c49646530e6b96002
       ah=sha1 key=20 f983430d6c161d0a4cd9007c7ae057f1ff011334
```

```
  enc: spi=8c72ba1a esp=aes key=16 6330f8c532a6ca5c5765f6a9a6034427
       ah=sha1 key=20 e5fe385ed5f0f6a33f1d507601b15743a8c70187
  dec:pkts/bytes=1/16536, enc:pkts/bytes=1686/223872
  npu_flag=02 npu_rgwy=172.16.202.2 npu_lgwy=172.16.203.1 npu_selid=2 dec_npuid=1 enc_
npuid=0
```

Packet distribution for aggregate IPsec tunnels using weighted round robin

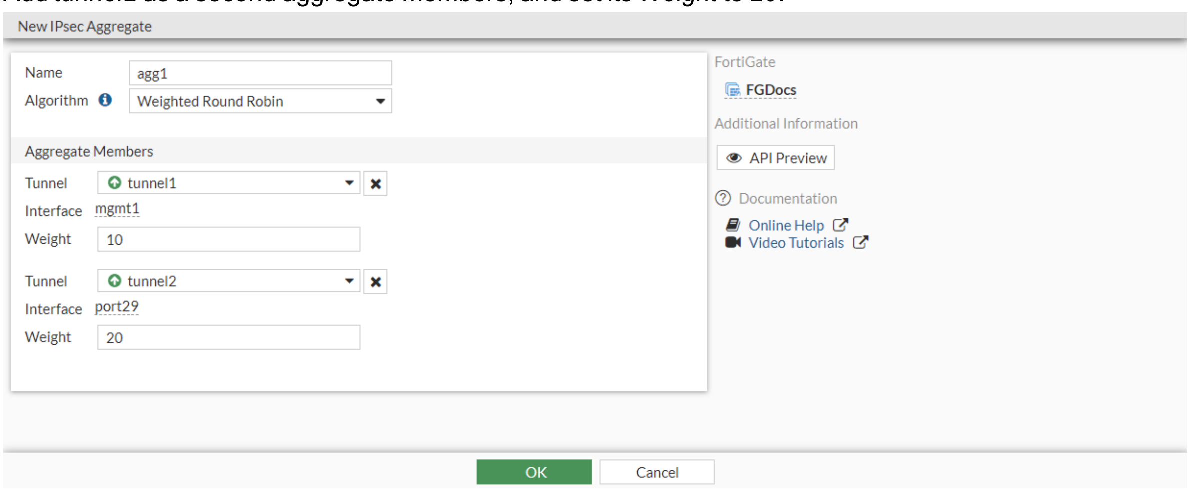
A weighted round robin algorithm can be used for IPsec aggregate tunnels to distribute traffic by the weight of each member tunnel.

In this example, the FortiGate has two IPsec tunnels put into IPsec aggregate. Traffic is distributed among the members, with one third over *tunnel1*, and two thirds over *tunnel2*. To achieve this, the weighted round robin algorithm is selected, *tunnel1* is assigned a weight of 10, and *tunnel2* is assigned a weight of 20.



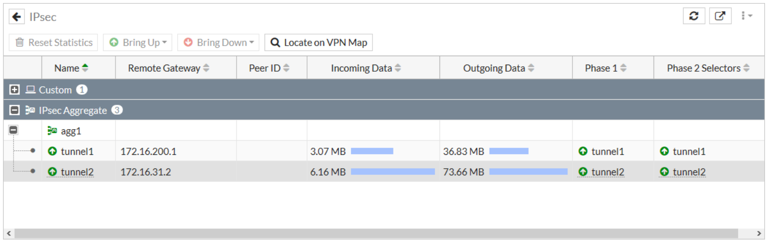
To create the IPsec aggregate in the GUI:

1. Go to *VPN > IPsec Tunnels* and click *Create New > IPsec Tunnel*.
2. Complete the wizard to create the *tunnel1* and *tunnel2* custom IPsec tunnels. Ensure that *Aggregate member* is *Enabled* for each tunnel under the *Network > Advanced* section.
3. Go to *VPN > IPsec Tunnels* and click *Create New > IPsec Aggregate*.
4. Enter a name for the aggregate, such as *agg1*, and ensure that *Algorithm* is *Weighted Round Robin*.
5. Add *tunnel1* as an aggregate members, and set *Weight* to *10*.
6. Add *tunnel2* as a second aggregate members, and set its *Weight* to *20*.



7. Click *OK*.

8. To view and monitor the aggregate tunnel statistics, go to the *IPsec* widget on the *Network* dashboard.



To create the IPsec aggregate in the CLI:

1. Create the *tunnel1* and *tunnel2* custom IPsec tunnels with aggregate-member enabled and aggregate-weight set for both tunnels:

```
config vpn ipsec phase1-interface
    edit "tunnel1"
        ...
        set aggregate-member enable
        set aggregate-weight 10
        ...
    next
    edit "tunnel2"
        ...
        set aggregate-member enable
        set aggregate-weight 20
        ...
    next
end
```

2. Create the IPsec aggregate:

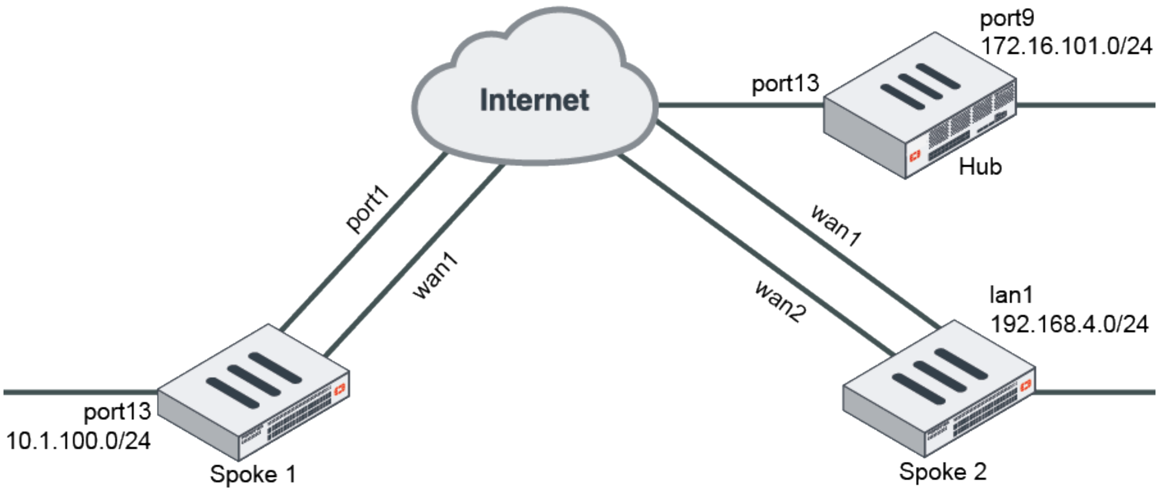
```
config system ipsec-aggregate
    edit "agg1"
        set member "tunnel1" "tunnel2"
        set algorithm weighted-round-robin
    next
end
```

Redundant hub and spoke VPN

A redundant hub and spoke configuration allows VPN connections to radiate from a central FortiGate unit (the hub) to multiple remote peers (the spokes). Traffic can pass between private networks behind the hub and private networks behind the remote peers. Traffic can also pass between remote peer private networks through the hub.

This is a sample configuration of hub and spoke IPsec VPN. The following applies for this scenario:

- The spokes have two WAN interfaces and two IPsec VPN tunnels for redundancy.
- The secondary VPN tunnel is up only when the primary tunnel is down by dead peer detection.



Because the GUI can only complete part of the configuration, we recommend using the CLI.

To configure redundant hub and spoke VPN using the FortiOS CLI:

1. Configure the hub.
   a. Configure the WAN, internal interface, and static route.
```
config system interface
   edit "port13"
      set alias "WAN"
      set ip 172.16.202.1 255.255.255.0
   next
   edit "port9"
      set alias "Internal"
      set ip 172.16.101.1 255.255.255.0
   next
end
config router static
   edit 1
      set gateway 172.16.202.2
      set device "port13"
   next
end
```
   b. Configure the IPsec phase1-interface and phase2-interface.
```
config vpn ipsec phase1-interface
   edit "hub"
      set type dynamic
      set interface "port13"
      set peertype any
      set net-device enable
      set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
      set dpd on-idle
      set psksecret sample
      set dpd-retryinterval 60
   next
end
config vpn ipsec phase2-interface
   edit "hub"
      set phase1name "hub"
```

```
      set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
          aes256gcm chacha20poly1305
   next
end
```

c. Configure the firewall policy.

```
config firewall policy
   edit 1
      set name "spoke-hub"
      set srcintf "hub"
      set dstintf "port9"
      set srcaddr "all"
      set dstaddr "172.16.101.0"
      set action accept
      set schedule "always"
      set service "ALL"
   next
   edit 2
      set name "spoke-spoke"
      set srcintf "hub"
      set dstintf "hub"
      set srcaddr "all"
      set dstaddr "all"
      set action accept
      set schedule "always"
      set service "ALL"
   next
end
```

2. Configure the spokes.
   a. Configure the WAN, internal interface, and static route.
      i. Configure Spoke1.

```
config system interface
   edit "port1"
      set ip 172.16.200.1 255.255.255.0
   next
   edit "wan1"
      set mode dhcp
      set distance 10
      set priority 100
   next
   edit "dmz"
      set ip 10.1.100.1 255.255.255.0
   next
end
config router static
   edit 1
      set gateway 172.16.200.2
      set device "port1"
   next
end
```

      ii. Configure Spoke2.

```
config system interface
   edit "wan1"
      set ip 172.16.200.3 255.255.255.0
   next
   edit "wan2"
      set mode dhcp
```

```
      set distance 10
      set priority 100
    next
    edit "lan1"
      set ip 192.168.4.1 255.255.255.0
    next
  end
  config router static
    edit 1
      set gateway 172.16.200.2
      set device "wan1"
    next
  end
```

b. Configure IPsec phase1-interface and phase2-interface.

   i. Configure Spoke1.

```
config vpn ipsec phase1-interface
  edit "primary"
    set interface "port1"
    set peertype any
    set net-device enable
    set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
    set remote-gw 172.16.202.1
    set psksecret sample
    next
  edit "secondary"
    set interface "wan1"
    set peertype any
    set net-device enable
    set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
    set remote-gw 172.16.202.1
    set monitor "primary"
    set psksecret sample
    next
end
config vpn ipsec phase2-interface
  edit "primary"
    set phase1name "primary"
    set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
        aes256gcm chacha20poly1305
    set auto-negotiate enable
    set src-subnet 10.1.100.0 255.255.255.0
  next
  edit "secondary"
    set phase1name "secondary"
    set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
        aes256gcm chacha20poly1305
    set auto-negotiate enable
    set src-subnet 10.1.100.0 255.255.255.0
  next
end
```

   ii. Configure Spoke2.

```
config vpn ipsec phase1-interface
  edit "primary"
    set interface "wan1"
    set peertype any
    set net-device enable
```

```
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set remote-gw 172.16.202.1
        set psksecret sample
        next
    edit "secondary"
        set interface "wan2"
        set peertype any
        set net-device enable
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set remote-gw 172.16.202.1
        set monitor "primary"
        set psksecret sample
        next
end
config vpn ipsec phase2-interface
    edit "primary"
        set phase1name "primary"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
            aes256gcm chacha20poly1305
        set auto-negotiate enable
        set src-subnet 192.168.4.0 255.255.255.0
    next
    edit "secondary"
        set phase1name "secondary"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
            aes256gcm chacha20poly1305
        set auto-negotiate enable
        set src-subnet 192.168.4.0 255.255.255.0
    next
end
```

c. Configure the firewall policy.

i. Configure Spoke1.

```
config firewall policy
    edit 1
        set srcintf "dmz"
        set dstintf "primary" "secondary"
        set srcaddr "10.1.100.0"
        set dstaddr "172.16.101.0"
        set action accept
        set schedule "always"
        set service "ALL"
    next
end
```

ii. Configure Spoke2.

```
config firewall policy
    edit 1
        set srcintf "lan1"
        set dstintf "primary" "secondary"
        set srcaddr "192.168.4.0"
        set dstaddr "172.16.101.0"
        set action accept
        set schedule "always"
        set service "ALL"
    next
end
```

d. Configure the static route.

i. Configure Spoke1.

```
config router static
    edit 3
        set dst 172.16.101.0 255.255.255.0
        set distance 1
        set device "primary"
    next
    edit 4
        set dst 172.16.101.0 255.255.255.0
        set distance 3
        set device "secondary"
    next
end
```

ii. Configure Spoke2.

```
config router static
    edit 3
        set dst 172.16.101.0 255.255.255.0
        set distance 1
        set device "primary"
    next
    edit 4
        set dst 172.16.101.0 255.255.255.0
        set distance 3
        set device "secondary"
    next
end
```

3. Run `diagnose` and `get` commands.

a. Run the `Spoke1 # diagnose vpn tunnel list` command. The system should return the following:

```
name=primary ver=1 serial=1 172.16.200.1:0->172.16.202.1:0 tun_id=172.16.202.1
bound_if=11 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/528 options[0210]=create_
     dev frag-rfc accept_traffic=1
proxyid_num=1 child_num=0 refcnt=15 ilast=0 olast=0 ad=/0
stat: rxp=1879 txp=1881 rxb=225480 txb=112860
dpd: mode=on-demand on=1 idle=20000ms retry=3 count=0 seqno=1
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=primary proto=0 sa=1 ref=2 serial=2 auto-negotiate
   src: 0:10.1.100.0/255.255.255.0:0 dst: 0:0.0.0.0/0.0.0.0:0 SA: ref=3 options=18227
        type=00 soft=0 mtu=1438 expire=41002/0B replaywin=2048
     seqno=758 esn=0 replaywin_lastseq=00000758 itn=0
   life: type=01 bytes=0/0 timeout=42901/43200 dec: spi=0908732f esp=aes key=16
        20770dfe67ea22dd8ec32c44d84ef4d5
     ah=sha1 key=20 edc89fc2ec06309ba13de95e7e486f9b795b8707
   enc: spi=a1d9eed1 esp=aes key=16 8eeea2526fba062e680d941083c8b5d1
     ah=sha1 key=20 f0f5deaf88b2a69046c3154e9f751739b3f411f5
   dec:pkts/bytes=1879/112740, enc:pkts/bytes=1879/225480
name=secondary ver=1 serial=2 172.17.200.1:0->172.16.202.1:0 tun_id=172.16.202.1
bound_if=5 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/528 options[0210]=create_
     dev frag-rfc accept_traffic=0
proxyid_num=1 child_num=0 refcnt=10 ilast=1892 olast=1892 ad=/0
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-demand on=0 idle=20000ms retry=3 count=0 seqno=0
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=secondary proto=0 sa=0 ref=2 serial=2 auto-negotiate
   src: 0:10.1.100.0/255.255.255.0:0 dst: 0:0.0.0.0/0.0.0.0:0
```

b. Run the `Spoke1 # get router info routing-table static` command. The system should return the following:

```
Routing table for VRF=0
................
S 172.16.101.0/24 [1/0] is directly connected, primary
```

Overlay Controller VPN (OCVPN)

Overlay Controller VPN (OCVPN) is a cloud based solution to simplify IPsec VPN setup. When OCVPN is enabled, IPsec phase1-interfaces, phase2-interfaces, static routes, and firewall policies are generated automatically on all FortiGates that belong to the same community network. A community network is defined as all FortiGates registered to FortiCare using the same FortiCare account.

If the network topology changes on any FortiGates in the community (such as changing a public IP address in DHCP mode, adding or removing protected subnets, failing over in dual WAN), the IPsec-related configuration for all devices is updated with Cloud assistance in self-learning mode. No intervention is required.

The following topics provide instructions on configuring OCVPN:

- Full mesh OCVPN on page 1499
- Hub-spoke OCVPN with ADVPN shortcut on page 1504
- Hub-spoke OCVPN with inter-overlay source NAT on page 1508
- OCVPN portal on page 1512
- SD-WAN integration with OCVPN on page 648
- Allow FortiClient to join OCVPN on page 1513
- Troubleshooting OCVPN on page 1517

Full mesh OCVPN

This example shows how to configure a full mesh Overlay Controller VPN (OCVPN), establishing full mesh IPsec tunnels between all of the FortiGates.

License

- Free license: Three devices full mesh, 10 overlays, 16 subnets per overlay.
- Full License: Maximum of 16 devices, 10 overlays, 16 subnets per overlay.

Prerequisites

- All FortiGates must be running FortiOS 6.2.0 or later.
- All FortiGates must have Internet access.
- All FortiGates must be registered on FortiCare using the same FortiCare account.

Restrictions

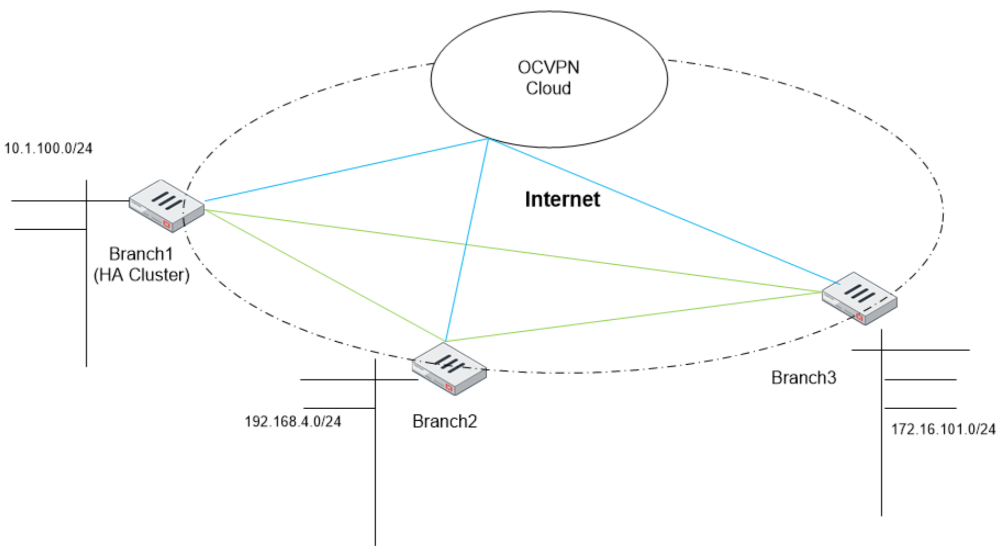
- Non-root VDOMs do not support OCVPN.
- FortiOS 6.2.x is not compatible with FortiOS 6.0.x.

Terminology

Poll-interval	How often FortiGate tries to fetch OCVPN-related data from OCVPN Cloud.
Role	The device OCVPN role of spoke, primary-hub, or secondary-hub.
Overlay	Defines network overlays and bind to subnets.
Subnet	Internal network subnet (IPsec protected subnet). Traffic to or from this subnet enters the IPsec tunnel encrypted by IPsec SA.

Sample topology

The following example shows three FortiGate units registered on FortiCare using the same FortiCare account. Each FortiGate unit has one internal subnet, and no NAT exists between the units.



Sample configuration

The following overlays and subnets are used:

- Branch1:
  - Overlay name: QA. Local subnets: 10.1.100.0/24
  - Overlay name: PM. Local subnets: 10.2.100.0/24
- Branch2:
  - Overlay name: QA. Local interfaces: lan1
  - Overlay name: PM. Local interfaces: lan2
- Branch3:
  - Overlay name: QA. Local subnets: 172.16.101.0/24
  - Overlay name: PM. Local subnets: 172.16.102.0/24

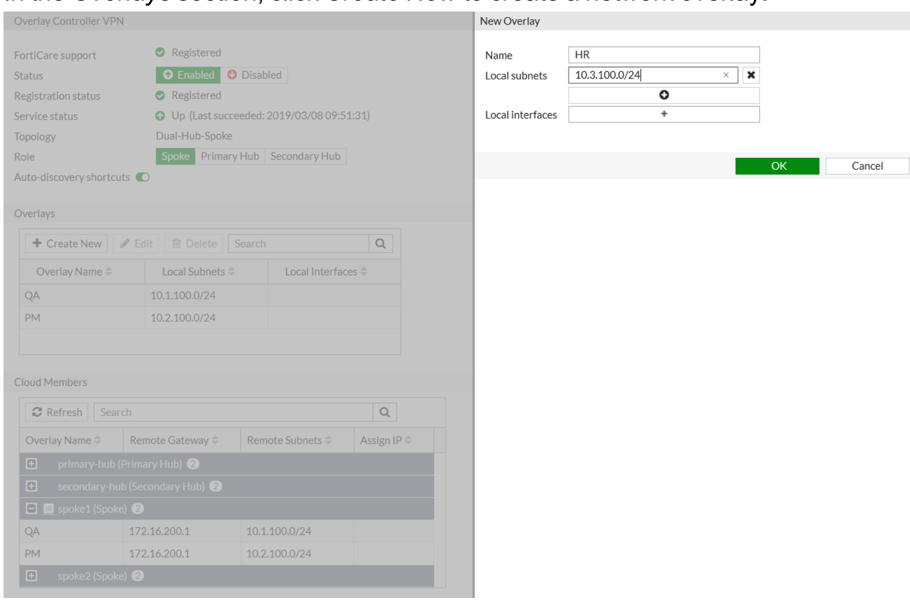
The overlay names on each device must be the same for local and remote selector pairs to be negotiated.

To register FortiGates on FortiCare:

1. Go to *System* > *FortiGuard*.
2. In the *License Information* section, click *Login to FortiCare*. The registration pane opens.
3. Enter the required information (email address, password, country/region, reseller).
4. Optionally, enable *Sign in to FortiGate Cloud using the same account*.
5. Click *OK*.

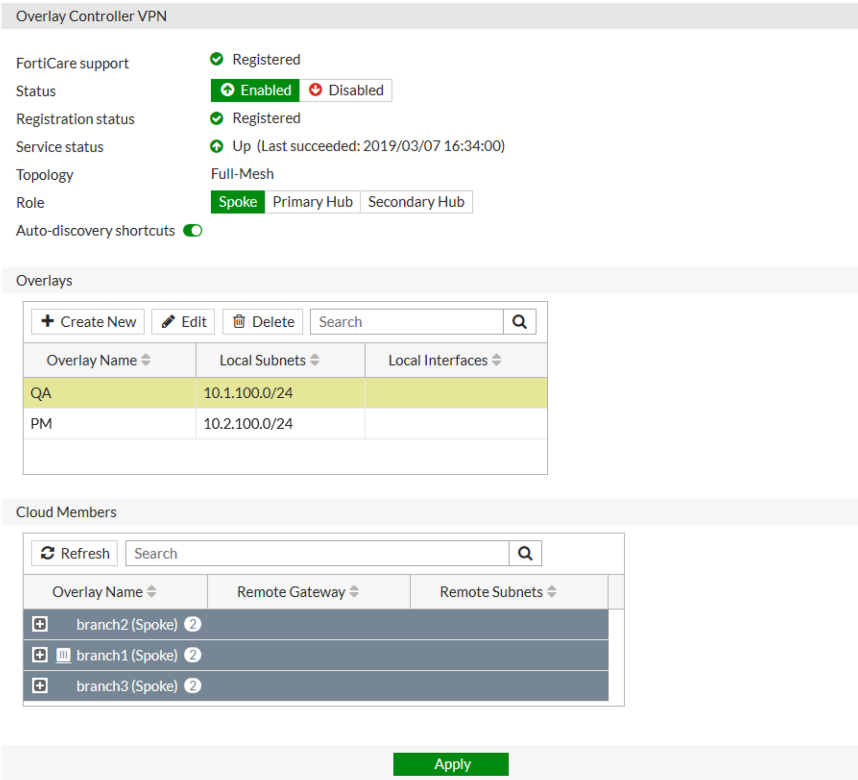
To enable OCVPN in the GUI:

1. Go to *VPN* > *Overlay Controller VPN*.
2. Create the first overlay by setting the following options:
   a. For *Status*, click *Enabled*.
   b. For *Role*, click *Spoke*.
   c. In the *Overlays* section, click *Create New* to create a network overlay.



3. Specify the *Name*, *Local subnets*, and/or *Local interfaces*.
   The local subnet must be routable and interfaces must have IP addresses.

4. Click *OK*.



5. Click *Apply* to commit the configuration.
6. Repeat this procedure to create all the overlays.

To enable OCVPN in the CLI:

1. Configure Branch1:

```
config vpn ocvpn
    set status enable
    set multipath disable
    config overlays
        edit 1
            set name "QA"
            config subnets
                edit 1
                    set subnet 10.1.100.0 255.255.255.0
                next
            end
        next
        edit 2
            set name "PM"
            config subnets
                edit 1
                    set subnet 10.2.100.0 255.255.255.0
                next
            end
        next
```

```
    end
end
```

2. Configure Branch2:

```
config vpn ocvpn
    set status enable
    set multipath disable
    config overlays
         edit 1
             set name "QA"
             config subnets
                  edit 1
                        set type interface
                        set interface "lan1"
                  next
             end
         next
         edit 2
             set name "PM"
             config subnets
                  edit 1
                        set type interface
                        set interface "lan2"
                  next
             end
         next
    end
end
```

3. Configure Branch3:

```
config vpn ocvpn
    set status enable
    set multipath disable
    config overlays
         edit 1
             set name "QA"
             config subnets
                 edit 1
                     set subnet 172.16.101.0 255.255.255.0
                 next
             end
         next
         edit 1
             set name "PM"
             config subnets
                 edit 1
                     set subnet 172.16.102.0 255.255.255.0
                 next
             end
         next
    end
end
```

Hub-spoke OCVPN with ADVPN shortcut

This topic shows a sample configuration of a hub-spoke One-Click VPN (OCVPN) with an Auto Discovery VPN (ADVPN) shortcut. OCVPN automatically detects the network topology based on members' information. To form a hub-spoke OCVPN, at least one device must announce its role as the primary hub, another device can work as the secondary hub (for redundancy), while others function as spokes.

License

- Free license: Hub-spoke network topology not supported.
- Full license: Maximum of 2 hubs, 10 overlays, 64 subnets per overlay; 1024 spokes, 10 overlays, 16 subnets per overlay.

Prerequisites

- All FortiGates must be running FortiOS 6.2.0 or later.
- All FortiGates must have Internet access.
- All FortiGates must be registered on FortiCare using the same FortiCare account.

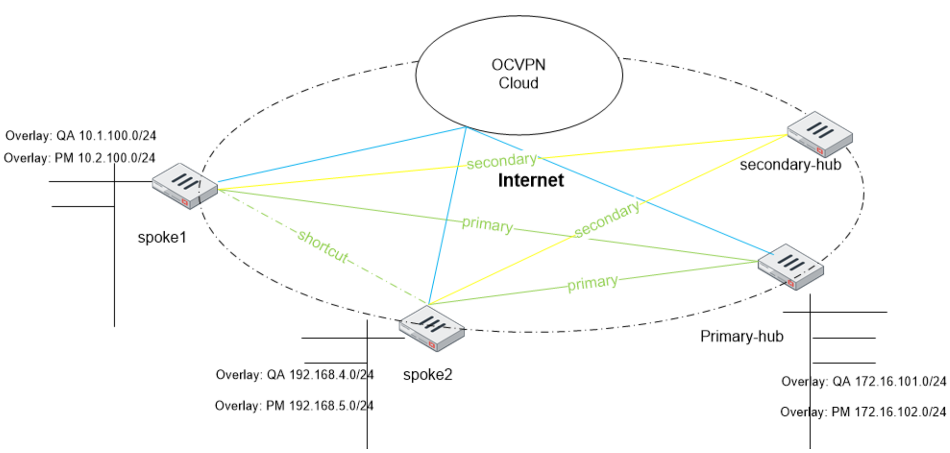
Restrictions

- Non-root VDOMs do not support OCVPN.
- FortiOS 6.2.x is not compatible with FortiOS 6.0.x.

OCVPN device roles

- Primary hub.
- Secondary hub.
- Spoke (OCVPN default role).

Sample topology



Sample configuration

The steps below use the following overlays and subnets for the sample configuration:

- Primary hub:
  - Overlay name: QA. Local subnets: 172.16.101.0/24
  - Overlay name: PM. Local subnets: 172.16.102.0/24
- Secondary hub:
  - Overlays are synced from primary hub.
- Spoke1:
  - Overlay name: QA. Local subnets: 10.1.100.0/24
  - Overlay name: PM. Local subnets: 10.2.100.0/24
- Spoke2:
  - Overlay name: QA. Local interfaces: lan1
  - Overlay name: PM. Local interfaces: lan2

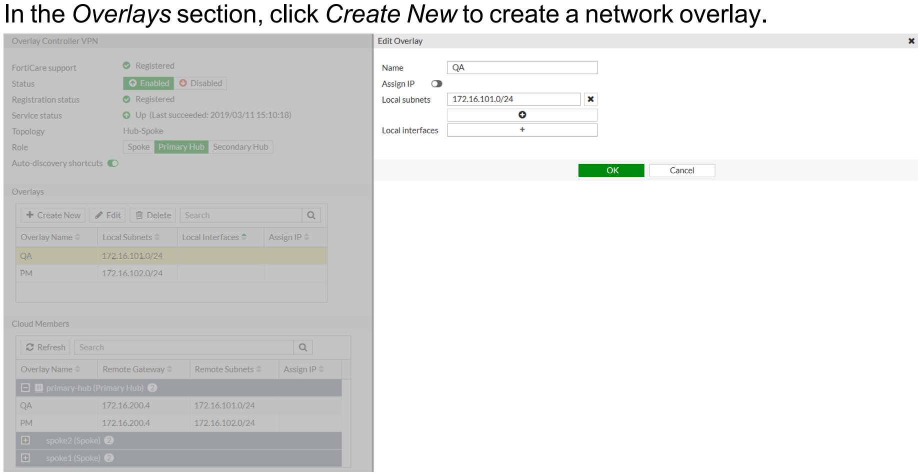
The overlay names on each device must be the same for local and remote selector pairs to be negotiated.

To register FortiGates on FortiCare:

1. Go to *System > FortiGuard*.
2. In the *License Information* section, click *Login to FortiCare*. The registration pane opens.
3. Enter the required information (email address, password, country/region, reseller).
4. Optionally, enable *Sign in to FortiGate Cloud using the same account*.
5. Click *OK*.

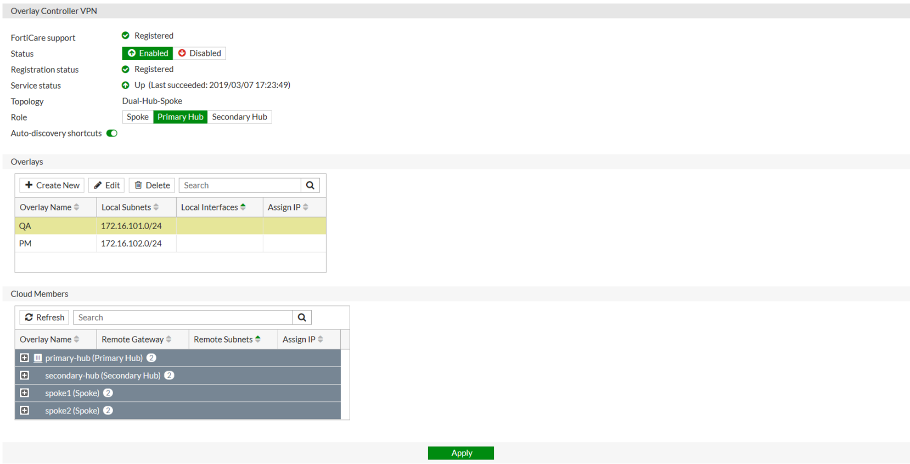
To enable hub-spoke OCVPN in the GUI:

1. Go to *VPN > Overlay Controller VPN*.
2. Configure the OCVPN primary hub by setting the following options:
   a. For *Status*, click *Enabled*.
   b. For *Role*, click *Primary Hub*.
   c. In the *Overlays* section, click *Create New* to create a network overlay.



d. Specify the *Name*, *Local subnets*, and/or *Local interfaces*. Then click *OK*.

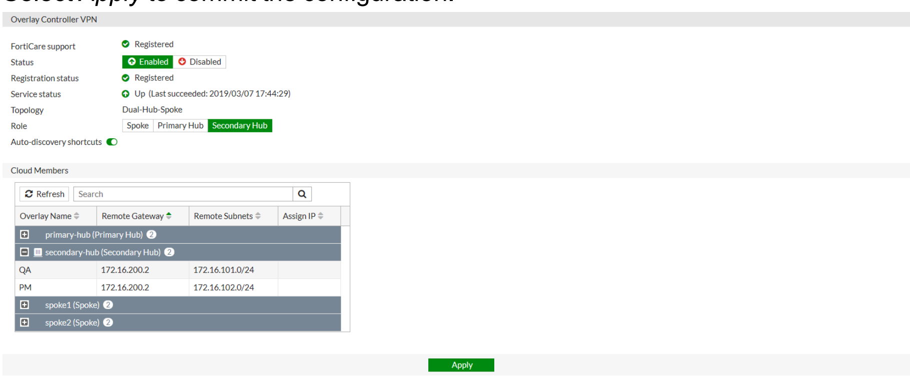
e. Click *Apply* to commit the configuration.



3. Configure the OCVPN secondary hub:
   Overlays are synced from the primary hub and cannot be defined in the secondary hub.

a. In the *Overlay Controller VPN* pane, select *Secondary Hub* for the *Role*.

b. Select *Apply* to commit the configuration.



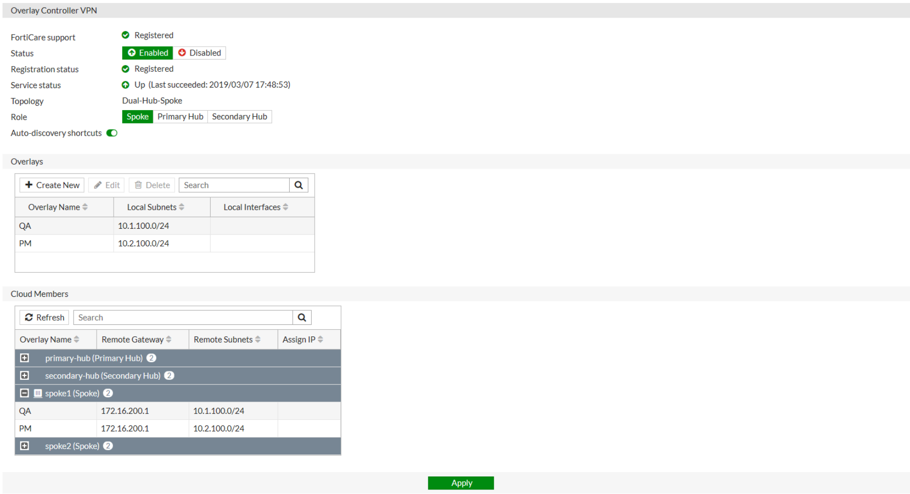
4. Configure the OCVPN spokes:

a. In the *Overlay Controller VPN* pane, select *Spoke* for the *Role*.

b. In the *Overlays* section, click *Create New* to create a network overlay.

c. Specify the *Name*, *Local subnets*, and/or *Local interfaces*.
   The local subnet must be routable and interfaces must have IP addresses.

d. Click *OK* and then click *Apply* to commit the configuration.



To enable hub-spoke OCVPN in the CLI:

1. Configure the OCVPN primary hub:

```
config vpn ocvpn
    set status enable
    set role primary-hub
    config overlays
        edit 1
            set name "QA"
            config subnets
                edit 1
                    set subnet 172.16.101.0 255.255.255.0
                next
            end
        next
        edit 2
            set name "PM"
            config subnets
                edit 1
                    set subnet 172.16.102.0 255.255.255.0
                next
            end
        next
    end
end
```

2. Configure the OCVPN secondary hub:

```
config vpn ocvpn
    set status enable
    set role secondary-hub
end
```

3. Configure the OCVPN spoke1:

```
config vpn ocvpn
    set status enable
    config overlays
```

```
            edit 1
                set name "QA"
                config subnets
                    edit 1
                        set subnet 10.1.100.0 255.255.255.0
                    next
                end
            next
            edit 2
                set name "PM"
                config subnets
                    edit 1
                        set subnet 10.2.100.0 255.255.255.0
                    next
                end
            next
        end
    end
```

4. Configure the OCVPN spoke2:

```
    config vpn ocvpn
        set status enable
        config overlays
            edit 1
                set name "QA"
                config subnets
                    edit 1
                        set subnet 192.168.4.0 255.255.255.0
                    next
                end
            next
            edit 2
                set name "PM"
                config subnets
                    edit 1
                        set subnet 192.168.5.0 255.255.255.0
                    next
                end
            next
        end
    end
```

Hub-spoke OCVPN with inter-overlay source NAT

This topic shows a sample configuration of hub-spoke OCVPN with inter-overlay source NAT. OCVPN isolates traffic between overlays by default. With NAT enabled on spokes and `assign-ip` enabled on hub, you can have inter-overlay communication.

Inter-overlay communication means devices from any source addresses and any source interfaces can communicate with any devices in overlays' subnets when the overlay option `assign-ip` is enabled.

You must first disable `auto-discovery` before you can enable NAT.

License

- Free license: Hub-spoke network topology not supported.
- Full License: Maximum of 2 hubs, 10 overlays, 64 subnets per overlay; 1024 spokes, 10 overlays, 16 subnets per overlay.

Prerequisites

- All FortiGates must be running FortiOS 6.2.0 or later.
- All FortiGates must have Internet access.
- All FortiGates must be registered on FortiCare using the same FortiCare account.

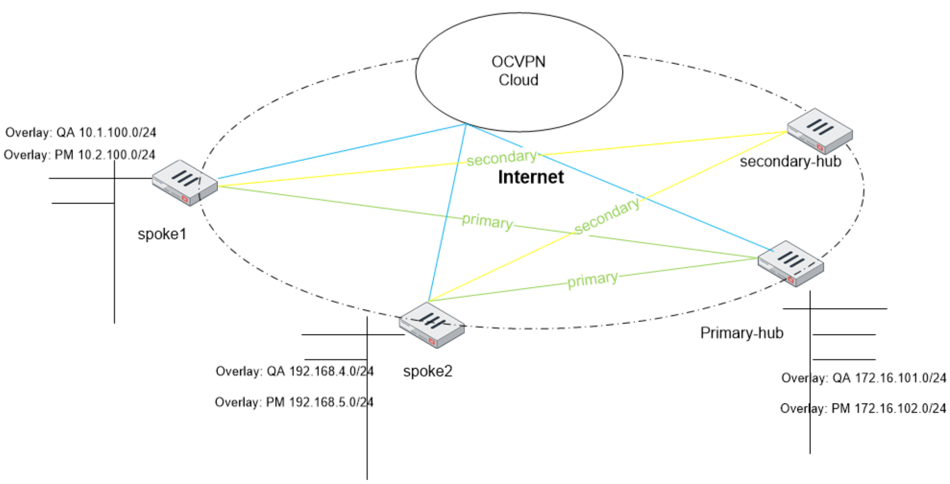
Restrictions

- Non-root VDOMs do not support OCVPN.
- FortiOS 6.2.x is not compatible with FortiOS 6.0.x.

OCVPN device roles

- Primary hub.
- Secondary hub.
- Spoke (OCVPN default role).

Sample topology



Sample configuration

You can only configure this feature using the CLI.

The overlay names on each device must be the same for local and remote selector pairs to be negotiated.

To enable inter-overlay source NAT in the CLI:

1. Configure the primary hub, enable overlay QA, and configure `assign-ip` and IP range:

```
config vpn ocvpn
    set status enable
    set role primary-hub
    config overlays
        edit 1
            set name "QA"
            set assign-ip enable
            set ipv4-start-ip 172.16.101.100
            set ipv4-end-ip 172.16.101.200
            config subnets
                edit 1
                    set subnet 172.16.101.0 255.255.255.0
                next
            end
        next
        edit 2
            set name "PM"
            set assign-ip enable
            config subnets
                edit 1
                    set subnet 172.16.102.0 255.255.255.0
                next
            end
        next
    end
end
```

2. Configure the secondary hub:

```
config vpn ocvpn
    set status enable
    set role secondary-hub
end
```

3. Configure spoke1 and enable NAT on the spoke:

```
config vpn ocvpn
    set status enable
    set auto-discovery disable
    set nat enable
    config overlays
        edit 1
            set name "QA"
            config subnets
                edit 1
                    set subnet 10.1.100.0 255.255.255.0
                next
            end
        next
        edit 2
            set name "PM"
            config subnets
                edit 1
                    set subnet 10.2.100.0 255.255.255.0
```

```
                next
            end
        next
    end
end
```

4. Configure spoke2 and enable NAT on the spoke:

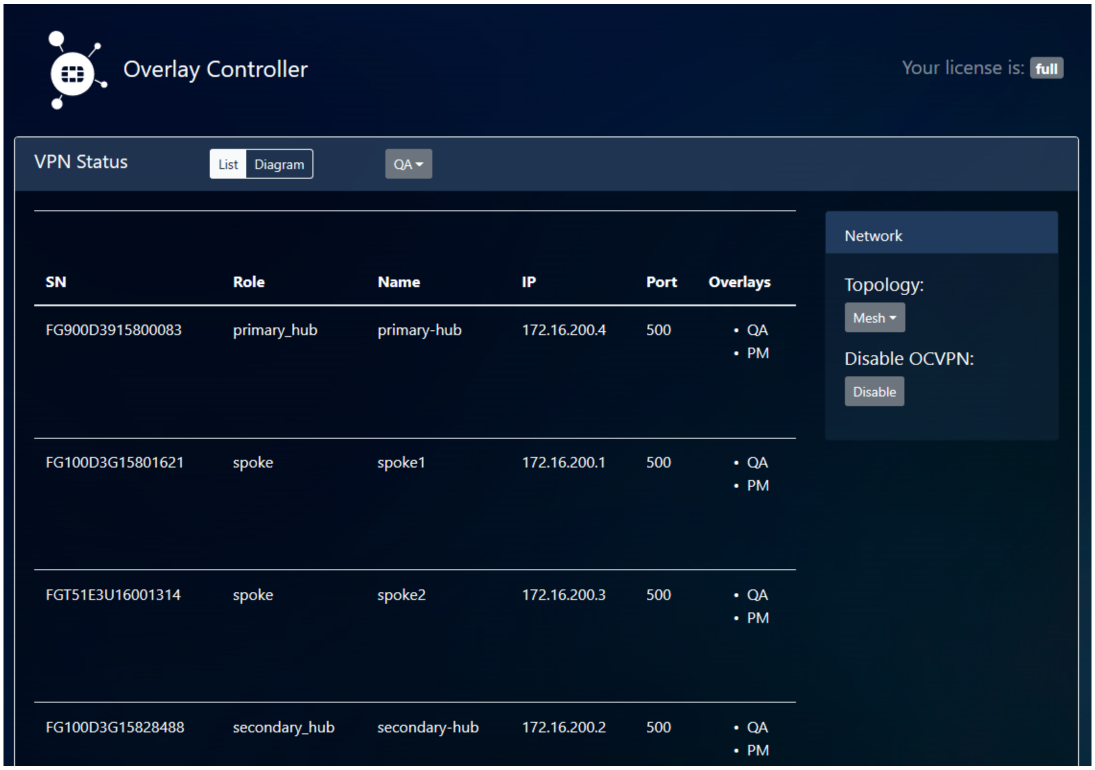
```
config vpn ocvpn
    set status enable
    set auto-discovery disable
    set nat enable
    config overlays
        edit 1
            set name "QA"
            config subnets
                edit 1
                    set subnet 192.168.4.0 255.255.255.0
                next
            end
        next
        edit 2
            set name "PM"
            config subnets
                edit 1
                    set subnet 192.168.5.0 255.255.255.0
                next
            end
        next
    end
end
```

A firewall policy with NAT is generated on the spoke:

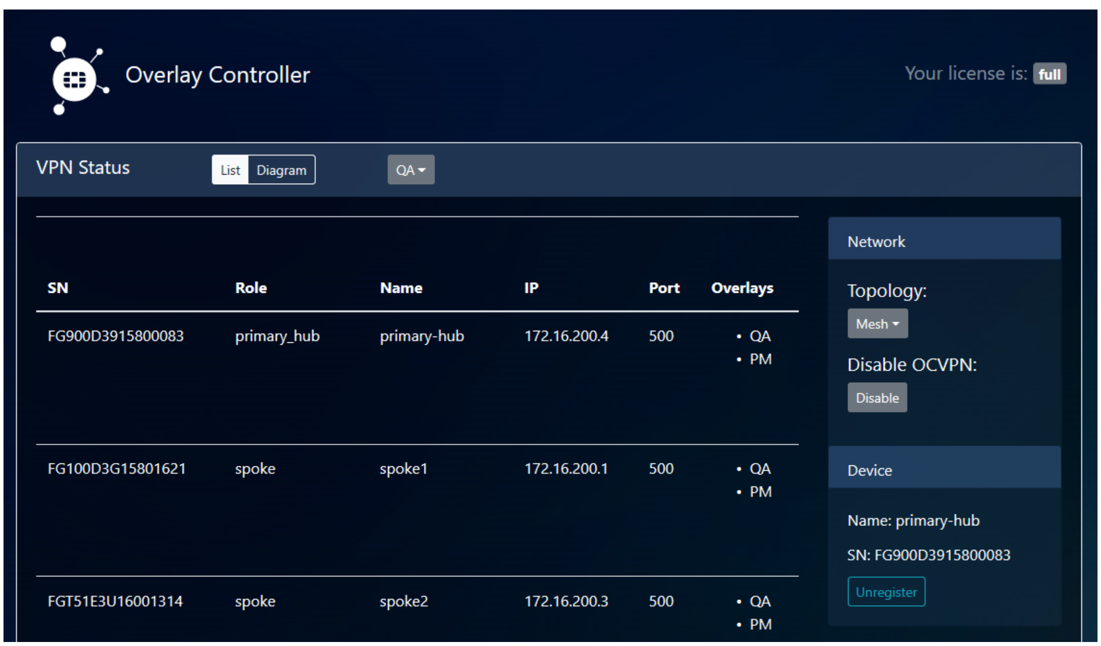
```
edit 9
   set name "_OCVPN2-1.1_nat"
   set uuid 3f7a84b8-3d36-51e9-ee97-8f418c91e666
   set srcintf "any"
   set dstintf "_OCVPN2-1.1"
   set srcaddr "all"
   set dstaddr "_OCVPN2-1.1_remote_networks"
   set action accept
   set schedule "always"
   set service "ALL"
   set comments "Generated by OCVPN Cloud Service."
   set nat enable
next
```

OCVPN portal

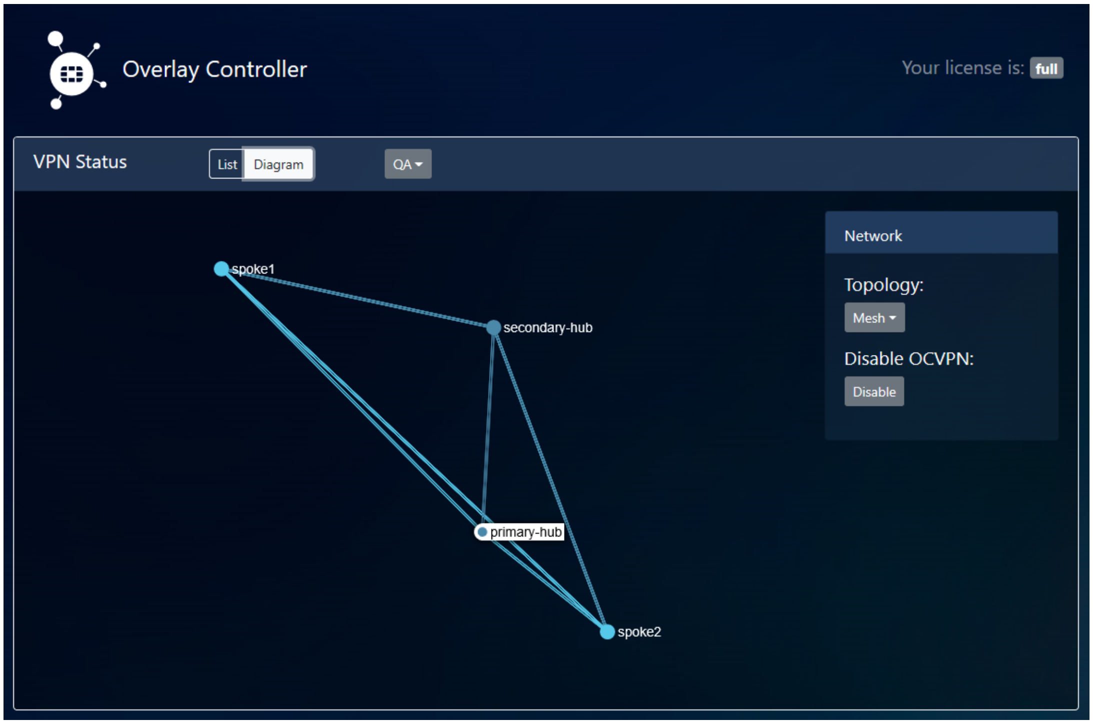
When you log into the OCVPN portal, the OCVPN license type and device information display. The device information includes the device serial number, OCVPN role, hostname, public IP address, port number, and overlays.



You can unregister an OCVPN device from the OCVPN portal under *Device* on the right pane.

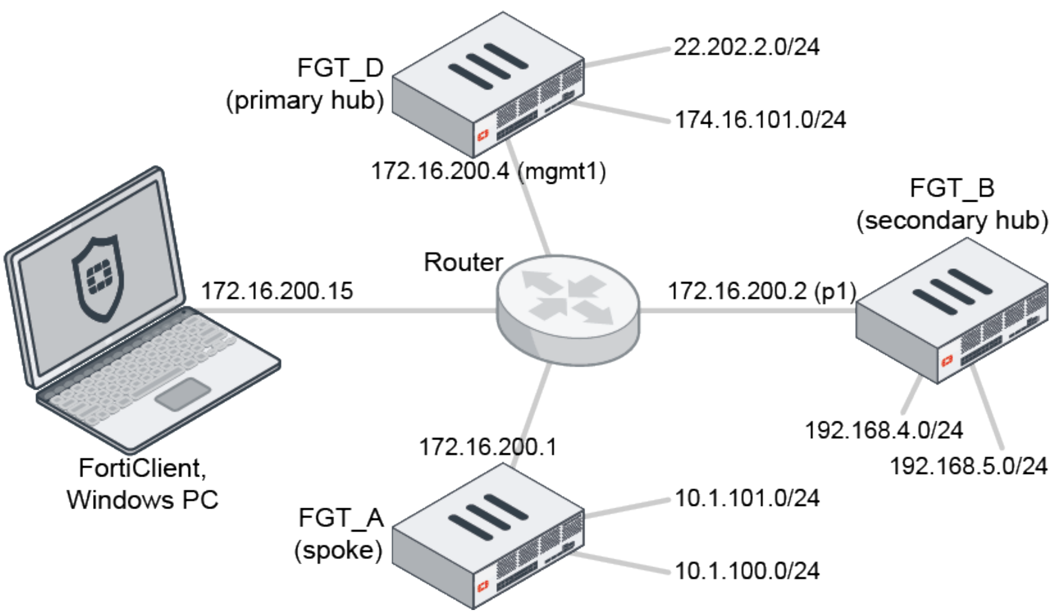


Use the OCVPN *Diagram* to show the OCVPN network topology.



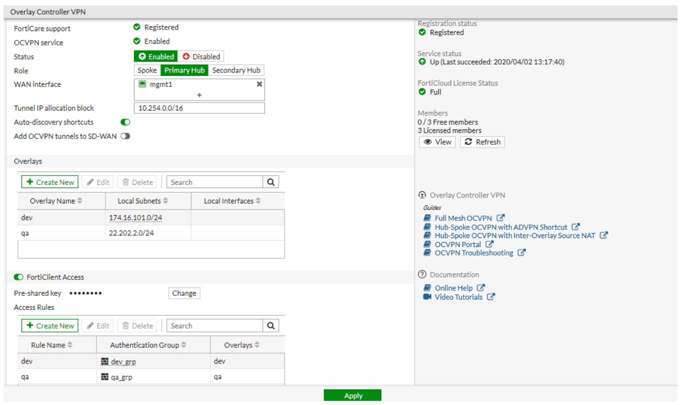
Allow FortiClient to join OCVPN

Administrators can configure remote access for FortiClient within an OCVPN hub. This provides simple configurations to allow a user group access to an overlay network.



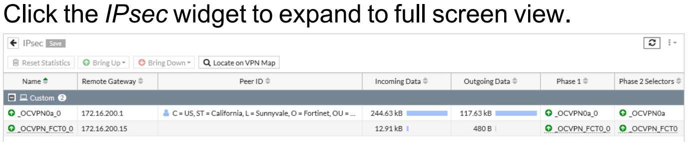
To configure remote FortiClient access to an OCVPN hub in the GUI:

1. On the primary hub, configure the users and user groups required for the FortiClient dialup user authentication and authorization. In this example, there are two user groups (*dev_grp* and *qa_grp*).
2. Go to *VPN > Overlay Controller VPN* and in the *Overlays* section, click *Create New*.
3. Enter a name and the local subnet (174.16.101.0/24 for *dev* and 22.202.2.0/24 for *qa*).
4. Enable *FortiClient Access*.
5. In the *Access Rules* section, click *Create New*.
6. Enter a name, and select the authentication groups and overlays.The authentication groups will be used by the IPsec phase 1 interface for authentication, and by firewall policies for authorization. The overlay allows access to the resource.
7. Click *OK*.
8. Create more rules if needed.
9. Click *Apply*.



To view the tunnel status and activity in the GUI:

1. Go to *Dashboard > Network*.
2. Click the *IPsec* widget to expand to full screen view.



To configure remote FortiClient access to an OCVPN hub in the CLI:

```
config vpn ocvpn
    set status enable
    set role primary-hub
    set wan-interface "mgmt1"
    set ip-allocation-block 10.254.0.0 255.255.0.0
    config overlays
        edit "dev"
            config subnets
                edit 1
                    set subnet 174.16.101.0 255.255.255.0
                next
            end
        next
```

```
        edit "qa"
            config subnets
                edit 1
                    set subnet 22.202.2.0 255.255.255.0
                next
            end
        next
    end
    config forticlient-access
        set status enable
        set psksecret xxxxxxxxxxxx
        config auth-groups
            edit "dev"
                set auth-group "dev_grp"
                set overlays "dev"
            next
            edit "qa"
                set auth-group "qa_grp"
                set overlays "qa"
            next
        end
    end
end
```

To view the tunnel status and activity in the CLI:

```
# diagnose vpn ike gateway list

vd: root/0
name: _OCVPN_FCT0_0
version: 1
interface: mgmt1 4
addr: 172.16.200.4:4500 -> 172.16.200.15:64916
tun_id: 172.16.200.15
created: 110s ago
xauth-user: usera
groups:
  dev_grp 1
assigned IPv4 address: 10.254.128.1/255.255.255.255
nat: peer
IKE SA: created 1/1  established 1/1  time 20/20/20 ms
IPsec SA: created 1/1  established 1/1  time 0/0/0 ms

  id/spi: 72 1ccd2abf2d981123/fd8da107f9e4d312
  direction: responder
  status: established 110-110s ago = 20ms
  proposal: aes256-sha256
  key: 105a0291b0c05219-3decdf78938a7bea-78943651e1720536-625114d66e46f668
  lifetime/rekey: 86400/86019
  DPD sent/recv: 00000000/00000af3
```

To view data on the PC running FortiClient:

```
C:\ route print
===========================================================================
```

```
IPv4 Route Table
===========================================================================
Active Routes:
Network Destination        Netmask          Gateway       Interface  Metric
          0.0.0.0          0.0.0.0      10.1.100.5     10.1.100.13    281
       10.1.100.0    255.255.255.0    10.254.128.2    10.254.128.1      1
      10.1.100.13  255.255.255.255         On-link     10.1.100.13    281
       10.1.101.0    255.255.255.0    10.254.128.2    10.254.128.1      1
        10.6.30.0    255.255.255.0         On-link      10.6.30.13    281
       10.6.30.13  255.255.255.255         On-link      10.6.30.13    281
      10.6.30.255  255.255.255.255         On-link      10.6.30.13    281
       10.254.0.0      255.255.0.0    10.254.128.2    10.254.128.1      1
     10.254.128.1  255.255.255.255         On-link    10.254.128.1    257
       22.202.2.0    255.255.255.0    10.254.128.2    10.254.128.1      1
        127.0.0.0        255.0.0.0         On-link       127.0.0.1    331
        127.0.0.1  255.255.255.255         On-link       127.0.0.1    331
  127.255.255.255  255.255.255.255         On-link       127.0.0.1    331
     172.16.200.4  255.255.255.255      10.1.100.5     10.1.100.13     25
     174.16.101.0    255.255.255.0    10.254.128.2    10.254.128.1      1
        224.0.0.0        240.0.0.0         On-link       127.0.0.1    331
        224.0.0.0        240.0.0.0         On-link    10.254.128.1    257
        224.0.0.0        240.0.0.0         On-link      10.6.30.13    281
        224.0.0.0        240.0.0.0         On-link     10.1.100.13    281
  255.255.255.255  255.255.255.255         On-link       127.0.0.1    331
  255.255.255.255  255.255.255.255         On-link    10.254.128.1    257
  255.255.255.255  255.255.255.255         On-link      10.6.30.13    281
  255.255.255.255  255.255.255.255         On-link     10.1.100.13    281
===========================================================================
Persistent Routes:
  Network Address          Netmask  Gateway Address  Metric
          0.0.0.0          0.0.0.0       10.1.100.5  Default
```

The PC can access the *dev* resource overlay, but not *qa*:

```
C:\Users\tester>ping 174.16.101.44

Pinging 174.16.101.44 with 32 bytes of data:
Reply from 174.16.101.44: bytes=32 time=1ms TTL=63
Reply from 174.16.101.44: bytes=32 time=1ms TTL=63
Reply from 174.16.101.44: bytes=32 time=1ms TTL=63
Reply from 174.16.101.44: bytes=32 time=1ms TTL=63

Ping statistics for 174.16.101.44:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 1ms, Average = 1ms

C:\Users\tester>ping 22.202.2.2

Pinging 22.202.2.2 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 22.202.2.2:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

Troubleshooting OCVPN

This document includes troubleshooting steps for the following OCVPN network topologies:

- Full mesh OCVPN.
- Hub-spoke OCVPN with ADVPN shortcut.
- Hub-spoke OCVPN with inter-overlay source NAT.

For OCVPN configurations in other network topologies, see the other OCVPN topics.

Troubleshooting full mesh network topology

- Branch_1 # `diagnose vpn ocvpn status`

```
Current State        : Registered
Topology             : Full-Mesh
Role                 : Spoke
Server Status        : Up
Registration time    : Thu Feb 28 18:42:25 2019
Update time          : Thu Feb 28 15:57:18 2019
Poll time            : Fri Mar  1 15:02:28 2019
```

- Branch_1 # `diagnose vpn ocvpn show-meta`

```
Topology :: auto
License  :: full
Members  :: 3
Max-free :: 3
```

- Branch_1 # `diagnose vpn ocvpn show-overlays`

```
QA
PM
```

- Branch_1 # `diagnose vpn ocvpn show-members`

```
Member: { "SN": "FG100D3G15801621", "IPv4": "172.16.200.1", "port": "500", "slot": 1000,
"overlay": [ { "id": 0, "name": "QA", "subnets": [ "10.1.100.0\/255.255.255.0" ], "ip_
range": "0.0.0.0-0.0.0.0" }, { "id": 1, "name": "PM", "subnets": [
"10.2.100.0\/255.255.255.0" ], "ip_range": "0.0.0.0-0.0.0.0" } ], "Name": "FortiGate-
100D", "topology_role": "spoke" }
Member: { "SN": "FG900D3915800083", "IPv4": "172.16.200.4", "port": "500", "slot": 1001,
"overlay": [ { "id": 0, "name": "QA", "subnets": [ "172.16.101.0\/255.255.255.0" ], "ip_
range": "0.0.0.0-0.0.0.0" }, { "id": 1, "name": "PM", "subnets": [
"172.16.102.0\/255.255.255.0" ], "ip_range": "0.0.0.0-0.0.0.0" } ], "Name": "Branch3",
"topology_role": "spoke" }
Member: { "SN": "FGT51E3U16001314", "IPv4": "172.16.200.199", "port": "500", "slot":
1002, "overlay": [ { "id": 0, "name": "QA", "subnets": [ "192.168.4.0\/255.255.255.0" ],
"ip_range": "0.0.0.0-0.0.0.0" }, { "id": 1, "name": "PM", "subnets": [
"192.168.5.0\/255.255.255.0" ], "ip_range": "0.0.0.0-0.0.0.0" } ], "Name": "Branch2",
"topology_role": "spoke" }
```

- Branch_1 # `diagnose vpn tunnel list`

```
list all ipsec tunnel in vd 0
------------------------------------------------------
name=_OCVPN2-3.1 ver=2 serial=4 172.16.200.1:0->172.16.200.199:0 tun_id=172.16.200.199
dst_mtu=1500
bound_if=11 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/528 options[0210]=create_dev
```

```
frag-rfc  accept_traffic=1

proxyid_num=2 child_num=0 refcnt=13 ilast=7 olast=0 ad=/0
stat: rxp=0 txp=7 rxb=0 txb=588
dpd: mode=on-demand on=1 idle=20000ms retry=3 count=0 seqno=6
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=_OCVPN2-3.1 proto=0 sa=1 ref=2 serial=8 auto-negotiate
  src: 0:10.1.100.0-10.1.100.255:0
  dst: 0:192.168.4.0-192.168.4.255:0
  SA:  ref=3 options=18627 type=00 soft=0 mtu=1438 expire=42923/0B replaywin=2048
       seqno=8 esn=0 replaywin_lastseq=00000000 itn=0 qat=0
  life: type=01 bytes=0/0 timeout=42931/43200
  dec: spi=c34bb752 esp=aes key=16 3c5ceeff3cac1eaa2702b5ccb713ab9b
       ah=sha1 key=20 5903e358b3d8938ee64f0412887a0fe741ccb105
  enc: spi=b5bd4fe1 esp=aes key=16 8ae97a8abe24dae725d614d2a6efdcb0
       ah=sha1 key=20 9ec200d9c0cef9e1b7cf76e05dbf344c70f53214
  dec:pkts/bytes=0/0, enc:pkts/bytes=7/1064
proxyid=_OCVPN2-3.1 proto=0 sa=0 ref=2 serial=1 auto-negotiate
  src: 0:10.1.100.0/255.255.255.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
------------------------------------------------------
name=_OCVPN2-4.1 ver=2 serial=6 172.16.200.1:0->172.16.200.4:0 tun_id=172.16.200.4 dst_
mtu=1500
bound_if=11 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/528 options[0210]=create_dev
frag-rfc  accept_traffic=1

proxyid_num=2 child_num=0 refcnt=11 ilast=19 olast=19 ad=/0
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-demand on=1 idle=20000ms retry=3 count=0 seqno=0
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=_OCVPN2-4.1 proto=0 sa=1 ref=2 serial=7 auto-negotiate
  src: 0:10.1.100.0-10.1.100.255:0
  dst: 0:172.16.101.0-172.16.101.255:0
  SA:  ref=3 options=18627 type=00 soft=0 mtu=1438 expire=42911/0B replaywin=2048
       seqno=1 esn=0 replaywin_lastseq=00000000 itn=0 qat=0
  life: type=01 bytes=0/0 timeout=42931/43200
  dec: spi=c34bb750 esp=aes key=16 8c9844a8bcd3fda6c7bd8a4f2ec81ef1
       ah=sha1 key=20 680c7144346f5b52126cbad9f325821b048c7192
  enc: spi=f2d1f2d4 esp=aes key=16 f9625fc8590152829eb39eecab3a3999
       ah=sha1 key=20 5df8447416da541fa54dde9fa3e5c35fbfc4723f
  dec:pkts/bytes=0/0, enc:pkts/bytes=0/0
proxyid=_OCVPN2-4.1 proto=0 sa=0 ref=2 serial=1 auto-negotiate
  src: 0:10.1.100.0/255.255.255.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
------------------------------------------------------
name=_OCVPN2-3.2 ver=2 serial=3 172.16.200.1:0->172.16.200.199:0 tun_id=172.16.200.199
dst_mtu=1500
bound_if=11 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/528 options[0210]=create_dev
frag-rfc  accept_traffic=1

proxyid_num=2 child_num=0 refcnt=11 ilast=6 olast=6 ad=/0
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-demand on=1 idle=20000ms retry=3 count=0 seqno=0
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=_OCVPN2-3.2 proto=0 sa=1 ref=2 serial=8 auto-negotiate
  src: 0:10.2.100.0-10.2.100.255:0
```

```
  dst: 0:192.168.5.0-192.168.5.255:0
  SA:  ref=3 options=18627 type=00 soft=0 mtu=1438 expire=42923/0B replaywin=2048
       seqno=1 esn=0 replaywin_lastseq=00000000 itn=0 qat=0
  life: type=01 bytes=0/0 timeout=42930/43200
  dec: spi=c34bb753 esp=aes key=16 58ddfad9a3699f1c49f3a9f369145c28
       ah=sha1 key=20 e749c7e6a7aaff119707c792eb73cd975127873b
  enc: spi=b5bd4fe2 esp=aes key=16 8f2366e653f5f9ad6587be1ce1905764
       ah=sha1 key=20 5347bf24e51219d483c0f7b058eceab202026204
  dec:pkts/bytes=0/0, enc:pkts/bytes=0/0
proxyid=_OCVPN2-3.2 proto=0 sa=0 ref=2 serial=1 auto-negotiate
  src: 0:10.2.100.0/255.255.255.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
------------------------------------------------------
name=_OCVPN2-4.2 ver=2 serial=5 172.16.200.1:0->172.16.200.4:0 tun_id=172.16.200.4 dst_
mtu=1500
bound_if=11 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/528 options[0210]=create_dev
frag-rfc  accept_traffic=1

proxyid_num=2 child_num=0 refcnt=11 ilast=17 olast=17 ad=/0
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-demand on=1 idle=20000ms retry=3 count=0 seqno=0
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=_OCVPN2-4.2 proto=0 sa=1 ref=2 serial=7 auto-negotiate
  src: 0:10.2.100.0-10.2.100.255:0
  dst: 0:172.16.102.0-172.16.102.255:0
  SA:  ref=3 options=18627 type=00 soft=0 mtu=1438 expire=42905/0B replaywin=2048
       seqno=1 esn=0 replaywin_lastseq=00000000 itn=0 qat=0
  life: type=01 bytes=0/0 timeout=42927/43200
  dec: spi=c34bb751 esp=aes key=16 41449ee5ea43d3e1f80df05fc632cd44
       ah=sha1 key=20 3ca2aea1c8764f35ccf987cdeca7cf6eb54331fb
  enc: spi=f2d1f2d5 esp=aes key=16 9010dd57e502c6296b27a4649a45a6ba
       ah=sha1 key=20 caf86a176ce04464221543f15fc3c63fc573b8ee
  dec:pkts/bytes=0/0, enc:pkts/bytes=0/0
proxyid=_OCVPN2-4.2 proto=0 sa=0 ref=2 serial=1 auto-negotiate
  src: 0:10.2.100.0/255.255.255.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
```

- Branch_1 # `get router info routing-table all`

```
Routing table for VRF=0
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default

S*      0.0.0.0/0 [10/0] via 172.16.200.254, port1
C       10.1.100.0/24 is directly connected, dmz
C       10.2.100.0/24 is directly connected, loop
C       11.101.1.0/24 is directly connected, wan1
C       11.102.1.0/24 is directly connected, wan2
S       192.168.5.0/24 [20/0] is directly connected, _OCVPN2-3.2
C       172.16.200.0/24 is directly connected, port1
S       172.16.101.0/24 [20/0] is directly connected, _OCVPN2-4.1
S       172.16.102.0/24 [20/0] is directly connected, _OCVPN2-4.2
S       192.168.4.0/24 [20/0] is directly connected, _OCVPN2-3.1
```

Troubleshooting hub-spoke with ADVPN shortcut

- Primary-Hub # `diagnose vpn ocvpn status`

```
Current State        : Registered
Topology             : Dual-Hub-Spoke
Role                 : Primary-Hub
Server Status        : Up
Registration time    : Sat Mar  2 11:31:54 2019
Poll time            : Sat Mar  2 11:46:02 2019
```

- Spoke1 # `diagnose vpn ocvpn status`

```
Current State        : Registered
Topology             : Dual-Hub-Spoke
Role                 : Spoke
Server Status        : Up
Registration time    : Sat Mar  2 11:41:22 2019
Poll time            : Sat Mar  2 11:46:44 2019
```

- Primary-Hub # `diagnose vpn ocvpn show-members`

```
Member: { "sn": "FG900D3915800083", "ip_v4": "172.16.200.4", "port": 500, "slot": 0,
"overlay": [ { "id": 0, "name": "QA", "subnets": [ "172.16.101.0\/255.255.255.0" ], "ip_
range": "0.0.0.0-0.0.0.0" }, { "id": 1, "name": "PM", "subnets": [
"172.16.102.0\/255.255.255.0" ], "ip_range": "0.0.0.0-0.0.0.0" } ], "name": "Primary-
Hub", "topology_role": "primary_hub", "eap": "disable", "auto_discovery": "enable" }
Member: { "sn": "FG100D3G15828488", "ip_v4": "172.16.200.2", "port": 500, "slot": 1,
"overlay": [ { "id": 0, "name": "QA", "subnets": [ "172.16.101.0\/255.255.255.0" ], "ip_
range": "0.0.0.0-0.0.0.0" }, { "id": 1, "name": "PM", "subnets": [
"172.16.102.0\/255.255.255.0" ], "ip_range": "0.0.0.0-0.0.0.0" } ], "name": "Secondary-
Hub", "topology_role": "secondary_hub", "eap": "disable", "auto_discovery": "enable" }
Member: { "sn": "FG100D3G15801621", "ip_v4": "172.16.200.1", "port": 500, "slot": 1000,
"overlay": [ { "id": 0, "name": "QA", "subnets": [ "10.1.100.0\/255.255.255.0" ], "ip_
range": "0.0.0.0-0.0.0.0" }, { "id": 1, "name": "PM", "subnets": [
"10.2.100.0\/255.255.255.0" ], "ip_range": "0.0.0.0-0.0.0.0" } ], "name": "Spoke1",
"topology_role": "spoke" }
Member: { "sn": "FGT51E3U16001314", "ip_v4": "172.16.200.3", "port": 500, "slot": 1001,
"overlay": [ { "id": 0, "name": "QA", "subnets": [ "192.168.4.0\/255.255.255.0" ], "ip_
range": "0.0.0.0-0.0.0.0" }, { "id": 1, "name": "PM", "subnets": [
"192.168.5.0\/255.255.255.0" ], "ip_range": "0.0.0.0-0.0.0.0" } ], "name": "Spoke2",
"topology_role": "spoke" }
```

- Primary-Hub # `diagnose vpn ocvpn show-meta`

```
Topology :: auto
License  :: full
Members  :: 4
Max-free :: 3
```

- Primary-Hub # `diagnose vpn ocvpn show-overlays`

```
QA
PM
```

- Spoke1 # `diagnose vpn tunnel list`

```
list all ipsec tunnel in vd 0
------------------------------------------------------
name=_OCVPN2-0.0 ver=2 serial=6 172.16.200.1:0->172.16.200.4:0 tun_id=172.16.200.4 dst_
```

```
mtu=1500
bound_if=11 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/528 options[0210]=create_dev
frag-rfc  accept_traffic=1

proxyid_num=1 child_num=0 refcnt=11 ilast=0 olast=0 ad=r/2
stat: rxp=1 txp=34 rxb=152 txb=2856
dpd: mode=on-idle on=1 idle=20000ms retry=3 count=0 seqno=46
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=_OCVPN2-0.0 proto=0 sa=1 ref=2 serial=1 auto-negotiate adr
  src: 0:10.1.100.0/255.255.255.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=3 options=1a227 type=00 soft=0 mtu=1438 expire=42895/0B replaywin=2048
       seqno=1 esn=0 replaywin_lastseq=00000000 itn=0 qat=0
  life: type=01 bytes=0/0 timeout=42901/43200
  dec: spi=048477c7 esp=aes key=16 240e064c0f1c980ca31980b9e7605c9d
       ah=sha1 key=20 6ff022cbebcaff4c5de62eefb2e6180c40a3adb2
  enc: spi=dfcffa86 esp=aes key=16 862208de164a02af377756c2bcabd588
       ah=sha1 key=20 af6e54781fd42d7a2ba2119ec95d0f95629c8448
  dec:pkts/bytes=0/0, enc:pkts/bytes=0/0
------------------------------------------------------
name=_OCVPN2-1.0 ver=2 serial=8 172.16.200.1:0->172.16.200.2:0 tun_id=172.16.200.2 dst_
mtu=1500
bound_if=11 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/528 options[0210]=create_dev
frag-rfc  accept_traffic=0

proxyid_num=1 child_num=0 refcnt=10 ilast=934 olast=934 ad=/0
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-idle on=0 idle=20000ms retry=3 count=0 seqno=1
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=_OCVPN2-1.0 proto=0 sa=0 ref=2 serial=1 auto-negotiate adr
  src: 0:10.1.100.0/255.255.255.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
------------------------------------------------------
name=_OCVPN2-0.1 ver=2 serial=5 172.16.200.1:0->172.16.200.4:0 tun_id=172.16.200.4 dst_
mtu=1500
bound_if=11 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/528 options[0210]=create_dev
frag-rfc  accept_traffic=1

proxyid_num=1 child_num=0 refcnt=11 ilast=12 olast=12 ad=r/2
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-idle on=1 idle=20000ms retry=3 count=0 seqno=46
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=_OCVPN2-0.1 proto=0 sa=1 ref=2 serial=1 auto-negotiate adr
  src: 0:10.2.100.0/255.255.255.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=3 options=1a227 type=00 soft=0 mtu=1438 expire=42895/0B replaywin=2048
       seqno=1 esn=0 replaywin_lastseq=00000000 itn=0 qat=0
  life: type=01 bytes=0/0 timeout=42901/43200
  dec: spi=048477c8 esp=aes key=16 701ec608767f4988b76c2f662464e654
       ah=sha1 key=20 93c65d106dc610d7ee3f04487f08601a9e00ffdd
  enc: spi=dfcffa87 esp=aes key=16 02b2d04dce3d81ebab69e128d45cb7ca
       ah=sha1 key=20 4a9283847f852c83a75691fad44d07d8409a2267
  dec:pkts/bytes=0/0, enc:pkts/bytes=0/0
------------------------------------------------------
name=_OCVPN2-1.1 ver=2 serial=7 172.16.200.1:0->172.16.200.2:0 tun_id=172.16.200.2 dst_
mtu=1500
```

```
bound_if=11 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/528 options[0210]=create_dev
frag-rfc  accept_traffic=0

proxyid_num=1 child_num=0 refcnt=10 ilast=934 olast=934 ad=/0
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-idle on=0 idle=20000ms retry=3 count=0 seqno=1
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=_OCVPN2-1.1 proto=0 sa=0 ref=2 serial=1 auto-negotiate adr
  src: 0:10.2.100.0/255.255.255.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
```

- Spoke1 # `get router info routing-table all`

```
Routing table for VRF=0
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default

S*      0.0.0.0/0 [10/0] via 172.16.200.254, port1
C       10.1.100.0/24 is directly connected, dmz
C       10.2.100.0/24 is directly connected, loop
C       11.101.1.0/24 is directly connected, wan1
C       11.102.1.0/24 is directly connected, wan2
S       172.16.102.0/24 [20/0] is directly connected, _OCVPN2-0.1
C       172.16.200.0/24 is directly connected, port1
S       172.16.101.0/24 [20/0] is directly connected, _OCVPN2-0.0
S       192.168.4.0/24 [20/0] is directly connected, _OCVPN2-0.0
S       192.168.5.0/24 [20/0] is directly connected, _OCVPN2-0.1
```

- Generate traffic from spoke1 to spoke2 to trigger the ADVPN shortcut and check the VPN tunnel and routing-table again on spoke1.

```
branch1 # diagnose vpn tunnel list
list all ipsec tunnel in vd 0
------------------------------------------------------
name=_OCVPN2-0.0_0 ver=2 serial=a 172.16.200.1:0->172.16.200.3:0 tun_id=172.16.200.3
dst_mtu=1500
bound_if=11 lgwy=static/1 tun=intf/0 mode=dial_inst/3 encap=none/720 options
[02d0]=create_dev no-sysctl rgwy-chg frag-rfc  accept_traffic=1

 parent=_OCVPN2-0.0 index=0
proxyid_num=1 child_num=0 refcnt=14 ilast=0 olast=0 ad=r/2
stat: rxp=7 txp=7 rxb=1064 txb=588
dpd: mode=on-idle on=1 idle=20000ms retry=3 count=0 seqno=0
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=_OCVPN2-0.0 proto=0 sa=1 ref=2 serial=1 auto-negotiate add-route adr
  src: 0:10.1.100.0-10.1.100.255:0
  dst: 0:192.168.4.0-192.168.4.255:0
  SA:  ref=3 options=1a227 type=00 soft=0 mtu=1438 expire=43180/0B replaywin=2048
       seqno=8 esn=0 replaywin_lastseq=00000008 itn=0 qat=0
  life: type=01 bytes=0/0 timeout=43187/43200
  dec: spi=048477c9 esp=aes key=16 27c35d53793013ef24cf887561e9f313
       ah=sha1 key=20 2c8cfd328c3b29104db0ca74a00c6063f46cafe4
  enc: spi=fb9e13fd esp=aes key=16 9d0d3bf6c84b7ddaf9d9196fe74002ed
```

```
       ah=sha1 key=20 d1f541db787dea384c6a4df16fc228abeb7ae334
  dec:pkts/bytes=7/588, enc:pkts/bytes=7/1064
------------------------------------------------------
name=_OCVPN2-0.0 ver=2 serial=6 172.16.200.1:0->172.16.200.4:0 tun_id=172.16.200.4 dst_
mtu=1500
bound_if=11 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/528 options[0210]=create_dev
frag-rfc  accept_traffic=1

proxyid_num=1 child_num=1 refcnt=12 ilast=7 olast=7 ad=r/2
stat: rxp=2 txp=35 rxb=304 txb=2940
dpd: mode=on-idle on=1 idle=20000ms retry=3 count=0 seqno=65
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=_OCVPN2-0.0 proto=0 sa=1 ref=2 serial=1 auto-negotiate adr
  src: 0:10.1.100.0/255.255.255.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=3 options=1a227 type=00 soft=0 mtu=1438 expire=42500/0B replaywin=2048
       seqno=2 esn=0 replaywin_lastseq=00000002 itn=0 qat=0
  life: type=01 bytes=0/0 timeout=42901/43200
  dec: spi=048477c7 esp=aes key=16 240e064c0f1c980ca31980b9e7605c9d
       ah=sha1 key=20 6ff022cbebcaff4c5de62eefb2e6180c40a3adb2
  enc: spi=dfcffa86 esp=aes key=16 862208de164a02af377756c2bcabd588
       ah=sha1 key=20 af6e54781fd42d7a2ba2119ec95d0f95629c8448
  dec:pkts/bytes=1/84, enc:pkts/bytes=1/152
------------------------------------------------------
name=_OCVPN2-1.0 ver=2 serial=8 172.16.200.1:0->172.16.200.2:0 tun_id=172.16.200.2 dst_
mtu=1500
bound_if=11 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/528 options[0210]=create_dev
frag-rfc  accept_traffic=0

proxyid_num=1 child_num=0 refcnt=10 ilast=1328 olast=1328 ad=/0
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-idle on=0 idle=20000ms retry=3 count=0 seqno=1
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=_OCVPN2-1.0 proto=0 sa=0 ref=2 serial=1 auto-negotiate adr
  src: 0:10.1.100.0/255.255.255.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
------------------------------------------------------
name=_OCVPN2-0.1 ver=2 serial=5 172.16.200.1:0->172.16.200.4:0 tun_id=172.16.200.4 dst_
mtu=1500
bound_if=11 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/528 options[0210]=create_dev
frag-rfc  accept_traffic=1

proxyid_num=1 child_num=0 refcnt=11 ilast=5 olast=5 ad=r/2
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-idle on=1 idle=20000ms retry=3 count=0 seqno=66
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=_OCVPN2-0.1 proto=0 sa=1 ref=2 serial=1 auto-negotiate adr
  src: 0:10.2.100.0/255.255.255.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=3 options=1a227 type=00 soft=0 mtu=1438 expire=42500/0B replaywin=2048
       seqno=1 esn=0 replaywin_lastseq=00000000 itn=0 qat=0
  life: type=01 bytes=0/0 timeout=42901/43200
  dec: spi=048477c8 esp=aes key=16 701ec608767f4988b76c2f662464e654
       ah=sha1 key=20 93c65d106dc610d7ee3f04487f08601a9e00ffdd
  enc: spi=dfcffa87 esp=aes key=16 02b2d04dce3d81ebab69e128d45cb7ca
       ah=sha1 key=20 4a9283847f852c83a75691fad44d07d8409a2267
```

```
  dec:pkts/bytes=0/0, enc:pkts/bytes=0/0
------------------------------------------------------
name=_OCVPN2-1.1 ver=2 serial=7 172.16.200.1:0->172.16.200.2:0 tun_id=172.16.200.2 dst_
mtu=1500
bound_if=11 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/528 options[0210]=create_dev
frag-rfc  accept_traffic=0

proxyid_num=1 child_num=0 refcnt=10 ilast=1328 olast=1328 ad=/0
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-idle on=0 idle=20000ms retry=3 count=0 seqno=1
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=_OCVPN2-1.1 proto=0 sa=0 ref=2 serial=1 auto-negotiate adr
  src: 0:10.2.100.0/255.255.255.0:0
  dst: 0:0.0.0.0/0.0.0.0:0


Routing table for VRF=0
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default

S*      0.0.0.0/0 [10/0] via 172.16.200.254, port1
C       10.1.100.0/24 is directly connected, dmz
C       10.2.100.0/24 is directly connected, loop
C       11.101.1.0/24 is directly connected, wan1
C       11.102.1.0/24 is directly connected, wan2
S       172.16.102.0/24 [20/0] is directly connected, _OCVPN2-0.1
C       172.16.200.0/24 is directly connected, port1
S       172.16.101.0/24 [20/0] is directly connected, _OCVPN2-0.0
S       192.168.4.0/24 [15/0] via 172.16.200.3, _OCVPN2-0.0_0
S       192.168.5.0/24 [20/0] is directly connected, _OCVPN2-0.1
```

- Simulate the primary hub being unavailable where all spokes' dialup VPN tunnels will switch to the secondary hub, to check VPN tunnel status and routing-table.

```
list all ipsec tunnel in vd 0
------------------------------------------------------
name=_OCVPN2-0.0 ver=2 serial=6 172.16.200.1:0->172.16.200.4:0 tun_id=172.16.200.4 dst_
mtu=1500
bound_if=11 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/528 options[0210]=create_dev
frag-rfc  accept_traffic=0

proxyid_num=1 child_num=0 refcnt=10 ilast=25 olast=25 ad=/0
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-idle on=0 idle=20000ms retry=3 count=0 seqno=82
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=_OCVPN2-0.0 proto=0 sa=0 ref=2 serial=1 auto-negotiate adr
  src: 0:10.1.100.0/255.255.255.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
------------------------------------------------------
name=_OCVPN2-1.0 ver=2 serial=8 172.16.200.1:0->172.16.200.2:0 tun_id=172.16.200.2 dst_
mtu=1500
bound_if=11 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/528 options[0210]=create_dev
frag-rfc  accept_traffic=1
```

```
proxyid_num=1 child_num=0 refcnt=11 ilast=14 olast=14 ad=r/2
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-idle on=1 idle=20000ms retry=3 count=0 seqno=9
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=_OCVPN2-1.0 proto=0 sa=1 ref=2 serial=1 auto-negotiate adr
  src: 0:10.1.100.0/255.255.255.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=3 options=1a227 type=00 soft=0 mtu=1438 expire=42723/0B replaywin=2048
       seqno=1 esn=0 replaywin_lastseq=00000000 itn=0 qat=0
  life: type=01 bytes=0/0 timeout=42898/43200
  dec: spi=048477cd esp=aes key=16 9bb363a32378b5897cd42890c92df811
       ah=sha1 key=20 2ed40583b9544e37867349b4adc7c013024d7e17
  enc: spi=f345fb42 esp=aes key=16 3ea31dff3310b245700a131db4565851
       ah=sha1 key=20 522862dfb232514b845e436133b148da0e67b7c4
  dec:pkts/bytes=0/0, enc:pkts/bytes=0/0
------------------------------------------------------
name=_OCVPN2-0.1 ver=2 serial=5 172.16.200.1:0->172.16.200.4:0 tun_id=172.16.200.4 dst_
mtu=1500
bound_if=11 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/528 options[0210]=create_dev
frag-rfc  accept_traffic=0

proxyid_num=1 child_num=0 refcnt=10 ilast=19 olast=19 ad=/0
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-idle on=0 idle=20000ms retry=3 count=0 seqno=83
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=_OCVPN2-0.1 proto=0 sa=0 ref=2 serial=1 auto-negotiate adr
  src: 0:10.2.100.0/255.255.255.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
------------------------------------------------------
name=_OCVPN2-1.1 ver=2 serial=7 172.16.200.1:0->172.16.200.2:0 tun_id=172.16.200.2 dst_
mtu=1500
bound_if=11 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/528 options[0210]=create_dev
frag-rfc  accept_traffic=1

proxyid_num=1 child_num=0 refcnt=11 ilast=12 olast=12 ad=r/2
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-idle on=1 idle=20000ms retry=3 count=0 seqno=9
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=_OCVPN2-1.1 proto=0 sa=1 ref=2 serial=1 auto-negotiate adr
  src: 0:10.2.100.0/255.255.255.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=3 options=1a227 type=00 soft=0 mtu=1438 expire=42728/0B replaywin=2048
       seqno=1 esn=0 replaywin_lastseq=00000000 itn=0 qat=0
  life: type=01 bytes=0/0 timeout=42902/43200
  dec: spi=048477cf esp=aes key=16 b6f0ca7564abcd8559b5b0ebb3fd04c1
       ah=sha1 key=20 4130d040554b39daca72adac7583b9cc83cce3c8
  enc: spi=f345fb43 esp=aes key=16 727582f20fcedff884ba693ed2164bcd
       ah=sha1 key=20 b0a625803fde701ed9d28d256079e908954b7fc8
  dec:pkts/bytes=0/0, enc:pkts/bytes=0/0


Routing table for VRF=0
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
```

```
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default

S*      0.0.0.0/0 [10/0] via 172.16.200.254, port1
C       10.1.100.0/24 is directly connected, dmz
C       10.2.100.0/24 is directly connected, loop
C       11.101.1.0/24 is directly connected, wan1
C       11.102.1.0/24 is directly connected, wan2
S       172.16.102.0/24 [21/0] is directly connected, _OCVPN2-1.1
C       172.16.200.0/24 is directly connected, port1
S       172.16.101.0/24 [21/0] is directly connected, _OCVPN2-1.0
S       192.168.4.0/24 [21/0] is directly connected, _OCVPN2-1.0
S       192.168.5.0/24 [21/0] is directly connected, _OCVPN2-1.1
```

Troubleshooting hub-spoke with inter-overlay source NAT

- Primary-Hub # `diagnose vpn ocvpn status`

```
Current State        : Registered
Topology             : Dual-Hub-Spoke
Role                 : Primary-Hub
Server Status        : Up
Registration time    : Sat Mar  2 11:31:54 2019
Update time          : Sat Mar  2 13:57:05 2019
Poll time            : Sat Mar  2 14:03:31 2019
```

- Spoke1 # `diagnose vpn ocvpn status`

```
Current State        : Registered
Topology             : Dual-Hub-Spoke
Role                 : Spoke
Server Status        : Up
Registration time    : Sat Mar  2 13:58:01 2019
Poll time            : Sat Mar  2 14:04:22 2019
```

- Primary-Hub # `diagnose vpn ocvpn show-members`


```
Member: { "sn": "FG900D3915800083", "ip_v4": "172.16.200.4", "port": 500, "slot": 0,
"overlay": [ { "id": 0, "name": "QA", "subnets": [ "172.16.101.0\/255.255.255.0" ], "ip_
range": "172.16.101.100-172.16.101.200" }, { "id": 1, "name": "PM", "subnets": [
"172.16.102.0\/255.255.255.0" ], "ip_range": "172.16.102.100-172.16.102.200" } ],
"name": "Primary-Hub", "topology_role": "primary_hub", "eap": "disable", "auto_
discovery": "enable" }
Member: { "sn": "FG100D3G15828488", "ip_v4": "172.16.200.2", "port": 500, "slot": 1,
"overlay": [ { "id": 0, "name": "QA", "subnets": [ "172.16.101.0\/255.255.255.0" ], "ip_
range": "0.0.0.0-0.0.0.0" }, { "id": 1, "name": "PM", "subnets": [
"172.16.102.0\/255.255.255.0" ], "ip_range": "0.0.0.0-0.0.0.0" } ], "name": "Secondary-
Hub", "topology_role": "secondary_hub", "eap": "disable", "auto_discovery": "enable" }
Member: { "sn": "FGT51E3U16001314", "ip_v4": "172.16.200.3", "port": 500, "slot": 1001,
"overlay": [ { "id": 0, "name": "QA", "subnets": [ "192.168.4.0\/255.255.255.0" ], "ip_
range": "0.0.0.0-0.0.0.0" }, { "id": 1, "name": "PM", "subnets": [
"192.168.5.0\/255.255.255.0" ], "ip_range": "0.0.0.0-0.0.0.0" } ], "name": "Spoke2",
"topology_role": "spoke" }
Member: { "sn": "FG100D3G15801621", "ip_v4": "172.16.200.1", "port": 500, "slot": 1000,
"overlay": [ { "id": 0, "name": "QA", "subnets": [ "10.1.100.0\/255.255.255.0" ], "ip_
range": "0.0.0.0-0.0.0.0" }, { "id": 1, "name": "PM", "subnets": [
```

```
"10.2.100.0\/255.255.255.0" ], "ip_range": "0.0.0.0-0.0.0.0" } ], "name": "Spoke1",
"topology_role": "spoke" }
```

- Primary-Hub # `diagnose vpn ocvpn show-meta`

```
Topology :: auto
License  :: full
Members  :: 4
Max-free :: 3
```

- Primary-Hub # `diagnose vpn ocvpn show-overlays`

```
QA
PM
```

- Spoke1 # `diagnose vpn tunnel list`

```
list all ipsec tunnel in vd 0
------------------------------------------------------
name=_OCVPN2-0.0 ver=2 serial=c 172.16.200.1:0->172.16.200.4:0 tun_id=172.16.200.4 dst_
mtu=1500
bound_if=11 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/528 options[0210]=create_dev
frag-rfc  accept_traffic=1

proxyid_num=3 child_num=0 refcnt=13 ilast=17 olast=17 ad=/0
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-idle on=1 idle=20000ms retry=3 count=0 seqno=29
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=_OCVPN2-0.0 proto=0 sa=1 ref=2 serial=1 auto-negotiate
  src: 0:10.1.100.0/255.255.255.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=3 options=18227 type=00 soft=0 mtu=1438 expire=42299/0B replaywin=2048
       seqno=1 esn=0 replaywin_lastseq=00000000 itn=0 qat=0
  life: type=01 bytes=0/0 timeout=42899/43200
  dec: spi=0484795d esp=aes key=16 10eeb76fadd49f00c333350d83509095
       ah=sha1 key=20 971bde5dcfca7e52fd1573cb3489e9c855f6154e
  enc: spi=dfcffaaa esp=aes key=16 d07a4dd683ee093af2dca9485aa436eb
       ah=sha1 key=20 65369be35d5ecad8cae63557318419cd6005c230
  dec:pkts/bytes=0/0, enc:pkts/bytes=0/0
proxyid=_OCVPN2-0.0_nat proto=0 sa=1 ref=2 serial=3 auto-negotiate
  src: 0:172.16.101.101-172.16.101.101:0
  dst: 0:0.0.0.0-255.255.255.255:0
  SA:  ref=3 options=18627 type=00 soft=0 mtu=1438 expire=42303/0B replaywin=2048
       seqno=1 esn=0 replaywin_lastseq=00000000 itn=0 qat=0
  life: type=01 bytes=0/0 timeout=42898/43200
  dec: spi=04847961 esp=aes key=16 ea181036b02e8bc8711fb520b3e98a60
       ah=sha1 key=20 b3c449d96d5d3f090975087a62447f6918ce7930
  enc: spi=dfcffaac esp=aes key=16 f7ea5e42e9443698e6b8b32161ace40e
       ah=sha1 key=20 a7e36dd1ec0bdb6eff0aa66e442707427400c700
  dec:pkts/bytes=0/0, enc:pkts/bytes=0/0
proxyid=_OCVPN2-0.0_nat proto=0 sa=0 ref=2 serial=2 auto-negotiate
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
------------------------------------------------------
name=_OCVPN2-1.0 ver=2 serial=e 172.16.200.1:0->172.16.200.2:0 tun_id=172.16.200.2 dst_
mtu=0
bound_if=11 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/528 options[0210]=create_dev
frag-rfc  accept_traffic=1
```

```
proxyid_num=2 child_num=0 refcnt=10 ilast=599 olast=599 ad=/0
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-idle on=0 idle=20000ms retry=3 count=0 seqno=0
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=_OCVPN2-1.0 proto=0 sa=0 ref=2 serial=1 auto-negotiate
  src: 0:10.1.100.0/255.255.255.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
proxyid=_OCVPN2-1.0_nat proto=0 sa=0 ref=2 serial=2 auto-negotiate
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
------------------------------------------------------
name=_OCVPN2-0.1 ver=2 serial=b 172.16.200.1:0->172.16.200.4:0 tun_id=172.16.200.4 dst_
mtu=1500
bound_if=11 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/528 options[0210]=create_dev
frag-rfc  accept_traffic=1

proxyid_num=3 child_num=0 refcnt=13 ilast=17 olast=17 ad=/0
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-idle on=1 idle=20000ms retry=3 count=0 seqno=29
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=_OCVPN2-0.1 proto=0 sa=1 ref=2 serial=1 auto-negotiate
  src: 0:10.2.100.0/255.255.255.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=3 options=18227 type=00 soft=0 mtu=1438 expire=42297/0B replaywin=2048
       seqno=1 esn=0 replaywin_lastseq=00000000 itn=0 qat=0
  life: type=01 bytes=0/0 timeout=42897/43200
  dec: spi=0484795e esp=aes key=16 106eaa95a2be64b566e7d1ca0aa88f6a
       ah=sha1 key=20 5dddfba7070b03d5a31931d41db06ff96e7bc542
  enc: spi=dfcffaab esp=aes key=16 29c774dbd7e54464ee298c381e71a94e
       ah=sha1 key=20 c3da7372789c0a53b3752e69baaba1a42d798820
  dec:pkts/bytes=0/0, enc:pkts/bytes=0/0
proxyid=_OCVPN2-0.1_nat proto=0 sa=1 ref=2 serial=3 auto-negotiate
  src: 0:172.16.102.101-172.16.102.101:0
  dst: 0:0.0.0.0-255.255.255.255:0
  SA:  ref=3 options=18627 type=00 soft=0 mtu=1438 expire=42307/0B replaywin=2048
       seqno=1 esn=0 replaywin_lastseq=00000000 itn=0 qat=0
  life: type=01 bytes=0/0 timeout=42902/43200
  dec: spi=04847962 esp=aes key=16 b7daa5807cfa86906592a012a9d2478f
       ah=sha1 key=20 39c8bb4c9e3f1e9e451f22c58a172ff01155055d
  enc: spi=dfcffaad esp=aes key=16 2ecc644def4cebe6b0c4b7729da43d8e
       ah=sha1 key=20 469c6f319e83bd73468f55d430566afcd6215138
  dec:pkts/bytes=0/0, enc:pkts/bytes=0/0
proxyid=_OCVPN2-0.1_nat proto=0 sa=0 ref=2 serial=2 auto-negotiate
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
------------------------------------------------------
name=_OCVPN2-1.1 ver=2 serial=d 172.16.200.1:0->172.16.200.2:0 tun_id=172.16.200.2 dst_
mtu=0
bound_if=11 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/528 options[0210]=create_dev
frag-rfc  accept_traffic=1

proxyid_num=2 child_num=0 refcnt=10 ilast=599 olast=599 ad=/0
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-idle on=0 idle=20000ms retry=3 count=0 seqno=0
natt: mode=none draft=0 interval=0 remote_port=0
```

```
proxyid=_OCVPN2-1.1 proto=0 sa=0 ref=2 serial=1 auto-negotiate
  src: 0:10.2.100.0/255.255.255.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
proxyid=_OCVPN2-1.1_nat proto=0 sa=0 ref=2 serial=2 auto-negotiate
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
```

- Spoke1 # `get router info routing-table all`

```
Routing table for VRF=0
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default

S*      0.0.0.0/0 [10/0] via 172.16.200.254, port1
C       10.1.100.0/24 is directly connected, dmz
C       10.2.100.0/24 is directly connected, loop
C       11.101.1.0/24 is directly connected, wan1
C       11.102.1.0/24 is directly connected, wan2
S       172.16.101.0/24 [20/0] is directly connected, _OCVPN2-0.1
C       172.16.101.101/32 is directly connected, _OCVPN2-0.1
C       172.16.200.0/24 is directly connected, port1
S       172.16.102.0/24 [20/0] is directly connected, _OCVPN2-0.0
C       172.16.102.101/32 is directly connected, _OCVPN2-0.0
S       192.168.4.0/24 [20/0] is directly connected, _OCVPN2-0.0
S       192.168.5.0/24 [20/0] is directly connected, _OCVPN2-0.1
```

- Spoke1 # `show firewall policy`

```
 ..............................

    edit 9
        set name "_OCVPN2-1.1_nat"
        set uuid 3f7a84b8-3d36-51e9-ee97-8f418c91e666
        set srcintf "any"
        set dstintf "_OCVPN2-1.1"
        set srcaddr "all"
        set dstaddr "_OCVPN2-1.1_remote_networks"
        set action accept
        set schedule "always"
        set service "ALL"
        set comments "Generated by OCVPN Cloud Service."
        set nat enable
    next
    edit 12
        set name "_OCVPN2-1.0_nat"
        set uuid 3fafec98-3d36-51e9-80c0-5d99325bad83
        set srcintf "any"
        set dstintf "_OCVPN2-1.0"
        set srcaddr "all"
        set dstaddr "_OCVPN2-1.0_remote_networks"
        set action accept
        set schedule "always"
        set service "ALL"
```

```
        set comments "Generated by OCVPN Cloud Service."
        set nat enable
    next
.................................
```

ADVPN

Auto-Discovery VPN (ADVPN) allows the central hub to dynamically inform spokes about a better path for traffic between two spokes.

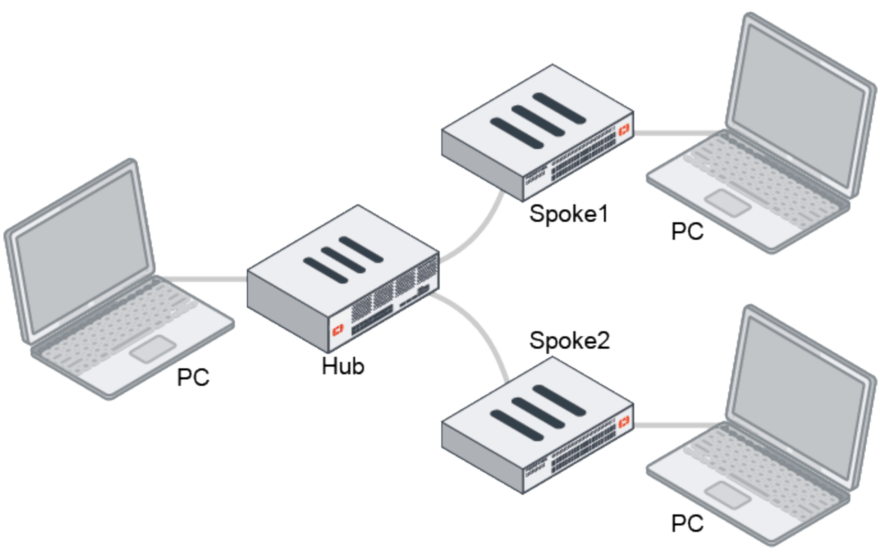
The following topics provide instructions on configuring ADVPN:

IPsec VPN wizard hub-and-spoke ADVPN support

When using the IPsec VPN wizard to create a hub and spoke VPN, multiple local interfaces can be selected. At the end of the wizard, changes can be reviewed, real-time updates can be made to the local address group and tunnel interface, and easy configuration keys can be copied for configuring the spokes.

When editing a VPN tunnel, the Hub & Spoke Topology section provides access to the easy configuration keys for the spokes, and allows you to add more spokes.

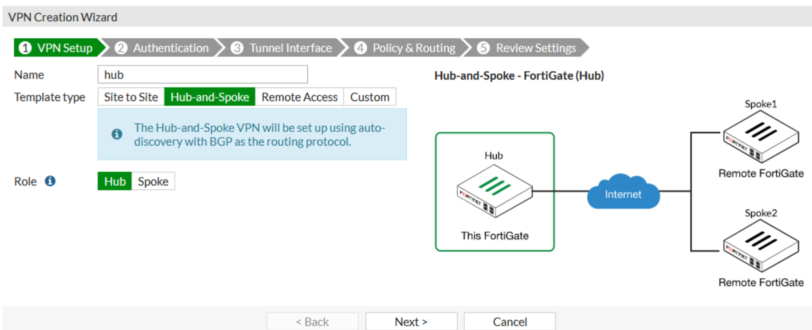
This example shows the configuration of a hub with two spokes.



To configure the hub:

1. Go to *VPN > IPsec Wizard*.
2. Go through the steps of the wizard:
   a. *VPN Setup*:

Name	hub
Template Type	Hub-and-Spoke
Role	Hub



   b. *Authentication*:

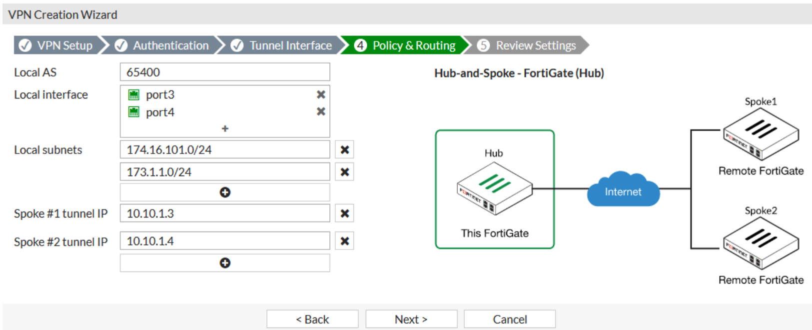
Incoming Interface	port1
Authentication method	Pre-shared Key
Pre-shared key	<key>

   c. *Tunnel Interface*:

Tunnel IP	10.10.1.1
Remote IP/netmask	10.10.1.2/24

   d. *Policy & Routing*:
   Multiple local interfaces and subnets can be configured.

Local AS	65400
Local interface	port3 port4
Local subnets	174.16.101.0/24 173.1.1.0/24
Spoke #1 tunnel IP	10.10.1.3
Spoke #2 tunnel IP	10.10.1.4

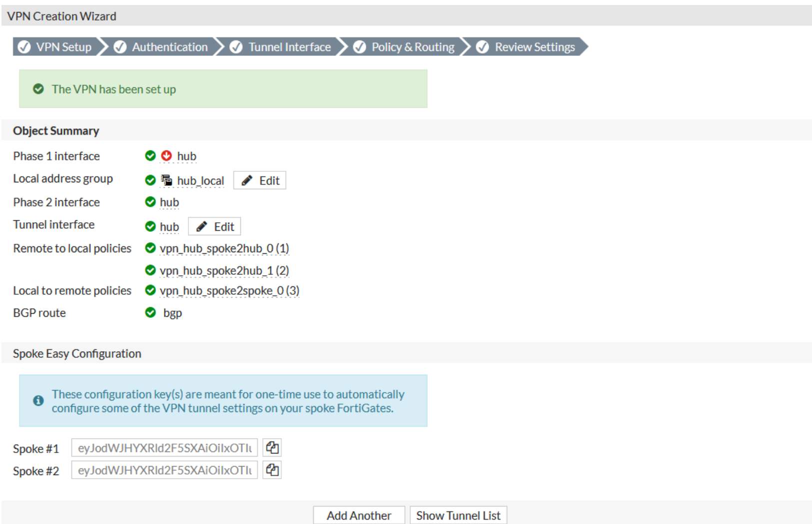


e. *Review Settings*:

Confirm that the settings look correct, then click *Create*.

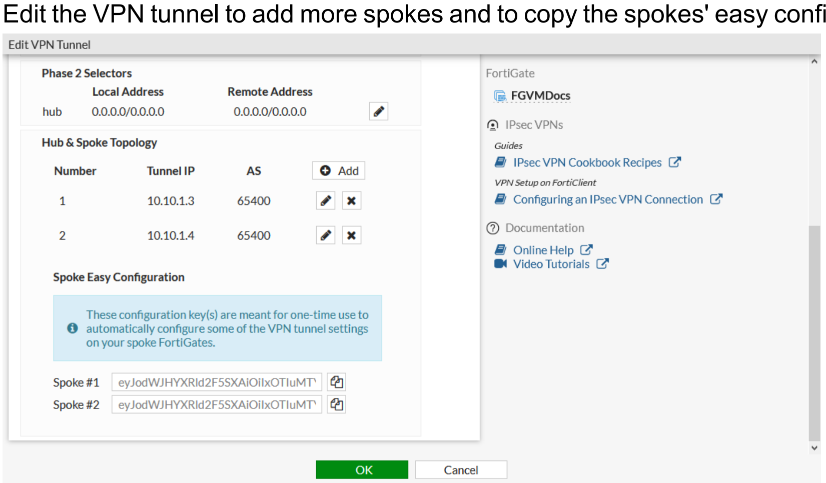
3. The summary shows details about the set up hub:

- The *Local address group* and *Tunnel interface* can be edited directly on this page.
- Spoke easy configuration keys can be used to quickly configure the spokes.



4. Click *Show Tunnel List* to go to *VPN > IPsec Tunnels*.

5. Edit the VPN tunnel to add more spokes and to copy the spokes' easy configuration keys.

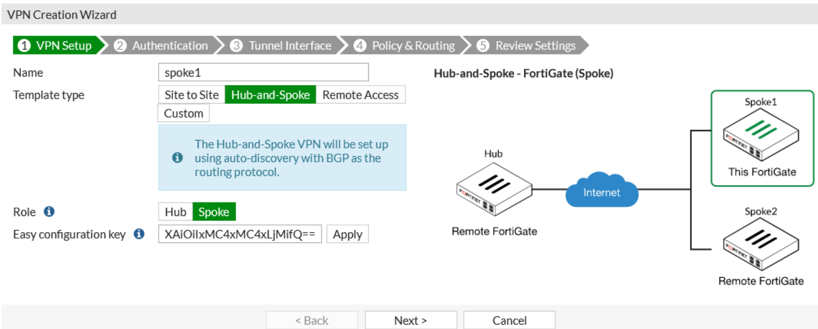


To configure the spokes:

1. Go to *VPN > IPsec Wizard*.
2. On the *VPN Setup* page of the wizard, enter the following:

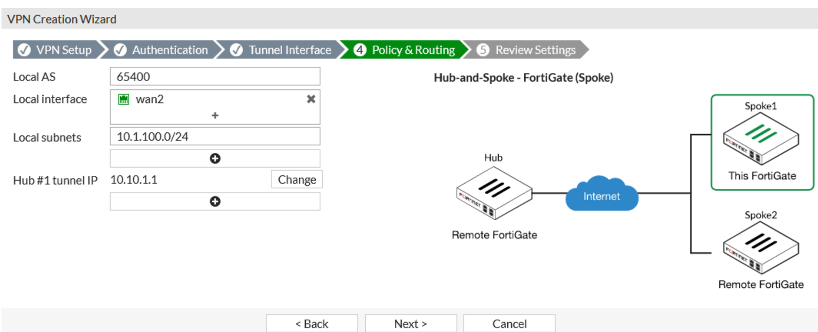
Name	spoke1
Template Type	Hub-and-Spoke
Role	Spoke

3. In the *Easy configuration key* field, paste the *Spoke #1* key from the hub FortiGate, click *Apply*, then click *Next*.



4. Adjust the *Authentication* settings as required, enter the *Pre-shared key*, then click *Next*.
5. Adjust the *Tunnel Interface* settings as required, then click *Next*.
6. Configure the *Policy & Routing* settings, then click *Next*:

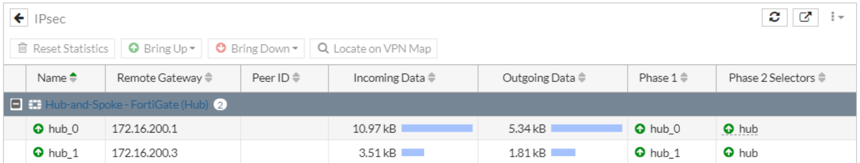
Local interface	wan2
Local subnets	10.1.100.0/24



7. Review the settings, then click *Create*.
8. The summary shows details about the set up spoke. The *Local address group* and *Tunnel interface* can be edited directly on this page.
9. Follow the same steps to configure the second spoke.

To check that the tunnels are created and working:

1. On the hub FortiGate, go to *Dashboard > Network* and expand the IPsec widget.
   The tunnels to the spokes are established.

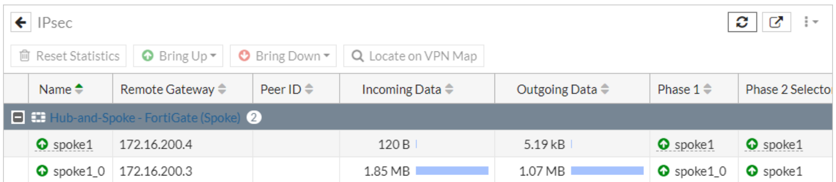


IPsec

Reset Statistics | Bring Up | Bring Down | Locate on VPN Map

Name	Remote Gateway	Peer ID	Incoming Data	Outgoing Data	Phase 1	Phase 2 Selectors
Hub-and-Spoke - FortiGate (Hub) 2						
hub_0	172.16.200.1		10.97 kB	5.34 kB	hub_0	hub
hub_1	172.16.200.3		3.51 kB	1.81 kB	hub_1	hub

2. On a spoke, go to *Dashboard* > *Network* and expand the IPsec widget.
   The tunnel to the hub and the spoke to spoke shortcut are established.



IPsec

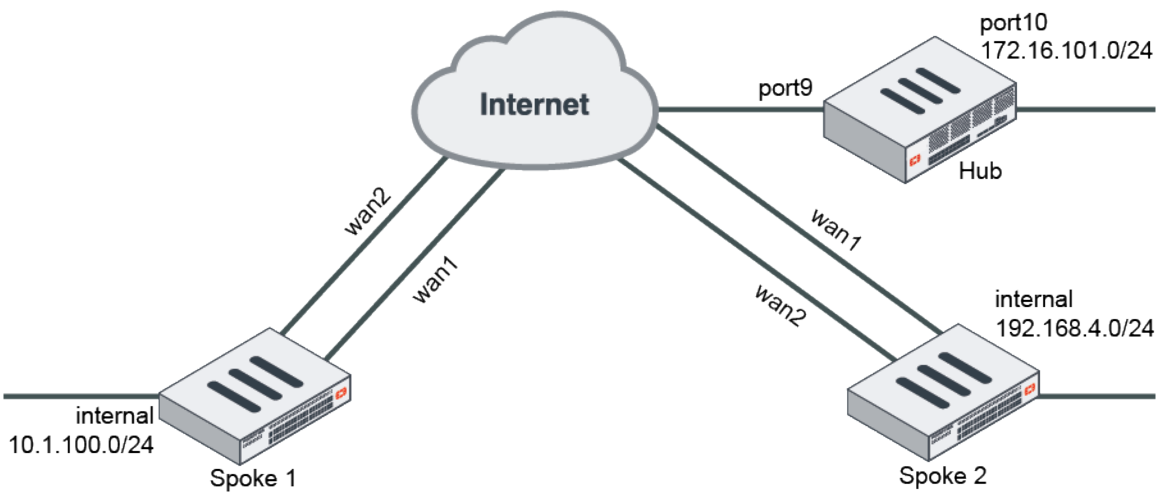
Reset Statistics | Bring Up | Bring Down | Locate on VPN Map

Name	Remote Gateway	Peer ID	Incoming Data	Outgoing Data	Phase 1	Phase 2 Selecto
Hub-and-Spoke - FortiGate (Spoke) 2						
spoke1	172.16.200.4		120 B	5.19 kB	spoke1	spoke1
spoke1_0	172.16.200.3		1.85 MB	1.07 MB	spoke1_0	spoke1

ADVPN with BGP as the routing protocol

This is a sample configuration of ADVPN with BGP as the routing protocol. The following options must be enabled for this configuration:

- On the hub FortiGate, IPsec phase1-interface `net-device disable` must be run.
- IBGP must be used between the hub and spoke FortiGates.
- `bgp neighbor-group/neighbor-range` must be reused.



To configure ADVPN with BGP as the routing protocol using the CLI:

1. Configure hub FortiGate WAN interface, internal interface, and a static route:

```
config system interface
    edit "port9"
        set alias "WAN"
        set ip 22.1.1.1 255.255.255.0
    next
    edit "port10"
        set alias "Internal"
        set ip 172.16.101.1 255.255.255.0
```

```
    next
end
config router static
    edit 1
        set gateway 22.1.1.2
        set device "port9"
    next
end
```

2. Configure the hub FortiGate:
   a. Configure the hub FortiGate IPsec phase1-interface and phase2-interface:

```
config vpn ipsec phase1-interface
    edit "advpn-hub"
        set type dynamic
        set interface "port9"
        set peertype any
        set net-device disable
        set proposal aes128-sha256 aes256-sha256 3des-sha256 aes128-sha1 aes256-sha1
3des-sha1
        set add-route disable
        set dpd on-idle
        set auto-discovery-sender enable
        set psksecret sample
        set dpd-retryinterval 5
    next
end
config vpn ipsec phase2-interface
    edit "advpn-hub"
        set phase1name "advpn-hub"
        set proposal aes128-sha1 aes256-sha1 3des-sha1 aes128-sha256 aes256-sha256
3des-sha256
    next
end
```

When `net-device` is disabled, a tunnel ID is generated for each dynamic tunnel. This ID, in the form of an IP address, is used as the gateway in the route entry to that tunnel. The `tunnel-search` option is removed in FortiOS 7.0.0 and later.

   b. Configure the hub FortiGate firewall policy:

```
config firewall policy
    edit 1
        set name "spoke2hub"
        set srcintf "advpn-hub"
        set dstintf "port10"
        set srcaddr "all"
        set dstaddr "172.16.101.0"
        set action accept
        set schedule "always"
        set service "ALL"
    next
    edit 2
        set name "spoke2spoke"
        set srcintf "advpn-hub"
```

```
        set dstintf "advpn-hub"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
    next
end
```

c. Configure the hub FortiGate's IPsec tunnel interface IP address:

```
config system interface
    edit "advpn-hub1"
        set ip 10.10.10.254 255.255.255.255
        set remote-ip 10.10.10.253 255.255.255.0
    next
end
```

d. Configure the hub FortiGate's BGP:

```
config router bgp
    set as 65412
    config neighbor-group
        edit "advpn"
            set link-down-failover enable
            set remote-as 65412
            set route-reflector-client enable
        next
    end
    config neighbor-range
        edit 1
            set prefix 10.10.10.0 255.255.255.0
            set neighbor-group "advpn"
        next
    end
    config network
        edit 1
            set prefix 172.16.101.0 255.255.255.0
        next
    end
end
```

3. Configure the spoke FortiGates:
   a. Configure the spoke FortiGates' WAN, internal interfaces, and static routes:
      i. Configure Spoke1:

```
config system interface
    edit "wan1"
        set alias "primary_WAN"
        set ip 15.1.1.2 255.255.255.0
    next
    edit "wan2"
        set alias "secondary_WAN"
        set ip 12.1.1.2 255.255.255.0
    next
    edit "internal"
        set ip 10.1.100.1 255.255.255.0
```

```
    next
end
config router static
    edit 1
        set gateway 12.1.1.1
        set device "wan2"
        set distance 15
    next
    edit 2
        set gateway 15.1.1.1
        set device "wan1"
    next
end
```

ii. Configure the Spoke2:

```
config system interface
    edit "wan1"
        set alias "primary_WAN"
        set ip 13.1.1.2 255.255.255.0
    next
    edit "wan2"
        set alias "secondary_WAN"
        set ip 17.1.1.2 255.255.255.0
    next
    edit "internal"
        set ip 192.168.4.1 255.255.255.0
    next
end
config router static
    edit 1
        set gateway 17.1.1.1
        set device "wan2"
        set distance 15
    next
    edit 2
        set gateway 13.1.1.1
        set device "wan1"
    next
end
```

b. Configure the spoke FortiGates' IPsec phase1-interface and phase2-interface:

i. Configure Spoke1:

```
config vpn ipsec phase1-interface
    edit "spoke1"
        set interface "wan1"
        set peertype any
        set net-device enable
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set add-route disable
        set dpd on-idle
        set auto-discovery-receiver enable
        set remote-gw 22.1.1.1
        set psksecret sample
        set dpd-retryinterval 5
    next
```

```
    edit "spoke1_backup"
        set interface "wan2"
        set peertype any
        set net-device enable
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set add-route disable
        set dpd on-idle
        set auto-discovery-receiver enable
        set remote-gw 22.1.1.1
        set monitor "spoke1"
        set psksecret sample
        set dpd-retryinterval 5
    next
end
config vpn ipsec phase2-interface
    edit "spoke1"
        set phase1name "spoke1"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256
aes128gcm aes256gcm chacha20poly1305
        set auto-negotiate enable
    next
    edit "spoke1_backup"
        set phase1name "spoke1_backup"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256
aes128gcm aes256gcm chacha20poly1305
        set auto-negotiate enable
    next
end
```

ii. Configure Spoke2:

```
config vpn ipsec phase1-interface
    edit "spoke2"
        set interface "wan1"
        set peertype any
        set net-device enable
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set add-route disable
        set dpd on-idle
        set auto-discovery-receiver enable
        set remote-gw 22.1.1.1
        set psksecret sample
        set dpd-retryinterval 5
    next
    edit "spoke2_backup"
        set interface "wan2"
        set peertype any
        set net-device enable
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set add-route disable
        set dpd on-idle
        set auto-discovery-receiver enable
        set remote-gw 22.1.1.1
        set monitor "spoke2"
        set psksecret sample
        set dpd-retryinterval 5
    next
```

```
end
config vpn ipsec phase2-interface
    edit "spoke2"
        set phase1name "spoke2"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256
aes128gcm aes256gcm chacha20poly1305
        set auto-negotiate enable
    next
    edit "spoke2_backup"
        set phase1name "spoke2_backup"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256
aes128gcm aes256gcm chacha20poly1305
        set auto-negotiate enable
    next
end
```

c. Configure the spoke FortiGates' firewall policies:

i. Configure Spoke1:

```
config firewall policy
    edit 1
        set name "outbound_advpn"
        set srcintf "internal"
        set dstintf "spoke1" "spoke1_backup"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
    next
    edit 2
        set name "inbound_advpn"
        set srcintf "spoke1" "spoke1_backup"
        set dstintf "internal"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
    next
end
```

ii. Configure Spoke2:

```
config firewall policy
    edit 1
        set name "outbound_advpn"
        set srcintf "internal"
        set dstintf "spoke2" "spoke2_backup"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
    next
    edit 2
        set name "inbound_advpn"
```

```
        set srcintf "spoke2" "spoke2_backup"
        set dstintf "internal"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
    next
end
```

d. Configure the spoke FortiGates' tunnel interface IP addresses:

i. Configure Spoke1:

```
config system interface
    edit "spoke1"
        set ip 10.10.10.1 255.255.255.255
        set remote-ip 10.10.10.254 255.255.255.0
    next
    edit "spoke1_backup"
        set ip 10.10.10.2 255.255.255.255
        set remote-ip 10.10.10.254 255.255.255.0
    next
end
```

ii. Configure Spoke2:

```
config system interface
    edit "spoke2"
        set ip 10.10.10.3 255.255.255.255
        set remote-ip 10.10.10.254 255.255.255.0
    next
    edit "spoke2_backup"
        set ip 10.10.10.4 255.255.255.255
        set remote-ip 10.10.10.254 255.255.255.0
    next
end
```

e. Configure the spoke FortiGates' BGP:

i. Configure Spoke1:

```
config router bgp
    set as 65412
    config neighbor
        edit "10.10.10.254"
            set advertisement-interval 1
            set link-down-failover enable
            set remote-as 65412
        next
    end
    config network
        edit 1
            set prefix 10.1.100.0 255.255.255.0
        next
    end
end
```

ii. Configure Spoke2:

```
config router bgp
    set as 65412
    config neighbor
        edit "10.10.10.254"
            set advertisement-interval 1
            set link-down-failover enable
            set remote-as 65412
        next
    end
    config network
        edit 1
            set prefix 192.168.4.0 255.255.255.0
        next
    end
end
```

4. Run `diagnose` and `get` commands on Spoke1 to check VPN and BGP states:
   a. Run the `diagnose vpn tunnel list` command on Spoke1. The system should return the following:

```
list all ipsec tunnel in vd 0
----
name=spoke1 ver=1 serial=2 15.1.1.2:0->22.1.1.1:0 tun_id=22.1.1.1
bound_if=7 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/536 options[0218]=npu
create_dev frag-rfc  accept_traffic=1

proxyid_num=1 child_num=1 refcnt=19 ilast=1 olast=1 ad=r/2
stat: rxp=1 txp=160 rxb=16428 txb=8969
dpd: mode=on-idle on=1 idle=5000ms retry=3 count=0 seqno=628
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=spoke1 proto=0 sa=1 ref=6 serial=1 auto-negotiate adr
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=6 options=1a227 type=00 soft=0 mtu=1438 expire=1225/0B replaywin=1024
       seqno=a1 esn=0 replaywin_lastseq=00000002 itn=0
  life: type=01 bytes=0/0 timeout=2369/2400
  dec: spi=c53a8f5b esp=aes key=16 cbe88682ad896a69290027b6dd8f7162
       ah=sha1 key=20 7bb704b388f83783ac76c2ab0b6c9f7dcf78e93b
  enc: spi=6e3633fc esp=aes key=16 1a0da3f4deed3d16becc9dda57537355
       ah=sha1 key=20 368544044bd9b82592d72476ff93d5055056da8d
  dec:pkts/bytes=1/16364, enc:pkts/bytes=160/19168
  npu_flag=03 npu_rgwy=22.1.1.1 npu_lgwy=15.1.1.2 npu_selid=1 dec_npuid=1 enc_npuid=1
----
name=spoke1_backup ver=1 serial=1 12.1.1.2:0->22.1.1.1:0 tun_id=22.1.1.1
bound_if=6 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/536 options[0218]=npu
create_dev frag-rfc  accept_traffic=0

proxyid_num=1 child_num=0 refcnt=11 ilast=0 olast=0 ad=/0
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-idle on=0 idle=5000ms retry=3 count=0 seqno=0
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=spoke1_backup proto=0 sa=0 ref=2 serial=1 auto-negotiate adr
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
```

   b. Run the `get router info bgp summary` command on Spoke1. The system should return the following:

```
BGP router identifier 7.7.7.7, local AS number 65412
BGP table version is 2
1 BGP AS-PATH entries
0 BGP community entries

Neighbor             V         AS [[QualityAssurance62/MsgRcvd]]
[[QualityAssurance62/MsgSent]]   [[QualityAssurance62/TblVer]]  InQ OutQ Up/Down
State/PfxRcd
10.10.10.254       1.          65412     143     142         1.     1.     1.
00:24:45                    2

Total number of neighbors 1
```

c. Run the `get router info routing-table bgp` command on Spoke1. The system should return the following:

```
Routing table for VRF=0
B       172.16.101.0/24 [200/0] via 10.10.10.254, spoke1, 00:23:57
B       192.168.4.0/24 [200/0] via 10.10.10.254, spoke1, 00:22:03
```

d. Generate traffic between the spokes and check the shortcut tunnel and routing table. Run the `diagnose vpn tunnel list` command on Spoke1. The system should return the following:

```
list all ipsec tunnel in vd 0
----
name=spoke1 ver=1 serial=2 15.1.1.2:0->22.1.1.1:0 tun_id=22.1.1.1
bound_if=7 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/536 options[0218]=npu
create_dev frag-rfc  accept_traffic=1

proxyid_num=1 child_num=1 refcnt=19 ilast=2 olast=2 ad=r/2
stat: rxp=1 txp=268 rxb=16428 txb=31243
dpd: mode=on-idle on=1 idle=5000ms retry=3 count=0 seqno=714
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=spoke1 proto=0 sa=1 ref=6 serial=1 auto-negotiate adr
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=6 options=1a227 type=00 soft=0 mtu=1438 expire=345/0B replaywin=1024
       seqno=10d esn=0 replaywin_lastseq=00000002 itn=0
  life: type=01 bytes=0/0 timeout=2369/2400
  dec: spi=c53a8f5b esp=aes key=16 cbe88682ad896a69290027b6dd8f7162
       ah=sha1 key=20 7bb704b388f83783ac76c2ab0b6c9f7dcf78e93b
  enc: spi=6e3633fc esp=aes key=16 1a0da3f4deed3d16becc9dda57537355
       ah=sha1 key=20 368544044bd9b82592d72476ff93d5055056da8d
  dec:pkts/bytes=1/16364, enc:pkts/bytes=268/48320
  npu_flag=03 npu_rgwy=22.1.1.1 npu_lgwy=15.1.1.2 npu_selid=1 dec_npuid=1 enc_npuid=1
----
name=spoke1_backup ver=1 serial=1 12.1.1.2:0->22.1.1.1:0 tun_id=22.1.1.1
bound_if=6 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/536 options[0218]=npu
create_dev frag-rfc  accept_traffic=0

proxyid_num=1 child_num=0 refcnt=11 ilast=8 olast=8 ad=/0
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-idle on=0 idle=5000ms retry=3 count=0 seqno=0
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=spoke1_backup proto=0 sa=0 ref=2 serial=1 auto-negotiate adr
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
```

```
----
name=spoke1_0 ver=1 serial=9 15.1.1.2:4500->13.1.1.2:4500 tun_id=13.1.1.2
bound_if=7 lgwy=static/1 tun=intf/0 mode=dial_inst/3 encap=none/728 options[02d8]=npu
create_dev no-sysctl rgwy-chg frag-rfc  accept_traffic=1

 parent=spoke1 index=0
proxyid_num=1 child_num=0 refcnt=17 ilast=4 olast=4 ad=r/2
stat: rxp=1 txp=100 rxb=112 txb=4686
dpd: mode=on-idle on=1 idle=5000ms retry=3 count=0 seqno=231
natt: mode=keepalive draft=32 interval=10 remote_port=4500
proxyid=spoke1 proto=0 sa=1 ref=5 serial=1 auto-negotiate adr
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=6 options=1a227 type=00 soft=0 mtu=1422 expire=447/0B replaywin=1024
       seqno=65 esn=0 replaywin_lastseq=00000002 itn=0
  life: type=01 bytes=0/0 timeout=2368/2400
  dec: spi=c53a8f5c esp=aes key=16 73fd9869547475db78851e6c057ad9b7
       ah=sha1 key=20 6ad3a5b1028f6b33c82ba494a370f13c7f462635
  enc: spi=79cb0f2b esp=aes key=16 52ab0acdc830d58c00e5956a6484654a
       ah=sha1 key=20 baa82aba4106dc60618f6fe95570728656799239
  dec:pkts/bytes=1/46, enc:pkts/bytes=100/11568
  npu_flag=03 npu_rgwy=13.1.1.2 npu_lgwy=15.1.1.2 npu_selid=5 dec_npuid=1 enc_npuid=1
```

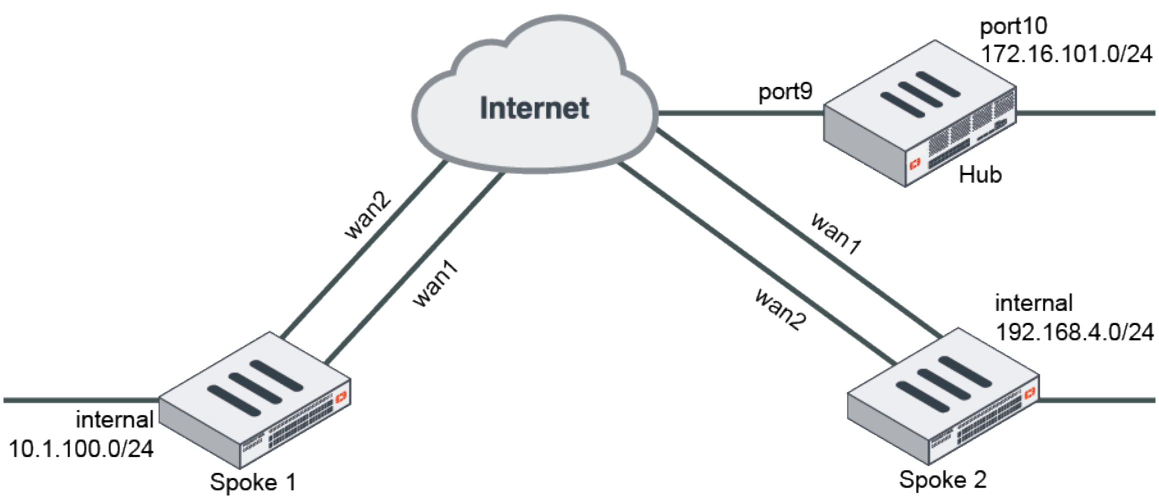
e. Run the `get router info routing-tale bgp` command. The system should return the following:

```
Routing table for VRF=0
B       172.16.101.0/24 [200/0] via 10.10.10.254, spoke1, 00:23:57
B       192.168.4.0/24 [200/0] via 10.10.10.3, spoke1_0 , 00:22:03
```

ADVPN with OSPF as the routing protocol

This is a sample configuration of ADVPN with OSPF as the routing protocol. The following options must be enabled for this configuration:

- On the hub FortiGate, IPsec phase1-interface `net-device enable` must be run.
- OSPF must be used between the hub and spoke FortiGates.



To configure ADVPN with OSPF as the routing protocol using the CLI:

1. Configure hub FortiGate's WAN, internal interface, and static route:

```
config system interface
    edit "port9"
        set alias "WAN"
        set ip 22.1.1.1 255.255.255.0
    next
    edit "port10"
        set alias "Internal"
        set ip 172.16.101.1 255.255.255.0
    next
end
config router static
    edit 1
        set gateway 22.1.1.2
        set device "port9"
    next
end
```

2. Configure the hub FortiGate:
   a. Configure the hub FortiGate IPsec phase1-interface and phase2-interface:

```
config vpn ipsec phase1-interface
    edit "advpn-hub"
        set type dynamic
        set interface "port9"
        set peertype any
        set net-device enable
        set proposal aes128-sha256 aes256-sha256 3des-sha256 aes128-sha1 aes256-sha1
3des-sha1
        set add-route disable
        set dpd on-idle
        set auto-discovery-sender enable
        set psksecret sample
        set dpd-retryinterval 5
    next
end
config vpn ipsec phase2-interface
    edit "advpn-hub"
        set phase1name "advpn-hub"
        set proposal aes128-sha1 aes256-sha1 3des-sha1 aes128-sha256 aes256-sha256
3des-sha256
    next
end
```

When `net-device` is disabled, a tunnel ID is generated for each dynamic tunnel. This ID, in the form of an IP address, is used as the gateway in the route entry to that tunnel. The `tunnel-search` option is removed in FortiOS 7.0.0 and later.

   b. Configure the hub FortiGate firewall policy:

```
config firewall policy
    edit 1
        set name "spoke2hub"
```

```
        set srcintf "advpn-hub"
        set dstintf "port10"
        set srcaddr "all"
        set dstaddr "172.16.101.0"
        set action accept
        set schedule "always"
        set service "ALL"
    next
    edit 2
        set name "spoke2spoke"
        set srcintf "advpn-hub"
        set dstintf "advpn-hub"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
    next
end
```

c. Configure the hub FortiGate's IPsec tunnel interface IP address:

```
config system interface
    edit "advpn-hub1"
        set ip 10.10.10.254 255.255.255.255
        set remote-ip 10.10.10.253 255.255.255.0
    next
end
```

d. Configure the hub FortiGate's OSPF:

```
config router ospf
    set router-id 1.1.1.1
    config area
        edit 0.0.0.0
        next
    end
    config network
        edit 1
            set prefix 10.10.10.0 255.255.255.0
        next
        edit 2
            set prefix 172.16.101.0 255.255.255.0
        next
    end
end
```

3. Configure the spoke FortiGates:
   a. Configure the spoke FortiGates' WAN, internal interfaces, and static routes:
      i. Configure Spoke1:

```
config system interface
    edit "wan1"
        set alias "primary_WAN"
        set ip 15.1.1.2 255.255.255.0
    next
    edit "wan2"
```

```
        set alias "secondary_WAN"
        set ip 12.1.1.2 255.255.255.0
    next
    edit "internal"
        set ip 10.1.100.1 255.255.255.0
    next
end
config router static
    edit 1
        set gateway 12.1.1.1
        set device "wan2"
        set distance 15
    next
    edit 2
        set gateway 15.1.1.1
        set device "wan1"
    next
end
```

ii. Configure the Spoke2:

```
config system interface
    edit "wan1"
        set alias "primary_WAN"
        set ip 13.1.1.2 255.255.255.0
    next
    edit "wan2"
        set alias "secondary_WAN"
        set ip 17.1.1.2 255.255.255.0
    next
    edit "internal"
        set ip 192.168.4.1 255.255.255.0
    next
end
config router static
    edit 1
        set gateway 17.1.1.1
        set device "wan2"
        set distance 15
    next
    edit 2
        set gateway 13.1.1.1
        set device "wan1"
    next
end
```

b. Configure the spoke FortiGates' IPsec phase1-interface and phase2-interface:

i. Configure Spoke1:

```
config vpn ipsec phase1-interface
    edit "spoke1"
        set interface "wan1"
        set peertype any
        set net-device enable
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set add-route disable
        set dpd on-idle
```

```
        set auto-discovery-receiver enable
        set remote-gw 22.1.1.1
        set psksecret sample
        set dpd-retryinterval 5
    next
    edit "spoke1_backup"
        set interface "wan2"
        set peertype any
        set net-device enable
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set add-route disable
        set dpd on-idle
        set auto-discovery-receiver enable
        set remote-gw 22.1.1.1
        set monitor "spoke1"
        set psksecret sample
        set dpd-retryinterval 5
    next
end
config vpn ipsec phase2-interface
    edit "spoke1"
        set phase1name "spoke1"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256
aes128gcm aes256gcm chacha20poly1305
        set auto-negotiate enable
    next
    edit "spoke1_backup"
        set phase1name "spoke1_backup"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256
aes128gcm aes256gcm chacha20poly1305
        set auto-negotiate enable
    next
end
```

ii. Configure Spoke2:

```
config vpn ipsec phase1-interface
    edit "spoke2"
        set interface "wan1"
        set peertype any
        set net-device enable
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set add-route disable
        set dpd on-idle
        set auto-discovery-receiver enable
        set remote-gw 22.1.1.1
        set psksecret sample
        set dpd-retryinterval 5
    next
    edit "spoke2_backup"
        set interface "wan2"
        set peertype any
        set net-device enable
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set add-route disable
        set dpd on-idle
        set auto-discovery-receiver enable
```

```
            set remote-gw 22.1.1.1
            set monitor "spoke2"
            set psksecret sample
            set dpd-retryinterval 5
        next
    end
    config vpn ipsec phase2-interface
        edit "spoke2"
            set phase1name "spoke2"
            set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256
    aes128gcm aes256gcm chacha20poly1305
            set auto-negotiate enable
        next
        edit "spoke2_backup"
            set phase1name "spoke2_backup"
            set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256
    aes128gcm aes256gcm chacha20poly1305
            set auto-negotiate enable
        next
    end
```

c. Configure the spoke FortiGates' firewall policies:

i. Configure Spoke1:

```
config firewall policy
    edit 1
        set name "outbound_advpn"
        set srcintf "internal"
        set dstintf "spoke1" "spoke1_backup"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
    next
    edit 2
        set name "inbound_advpn"
        set srcintf "spoke1" "spoke1_backup"
        set dstintf "internal"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
    next
end
```

ii. Configure Spoke2:

```
config firewall policy
    edit 1
        set name "outbound_advpn"
        set srcintf "internal"
        set dstintf "spoke2" "spoke2_backup"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
```

```
        set schedule "always"
        set service "ALL"
    next
    edit 2
        set name "inbound_advpn"
        set srcintf "spoke2" "spoke2_backup"
        set dstintf "internal"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
    next
end
```

d. Configure the spoke FortiGates' tunnel interface IP addresses:

i. Configure Spoke1:

```
config system interface
    edit "spoke1"
        set ip 10.10.10.1 255.255.255.255
        set remote-ip 10.10.10.254 255.255.255.0
    next
    edit "spoke1_backup"
        set ip 10.10.10.2 255.255.255.255
        set remote-ip 10.10.10.254 255.255.255.0
    next
end
```

ii. Configure Spoke2:

```
config system interface
    edit "spoke2"
        set ip 10.10.10.3 255.255.255.255
        set remote-ip 10.10.10.254 255.255.255.0
    next
    edit "spoke2_backup"
        set ip 10.10.10.4 255.255.255.255
        set remote-ip 10.10.10.254 255.255.255.0
    next
end
```

e. Configure the spoke FortiGates' OSPF:

i. Configure Spoke1:

```
config router ospf
    set router-id 7.7.7.7
    config area
        edit 0.0.0.0
        next
    end
    config network
        edit 1
            set prefix 10.10.10.0 255.255.255.0
        next
        edit 2
            set prefix 10.1.100.0 255.255.255.0
        next
```

```
        end
    end
```

ii. Configure Spoke2:

```
config router ospf
    set router-id 8.8.8.8
    config area
        edit 0.0.0.0
        next
    end
    config network
        edit 1
            set prefix 10.10.10.0 255.255.255.0
        next
        edit 2
            set prefix 192.168.4.0 255.255.255.0
        next
    end
end
```

4. Run `diagnose` and `get` commands on Spoke1 to check VPN and OSPF states:
   a. Run the `diagnose vpn tunnel list` command on Spoke1. The system should return the following:

```
list all ipsec tunnel in vd 0
----
name=spoke1 ver=1 serial=2 15.1.1.2:0->22.1.1.1:0 tun_id=22.1.1.1
bound_if=7 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/536 options[0218]=npu
create_dev frag-rfc  accept_traffic=1

proxyid_num=1 child_num=1 refcnt=19 ilast=5 olast=2 ad=r/2
stat: rxp=1 txp=263 rxb=16452 txb=32854
dpd: mode=on-idle on=1 idle=5000ms retry=3 count=0 seqno=2283
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=spoke1 proto=0 sa=1 ref=5 serial=1 auto-negotiate adr
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=6 options=1a227 type=00 soft=0 mtu=1438 expire=1057/0B replaywin=1024
       seqno=108 esn=0 replaywin_lastseq=00000003 itn=0
  life: type=01 bytes=0/0 timeout=2371/2400
  dec: spi=c53a8f78 esp=aes key=16 7cc50c5c9df1751f6497a4ad764c5e9a
       ah=sha1 key=20 269292ddbf7309a6fc05871e63ed8a5297b5c9a1
  enc: spi=6e363612 esp=aes key=16 42bd49bced1e85cf74a24d97f10eb601
       ah=sha1 key=20 13964f166aad48790c2e551d6df165d7489f524b
  dec:pkts/bytes=1/16394, enc:pkts/bytes=263/50096
  npu_flag=03 npu_rgwy=22.1.1.1 npu_lgwy=15.1.1.2 npu_selid=1 dec_npuid=1 enc_npuid=1
----
name=spoke1_backup ver=1 serial=1 12.1.1.2:0->22.1.1.1:0 tun_id=22.1.1.1
bound_if=6 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/536 options[0218]=npu
create_dev frag-rfc  accept_traffic=0

proxyid_num=1 child_num=0 refcnt=11 ilast=8 olast=8 ad=/0
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-idle on=0 idle=5000ms retry=3 count=0 seqno=0
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=spoke1_backup proto=0 sa=0 ref=2 serial=1 auto-negotiate adr
```

```
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
```

b. Run the `get router info ospf neighbor` command on Spoke1. The system should return the following:

```
OSPF process 0, VRF 0: Neighbor ID Pri State Dead Time Address Interface 8.8.8.8 1.
Full/ - 00:00:35 10.10.10.254 spoke1 1.1.1.1 1. Full/ - 00:00:35 10.10.10.254 spoke1
```

c. Run the `get router info routing-table ospf` command on Spoke1. The system should return the following:

```
Routing table for VRF=0
O       172.16.101.0/24 [110/110] via 10.10.10.254, spoke1, 00:23:23
O       192.168.4.0/24   [110/110] via 10.10.10.254, spoke1, 00:22:35
```

d. Generate traffic between the spokes, then check the shortcut tunnel and routing table. Run the `diagnose vpn tunnel list` command on Spoke1. The system should return the following:

```
list all ipsec tunnel in vd 0
----
----
name=spoke1 ver=1 serial=2 15.1.1.2:0->22.1.1.1:0 tun_id=22.1.1.1
bound_if=7 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/536 options[0218]=npu
create_dev frag-rfc  accept_traffic=1

proxyid_num=1 child_num=1 refcnt=19 ilast=2 olast=2 ad=r/2
stat: rxp=1 txp=313 rxb=16452 txb=35912
dpd: mode=on-idle on=1 idle=5000ms retry=3 count=0 seqno=2303
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=spoke1 proto=0 sa=1 ref=3 serial=1 auto-negotiate adr
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=6 options=1a227 type=00 soft=0 mtu=1438 expire=782/0B replaywin=1024
       seqno=13a esn=0 replaywin_lastseq=00000003 itn=0
  life: type=01 bytes=0/0 timeout=2371/2400
  dec: spi=c53a8f78 esp=aes key=16 7cc50c5c9df1751f6497a4ad764c5e9a
       ah=sha1 key=20 269292ddbf7309a6fc05871e63ed8a5297b5c9a1
  enc: spi=6e363612 esp=aes key=16 42bd49bced1e85cf74a24d97f10eb601
       ah=sha1 key=20 13964f166aad48790c2e551d6df165d7489f524b
  dec:pkts/bytes=1/16394, enc:pkts/bytes=313/56432
  npu_flag=03 npu_rgwy=22.1.1.1 npu_lgwy=15.1.1.2 npu_selid=1 dec_npuid=1 enc_npuid=1
----
name=spoke1_backup ver=1 serial=1 12.1.1.2:0->22.1.1.1:0 tun_id=22.1.1.1
bound_if=6 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/536 options[0218]=npu
create_dev frag-rfc  accept_traffic=0

proxyid_num=1 child_num=0 refcnt=11 ilast=13 olast=13 ad=/0
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-idle on=0 idle=5000ms retry=3 count=0 seqno=0
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=spoke1_backup proto=0 sa=0 ref=2 serial=1 auto-negotiate adr
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
----
name=spoke1_0 ver=1 serial=e 15.1.1.2:4500->13.1.1.2:4500 tun_id=13.1.1.2
bound_if=7 lgwy=static/1 tun=intf/0 mode=dial_inst/3 encap=none/728 options[02d8]=npu
create_dev no-sysctl rgwy-chg frag-rfc  accept_traffic=1
```

```
 parent=spoke1 index=0
proxyid_num=1 child_num=0 refcnt=19 ilast=4 olast=2 ad=r/2
stat: rxp=641 txp=1254 rxb=278648 txb=161536
dpd: mode=on-idle on=1 idle=5000ms retry=3 count=0 seqno=184
natt: mode=keepalive draft=32 interval=10 remote_port=4500
proxyid=spoke1_backup proto=0 sa=1 ref=10 serial=1 auto-negotiate adr
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=6 options=1a227 type=00 soft=0 mtu=1422 expire=922/0B replaywin=1024
       seqno=452 esn=0 replaywin_lastseq=00000280 itn=0
  life: type=01 bytes=0/0 timeout=2370/2400
  dec: spi=c53a8f79 esp=aes key=16 324f8cf840ba6722cc7abbba46b34e0e
       ah=sha1 key=20 a40e9aac596b95c4cd83a7f6372916a5ef5aa505
  enc: spi=ef3327b5 esp=aes key=16 5909d6066b303de4520d2b5ae2db1b61
       ah=sha1 key=20 1a42f5625b5a335d8d5282fe83b5d6c6ff26b2a4
  dec:pkts/bytes=641/278568, enc:pkts/bytes=1254/178586
  npu_flag=03 npu_rgwy=13.1.1.2 npu_lgwy=15.1.1.2 npu_selid=a dec_npuid=1 enc_npuid=1
```

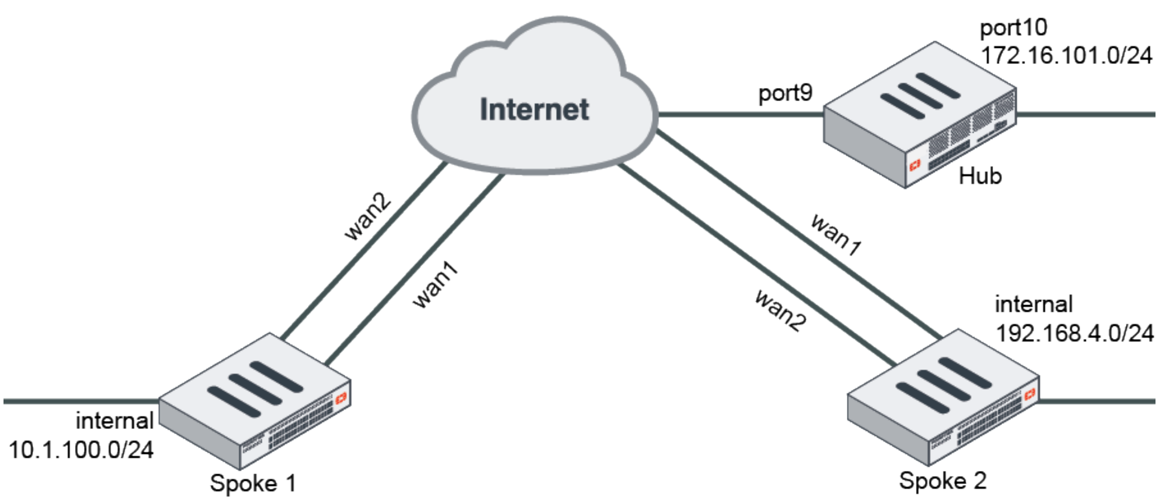
e. Run the `get router info routing-tale ospf` command. The system should return the following:

```
Routing table for VRF=0
O       172.16.101.0/24 [110/110] via 10.10.10.254, spoke1, 00:27:14
O       192.168.4.0/24  [110/110] via  10.10.10.3, spoke1_0, 00:26:26
```

ADVPN with RIP as the routing protocol

This is a sample configuration of ADVPN with RIP as routing protocol. The following options must be enabled for this configuration:

- On the hub FortiGate, IPsec phase1-interface `net-device disable` must be run.
- RIP must be used between the hub and spoke FortiGates.
- `split-horizon-status enable` must be run on the hub FortiGate.



To configure ADVPN with RIP as the routing protocol using the CLI:

1. In the CLI, configure hub FortiGate's WAN, internal interface, and static route:

```
config system interface
    edit "port9"
        set alias "WAN"
        set ip 22.1.1.1 255.255.255.0
    next
    edit "port10"
        set alias "Internal"
        set ip 172.16.101.1 255.255.255.0
    next
end
config router static
    edit 1
        set gateway 22.1.1.2
        set device "port9"
    next
end
```

2. Configure the hub FortiGate:
   a. Configure the hub FortiGate IPsec phase1-interface and phase2-interface:

```
config vpn ipsec phase1-interface
    edit "advpn-hub"
        set type dynamic
        set interface "port9"
        set peertype any
        set net-device disable
        set proposal aes128-sha256 aes256-sha256 3des-sha256 aes128-sha1 aes256-sha1
3des-sha1
        set add-route disable
        set dpd on-idle
        set auto-discovery-sender enable
        set psksecret sample
        set dpd-retryinterval 5
    next
end
config vpn ipsec phase2-interface
    edit "advpn-hub"
        set phase1name "advpn-hub"
        set proposal aes128-sha1 aes256-sha1 3des-sha1 aes128-sha256 aes256-sha256
3des-sha256
    next
end
```

When `net-device` is disabled, a tunnel ID is generated for each dynamic tunnel. This ID, in the form of an IP address, is used as the gateway in the route entry to that tunnel. The `tunnel-search` option is removed in FortiOS 7.0.0 and later.

   b. Configure the hub FortiGate firewall policy:

```
config firewall policy
    edit 1
        set name "spoke2hub"
```

```
        set srcintf "advpn-hub"
        set dstintf "port10"
        set srcaddr "all"
        set dstaddr "172.16.101.0"
        set action accept
        set schedule "always"
        set service "ALL"
    next
    edit 2
        set name "spoke2spoke"
        set srcintf "advpn-hub"
        set dstintf "advpn-hub"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
    next
end
```

c. Configure the hub FortiGate's IPsec tunnel interface IP address:

```
config system interface
    edit "advpn-hub1"
        set ip 10.10.10.254 255.255.255.255
        set remote-ip 10.10.10.253 255.255.255.0
    next
end
```

d. Configure the hub FortiGate's RIP:

```
config router rip
    set default-information-originate enable
    config network
        edit 1
            set prefix 10.10.10.0 255.255.255.0
        next
        edit 2
            set prefix 172.16.101.0 255.255.255.0
        next
    end
    config interface
        edit "advpn-hub"
            set split-horizon-status disable
        next
    end
end
```

3. Configure the spoke FortiGates:
   a. Configure the spoke FortiGates' WAN, internal interfaces, and static routes:
      i. Configure Spoke1:

```
config system interface
    edit "wan1"
        set alias "primary_WAN"
        set ip 15.1.1.2 255.255.255.0
    next
```

```
    edit "wan2"
        set alias "secondary_WAN"
        set ip 12.1.1.2 255.255.255.0
    next
    edit "internal"
        set ip 10.1.100.1 255.255.255.0
    next
end
config router static
    edit 1
        set gateway 12.1.1.1
        set device "wan2"
        set distance 15
    next
    edit 2
        set gateway 15.1.1.1
        set device "wan1"
    next
end
```

ii. Configure the Spoke2:

```
config system interface
    edit "wan1"
        set alias "primary_WAN"
        set ip 13.1.1.2 255.255.255.0
    next
    edit "wan2"
        set alias "secondary_WAN"
        set ip 17.1.1.2 255.255.255.0
    next
    edit "internal"
        set ip 192.168.4.1 255.255.255.0
    next
end
config router static
    edit 1
        set gateway 17.1.1.1
        set device "wan2"
        set distance 15
    next
    edit 2
        set gateway 13.1.1.1
        set device "wan1"
    next
end
```

b. Configure the spoke FortiGates' IPsec phase1-interface and phase2-interface:

i. Configure Spoke1:

```
config vpn ipsec phase1-interface
    edit "spoke1"
        set interface "wan1"
        set peertype any
        set net-device enable
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set add-route disable
```

```
        set dpd on-idle
        set auto-discovery-receiver enable
        set remote-gw 22.1.1.1
        set psksecret sample
        set dpd-retryinterval 5
    next
    edit "spoke1_backup"
        set interface "wan2"
        set peertype any
        set net-device enable
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set add-route disable
        set dpd on-idle
        set auto-discovery-receiver enable
        set remote-gw 22.1.1.1
        set monitor "spoke1"
        set psksecret sample
        set dpd-retryinterval 5
    next
end
config vpn ipsec phase2-interface
    edit "spoke1"
        set phase1name "spoke1"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256
aes128gcm aes256gcm chacha20poly1305
        set auto-negotiate enable
    next
    edit "spoke1_backup"
        set phase1name "spoke1_backup"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256
aes128gcm aes256gcm chacha20poly1305
        set auto-negotiate enable
    next
end
```

ii. Configure Spoke2:

```
config vpn ipsec phase1-interface
    edit "spoke2"
        set interface "wan1"
        set peertype any
        set net-device enable
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set add-route disable
        set dpd on-idle
        set auto-discovery-receiver enable
        set remote-gw 22.1.1.1
        set psksecret sample
        set dpd-retryinterval 5
    next
    edit "spoke2_backup"
        set interface "wan2"
        set peertype any
        set net-device enable
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set add-route disable
        set dpd on-idle
```

```
        set auto-discovery-receiver enable
        set remote-gw 22.1.1.1
        set monitor "spoke2"
        set psksecret sample
        set dpd-retryinterval 5
    next
end
config vpn ipsec phase2-interface
    edit "spoke2"
        set phase1name "spoke2"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256
aes128gcm aes256gcm chacha20poly1305
        set auto-negotiate enable
    next
    edit "spoke2_backup"
        set phase1name "spoke2_backup"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256
aes128gcm aes256gcm chacha20poly1305
        set auto-negotiate enable
    next
end
```

c. Configure the spoke FortiGates' firewall policies:

i. Configure Spoke1:

```
config firewall policy
    edit 1
        set name "outbound_advpn"
        set srcintf "internal"
        set dstintf "spoke1" "spoke1_backup"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
    next
    edit 2
        set name "inbound_advpn"
        set srcintf "spoke1" "spoke1_backup"
        set dstintf "internal"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
    next
end
```

ii. Configure Spoke2:

```
config firewall policy
    edit 1
        set name "outbound_advpn"
        set srcintf "internal"
        set dstintf "spoke2" "spoke2_backup"
        set srcaddr "all"
        set dstaddr "all"
```

```
            set action accept
            set schedule "always"
            set service "ALL"
        next
        edit 2
            set name "inbound_advpn"
            set srcintf "spoke2" "spoke2_backup"
            set dstintf "internal"
            set srcaddr "all"
            set dstaddr "all"
            set action accept
            set schedule "always"
            set service "ALL"
        next
    end
```

d. Configure the spoke FortiGates' tunnel interface IP addresses:

   i. Configure Spoke1:

```
config system interface
    edit "spoke1"
        set ip 10.10.10.1 255.255.255.255
        set remote-ip 10.10.10.254 255.255.255.0
    next
    edit "spoke1_backup"
        set ip 10.10.10.2 255.255.255.255
        set remote-ip 10.10.10.254 255.255.255.0
    next
end
```

   ii. Configure Spoke2:

```
config system interface
    edit "spoke2"
        set ip 10.10.10.3 255.255.255.255
        set remote-ip 10.10.10.254 255.255.255.0
    next
    edit "spoke2_backup"
        set ip 10.10.10.4 255.255.255.255
        set remote-ip 10.10.10.254 255.255.255.0
    next
end
```

e. Configure the spoke FortiGates' RIP:

   i. Configure Spoke1:

```
config router rip
    config network
        edit 1
            set prefix 10.10.10.0 255.255.255.0
        next
        edit 2
            set prefix 10.1.100.0 255.255.255.0
        next
    end
end
```

   ii. Configure Spoke2:

```
config router rip
    config network
        edit 1
            set prefix 10.10.10.0 255.255.255.0
        next
        edit 2
            set prefix 192.168.4.0 255.255.255.0
        next
    end
end
```

4. Run `diagnose` and `get` commands on Spoke1:
   a. Run the `diagnose vpn tunnel list` command on Spoke1. The system should return the following:

```
list all ipsec tunnel in vd 0
----
name=spoke1 ver=1 serial=2 15.1.1.2:0->22.1.1.1:0 tun_id=22.1.1.1
bound_if=7 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/536 options[0218]=npu
create_dev frag-rfc  accept_traffic=1

proxyid_num=1 child_num=1 refcnt=17 ilast=2 olast=2 ad=r/2
stat: rxp=1 txp=87 rxb=200 txb=6208
dpd: mode=on-idle on=1 idle=5000ms retry=3 count=0 seqno=1040
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=spoke1 proto=0 sa=1 ref=4 serial=1 auto-negotiate adr
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=7 options=1a227 type=00 soft=0 mtu=1438 expire=1793/0B replaywin=1024
       seqno=57 esn=0 replaywin_lastseq=00000002 itn=0
  life: type=01 bytes=0/0 timeout=2370/2400
  dec: spi=c53a8f60 esp=aes key=16 6b54e32d54d039196a74d96e96d1cf14
       ah=sha1 key=20 e4903474614eafc96eda6400a3a5e88bbcb26a7f
  enc: spi=6e36349d esp=aes key=16 914a40a7993eda75c4dea2f42905f27d
       ah=sha1 key=20 8040eb08342edea2dae5eee058fd054a46688267
  dec:pkts/bytes=1/132, enc:pkts/bytes=86/11696
  npu_flag=03 npu_rgwy=22.1.1.1 npu_lgwy=15.1.1.2 npu_selid=1 dec_npuid=1 enc_npuid=1
----
name=spoke1_backup ver=1 serial=1 12.1.1.2:0->22.1.1.1:0 tun_id=22.1.1.1
bound_if=6 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/536 options[0218]=npu
create_dev frag-rfc  accept_traffic=0

proxyid_num=1 child_num=0 refcnt=11 ilast=0 olast=0 ad=/0
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-idle on=0 idle=5000ms retry=3 count=0 seqno=0
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=spoke1_backup proto=0 sa=0 ref=2 serial=1 auto-negotiate adr
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
```

   b. Run the `get router info rip database` command on Spoke1. The system should return the following:

```
Codes: R - RIP, Rc - RIP connected, Rs - RIP static, K - Kernel,
       C - Connected, S - Static, O - OSPF, I - IS-IS, B - BGP

   Network            Next Hop         Metric From            If      Time
Rc 10.1.100.0/24                             1.                 internal
Rc 10.10.10.2/32                               1.                 spoke1
```

```
R  172.16.101.0/24    10.10.10.254            1. 10.10.10.254      spoke1 02:28
R  192.168.4.0/24      10.10.10.254             1. 10.10.10.254       spoke1 02:44
```

c. Run the `get router info routing-table rip` command on Spoke1. The system should return the following:

```
Routing table for VRF=0
R       172.16.101.0/24 [120/2] via 10.10.10.254, spoke1, 00:08:38
R       192.168.4.0/24 [120/3] via 10.10.10.254, spoke1, 00:08:38
```

d. Generate traffic between the spokes, then check the shortcut tunnel and routing table. Run the `diagnose vpn tunnel list` command on Spoke1. The system should return the following:

```
list all ipsec tunnel in vd 0
----
name=spoke1 ver=1 serial=2 15.1.1.2:0->22.1.1.1:0 tun_id=22.1.1.1
bound_if=7 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/536 options[0218]=npu
create_dev frag-rfc  accept_traffic=1

proxyid_num=1 child_num=0 refcnt=19 ilast=3 olast=3 ad=r/2
stat: rxp=1 txp=78 rxb=200 txb=5546
dpd: mode=on-idle on=1 idle=5000ms retry=3 count=0 seqno=1039
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=spoke1 proto=0 sa=1 ref=5 serial=1 auto-negotiate adr
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=7 options=1a227 type=00 soft=0 mtu=1438 expire=1807/0B replaywin=1024
       seqno=4e esn=0 replaywin_lastseq=00000002 itn=0
  life: type=01 bytes=0/0 timeout=2370/2400
  dec: spi=c53a8f60 esp=aes key=16 6b54e32d54d039196a74d96e96d1cf14
       ah=sha1 key=20 e4903474614eafc96eda6400a3a5e88bbcb26a7f
  enc: spi=6e36349d esp=aes key=16 914a40a7993eda75c4dea2f42905f27d
       ah=sha1 key=20 8040eb08342edea2dae5eee058fd054a46688267
  dec:pkts/bytes=1/132, enc:pkts/bytes=77/10456
  npu_flag=03 npu_rgwy=22.1.1.1 npu_lgwy=15.1.1.2 npu_selid=1 dec_npuid=1 enc_npuid=1
----
name=spoke1_backup ver=1 serial=1 12.1.1.2:0->22.1.1.1:0 tun_id=22.1.1.1
bound_if=6 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/536 options[0218]=npu
create_dev frag-rfc  accept_traffic=0

proxyid_num=1 child_num=0 refcnt=11 ilast=20 olast=20 ad=/0
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-idle on=0 idle=5000ms retry=3 count=0 seqno=0
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=spoke1_backup proto=0 sa=0 ref=2 serial=1 auto-negotiate adr
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
----
name=spoke1_0 ver=1 serial=a 15.1.1.2:4500->13.1.1.2:4500 tun_id=13.1.1.2
bound_if=7 lgwy=static/1 tun=intf/0 mode=dial_inst/3 encap=none/728 options[02d8]=npu
create_dev no-sysctl rgwy-chg frag-rfc  accept_traffic=1

 parent=spoke1 index=0
proxyid_num=1 child_num=0 refcnt=20 ilast=2 olast=0 ad=r/2
stat: rxp=1 txp=7 rxb=112 txb=480
dpd: mode=on-idle on=1 idle=5000ms retry=3 count=0 seqno=0
natt: mode=keepalive draft=32 interval=10 remote_port=4500
```

```
proxyid=spoke1 proto=0 sa=1 ref=8 serial=1 auto-negotiate adr
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=6 options=1a227 type=00 soft=0 mtu=1422 expire=2358/0B replaywin=1024
       seqno=8 esn=0 replaywin_lastseq=00000002 itn=0
  life: type=01 bytes=0/0 timeout=2367/2400
  dec: spi=c53a8f61 esp=aes key=16 c66aa7ae9657068108ed47c048ff56b6
       ah=sha1 key=20 60661c68e20bbc913c2564ade85e01ea3769e703
  enc: spi=79cb0f30 esp=aes key=16 bf6c898c2e1c64baaa679ed5d79c3b58
       ah=sha1 key=20 146ca78be6c34eedb9cd66cc328216e08682ecb1
  dec:pkts/bytes=1/46, enc:pkts/bytes=7/992
  npu_flag=03 npu_rgwy=13.1.1.2 npu_lgwy=15.1.1.2 npu_selid=6 dec_npuid=1 enc_npuid=1
```

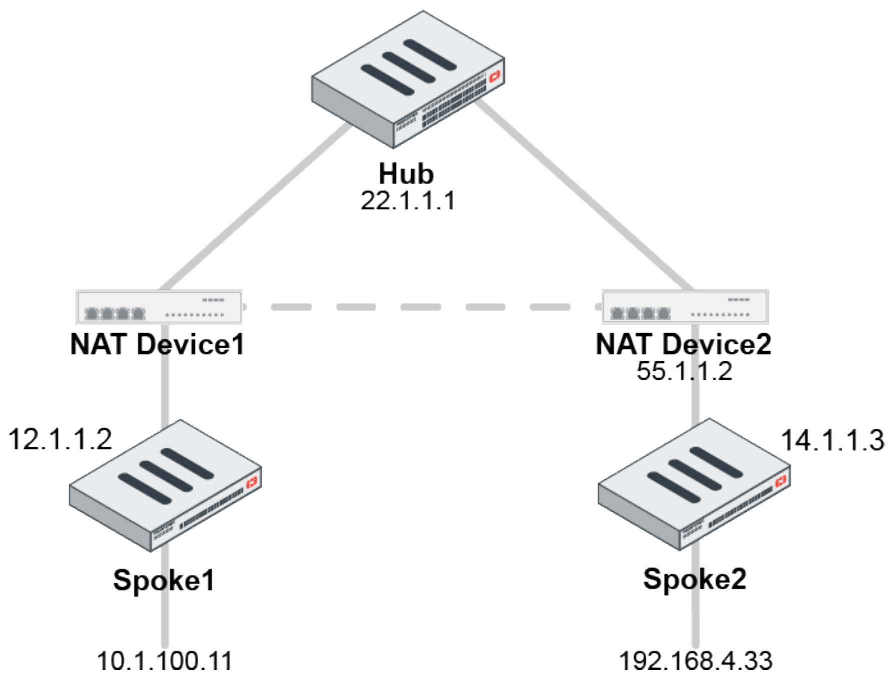
e. Run the `get router info routing-tale rip` command. The system should return the following:

```
Routing table for VRF=0
R       172.16.101.0/24 [120/2] via 10.10.10.254, spoke1, 00:09:04
R       192.168.4.0/24 [120/2] via 10.10.10.3, spoke1_0, 00:00:02
```

UDP hole punching for spokes behind NAT

UDP hole punching allows ADVPN shortcuts to be established through a UDP hole on a NAT device. The NAT device must support RFC 4787 Endpoint-Independent Mapping.

In the following example, device 10.1.100.11 behind Spoke1 needs to reach device 192.168.4.33 behind Spoke2. Spoke1 and Spoke2 are behind NAT devices and have established IPsec tunnels to the Hub. The hole punching creates a shortcut between Spoke1 and Spoke2 that bypasses the Hub.



To verify the ADVPN shortcut is established between both spokes behind NAT:

```
# diagnose debug enable
# diagnose debug application ike -1
ike 0: comes 22.1.1.1:4500->12.1.1.2:4500,ifindex=6....
```

```
ike 0: IKEv1 exchange=Informational id=3c10fb6a76f1e264/6c7b397100dffc63:58ac7c02 len=204
ike 0:toHub1:35: notify msg received: SHORTCUT-OFFER
ike 0:toHub1: shortcut-offer 10.1.100.11->192.168.4.33 psk 64 ppk 0 ver 1 mode 0
ike 0 looking up shortcut by addr 192.168.4.33, name toHub1
ike 0:toHub1: send shortcut-query 1438189781753480593 d3fdd1bfbc94caee/0000000000000000
12.1.1.2 10.1.100.11->192.168.4.33 psk 64 ttl 32 nat 1 ver 1 mode 0
ike 0:toHub1:35: sent IKE msg (SHORTCUT-QUERY): 12.1.1.2:4500->22.1.1.1:4500, len=236,
id=3c10fb6a76f1e264/6c7b397100dffc63:12e263f7
ike 0: comes 22.1.1.1:4500->12.1.1.2:4500,ifindex=6....
ike 0: IKEv1 exchange=Informational id=3c10fb6a76f1e264/6c7b397100dffc63:4976e1ac len=236
ike 0:toHub1:35: notify msg received: SHORTCUT-REPLY
ike 0:toHub1: recv shortcut-reply 1438189781753480593 d3fdd1bfbc94caee/16a1eb5b0f37ee23
14.1.1.3 to 10.1.100.11 psk 64 ppk 0 ver 1 mode 0 nat 55.1.1.2:64916
ike 0:toHub1: iif 22 192.168.4.33->10.1.100.11 route lookup oif 21
ike 0:toHub1: shortcut-reply received from 55.1.1.2:64916, local-nat=yes, peer-nat=yes
ike 0:toHub1: NAT hole punching to peer at 55.1.1.2:64916
ike 0:toHub1: created connection: 0x5e71f58 6 12.1.1.2->55.1.1.2:64916.
<==55.1.1.2:64916 this is UDP hole of NAT device
ike 0:toHub1: adding new dynamic tunnel for 55.1.1.2:64916
ike 0:toHub1_0: added new dynamic tunnel for 55.1.1.2:64916
ike 0:toHub1_0:48: initiator: main mode is sending 1st message...
ike 0:toHub1_0:48: cookie d3fdd1bfbc94caee/16a1eb5b0f37ee23
ike 0:toHub1_0:48: sent IKE msg (ident_i1send): 12.1.1.2:4500->55.1.1.2:64916, len=632,
id=d3fdd1bfbc94caee/16a1eb5b0f37ee23
ike 0: comes 55.1.1.2:64916->12.1.1.2:4500,ifindex=6....
ike 0: IKEv1 exchange=Identity Protection id=d3fdd1bfbc94caee/16a1eb5b0f37ee23 len=252
ike 0:toHub1_0:48: initiator: main mode get 1st response...
…
…
ike 0:toHub1_0:48: negotiation result
ike 0:toHub1_0:48: proposal id = 1:
ike 0:toHub1_0:48:   protocol id = ISAKMP:
ike 0:toHub1_0:48:      trans_id = KEY_IKE.
ike 0:toHub1_0:48:      encapsulation = IKE/none
ike 0:toHub1_0:48:         type=OAKLEY_ENCRYPT_ALG, val=AES_CBC, key-len=128
ike 0:toHub1_0:48:         type=OAKLEY_HASH_ALG, val=SHA2_256.
ike 0:toHub1_0:48:         type=AUTH_METHOD, val=PRESHARED_KEY.
ike 0:toHub1_0:48:         type=OAKLEY_GROUP, val=MODP2048.
ike 0:toHub1_0:48: ISAKMP SA lifetime=86400
ike 0:toHub1_0:48: sent IKE msg (ident_i2send): 12.1.1.2:4500->55.1.1.2:64916, len=380,
id=d3fdd1bfbc94caee/16a1eb5b0f37ee23
ike 0: comes 55.1.1.2:64916->12.1.1.2:4500,ifindex=6....
ike 0: IKEv1 exchange=Identity Protection id=d3fdd1bfbc94caee/16a1eb5b0f37ee23 len=380
ike 0:toHub1_0:48: initiator: main mode get 2nd response...
…
…
ike 0:toHub1_0:48: add INITIAL-CONTACT
ike 0:toHub1_0:48: add INTERFACE-ADDR4 10.10.1.100
ike 0:toHub1_0:48: sent IKE msg (ident_i3send): 12.1.1.2:4500->55.1.1.2:64916, len=140,
id=d3fdd1bfbc94caee/16a1eb5b0f37ee23
ike 0: comes 55.1.1.2:64916->12.1.1.2:4500,ifindex=6....
ike 0: IKEv1 exchange=Identity Protection id=d3fdd1bfbc94caee/16a1eb5b0f37ee23 len=124
ike 0:toHub1_0:48: initiator: main mode get 3rd response...
ike 0:toHub1_0:48: received p1 notify type INTERFACE-ADDR4
ike 0:toHub1_0:48: INTERFACE-ADDR4 10.10.1.102
ike 0:toHub1_0:48: peer identifier IPV4_ADDR 14.1.1.3
```

```
ike 0:toHub1_0:48: PSK authentication succeeded
ike 0:toHub1_0:48: authentication OK
ike 0:toHub1_0:48: established IKE SA d3fdd1bfbc94caee/16a1eb5b0f37ee23
ike 0:toHub1_0:48: auto-discovery receiver
ike 0:toHub1_0:48: auto-discovery 2
ike 0:toHub1_0: add R/32 route 10.10.1.102 via 10.10.1.102, intf=toHub1(22)
ike 0:toHub1_0: add peer route 10.10.1.102
ike 0:toHub1: schedule auto-negotiate
ike 0:toHub1_0:48: no pending Quick-Mode negotiations
ike 0:toHub1_0:toHub1: IPsec SA connect 6 12.1.1.2->55.1.1.2:64916
ike 0:toHub1_0:toHub1: using existing connection
ike 0:toHub1_0:toHub1: traffic triggered, serial=1 1:10.1.100.11:2048->1:192.168.4.33:0
ike 0:toHub1:toHub1: config found
ike 0:toHub1_0:toHub1: IPsec SA connect 6 12.1.1.2->55.1.1.2:64916 negotiating
ike 0:toHub1_0:48: cookie d3fdd1bfbc94caee/16a1eb5b0f37ee23:8465e467
ike 0:toHub1_0:48:toHub1:109: natt flags 0x1f, encmode 1->3
ike 0:toHub1_0:48:toHub1:109: initiator selectors 0 0:0.0.0.0/0.0.0.0:0:0-
>0:0.0.0.0/0.0.0.0:0:0
ike 0:toHub1_0:48: sent IKE msg (quick_i1send): 12.1.1.2:4500->55.1.1.2:64916, len=620,
id=d3fdd1bfbc94caee/16a1eb5b0f37ee23:8465e467
ike 0: comes 55.1.1.2:64916->12.1.1.2:4500,ifindex=6....
ike 0: IKEv1 exchange=Quick id=d3fdd1bfbc94caee/16a1eb5b0f37ee23:8465e467 len=444
ike 0:toHub1_0:48:toHub1:109: responder selectors 0:0.0.0.0/0.0.0.0:0->0:0.0.0.0/0.0.0.0:0
ike 0:toHub1_0:48:toHub1:109: my proposal:
…
…
ike 0:toHub1_0:48:toHub1:109: add IPsec SA: SPIs=79654cf1/5e9936a5
ike 0:toHub1_0:48:toHub1:109: IPsec SA dec spi 79654cf1 key
16:5E21180992B8892DE5142E1F53ABD29E auth 20:49AA4AE14994A39A138392AC517B6E79D98CA673
ike 0:toHub1_0:48:toHub1:109: IPsec SA enc spi 5e9936a5 key
16:BE16B8EF4E75F7B3CF97A1D58D996890 auth 20:2F46B57CAC6F3185BB182F9280312263325F6BAF
ike 0:toHub1_0:48:toHub1:109: added IPsec SA: SPIs=79654cf1/5e9936a5
ike 0:toHub1_0:48:toHub1:109: sending SNMP tunnel UP trapp
```

To verify the spoke-to-spoke IPsec phase 1 tunnel shortcut is established:

```
# diagnose vpn ike gateway list
vd: root/0
name: toHub1
version: 1
interface: wan2 6
addr: 12.1.1.2:4500 -> 22.1.1.1:4500
tun_id: 22.1.1.1
created: 503s ago
assigned IPv4 address: 10.10.1.100/255.255.255.0
nat: me
auto-discovery: 2 receiver
IKE SA: created 1/1  established 1/1  time 0/0/0 ms
IPsec SA: created 1/3  established 1/3  time 0/0/0 ms

  id/spi: 35 3c10fb6a76f1e264/6c7b397100dffc63
  direction: initiator
  status: established 503-503s ago = 0ms
  proposal: aes128-sha256
  key: 7fca86063ea2e72f-4efea6f1bec23948
  lifetime/rekey: 86400/85596
```

```
  DPD sent/recv: 00000000/00000000

vd: root/0
name: toHub1_0
version: 1
interface: wan2 6
addr: 12.1.1.2:4500 -> 55.1.1.2:64916
created: 208s ago
nat: me peer
auto-discovery: 2 receiver
IKE SA: created 1/1  established 1/1  time 20/20/20 ms
IPsec SA: created 1/1  established 1/1  time 10/10/10 ms

  id/spi: 48 d3fdd1bfbc94caee/16a1eb5b0f37ee23
  direction: initiator
  status: established 208-208s ago = 20ms
  proposal: aes128-sha256
  key: 9bcac400d8e14e11-fffde33eaa3a8263
  lifetime/rekey: 86400/85891
  DPD sent/recv: 0000000a/00000000
```

Other VPN topics

The following topics provide instructions on configuring other VPN topics.

- VPN and ASIC offload on page 1564
- Encryption algorithms on page 1574
- Fragmenting IP packets before IPsec encapsulation on page 1581
- Configure DSCP for IPsec tunnels on page 1582
- VXLAN over IPsec tunnel with virtual wire pair on page 1584
- VXLAN over IPsec using a VXLAN tunnel endpoint on page 1587
- Defining gateway IP addresses in IPsec with mode-config and DHCP on page 1592
- FQDN support for remote gateways on page 1594
- Windows IKEv2 native VPN with user certificate on page 1596

VPN and ASIC offload

This topic provides a brief introduction to VPN traffic offloading.

IPsec traffic processed by NPU

1. Check the device ASIC information. For example, a FortiGate 900D has an NP6 and a CP8.

```
# get hardware status
Model name: [[QualityAssurance62/FortiGate]]-900D
ASIC version: CP8
ASIC SRAM: 64M
CPU: Intel(R) Xeon(R) CPU E3-1225 v3 @ 3.20GHz
Number of CPUs: 4
RAM: 16065 MB
Compact Flash: 1925 MB /dev/sda
```

```
Hard disk: 244198 MB /dev/sdb
USB Flash: not available
Network Card chipset: [[QualityAssurance62/FortiASIC]] NP6 Adapter (rev.)
```

2. Check port to NPU mapping.

```
# diagnose npu np6 port-list
Chip   XAUI Ports            Max   Cross-chip
                             Speed offloading
----
np6_0  0
        1.    port17           1G    Yes
        1.    port18           1G    Yes
        1.    port19           1G    Yes
        1.    port20           1G    Yes
        1.    port21           1G    Yes
        1.    port22           1G    Yes
        1.    port23           1G    Yes
        1.    port24           1G    Yes
        1.    port27           1G    Yes
        1.    port28           1G    Yes
        1.    port25           1G    Yes
        1.    port26           1G    Yes
        1.    port31           1G    Yes
        1.    port32           1G    Yes
        1.    port29           1G    Yes
        1.    port30           1G    Yes
        1.    portB            10G   Yes
        1.
----
np6_1  0
        1.    port1            1G    Yes
        1.    port2            1G    Yes
        1.    port3            1G    Yes
        1.    port4            1G    Yes
        1.    port5            1G    Yes
        1.    port6            1G    Yes
        1.    port7            1G    Yes
        1.    port8            1G    Yes
        1.    port11           1G    Yes
        1.    port12           1G    Yes
        1.    port9            1G    Yes
        1.    port10           1G    Yes
        1.    port15           1G    Yes
        1.    port16           1G    Yes
        1.    port13           1G    Yes
        1.    port14           1G    Yes
        1.    portA            10G   Yes
        1.
----
```

3. Configure the option in IPsec phase1 settings to control NPU encrypt/decrypt IPsec packets (enabled by default).

```
config vpn ipsec phase1/phase1-interface
    edit "vpn_name"
        set npu-offload enable/disable
```

```
    next
end
```

4. Check NPU offloading. The NPU encrypted/decrypted counter should tick. The `npu_flag 03` flag means that the traffic processed by the NPU is bi-directional.

```
# diagnose vpn tunnel list
list all ipsec tunnel in vd 0
----
name=test ver=2 serial=1 173.1.1.1:0->11.101.1.1:0 tun_id=11.101.1.1
bound_if=42 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/8 options[0008]=npu
proxyid_num=1 child_num=0 refcnt=14 ilast=2 olast=2 ad=/0
stat: rxp=12231 txp=12617 rxb=1316052 txb=674314
dpd: mode=on-demand on=1 idle=20000ms retry=3 count=0 seqno=0
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=test proto=0 sa=1 ref=4 serial=7
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=6 options=10626 type=00 soft=0 mtu=1438 expire=42921/0B replaywin=2048
       seqno=802 esn=0 replaywin_lastseq=00000680 itn=0
  life: type=01 bytes=0/0 timeout=42930/43200
  dec: spi=e313ac46 esp=aes key=16 0dcb52642eed18b852b5c65a7dc62958
       ah=md5 key=16 c61d9fe60242b9a30e60b1d01da77660
  enc: spi=706ffe03 esp=aes key=16 6ad98c204fa70545dbf3d2e33fb7b529
       ah=md5 key=16 dcc3b866da155ef73c0aba15ec530e2e
  dec:pkts/bytes=1665/16352, enc:pkts/bytes=2051/16826
  npu_flag=03 npu_rgwy=11.101.1.1 npu_lgwy=173.1.1.1 npu_selid=6 dec_npuid=2 enc_npuid=2

FGT_900D # diagnose vpn ipsec st
All ipsec crypto devices in use:
NP6_0:
    Encryption (encrypted/decrypted)
        null             : 0                 1.
        des              : 0                 1.
        3des             : 0                 1.
        aes              : 0                 1.
        aes-gcm          : 0                 1.
        aria             : 0                 1.
        seed             : 0                 1.
        chacha20poly1305 : 0                 1.
    Integrity (generated/validated)
        null             : 0                 1.
        md5              : 0                 1.
        sha1             : 0                 1.
        sha256           : 0                 1.
        sha384           : 0                 1.
        sha512           : 0                 1.

NP6_1:
    Encryption (encrypted/decrypted)
        null             : 14976            15357
        des              : 0                 1.
        3des             : 0                 1.
        aes              : 1664             2047
        aes-gcm          : 0                 1.
        aria             : 0                 1.
        seed             : 0                 1.
```

```
            chacha20poly1305 : 0                 1.
        Integrity (generated/validated)
            null             : 0                 1.
            md5              : 1664             2047
            sha1             : 14976            15357
            sha256           : 0                 1.
            sha384           : 0                 1.
            sha512           : 0                 1.

    NPU Host Offloading:
        Encryption (encrypted/decrypted)
            null             : 3                 1.
            des              : 0                 1.
            3des             : 0                 1.
            aes              : 3                 1.
            aes-gcm          : 0                 1.
            aria             : 0                 1.
            seed             : 0                 1.
            chacha20poly1305 : 0                 1.
        Integrity (generated/validated)
            null             : 0                 1.
            md5              : 3                 1.
            sha1             : 3                 1.
            sha256           : 0                 1.
            sha384           : 0                 1.
            sha512           : 0                 1.

    CP8:
        Encryption (encrypted/decrypted)
            null             : 1                 1.
            des              : 0                 1.
            3des             : 0                 1.
            aes              : 1                 1.
            aes-gcm          : 0                 1.
            aria             : 0                 1.
            seed             : 0                 1.
            chacha20poly1305 : 0                 1.
        Integrity (generated/validated)
            null             : 0                 1.
            md5              : 1                 1.
            sha1             : 1                 1.
            sha256           : 0                 1.
            sha384           : 0                 1.
            sha512           : 0                 1.

    SOFTWARE:
        Encryption (encrypted/decrypted)
            null             : 0                 1.
            des              : 0                 1.
            3des             : 0                 1.
            aes              : 0                 1.
            aes-gcm          : 29882            29882
            aria             : 21688            21688
            seed             : 153774           153774
            chacha20poly1305 : 29521            29521
        Integrity (generated/validated)
```

```
        null             : 59403            59403
        md5              : 0                 1.
        sha1             : 175462           175462
        sha256           : 0                 1.
        sha384           : 0                 1.
        sha512           : 0                 1.
```

5. If traffic cannot be offloaded by the NPU, the CP will try to encrypt/decrypt the IPsec packets.

IPsec traffic processed by CP

1. Check the NPU flag and CP counter.

```
# diagnose vpn tunnel list
list all ipsec tunnel in vd 0
----
name=test ver=2 serial=1 173.1.1.1:0->11.101.1.1:0 tun_id=11.101.1.1
bound_if=42 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/0
proxyid_num=1 child_num=0 refcnt=13 ilast=0 olast=0 ad=/0
stat: rxp=8418 txp=8418 rxb=1251248 txb=685896
dpd: mode=on-demand on=1 idle=20000ms retry=3 count=0 seqno=0
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=test proto=0 sa=1 ref=3 serial=7
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=3 options=10226 type=00 soft=0 mtu=1438 expire=42037/0B replaywin=2048
       seqno=20e3 esn=0 replaywin_lastseq=000020e3 itn=0
  life: type=01 bytes=0/0 timeout=42928/43200
  dec: spi=e313ac48 esp=aes key=16 393770842f926266530db6e43e21c4f8
       ah=md5 key=16 b2e4e025e8910e95c1745e7855479cca
  enc: spi=706ffe05 esp=aes key=16 7ef749610335f9f50e252023926de29e
       ah=md5 key=16 0b81e4d835919ab2b8ba8edbd01aec9d
  dec:pkts/bytes=8418/685896, enc:pkts/bytes=8418/1251248
  npu_flag=00 npu_rgwy=11.101.1.1 npu_lgwy=173.1.1.1 npu_selid=6 dec_npuid=0 enc_npuid=0

FGT-D # diagnose vpn ipsec status
All ipsec crypto devices in use:
NP6_0:
    Encryption (encrypted/decrypted)
        null             : 0                 1.
        des              : 0                 1.
        3des             : 0                 1.
        aes              : 0                 1.
        aes-gcm          : 0                 1.
        aria             : 0                 1.
        seed             : 0                 1.
        chacha20poly1305 : 0                 1.
    Integrity (generated/validated)
        null             : 0                 1.
        md5              : 0                 1.
        sha1             : 0                 1.
        sha256           : 0                 1.
        sha384           : 0                 1.
        sha512           : 0                 1.

NP6_1:
    Encryption (encrypted/decrypted)
```

```
        null             : 14976           15357
        des              : 0                1.
        3des             : 0                1.
        aes              : 1664            2047
        aes-gcm          : 0                1.
        aria             : 0                1.
        seed             : 0                1.
        chacha20poly1305 : 0                1.
    Integrity (generated/validated)
        null             : 0                1.
        md5              : 1664            2047
        sha1             : 14976           15357
        sha256           : 0                1.
        sha384           : 0                1.
        sha512           : 0                1.

NPU Host Offloading:
    Encryption (encrypted/decrypted)
        null             : 3                1.
        des              : 0                1.
        3des             : 0                1.
        aes              : 3                1.
        aes-gcm          : 0                1.
        aria             : 0                1.
        seed             : 0                1.
        chacha20poly1305 : 0                1.
    Integrity (generated/validated)
        null             : 0                1.
        md5              : 3                1.
        sha1             : 3                1.
        sha256           : 0                1.
        sha384           : 0                1.
        sha512           : 0                1.

CP8:
    Encryption (encrypted/decrypted)
        null             : 1                1.
        des              : 0                1.
        3des             : 0                1.
        aes              : 8499            8499
        aes-gcm          : 0                1.
        aria             : 0                1.
        seed             : 0                1.
        chacha20poly1305 : 0                1.
    Integrity (generated/validated)
        null             : 0                1.
        md5              : 8499            8499
        sha1             : 1                1.
        sha256           : 0                1.
        sha384           : 0                1.
        sha512           : 0                1.

SOFTWARE:
    Encryption (encrypted/decrypted)
        null             : 0                1.
        des              : 0                1.
```

```
        3des              : 0                 1.
        aes               : 0                 1.
        aes-gcm           : 29882            29882
        aria              : 21688            21688
        seed              : 153774           153774
        chacha20poly1305 : 29521             29521
    Integrity (generated/validated)
        null              : 59403            59403
        md5               : 0                 1.
        sha1              : 175462           175462
        sha256            : 0                 1.
        sha384            : 0                 1.
        sha512            : 0                 1.
```

2. Two options are used to control if the CP processes packets. If disabled, packets are processed by the CPU.

```
config system global
    set ipsec-asic-offload disable
    set ipsec-hmac-offload disable
end
```

IPsec traffic processed by CPU

IPsec traffic might be processed by the CPU for the following reasons:

- Some low end models do not have NPUs.
- NPU offloading and CP IPsec traffic processing manually disabled.
- Some types of proposals - SEED, ARIA, chacha20poly1305 - are not supported by the NPU or CP.
- NPU flag set to 00 and software encrypt/decrypt counter ticked.

```
# diagnose vpn tunnel list
list all ipsec tunnel in vd 0
----
name=test ver=2 serial=1 173.1.1.1:0->11.101.1.1:0 tun_id=11.101.1.1
bound_if=42 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/0
proxyid_num=1 child_num=0 refcnt=14 ilast=0 olast=0 ad=/0
stat: rxp=12162 txp=12162 rxb=1691412 txb=1008216
dpd: mode=on-demand on=1 idle=20000ms retry=3 count=0 seqno=0
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=test proto=0 sa=1 ref=4 serial=8
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=3 options=10602 type=00 soft=0 mtu=1453 expire=42903/0B replaywin=2048
       seqno=2d70 esn=0 replaywin_lastseq=00002d70 itn=0
  life: type=01 bytes=0/0 timeout=42931/43200
  dec: spi=e313ac4d esp=chacha20poly1305 key=36
812d1178784c1130d1586606e44e1b9ab157e31a09edbed583be1e9cc82e8c9f2655a2cf
       ah=null key=0
  enc: spi=706ffe0a esp=chacha20poly1305 key=36
f2727e001e2243549b140f1614ae3df82243adb070e60c33911f461b389b05a7a642e11a
       ah=null key=0
  dec:pkts/bytes=11631/976356, enc:pkts/bytes=11631/1627692
  npu_flag=00 npu_rgwy=11.101.1.1 npu_lgwy=173.1.1.1 npu_selid=7 dec_npuid=0 enc_npuid=0

FGT_900D # diagnose vpn ipsec status
All ipsec crypto devices in use:
NP6_0:
```

```
    Encryption (encrypted/decrypted)
        null              : 0                 1.
        des               : 0                 1.
        3des              : 0                 1.
        aes               : 0                 1.
        aes-gcm           : 0                 1.
        aria              : 0                 1.
        seed              : 0                 1.
        chacha20poly1305 : 0                 1.
    Integrity (generated/validated)
        null              : 0                 1.
        md5               : 0                 1.
        sha1              : 0                 1.
        sha256            : 0                 1.
        sha384            : 0                 1.
        sha512            : 0                 1.

NP6_1:
    Encryption (encrypted/decrypted)
        null              : 14976            15357
        des               : 0                 1.
        3des              : 0                 1.
        aes               : 1664             2047
        aes-gcm           : 0                 1.
        aria              : 0                 1.
        seed              : 0                 1.
        chacha20poly1305 : 0                 1.
    Integrity (generated/validated)
        null              : 0                 1.
        md5               : 1664             2047
        sha1              : 14976            15357
        sha256            : 0                 1.
        sha384            : 0                 1.
        sha512            : 0                 1.

NPU Host Offloading:
    Encryption (encrypted/decrypted)
        null              : 3                 1.
        des               : 0                 1.
        3des              : 0                 1.
        aes               : 3                 1.
        aes-gcm           : 0                 1.
        aria              : 0                 1.
        seed              : 0                 1.
        chacha20poly1305 : 0                 1.
    Integrity (generated/validated)
        null              : 0                 1.
        md5               : 3                 1.
        sha1              : 3                 1.
        sha256            : 0                 1.
        sha384            : 0                 1.
        sha512            : 0                 1.

CP8:
    Encryption (encrypted/decrypted)
        null              : 1                 1.
```

```
        des               : 0                 1.
        3des              : 0                 1.
        aes               : 8865             8865
        aes-gcm           : 0                 1.
        aria              : 0                 1.
        seed              : 0                 1.
        chacha20poly1305 : 0                 1.
    Integrity (generated/validated)
        null              : 0                 1.
        md5               : 8865             8865
        sha1              : 1                 1.
        sha256            : 0                 1.
        sha384            : 0                 1.
        sha512            : 0                 1.

SOFTWARE:
    Encryption (encrypted/decrypted)
        null              : 0                 1.
        des               : 0                 1.
        3des              : 0                 1.
        aes               : 531              531
        aes-gcm           : 29882            29882
        aria              : 21688            21688
        seed              : 153774           153774
        chacha20poly1305 : 41156            41156
    Integrity (generated/validated)
        null              : 71038            71038
        md5               : 531              531
        sha1              : 175462           175462
        sha256            : 0                 1.
        sha384            : 0                 1.
        sha512            : 0                 1.
```

Disable automatic ASIC offloading

When `auto-asic-offload` is set to `disable` in the firewall policy, traffic is not offloaded and the NPU hosting counter is ticked.

```
# diagnose vpn ipsec status
All ipsec crypto devices in use:
NP6_0:
    Encryption (encrypted/decrypted)
        null              : 0                 1.
        des               : 0                 1.
        3des              : 0                 1.
        aes               : 0                 1.
        aes-gcm           : 0                 1.
        aria              : 0                 1.
        seed              : 0                 1.
        chacha20poly1305 : 0                 1.
    Integrity (generated/validated)
        null              : 0                 1.
        md5               : 0                 1.
        sha1              : 0                 1.
        sha256            : 0                 1.
        sha384            : 0                 1.
```

```
        sha512           : 0                  1.

    NP6_1:
        Encryption (encrypted/decrypted)
            null             : 14976            15357
            des              : 0                  1.
            3des             : 0                  1.
            aes              : 110080           2175
            aes-gcm          : 0                  1.
            aria             : 0                  1.
            seed             : 0                  1.
            chacha20poly1305 : 0                  1.
        Integrity (generated/validated)
            null             : 0                  1.
            md5              : 110080           2175
            sha1             : 14976            15357
            sha256           : 0                  1.
            sha384           : 0                  1.
            sha512           : 0                  1.

    NPU Host Offloading:
        Encryption (encrypted/decrypted)
            null             : 3                  1.
            des              : 0                  1.
            3des             : 0                  1.
            aes              : 111090             1.
            aes-gcm          : 0                  1.
            aria             : 0                  1.
            seed             : 0                  1.
            chacha20poly1305 : 0                  1.
        Integrity (generated/validated)
            null             : 0                  1.
            md5              : 111090             1.
            sha1             : 3                  1.
            sha256           : 0                  1.
            sha384           : 0                  1.
            sha512           : 0                  1.

    CP8:
        Encryption (encrypted/decrypted)
            null             : 1                  1.
            des              : 0                  1.
            3des             : 0                  1.
            aes              : 8865             8865
            aes-gcm          : 0                  1.
            aria             : 0                  1.
            seed             : 0                  1.
            chacha20poly1305 : 0                  1.
        Integrity (generated/validated)
            null             : 0                  1.
            md5              : 8865             8865
            sha1             : 1                  1.
            sha256           : 0                  1.
            sha384           : 0                  1.
            sha512           : 0                  1.
```

```
SOFTWARE:
    Encryption (encrypted/decrypted)
        null              : 0                  1.
        des               : 0                  1.
        3des              : 0                  1.
        aes               : 539               539
        aes-gcm           : 29882             29882
        aria              : 21688             21688
        seed              : 153774            153774
        chacha20poly1305 : 41259             41259
    Integrity (generated/validated)
        null              : 71141             71141
        md5               : 539               539
        sha1              : 175462            175462
        sha256            : 0                  1.
        sha384            : 0                  1.
        sha512            : 0                  1.
```

Encryption algorithms

This topic provides a brief introduction to IPsec phase 1 and phase 2 encryption algorithms and includes the following sections:

- IKEv1 phase 1 encryption algorithm
- IKEv1 phase 2 encryption algorithm
- IKEv2 phase 1 encryption algorithm
- IKEv2 phase 2 encryption algorithm
- HMAC settings

IKEv1 phase 1 encryption algorithm

The default encryption algorithm is:

`aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1`

DES is a symmetric-key algorithm, which means the same key is used for encrypting and decrypting data. FortiOS supports:

- des-md5
- des-sha1
- des-sha256
- des-sha384
- des-sha512

3DES applies the DES algorithm three times to each data. FortiOS supports:

- 3des-md5
- 3des-sha1
- 3des-sha256
- 3des-sha384
- 3des-sha512

AES is a symmetric-key algorithm with different key lengths (128, 192, and 256 bits). FortiOS supports:

- aes128-md5
- aes128-sha1
- aes128-sha256
- aes128-sha384
- aes128-sha512
- aes192-md5
- aes192-sha1
- aes192-sha256
- aes192-sha384
- aes192-sha512
- aes256-md5
- aes256-sha1
- aes256-sha256
- aes256-sha384
- aes256-sha512

The ARIA algorithm is based on AES with different key lengths (128, 192, and 256 bits). FortiOS supports:

- aria128-md5
- aria128-sha1
- aria128-sha256
- aria128-sha384
- aria128-sha512
- aria192-md5
- aria192-sha1
- aria192-sha256
- aria192-sha384
- aria192-sha512
- aria256-md5
- aria256-sha1
- aria256-sha256
- aria256-sha384
- aria256-sha512

SEED is a symmetric-key algorithm. FortiOS supports:

- seed128-md5
- seed128-sha1
- seed128-sha256
- seed128-sha384
- seed128-sha512

Suite-B is a set of AES encryption with ICV in GCM mode. FortiOS supports Suite-B on new kernel platforms only. IPsec traffic **cannot** offload to NPU. CP9 supports Suite-B offloading, otherwise packets are encrypted and decrypted by software. FortiOS supports:

- suite-b-gcm-128
- suite-b-gcm-256

IKEv1 phase 2 encryption algorithm

The default encryption algorithm is:

`aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm aes256gcm chacha20poly1305`

With null encryption, IPsec traffic can offload NPU/CP. FortiOS supports:

- null-md5
- null-sha1
- null-sha256
- null-sha384
- null-sha512

With the DES encryption algorithm, IPsec traffic can offload NPU/CP. FortiOS supports:

- des-null
- des-md5
- des-sha1
- des-sha256
- des-sha384
- des-sha512

With the 3DES encryption algorithm, IPsec traffic can offload NPU/CP. FortiOS supports:

- 3des-null
- 3des-md5
- 3des-sha1
- 3des-sha256
- 3des-sha384
- 3des-sha512

With the AES encryption algorithm, IPsec traffic can offload NPU/CP. FortiOS supports:

- aes128-null
- aes128-md5
- aes128-sha1
- aes128-sha256
- aes128-sha384
- aes128-sha512
- aes192-null
- aes192-md5
- aes192-sha1
- aes192-sha256
- aes192-sha384
- aes192-sha512
- aes256-null
- aes256-md5
- aes256-sha1
- aes256-sha256
- aes256-sha384
- aes256-sha512

With the AESGCM encryption algorithm, IPsec traffic **cannot** offload NPU/CP. FortiOS supports:

- aes128gcm
- aes256gcm

With the chacha20poly1305 encryption algorithm, IPsec traffic **cannot** offload NPU/CP. FortiOS supports:

- chacha20poly1305

With the ARIA encryption algorithm, IPsec traffic **cannot** offload NPU/CP. FortiOS supports:

- aria128-null
- aria128-md5
- aria128-sha1
- aria128-sha256
- aria128-sha384
- aria128-sha512
- aria192-null
- aria192-md5
- aria192-sha1
- aria192-sha256
- aria192-sha384
- aria192-sha512
- aria256-null
- aria256-md5
- aria256-sha1
- aria256-sha256
- aria256-sha384
- aria256-sha512

With the SEED encryption algorithm, IPsec traffic **cannot** offload NPU/CP. FortiOS supports:

- seed-null
- seed-md5
- seed-sha1
- seed-sha256
- seed-sha384
- seed-sha512

IKEv2 phase 1 encryption algorithm

The default encryption algorithm is:

```
aes128-sha256 aes256-sha256 aes128gcm-prfsha256 aes256gcm-prfsha384 chacha20poly1305-
     prfsha256
```

DES is a symmetric-key algorithm, which means the same key is used for encrypting and decrypting data. FortiOS supports:

- des-md5
- des-sha1
- des-sha256

- des-sha384
- des-sha512

3DES applies the DES algorithm three times to each data. FortiOS supports:

- 3des-md5
- 3des-sha1
- 3des-sha256
- 3des-sha384
- 3des-sha512

AES is a symmetric-key algorithm with different key lengths (128, 192, and 256 bits). FortiOS supports:

- aes128-md5
- aes128-sha1
- aes128-sha256
- aes128-sha384
- aes128-sha512
- aes128gcm-prfsha1
- aes128gcm-prfsha256
- aes128gcm-prfsha384
- aes128gcm-prfsha512
- aes192-md5
- aes192-sha1
- aes192-sha256
- aes192-sha384
- aes192-sha512
- aes256-md5
- aes256-sha1
- aes256-sha256
- aes256-sha384
- aes256-sha512
- aes256gcm-prfsha1
- aes256gcm-prfsha256
- aes256gcm-prfsha384
- aes256gcm-prfsha512

The ARIA algorithm is based on AES with different key lengths (128, 192, and 256 bits). FortiOS supports:

- aria128-md5
- aria128-sha1
- aria128-sha256
- aria128-sha384
- aria128-sha512
- aria192-md5
- aria192-sha1
- aria192-sha256
- aria192-sha384
- aria192-sha512

- aria256-md5
- aria256-sha1
- aria256-sha256
- aria256-sha384
- aria256-sha512

With the chacha20poly1305 encryption algorithm, FortiOS supports:

- chacha20poly1305-prfsha1
- chacha20poly1305-prfsha256
- chacha20poly1305-prfsha384
- chacha20poly1305-prfsha512

SEED is a symmetric-key algorithm. FortiOS supports:

- seed128-md5
- seed128-sha1
- seed128-sha256
- seed128-sha384
- seed128-sha512

Suite-B is a set of AES encryption with ICV in GCM mode. FortiOS supports Suite-B on new kernel platforms only. IPsec traffic **cannot** offload to NPU. CP9 supports Suite-B offloading, otherwise packets are encrypted and decrypted by software. FortiOS supports:

- suite-b-gcm-128
- suite-b-gcm-256

IKEv2 phase 2 encryption algorithm

The default encryption algorithm is:

```
aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm aes256gcm chacha20poly1305
```

With null encryption, IPsec traffic can offload NPU/CP. FortiOS supports:

- null-md5
- null-sha1
- null-sha256
- null-sha384
- null-sha512

With the DES encryption algorithm, IPsec traffic can offload NPU/CP. FortiOS supports:

- des-null
- des-md5
- des-sha1
- des-sha256
- des-sha384
- des-sha512

With the 3DES encryption algorithm, IPsec traffic can offload NPU/CP. FortiOS supports:

- 3des-null
- 3des-md5

- 3des-sha1
- 3des-sha256
- 3des-sha384
- 3des-sha512

With the AES encryption algorithm, IPsec traffic can offload NPU/CP. FortiOS supports:

- aes128-null
- aes128-md5
- aes128-sha1
- aes128-sha256
- aes128-sha384
- aes128-sha512
- aes192-null
- aes192-md5
- aes192-sha1
- aes192-sha256
- aes192-sha384
- aes192-sha512
- aes256-null
- aes256-md5
- aes256-sha1
- aes256-sha256
- aes256-sha384
- aes256-sha512

With the AESGCM encryption algorithm, IPsec traffic **cannot** offload NPU. CP9 supports AESGCM offloading. FortiOS supports:

- aes128gcm
- aes256gcm

With the chacha20poly1305 encryption algorithm, IPsec traffic **cannot** offload NPU/CP. FortiOS supports:

- chacha20poly1305

With the ARIA encryption algorithm, IPsec traffic **cannot** offload NPU/CP. FortiOS supports:

- aria128-null
- aria128-md5
- aria128-sha1
- aria128-sha256
- aria128-sha384
- aria128-sha512
- aria192-null
- aria192-md5
- aria192-sha1
- aria192-sha256
- aria192-sha384
- aria192-sha512
- aria256-null

- aria256-md5
- aria256-sha1
- aria256-sha256
- aria256-sha384
- aria256-sha512

With the SEED encryption algorithm, IPsec traffic **cannot** offload NPU/CP. FortiOS supports:

- seed-null
- seed-md5
- seed-sha1
- seed-sha256
- seed-sha384
- seed-sha512

HMAC settings

The FortiGate uses the HMAC based on the authentication proposal that is chosen in phase 1 or phase 2 of the IPsec configuration. Each proposal consists of the encryption-hash pair (such as 3des-sha256). The FortiGate matches the most secure proposal to negotiate with the peer.

To view the chosen proposal and the HMAC hash used:

```
# diagnose vpn ike gateway list

vd: root/0
name: MPLS
version: 1
interface: port1 3
addr: 192.168.2.5:500 -> 10.10.10.1:500
tun_id: 10.10.10.1
virtual-interface-addr: 172.31.0.2 -> 172.31.0.1
created: 1015820s ago
IKE SA: created 1/13 established 1/13 time 10/1626/21010 ms
IPsec SA: created 1/24 established 1/24 time 0/11/30 ms

  id/spi: 124 43b087dae99f7733/6a8473e58cd8990a
  direction: responder
  status: established 68693-68693s ago = 10ms
  proposal: 3des-sha256
  key: e0fa6ab8dc509b33-aa2cc549999b1823-c3cb9c337432646e
  lifetime/rekey: 86400/17436
  DPD sent/recv: 000001e1/00000000
```

Fragmenting IP packets before IPsec encapsulation

The `ip-fragmentation` command controls packet fragmentation before IPsec encapsulation, which can benefit packet loss in some environments.

The following options are available for the `ip-fragmentation` variable.

Option	Description
pre-encapsulation	Fragment before IPsec encapsulation.
post-encapsulation (default value)	Fragment after IPsec encapsulation (RFC compliant).

To configure packet fragmentation using the CLI:

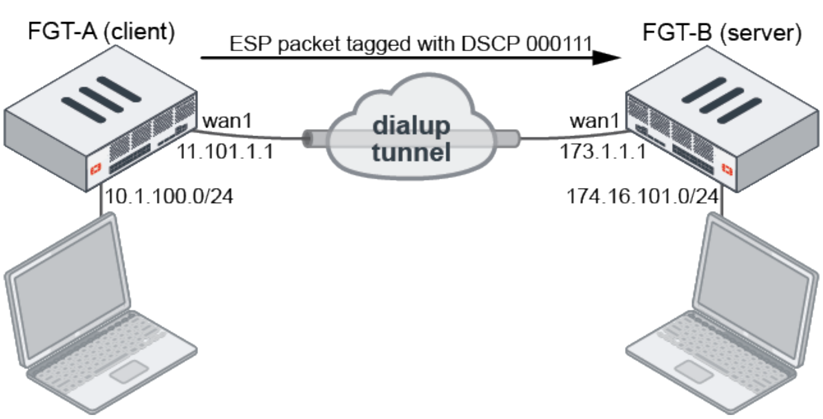
```
config vpn ipsec phase1-interface
   edit "demo"
      set interface "port1"
      set authmethod signature
      set peertype any
      set net-device enable
      set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
      set ip-fragmentation pre-encapsulation
      set remote-gw 172.16.200.4
      set certificate "Fortinet_Factory"
   next
end
```

Configure DSCP for IPsec tunnels

Configuring the differentiated services (DiffServ) code in phase2 of an IPsec tunnel allows the tag to be applied to the Encapsulating Security Payload (ESP) packet.

- If `diffserv` is disabled in the IPsec phase2 configuration, then the ESP packets' DSCP value is copied from the inner IP packet DSCP.
- If `diffserv` is enabled in the IPsec phase2 configuration, then ESP packets' DSCP value is set to the configured value.

Offloading traffic to the NPU must be disabled for the tunnel.



In this example, NPU offloading is disabled, diffserv is enabled, and the diffserv code is set to 000111 on FGT-A. Only one side of the tunnel needs to have diffserv enabled.

To configure IPsec on FGT-A:

1. Configure the phase1-interface:

```
config vpn ipsec phase1-interface
    edit "s2s"
        set interface "wan1"
        set peertype any
        set net-device disable
        set proposal aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1
        set npu-offload disable
        set dhgrp 14 5
        set wizard-type static-fortigate
        set remote-gw 173.1.1.1
        set psksecret ***********
    next
end
```

2. Configure the phase2-interface:

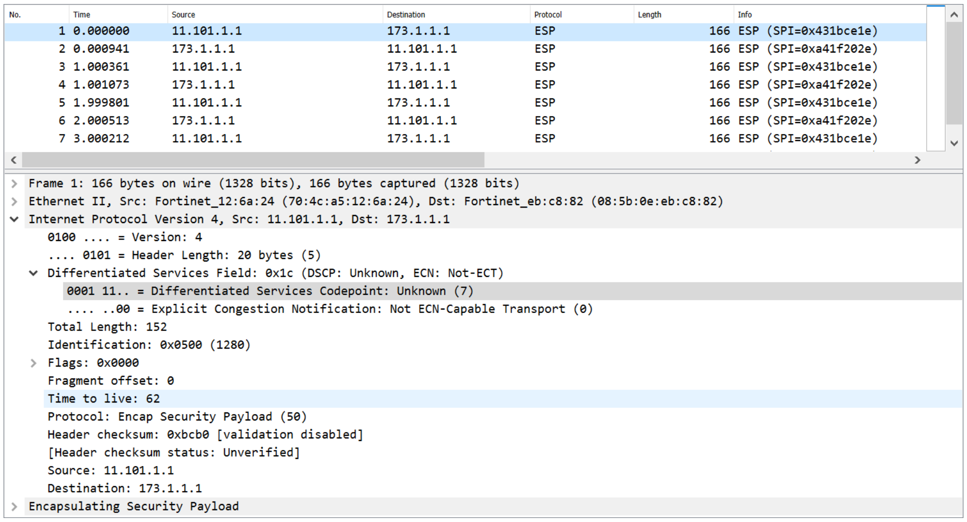
```
config vpn ipsec phase2-interface
    edit "s2s"
        set phase1name "s2s"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
aes256gcm chacha20poly1305
        set dhgrp 14 5
        set diffserv enable
        set diffservcode 000111
        set src-addr-type name
        set dst-addr-type name
        set src-name "s2s_local"
        set dst-name "s2s_remote"
    next
end
```

3. Check the state of the IPsec tunnel:

```
FGT-A # diagnose vpn tunnel list
list all ipsec tunnel in vd 0
------------------------------------------------------
name=s2s ver=1 serial=1 11.101.1.1:0->173.1.1.1:0 tun_id=173.1.1.1 dst_mtu=1500
bound_if=17 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/512 options[0200]=frag-rfc
run_state=0 accept_traffic=1 overlay_id=0

proxyid_num=1 child_num=0 refcnt=11 ilast=12 olast=2978 ad=/0
stat: rxp=4 txp=4 rxb=608 txb=336
dpd: mode=on-demand on=1 idle=20000ms retry=3 count=0 seqno=0
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=s2s proto=0 sa=1 ref=2 serial=2 dscp
  src: 0:10.1.100.0/255.255.255.0:0
  dst: 0:174.16.101.0/255.255.255.0:0
  SA:  ref=3 options=110226 type=00 soft=0 mtu=1438 expire=39916/0B replaywin=2048
       seqno=5 esn=0 replaywin_lastseq=00000005 itn=0 qat=0 hash_search_len=1
  life: type=01 bytes=0/0 timeout=42899/43200
  dec: spi=a41f202e esp=aes key=16 8a02875b80b884d961af227fe8b5cdee
       ah=sha1 key=20 fc9760b79e79dbbeef630ec0c5dca74777976208
  enc: spi=431bce1e esp=aes key=16 851117af24212da89e466d8bea9632bb
       ah=sha1 key=20 0807cc0af2dc4ea049a6b1a4af410ccc71e2156d
```

```
  dec:pkts/bytes=4/336, enc:pkts/bytes=4/608
  npu_flag=00 npu_rgwy=173.1.1.1 npu_lgwy=11.101.1.1 npu_selid=1 dec_npuid=0 enc_npuid=0
run_tally=1
```

4. Use a packet analyzer, or sniffer, to check the ESP packets:

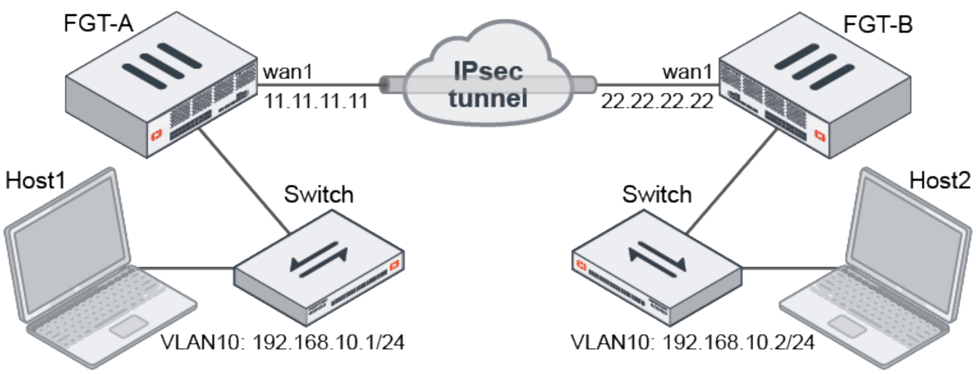


No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	11.101.1.1	173.1.1.1	ESP	166	ESP (SPI=0x431bce1e)
2	0.000941	173.1.1.1	11.101.1.1	ESP	166	ESP (SPI=0xa41f202e)
3	1.000361	11.101.1.1	173.1.1.1	ESP	166	ESP (SPI=0x431bce1e)
4	1.001073	173.1.1.1	11.101.1.1	ESP	166	ESP (SPI=0xa41f202e)
5	1.999801	11.101.1.1	173.1.1.1	ESP	166	ESP (SPI=0x431bce1e)
6	2.000513	173.1.1.1	11.101.1.1	ESP	166	ESP (SPI=0xa41f202e)
7	3.000212	11.101.1.1	173.1.1.1	ESP	166	ESP (SPI=0x431bce1e)

```
> Frame 1: 166 bytes on wire (1328 bits), 166 bytes captured (1328 bits)
> Ethernet II, Src: Fortinet_12:6a:24 (70:4c:a5:12:6a:24), Dst: Fortinet_eb:c8:82 (08:5b:0e:eb:c8:82)
v Internet Protocol Version 4, Src: 11.101.1.1, Dst: 173.1.1.1
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  v Differentiated Services Field: 0x1c (DSCP: Unknown, ECN: Not-ECT)
      0001 11.. = Differentiated Services Codepoint: Unknown (7)
      .... ..00 = Explicit Congestion Notification: Not ECN-Capable Transport (0)
    Total Length: 152
    Identification: 0x0500 (1280)
  > Flags: 0x0000
    Fragment offset: 0
    Time to live: 62
    Protocol: Encap Security Payload (50)
    Header checksum: 0xbcb0 [validation disabled]
    [Header checksum status: Unverified]
    Source: 11.101.1.1
    Destination: 173.1.1.1
> Encapsulating Security Payload
```

VXLAN over IPsec tunnel with virtual wire pair

In this example, a site-to-site VPN tunnel is formed between two FortiGates. Multiple VLANs are configured that match on each FortiGate. Host1 and Host2 are connected to VLAN10 on the switches.



To configure FGT-A in the CLI:

1. Configure the WAN interface:

```
config system interface
    edit "wan1"
        set vdom "root"
        set ip 11.11.11.11 255.255.255.0
        set allowaccess ping https ssh http fgfm
        set type physical
        set role wan
        set snmp-index 1
```

```
    next
end
```

2. Configure a static route to send all traffic out the WAN interface:

```
config router static
    edit 1
        set gateway 11.11.11.1
        set device "wan1"
    next
end
```

3. Configure the IPsec tunnel:

```
config vpn ipsec phase1-interface
    edit "ipsec"
        set interface "wan1"
        set peertype any
        set proposal aes256-sha1
        set remote-gw 22.22.22.22
        set psksecret **********
    next
end
config vpn ipsec phase2-interface
    edit "ipsec"
        set phase1name "ipsec"
        set proposal aes256-sha1
        set auto-negotiate enable
    next
end
```

4. Configure the VXLAN interface and bind it to the IPsec interface:

```
config system vxlan
    edit "vxlan"
        set interface "ipsec"
        set vni 10
        set remote-ip "22.22.22.22"
    next
end
```

The remote IP address is the peer side WAN IP address.

5. Configure a virtual wire pair with the LAN and VXLAN interfaces as members:

```
config system virtual-wire-pair
    edit "vwp"
        set member "port1" "vxlan"
        set wildcard-vlan enable
    next
end
```

The interfaces added to the virtual wire pair cannot be part of a switch, such as the default internal interface.

By enabling wildcard VLANs on the virtual wire pair, all VLAN tagged traffic that is allowed by the virtual wire pair firewall policies passes through the pair.

6. Configure a firewall policy to allow traffic between the LAN and VXLAN interfaces:

```
config firewall policy
    edit 4
```

```
        set name "vwp-pol"
        set srcintf "port1" "vxlan"
        set dstintf "port1" "vxlan"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
    next
end
```

To configure FGT-B in the CLI:

1. Configure the WAN interface:

```
config system interface
    edit "wan1"
        set vdom "root"
        set ip 22.22.22.22 255.255.255.0 255.255.255.0
        set allowaccess ping https ssh http fgfm
        set type physical
        set role wan
        set snmp-index 1
    next
end
```

2. Configure a static route to send all traffic out the WAN interface:

```
config router static
    edit 1
        set gateway 22.22.22.2
        set device "wan1"
    next
end
```

3. Configure the IPsec tunnel:

```
config vpn ipsec phase1-interface
    edit "ipsec"
        set interface "wan1"
        set peertype any
        set proposal aes256-sha1
        set remote-gw 11.11.11.11
        set psksecret **********
    next
end
config vpn ipsec phase2-interface
    edit "ipsec"
        set phase1name "ipsec"
        set proposal aes256-sha1
        set auto-negotiate enable
    next
end
```

4. Configure the VXLAN interface and bind it to the IPsec interface:

```
config system vxlan
    edit "vxlan"
```

```
        set interface "ipsec"
        set vni 10
        set remote-ip "11.11.11.11"
    next
end
```

The remote IP address is the peer side WAN IP address.

5. Configure a virtual wire pair with the LAN and VXLAN interfaces as members:

```
config system virtual-wire-pair
    edit "vwp"
        set member "port1" "vxlan"
        set wildcard-vlan enable
    next
end
```

6. Configure a firewall policy to allow traffic between the LAN and VXLAN interfaces:

```
config firewall policy
    edit 4
        set name "vwp-pol"
        set srcintf "port1" "vxlan"
        set dstintf "port1" "vxlan"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
    next
end
```

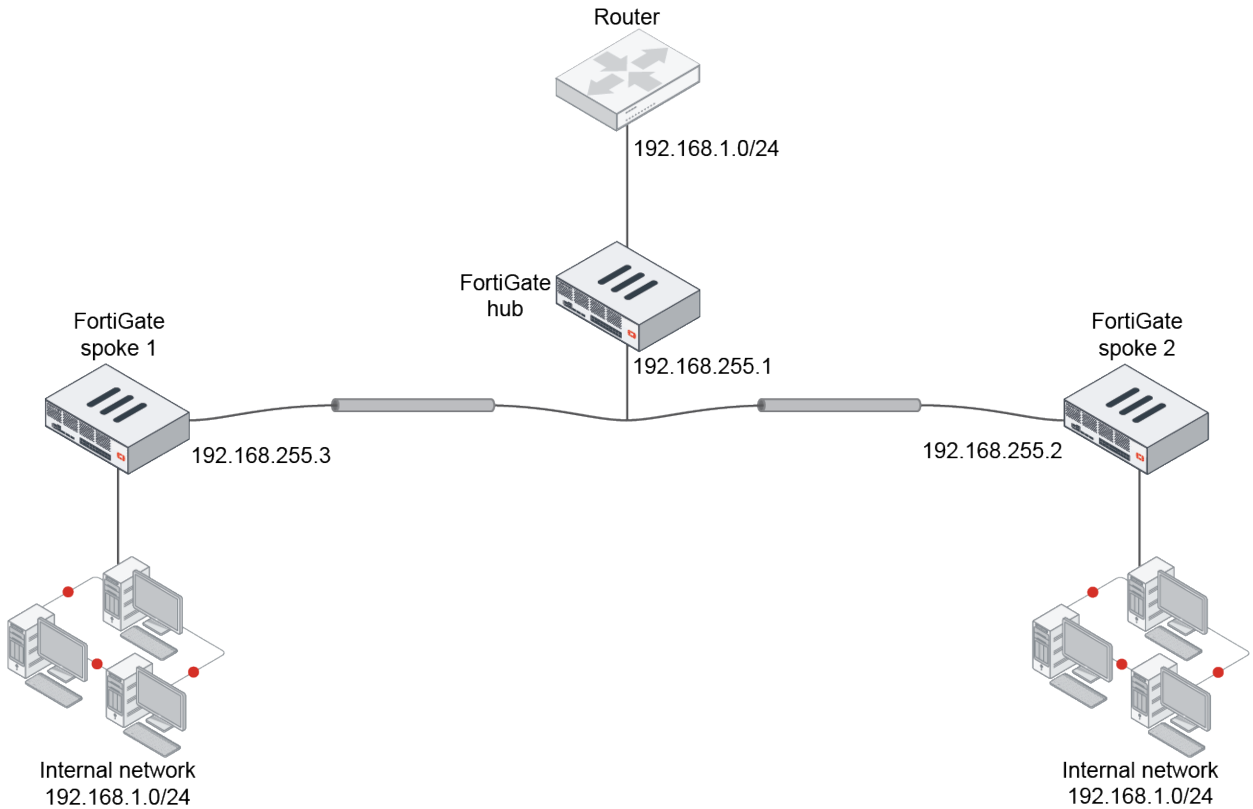
Test the configuration

To test the configuration, ping Host2 (VLAN10: 192.168.10.2/24) from Host1 (VLAN10: 192.168.10.1/24):

```
C:\>ping 192.168.10.2

Pinging 192.168.10.2 with 32 bytes of data:
Reply from 192.168.10.2: bytes=32 time=8ms TTL=56
Reply from 192.168.10.2: bytes=32 time=8ms TTL=56
Reply from 192.168.10.2: bytes=32 time=8ms TTL=56
Reply from 192.168.10.2: bytes=32 time=11ms TTL=56

Ping statistics for 192.168.10.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 8ms, Maximum = 11ms, Average = 8ms
```

VXLAN over IPsec using a VXLAN tunnel endpoint

This example describes how to implement VXLAN over IPsec VPN using a VXLAN tunnel endpoint (VTEP).



This example shows a specific configuration that uses a hub-and-spoke topology. However, the same logic can be applied to a static VPN with or without XAuth. In this hub-and-spoke topology, dialup VPN is convenient because it uses a single phase 1 dialup definition on the hub FortiGate. Additional spoke tunnels are added without any changes to the hub, other than adding a user account for each additional spoke. Spoke-to-spoke communication is established through the hub. This example assumes the authentication users and user groups have already been created.

IPsec tunnel interfaces are used to support VXLAN tunnel termination. An IP address is set for each tunnel interface. Ping access is allowed for troubleshooting purposes.

VTEPs are created on each of the hub and spokes in order to forward VXLAN traffic through the IPsec tunnels. VXLAN encapsulates OSI layer 2 Ethernet frames within layer 3 IP packets. You will need to either combine the internal port and VXLAN interface into a soft switch, or create a virtual wire pair so that devices behind port1 have direct layer 2 access to remote peers over the VXLAN tunnel. This example uses a switch interface on the hub and a virtual wire pair on the spokes to demonstrate the two different methods.

Finally, in order to apply an IPsec VPN interface on the VXLAN interface setting, `net-device` must be disabled in the IPsec VPN phase 1 settings. All VXLAN interfaces in this example share the same VXLAN network ID (`vni`).

To configure the hub FortiGate:

1. Configure the phase 1 and phase 2 interfaces:

```
config vpn ipsec phase1-interface
   edit "SPOKES"
      set type dynamic
      set interface "port2"
      set mode aggressive
      set peertype one
```

```
        set net-device disable
        set proposal aes256-sha256
        set xauthtype auto
        set authusrgrp "SPOKES"
        set peerid "SPOKES"
        set psksecret <secret>
    next
end
config vpn ipsec phase2-interface
    edit "SPOKES"
        set phase1name "SPOKES"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
aes256gcm chacha20poly1305
    next
end
```

2. Configure the IPsec VPN policy that allows VXLAN traffic between spokes:

```
config firewall policy
    edit 1
        set name "VXLAN_SPOKE_to_SPOKE"
        set srcintf "SPOKES"
        set dstintf "SPOKES"
        set srcaddr "NET_192.168.255.0"
        set dstaddr "NET_192.168.255.0"
        set action accept
        set schedule "always"
        set service "UDP_4789"
        set logtraffic all
        set fsso disable
    next
end
```

3. Configure the IPsec tunnel interfaces (the remote IP address is not used, but it is necessary for this configuration):

```
config system interface
    edit "SPOKES"
        set vdom "root"
        set ip 192.168.255.1 255.255.255.255
        set allowaccess ping
        set type tunnel
        set remote-ip 192.168.255.254 255.255.255.0
        set snmp-index 12
        set interface "port2"
    next
end
```

4. Configure the VXLAN interface (the remote IP is the tunnel interfaces IPs of the spokes):

```
config system VXLAN
    edit "SPOKES_VXLAN"
        set interface "SPOKES"
        set vni 1
        set remote-ip "192.168.255.2" "192.168.255.3"
    next
end
```

To configure the spoke FortiGates:

1. Configure the phase 1 and phase 2 interfaces:

```
config vpn ipsec phase1-interface
   edit "HUB"
      set interface "port2"
      set mode aggressive
      set peertype any
      set net-device disable
      set proposal aes256-sha256
      set localid "SPOKES"
      set xauthtype client
      set authusr "SPOKE1"
      set authpasswd <secret>
      set remote-gw <hub public IP>
      set psksecret <secret>
   next
end
config vpn ipsec phase2-interface
   edit "HUB"
      set phase1name "HUB"
      set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm
aes256gcm chacha20poly1305
      set auto-negotiate enable
      set src-subnet 192.168.255.2 255.255.255.255
   next
end
```

The hub FortiGate inserts a reverse route pointing to newly established tunnel interfaces for any of the subnets that the spoke FortiGate's source quick mode selectors provides. This is why you should set the tunnel IP address here.

2. Configure the IPsec VPN policy:

```
config firewall policy
   edit 1
      set name "VTEP_IPSEC_POLICY"
      set srcintf "HUB"
      set dstintf "HUB"
      set srcaddr "none"
      set dstaddr "none"
      set action accept
      set schedule "always"
      set service "PING"
      set logtraffic disable
      set fsso disable
   next
end
```

3. Configure the IPsec tunnel interfaces:

```
config system interface
   edit "HUB"
      set vdom "root"
      set ip 192.168.255.2 255.255.255.255
      set allowaccess ping
```

```
        set type tunnel
        set remote-ip 192.168.255.1 255.255.255.0
        set snmp-index 12
        set interface "port2"
    next
end
```

4. Configure the VXLAN interface (the remote IP is the tunnel interface IP of the hub):

```
config system VXLAN
    edit "HUB_VXLAN"
        set interface "HUB"
        set vni 1
        set remote-ip "192.168.255.1"
    next
end
```

To establish a VXLAN tunnel between spokes, you can add a spoke's tunnel IP address in `remote-ip`.

> To add more remote IP addresses to a VXLAN interface, the interface cannot be in use. You may want to provision future spokes' remote IP addresses at this point to avoid traffic disruption. Otherwise, you must delete the reference (the policy in this example) before adding remote IP addresses.

To bind the VXLAN interface to the internal interface:

1. Configure a switch interface on the hub:

```
config system switch-interface
    edit "SW"
        set vdom "root"
        set member "port1" "SPOKES_VXLAN"
        set intra-switch-policy {implicit | explicit}
    next
end
```

> Allowing intra-switch traffic is implicitly allowed by default. Use `set intra-switch-policy explicit` to require firewall policies to allow traffic between switch interfaces.

2. Configure a virtual wire pair on the spokes:

```
config system virtual-wire-pair
    edit "VWP"
        set member "HUB_VXLAN" "port1"
    next
end
```

> The virtual wire pair requires an explicit policy to allow traffic between interfaces.

To test the configuration:

1. Ping the hub FortiGate from the spoke FortiGate:

```
user@pc-spoke1:~$ ping 192.168.1.1 -c 3PING 192.168.1.1 (192.168.1.1) 56(84) bytes of
data.
64 bytes from 192.168.1.1: icmp_seq=1 ttl=64 time=1.24 ms
64 bytes from 192.168.1.1: icmp_seq=2 ttl=64 time=0.672 ms
64 bytes from 192.168.1.1: icmp_seq=3 ttl=64 time=0.855 ms
--- 192.168.1.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2002ms
rtt min/avg/max/mdev = 0.672/0.923/1.243/0.239 ms
```

2. Sniff traffic on the hub FortiGate:

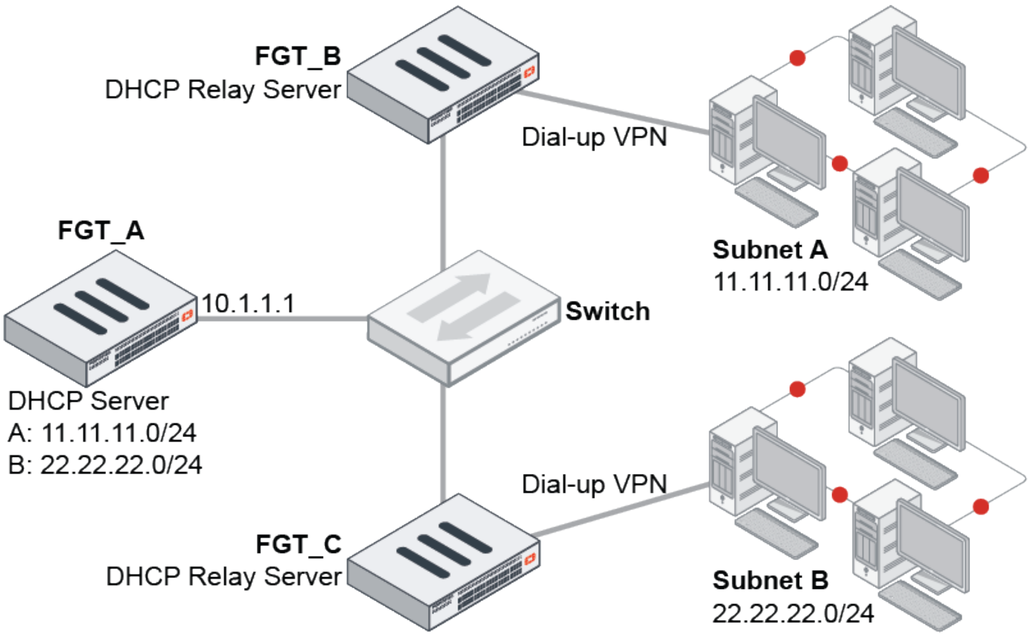
```
# diagnose sniffer packet any 'icmp or (udp and port 4789)' 4 0 ainterfaces=[any]
filters=[icmp or (udp and port 4789)]
15:00:01.438230 SPOKES in 192.168.255.2.4790 -&gt; 192.168.255.1.4789: udp 106
<<<<1
15:00:01.438256 SPOKES_VXLAN in 192.168.1.2 -&gt; 192.168.1.1: icmp: echo request
<<<<2
15:00:01.438260 port1 out 192.168.1.2 -&gt; 192.168.1.1: icmp: echo request
<<<<3
15:00:01.438532 port1 in 192.168.1.1 -&gt; 192.168.1.2: icmp: echo reply
15:00:01.438536 SPOKES_VXLAN out 192.168.1.1 -&gt; 192.168.1.2: icmp: echo reply
15:00:01.438546 SPOKES out 192.168.255.1.4851 -&gt; 192.168.255.2.4789: udp 106
```

Defining gateway IP addresses in IPsec with mode-config and DHCP

For an IPsec tunnel, the gateway IP address (giaddr) can be defined on a DHCP relay agent. Both IPv4 and IPv6 addresses are supported. An IPsec tunnel with mode-config and DHCP relay cannot specify a DHCP subnet range to the DHCP server.

The DHCP server assigns an IP address based on the giaddr set on the IPSec phase1 interface and sends an offer to this subnet. The DHCP server must have a route to the specified subnet giaddr.

Example



To define the gateway IP address on the DHCP relay server:

1. Configure the VPN IPsec phase1 interface:

```
config vpn ipsec phase1-interface
    edit "ipv4"
        set type dynamic
        set interface "port2"
        set peertype any
        set net-device disable
        set mode-cfg enable
        set proposal des-md5 des-sha1
        set dpd on-idle
        set dhgrp 5
        set assign-ip-from dhcp
        set dhcp-ra-giaddr 11.11.11.1
        set psksecret ***********
        set dpd-retryinterval 60
    next
end
```

IPv6 could also be configured:

```
config vpn ipsec phase1-interface
    edit "ipv6"
        set type dynamic
        set interface "port2"
        set peertype any
        set net-device disable
        set mode-cfg enable
        set proposal des-md5 des-sha1
        set dpd on-idle
```

```
            set dhgrp 5
            set assign-ip-from dhcp
            set dhcp6-ra-linkaddr 2000:11:11:11::1
            set psksecret **********
            set dpd-retryinterval 60
        next
    end
```

2. Enable DHCP proxy and configure the DHCP server IP address:

```
config system settings
    set dhcp-proxy enable
    set dhcp-server-ip "10.1.1.1"
end
```

3. Repeat the above steps for FGT_C and subnet B.

FQDN support for remote gateways

FortiGate supports FQDN when defining an IPsec remote gateway with a dynamically assigned IPv6 address. When FortiGate attempts to connect to the IPv6 device, FQDN will resolve the IPv6 address even when the address changes.

Using FQDN to configure the remote gateway is useful when the remote end has a dynamic IPv6 address assigned by their ISP or DHCPv6 server.

To set the VPN to DDNS and configure FQDN:

```
config vpn ipsec phase1-interface
    edit "ddns6"
        set type ddns
        set interface "agg1"
        set ip-version 6
        set ike-version 2
        set peertype any
        set net-device disable
        set proposal aes128-sha256 aes256-sha256 aes128gcm-prfsha256 aes256gcm-prfsha384 
chacha20poly1305-prfsha256
        set dpd on-idle
        set remotegw-ddns "rgwa61.vpnlab.org"
        set psksecret **********
    next
end
```

```
config vpn ipsec phase2-interface
    edit "ddns6"
        set phase1name "ddns6"
        set proposal aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm aes256gcm 
chacha20poly1305
        set src-addr-type subnet6
        set dst-addr-type subnet6
        set src-subnet6 2003:1:1:1::/64
    next
end
```

FQDN resolves the IPv6 address

```
# diagnose test application dnsproxy 7

vfid=0, name=rgwa61.vpnlab.org, ttl=3600:3547:1747
         2003:33:1:1::22 (ttl=3600)
```

FortiGate uses FQDN to connect to the IPv6 device

```
# diagnose vpn tunnel list name ddns6
list ipsec tunnel by names in vd 0
------------------------------------------------------
name=ddns6 ver=2 serial=2 2003:33:1:1::1:0->2003:33:1:1::22:0 dst_mtu=1500
bound_if=32 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/520 options[0208]=npu frag-rfc
run_state=0 accept_traffic=1 overlay_id=0

proxyid_num=1 child_num=0 refcnt=10 ilast=9 olast=9 ad=/0
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-idle on=1 idle=20000ms retry=3 count=0 seqno=72340
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=ddns6 proto=0 sa=1 ref=2 serial=1
  src: 0:2003:1:1:1::/64:0
  dst: 0:::/0:0
  SA:  ref=3 options=10226 type=00 soft=0 mtu=1422 expire=42680/0B replaywin=2048
       seqno=1 esn=0 replaywin_lastseq=00000000 itn=0 qat=0 hash_search_len=1
  life: type=01 bytes=0/0 timeout=42901/43200
  dec: spi=ac7a5718 esp=aes key=16 9976b66280cc49f500d8edca093e03fb
       ah=sha1 key=20 4d94d76fc18df5a180c52e0a6cd5f430fde48fe8
  enc: spi=7ab888ec esp=aes key=16 841a95d3ee5ea5108a2ba269b74998d1
       ah=sha1 key=20 ed0b52d27776e30149ee36af4fd4626681c2a3a1
  dec:pkts/bytes=0/0, enc:pkts/bytes=0/0
  npu_flag=00 npu_rgwy=2003:33:1:1::22 npu_lgwy=2003:33:1:1::1 npu_selid=0 dec_npuid=0 enc_
npuid=0
run_tally=1
```

The tunnel can still connect to the FQDN address when the IPv6 address changes

```
# diagnose debug application ike -1
# diagnose debug enable
ike 0:ddns6: set oper down
ike 0:ddns6: carrier down
ike shrank heap by 159744 bytes
ike 0: cache rebuild start
ike 0:ddns6: sending DNS request for remote peer rgwa61.vpnlab.org
ike 0: send IPv6 DNS query : rgwa61.vpnlab.org
ike 0: cache rebuild done
ike 0:ddns6: remote IPv6 DDNS gateway is empty, retry to resolve it
ike 0: DNS response received for remote gateway rgwa61.vpnlab.org
ike 0: DNS rgwa61.vpnlab.org -> 2003:33:1:1::33
ike 2:test:46932: could not send IKE Packet(P1_RETRANSMIT):50.1.1.1:500->50.1.1.2:500,
len=716: error 101:Network is unreachable
ike 0:ddns6: remote IPv6 DDNS gateway is empty, retry to resolve it
ike 0:ddns6: 'rgwa61.vpnlab.org' resolved to 2003:33:1:1::33
ike 0: cache rebuild start
ike 0:ddns6: local:2003:33:1:1::1, remote:2003:33:1:1::33
ike 0:ddns6: cached as static-ddns.
```

```
ike 0: cache rebuild done
ike 0:ddns6: auto-negotiate connection
ike 0:ddns6: created connection: 0x155aa510 32 2003:33:1:1::1->2003:33:1:1::33:500.

.............................................................................................
.........................
ike 0:ddns6:46933:ddn6:47779: add IPsec SA: SPIs=ac7a5719/7ab888ed
ike 0:ddns6:46933:ddn6:47779: IPsec SA dec spi ac7a5719 key
16:0F27F1D1D02496F90D15A30E2C032678 auth 20:46564E0E86A054374B31E58F95E4458340121BCE
ike 0:ddns6:46933:ddn6:47779: IPsec SA enc spi 7ab888ed key
16:926B12908EE670E1A5DDA6AD8E96607B auth 20:42BF438DC90867B837B0490EAB08E329AB62CBE3
ike 0:ddns6:46933:ddn6:47779: added IPsec SA: SPIs=ac7a5719/7ab888ed
ike 0:ddns6:46933:ddn6:47779: sending SNMP tunnel UP trap
ike 0:ddns6: carrier up
```

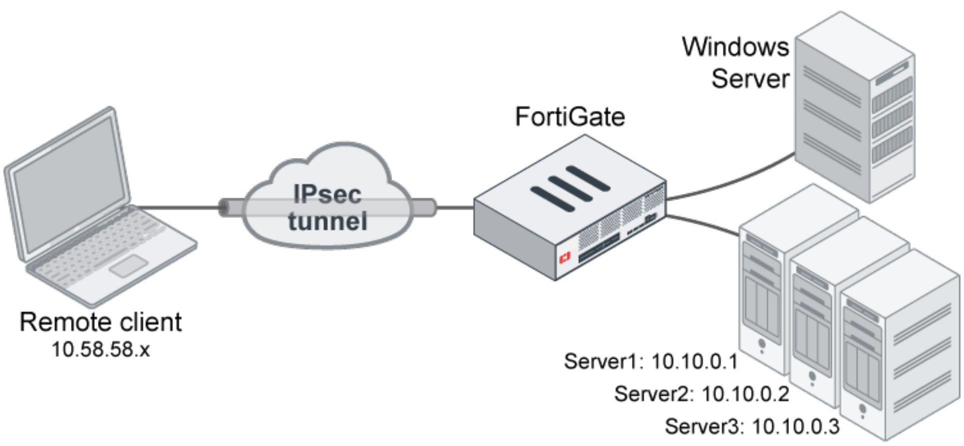
Windows IKEv2 native VPN with user certificate

In this example, IKEv2 with Extensible Authentication Protocol – Transport Layer Security (EAP-TLS) using mutual certificate authentication is configured. Mutual certificate authentication means that both the client and server use certificates to identify themselves. EAP uses RADIUS, which is handled by the Network Policy Server (NPS) on the Windows server. Certificates are generated and distributed through Active Directory Certificate Services (AD CS). An additional certificate is used to identify the IPsec gateway.

This example assumes that the following Windows server roles are installed and available:

- NPS (RADIUS)
- AD CS with a generated CA
- Group Policy Management
- DNS server

It is also assumed that a connection is established between the NPS and FortiGate, and a DNS entry exists for the NPS that the FortiGate can resolve.



Certificates

The following certificates are required:

- CA certificate for EAP-TLS to sign the client and server certificates.

  The CA certificate must be able to sign other certificates. It is created after AD CSs CA role installation. It is named lab-local-CA, as lab.local is the domain that is used in this example. The CA certificate is automatically installed on the server that is hosting the AD CS role. In this example, that server is also hosting the NPS and DNS server.

  The *Key Usage* specifies *Certificate Signing*.
- Client certificate for EAP-TLS used by the windows client.

  The client certificate is stored in the personal user certificate store and is used to authenticate the user. The certificate has *Client Authentication* and a SAN of the user's FQDN, and is signed by the CA. The CA is stored in *Current User > Trusted Root Certification Authorities*.
- Server certificate for EAP-TLS used by the server providing RADIUS authentication.

  The NPS certificate must be in the hosting server's certificate store so that the NPS can access it. It has *Server Authentication* and a SAN DNS name to match the server's IP address. The user must use the FQDN to connect to the VPN. If the IP address that the name resolves to is used, the certificate will not be considered valid.
- VPN certificate used to identify the FortiGate dialup gateway.

  The VPN certificate and private key are installed to the FortiGate using a CSR generated by the FortiGate

Configure the Windows server

The Windows server includes AD-CS, a RADIUS server, and a DNS server.

After the AD CS role has been installed and configured, the CA is ready to sign certificates.

Users and groups are defined first. The groups are configured to automatically receive certificates and relay membership to the FortiGate for granular access control through group matching in policies.

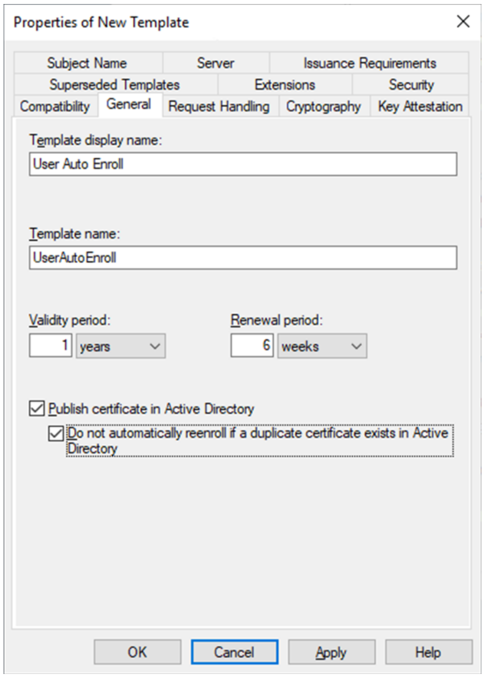
RADIUS is used to authorize connecting users. The RADIUS server returns users' groups with the access-accept response, to indicate to the FortiGate what groups the users belong to.

To create security groups and users:

1. Open *Active Directory Users and Computers*.
2. Create two groups, *Group1* and *Group2*.
3. Create two users, *User1* and *User2*.
4. Add *User1* to *Group1* and *User2* to *Group2*.

To create a certificate template to enable automatic enrollment for the user groups:

1. Open *Certificate Authority*.
2. In the navigation pane, expand the new CA, right-click *Certificate Template* and click *Manage*.
3. Configure a new certificate template:
   a. Right-click the *User* template and click *Duplicate Template*.
   b. On the *General* tab, enter a *Template display name*, such as *User Auto Enroll*.
   c. Enable *Publish certificate in Active Directory* and *Do not automatically reenroll...*.

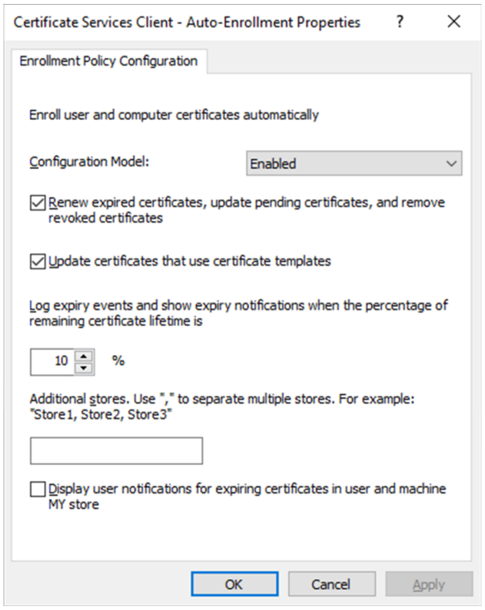


d. Configure the remaining settings as required, then go to the *Request Handling* tab.
e. Disable *Allow private key to be exported* and select *Enroll subject without requiring any user input*.
f. On the *Security* tab, in *Group or user name*, click *Add*.
g. Add *Group1* and *Group2*.
h. Select each group and, under *Permissions*, enable *Read*, *Enroll*, and *Autoenroll*.
i. On the *Extensions* tab, click *Application Policies* then click *Edit*.
j. Remove all of the policies expect for *Client Authentication*.
k. Click *OK* then close the *Certificate Templates* console.

4. In the navigation pane, right-click *Certificate Template* and click *New* > *Certificate Template to Issue*.
5. Select the new certificate template, *User Auto Enroll*, then click *OK*.

To create a group policy to enable automatic enrollment:

1. Open the *Group Policy Management* console.
2. In the navigation pane, go to *Forest:lab.local* > *Domains* > *lab.local*, and then click *Group Policy Objects*.
3. Click *Action*, and then click *New*.
4. Set a *Name* for the new GPO then click *OK*.
5. Right-click the new GPO and click *Edit*.
6. In the *Group Policy Management Editor* navigation pane, go to *User configuration* > *Windows Settings* > *Security Settings* > *Public Key Policies*.
7. In the content pane, double-click *Certificate Services Client - Auto-Enrollment*.
8. Set *Configuration Model* to *Enabled*.
9. Enable *Renew expired certificates...* and *Update certificates...*.



10. Click *OK*.

To verify that users are receiving certificates:

1. Log into an endpoint with a domain user.
2. On the server, open Certification Authority.
3. Expand the CA and select *Issued Certificates*.
4. Verify that the user logged into the endpoint is listed under *Requested Name*. You can also check the local user certificate store on the endpoint.

To generate and sign a CSR and import the signed certificate to the FortiGate:

1. On the FortiGate and go to *System* > *Certificates* and click *Create/Import* > *Generate CSR*.
2. Configure the CSR:

Certificate Name	vpn.lab.local
ID Type	Domain Name
Domain Name	vpn.lab.local
Subject Alternative Name	DNS:vpn.lab.local

3. Configure the remaining settings as required, then click *OK*.
4. Download the CSR to a location that is accessible to the CA server, in this example: C:\CSR\
5. Sign the CSR with the previously created CA:
   a. Open the command prompt as an administrator and enter the following:

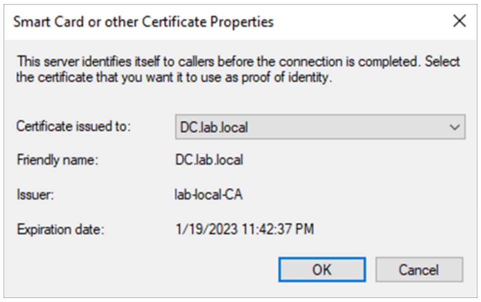
   ```
   certreq -submit -attrib "CertificateTemplate:WebServer" C:\CSR\vpn.lab.local.csr
   ```

   The *Certification Authority List* window opens.
   b. Select the CA and click *OK*.
   c. Save the signed certificate with a .cer file extension to a location that is accessible from the FortiGate.

6. Import the signed certificate to the FortiGate:
   a. On the FortiGate, go to *System* > *Certificates* and click *Create/Import* > *Certificate*.
   b. Click *Import Certificate*.
   c. Set *Type* to *Local Certificate*.
   d. Click *Upload* and locate and select the signed certificate
   e. Click *Create* then click *OK*.

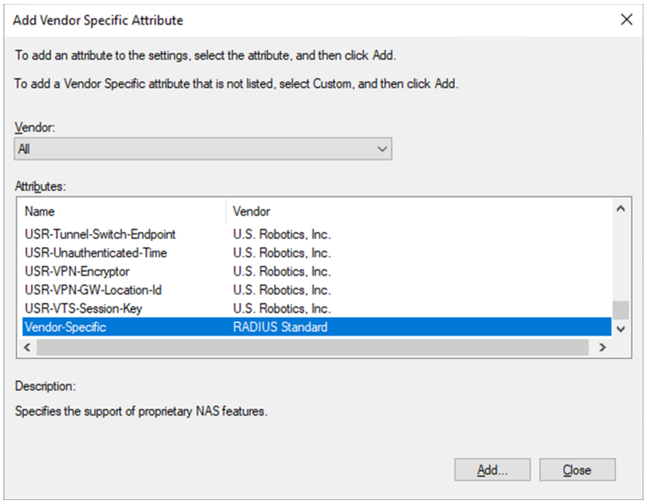
To configure network policies on the RADIUS server:

1. Open the *Network Policy Server* and, in the console tree, expand *Policies*.
2. Right-click on *Network Policies* and click *New*.
3. Enter a *Policy name*, such as *VPN-Group1*, then click *Next*.
4. Under *Condition description* click *Add*:
   a. Select *User Groups*, then click *Add*.
   b. Click *Add Groups*.
   c. Enter the group name, *Group1*, click *Check Names* to confirm the group.
   d. Click *OK* in both windows.
5. Click *Next*.
6. Make sure that *Access granted* is selected, then click *Next*.
7. On the *Configure Authentication Methods* page, click *Add* and add the EAP type *Microsoft: Smart Care or other certificate*.
8. Edit the EAP type, select the previously generated certificate, then click *OK*.



9. Deselect all of the *Less secure authentication methods* then click *Next*.
10. Configure constraints as needed, then click *Next*.
11. On the *Configure Settings* page, under *RADIUS Attributes*, select *Vendor Specific*, then click *Add*:

a. In the *Attributes* list, select *Vendor-Specific*, then click *Add*.

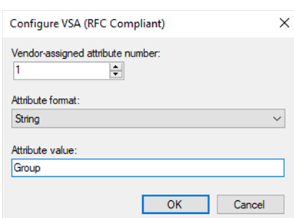


b. In the *Attribute Information* window, click *Add*.

c. In the *Vendor-Specific Attribute Information* window, enter the *Vendor Code*, *12356*, and select *Yes. It conforms*.

d. Click *Configure Attribute* and configure the following:

Vendor-assigned attribute number	1
Attribute format	String
Attribute value	Group



e. Click *OK* on all three windows and on the *Add Vendor Specific Attribute* window click *Close*.

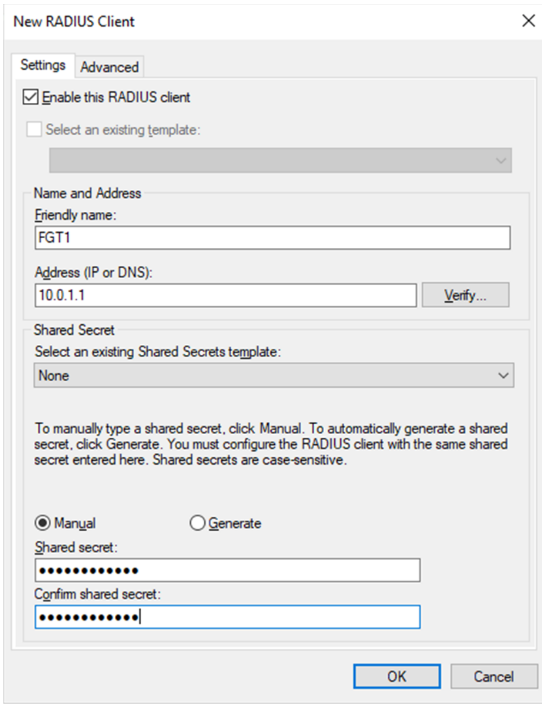
12. Click *Next*.
13. On the *Completing New Network Policy* page, review the configuration, then click *Finish*.
14. Duplicate the policy for *Group2*, and call the new policy *VPN-Group2*.
15. Reorder the policies so that *VPN-Group1* and *VPN-Group2* are one and two in the processing order.

To add the FortiGate as a RADIUS client:

1. Open the *Network Policy Server* and, in the console tree, expand *RADIUS Clients and Servers*.
2. Right-click on *RADIUS Clients* and click *New*.
3. Add the FortiGate as a RADIUS client:

Friendly name	FGT1

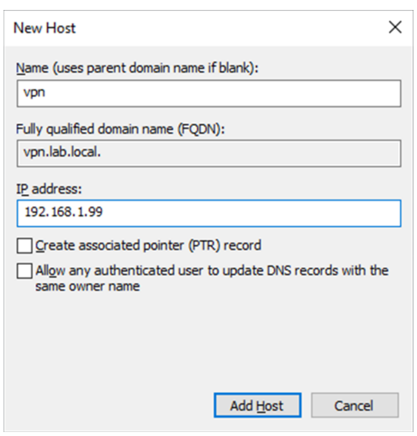
Address	10.0.1.1
Shared Secret	Manually enter the shared secret.



4. Click *OK*.

To create a DNS entry for the VPN connection:

1. Open the *DNS Manager*.
2. Go to *DC > Forward Lookup Zones* and select *lab.local*.
3. Right click in the content pane and select *New Host (A or AAAA)*.
4. Enter the VPN name. The FQDN should be auto-filled with *vpn.lab.local*.



5. Enter an IP address.
6. Click *Add Host*.

Configure the FortiGate

An IPsec VPN tunnel is configured to connect to the NPS (RADIUS) server for EAP authentication. For information about IPsec VPN, see IPsec VPNs on page 1328.

A RADIUS server is added to relay VPN authentication requests to the NPS server. For information about RADIUS servers, see RADIUS servers on page 1818.

Three groups are created that point to the RADIUS server for authentication: one group each for user group *Group1*, user group *Group2*, and the remote server. For information about groups, see User groups on page 1792.

Three firewall policies are created to test the functionality of the three user groups (see Policies on page 758):

- Policy 1 allows VPN clients to communicate with each other.
- Policy 2 allows VPN clients in the *Group1* user group to communicate with *Server1* and *Server3*.
- Policy 3 allows VPN clients in the *Group2* user group to communicate with *Server1* and *Server2*.

To configure IPsec VPN in the GUI:

1. Go to *VPN > IPsec Wizard*.
2. Enter a name for the VPN, such as *VPN1*.
3. Set *Template type* to *Custom*, then click *Next*.
4. In the *Network* section, configure the following:

Remote Gateway	Dialup User
Interface	port1
Mode Config	Enable
Assign IP From	Range
Client Address Range	10.58.58.1-10.58.58.10
DNS Server	192.168.1.100
Enable IPv4 Split Tunnel	Enable
Accessible Networks	Select the networks that VPN users will have access to.

5. In the *Authentication* section, configure the following:

Method	Signature
Certificate Name	vpn.lab.local
Version	2
Accept Types	Any peer ID

6. In the *Phase 1 Proposal* section, configure the following:

Encryption / Authentication	AES128 / SHA256
Encryption / Authentication	AES256 / SHA256
Encryption / Authentication	AES128 / SHA1

Diffie-Hellman Groups	14, 5, 2
Local ID	vpn.lab.local

7. In the *Phase 2 Selectors* section, configure the following:

Local Address	Named Address - all
Remote Address	Named Address - all
Encryption / Authentication	AES128 / SHA256
Encryption / Authentication	AES256 / SHA256
Encryption / Authentication	AES128 / SHA1
Enable Perfect Forward Secrecy (PFS)	Disable
Autokey Keep Alive	Enable

8. Enable EAP settings in the CLI:

```
config vpn ipsec phase1-interface
    edit VPN1
        set eap enable
        set eap-identity send-request
    next
end
```

To configure IPsec VPN in the CLI:

```
config vpn ipsec phase1-interface
    edit "VPN1"
        set type dynamic
        set interface "port1"
        set ike-version 2
        set authmethod signature
        set peertype any
        set net-device disable
        set mode-cfg enable
        set ipv4-dns-server1 192.168.1.100
        set proposal aes128-sha256 aes256-sha256 aes128-sha1
        set localid "vpn.lab.local"
        set dpd on-idle
        set dhgrp 14 5 2
        set eap enable
        set eap-identity send-request
        set certificate "vpn.lab.local"
        set ipv4-start-ip 10.58.58.1
        set ipv4-end-ip 10.58.58.10
        set ipv4-split-include "10/8_net"
        set dpd-retryinterval 60
    next
end
config vpn ipsec phase2-interface
    edit "VPN1"
        set phase1name "VPN1"
```

```
        set proposal aes128-sha256 aes256-sha256 aes128-sha1
        set pfs disable
        set keepalive enable
        set src-addr-type name
        set dst-addr-type name
        set src-name "all"
        set dst-name "all"
    next
end
```

To add the RADIUS server in the GUI:

1. Go to *User & Authentication > RADIUS Servers* and click *Create New*.
2. Enter a name for the server, such as *NPS*.
3. Enter the *Primary Server IP/Name* and *Secret*.
   The *Test User Credentials* option will not work, as it does not use certificates for the test.
4. Click *OK*.

To add the RADIUS server in the CLI:

```
config user radius
    edit "NPS"
        set server <ip>
        set secret **********
    next
end
```

To configure the user groups in the GUI:

1. Go to *User & Authentication > User Groups* and click *Create New*.
2. Enter a name for the group, such as *Group1*.
3. In the *Remote Groups* table, click *Add*:
   a. Set *Remote Server* to the just created RADIUS server, *NPS*.
   b. Set *Groups* to *Specify* and enter *Group1*.
   c. Click *OK*.
4. Click *OK*.
5. Create a second group called *Group2* with the same *Remote Server* and *Group Name* set to *Group2*.
6. Create a third group called RADIUS with the same *Remote Server* but no *Group Name*.

To configure the user groups in the CLI:

```
config user group
    edit "Group1"
        set member "NPS"
        config match
            edit 1
                set server-name "NPS"
                set group-name "Group1"
            next
        end
    next
```

```
        edit "Group2"
            set member "NPS"
            config match
                edit 1
                    set server-name "NPS"
                    set group-name "Group2"
                next
            end
        next
        edit "RADIUS"
            set member "NPS"
        next
    end
```

To configure the policies in the GUI:

1. Go to *Policy & Objects > Firewall Policy* and click *Create New*.
2. Configure policy 1:

Name	VPN-VPN
Incoming Interface	VPN1
Outgoing Interface	VPN1
Source	all, RADIUS
Destination	all
Schedule	always
Service	ALL
NAT	Disable

3. Click *OK*.
4. Click *Create New* again and configure policy 2:

Name	VPN Group1
Incoming Interface	VPN1
Outgoing Interface	Server1, Server3
Source	all, Group1
Destination	10.10.0.1, 10.10.0.3
Schedule	always
Service	ALL
NAT	Disable

5. Click *OK*.

6. Click *Create New* again and configure policy 3:

Name	VPN Group2
Incoming Interface	VPN1
Outgoing Interface	Server1, Server2
Source	all, Group2
Destination	10.10.0.1, 10.10.0.2
Schedule	always
Service	ALL
NAT	Disable

7. Click *OK*.

To configure the policies in the CLI:

```
config firewall policy
    edit 1
        set name "VPN-VPN"
        set srcintf "VPN1"
        set dstintf "VPN1"
        set action accept
        set srcaddr "all" "RADIUS"
        set dstaddr "all"
        set schedule "always"
        set service "ALL"
        set nat disable
    next
    edit 2
        set name "VPN Group1"
        set srcintf "VPN1"
        set dstintf "Server1" "Server3"
        set action accept
        set srcaddr "all" "Group1"
        set dstaddr "10.10.0.1" "10.10.0.3"
        set schedule "always"
        set service "ALL"
        set nat disable
    next
    edit 3
        set name "VPN Group2"
        set srcintf "VPN1"
        set dstintf "Server1" "Server2"
        set action accept
        set srcaddr "all" "Group2"
        set dstaddr "10.10.0.1" "10.10.0.2"
        set schedule "always"
        set service "ALL"
        set nat disable
    next
end
```

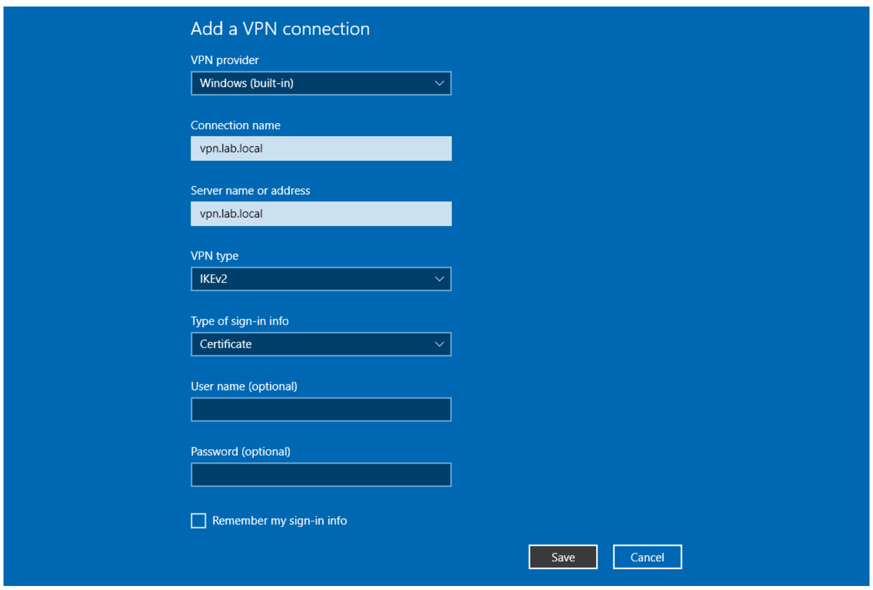
Configure the Windows client

The configuration is done on a Windows 10 Enterprise endpoint.

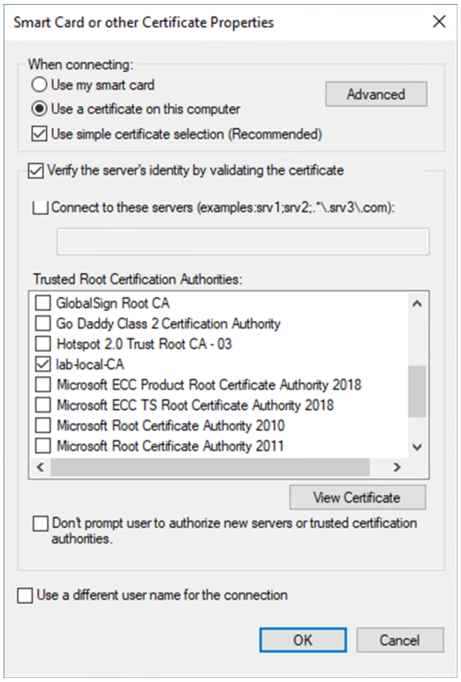
To add VPN connection and configure a VPN interface:

1. Open the *Settings* page and go to *Network & Internet* > *VPN*.
2. Click *Add a VPN connection*.
3. Configure the following:

VPN provider	Windows (built-in)
Connection name	vpn.lab.local
Server name or address	vpn.lab.local
VPN type	IKEv2
Type of sign-in info	Certificate



4. Click *Save*.
5. Go to *Network & Internet* > *Status* and, under *Advanced network settings*, click *Change adapter options*.
6. Select the VPN connection then click *Change settings of this connection*, or right-click on the connection and select *Properties*:
   a. Go to the *Settings* tab and, in the *Authentication* section, click *Properties*.
   b. Select *Use a certificate on this computer* and enable *Use simple certification selection*.
   c. Enable *Verify the server's identity by validating the certificate*.
   d. Optionally, enable *Connect to these servers* and enter your NPS server's FQDN, in this case *DC.lab.local*.
   e. In the *Trusted Root Certificate Authorities* list, select the CA *lab-local-CA*.



f. Click *OK*, then click *OK* again.

To test the connection:

1. Log in to the Windows endpoint as user1.
2. Open the network settings and connect to the *vpn.lab.local* VPN.
3. Ping each of the three servers to confirm that you can connect to server1 (10.10.0.1) and server3 (10.10.0.3), but not server2 (10.10.0.2).
4. Log out of the Windows endpoint, then log back in as user2.
5. Open the network settings and connect to the *vpn.lab.local* VPN.
6. Ping each of the three servers to confirm that you can connect to server1 (10.10.0.1) and server2 (10.10.0.2), but not server3 (10.10.0.3).

VPN IPsec troubleshooting

See the following IPsec troubleshooting examples:

- Understanding VPN related logs
- IPsec related diagnose commands on page 1611

Understanding VPN related logs

This section provides some IPsec log samples.

IPsec phase1 negotiating

```
logid="0101037127" type="event" subtype="vpn" level="notice" vd="root" eventtime=1544132571
logdesc="Progress IPsec phase 1" msg="progress IPsec phase 1" action="negotiate"
```

```
remip=11.101.1.1

    locip=173.1.1.1 remport=500 locport=500 outintf="port13"
    cookies="e41eeecb2c92b337/0000000000000000" user="N/A" group="N/A" xauthuser="N/A"
    xauthgroup="N/A" assignip=N/A vpntunnel="to_HQ" status="success" init="local"
    mode="aggressive" dir="outbound" stage=1 role="initiator" result="OK"
```

IPsec phase1 negotiated

```
logid="0101037127" type="event" subtype="vpn" level="notice" vd="root" eventtime=1544132571
logdesc="Progress IPsec phase 1" msg="progress IPsec phase 1" action="negotiate"
remip=11.101.1.1

    locip=173.1.1.1 remport=500 locport=500 outintf="port13"
    cookies="e41eeecb2c92b337/1230131a28eb4e73" user="N/A" group="N/A" xauthuser="N/A"
    xauthgroup="N/A" assignip=N/A vpntunnel="to_HQ" status="success" init="local"

mode="aggressive" dir="outbound" stage=2 role="initiator" result="DONE"
```

IPsec phase1 tunnel up

```
logid="0101037138" type="event" subtype="vpn" level="notice" vd="root" eventtime=1544132604
logdesc="IPsec connection status changed" msg="IPsec connection status change"
action="tunnel-up" remip=11.101.1.1 locip=173.1.1.1 remport=500 locport=500 outintf="port13"
cookies="5b1c59fab2029e43/bf517e686d3943d2" user="N/A" group="N/A" xauthuser="N/A"
xauthgroup="N/A" assignip=11.11.11.1 vpntunnel="to_HQ" tunnelip=N/A tunnelid=1530910918
tunneltype="ipsec" duration=0 sentbyte=0 rcvdbyte=0 nextstat=0
```

IPsec phase2 negotiate

```
logid="0101037129" type="event" subtype="vpn" level="notice" vd="root" eventtime=1544132604
logdesc="Progress IPsec phase 2" msg="progress IPsec phase 2" action="negotiate"
remip=11.101.1.1

locip=173.1.1.1 remport=500 locport=500 outintf="port13"
cookies="5b1c59fab2029e43/bf517e686d3943d2" user="N/A" group="N/A" xauthuser="N/A"
xauthgroup="N/A" assignip=11.11.11.1 vpntunnel="to_HQ" status="success" init="local"

mode="quick" dir="outbound" stage=1 role="initiator" result="OK"
```

IPsec phase2 tunnel up

```
logid="0101037139" type="event" subtype="vpn" level="notice" vd="root" eventtime=1544132604
logdesc="IPsec phase 2 status changed" msg="IPsec phase 2 status change" action="phase2-up"

    remip=11.101.1.1 locip=173.1.1.1 remport=500 locport=500 outintf="port13"
    cookies="5b1c59fab2029e43/bf517e686d3943d2" user="N/A" group="N/A" xauthuser="N/A"
    xauthgroup="N/A" assignip=11.11.11.1 vpntunnel="to_HQ"

phase2_name="to_HQ"
```

IPsec phase2 sa install

```
logid="0101037133" type="event" subtype="vpn" level="notice" vd="root" eventtime=1544132604
logdesc="IPsec SA installed" msg="install IPsec SA" action="install_sa" remip=11.101.1.1
locip=173.1.1.1
```

```
    remport=500 locport=500 outintf="port13" cookies="5b1c59fab2029e43/bf517e686d3943d2"
    user="N/A" group="N/A" xauthuser="N/A" xauthgroup="N/A" assignip=11.11.11.1
    vpntunnel="to_HQ" role="initiator" in_spi="ca646448" out_spi="747c10c6"
```

IPsec tunnel statistics

```
logid="0101037141" type="event" subtype="vpn" level="notice" vd="root" eventtime=1544131118
logdesc="IPsec tunnel statistics" msg="IPsec tunnel statistics" action="tunnel-stats"
remip=10.1.100.15 locip=172.16.200.4 remport=500 locport=500 outintf="mgmt1"
cookies="3539884dbd8f3567/c32e4c1beca91b36"
    user="N/A" group="N/A" xauthuser="N/A" xauthgroup="N/A" assignip=N/A
    vpntunnel="L2tpoIPsec_0" tunnelip=10.1.100.15 tunnelid=1530910802 tunneltype="ipsec"
    duration=6231 sentbyte=57343 rcvdbyte=142640 nextstat=60
```

IPsec phase2 tunnel down

```
logid="0101037138" type="event" subtype="vpn" level="notice" vd="root" eventtime=1544132571
logdesc="IPsec connection status changed" msg="IPsec connection status change"
action="tunnel-down" remip=11.101.1.1 locip=173.1.1.1 remport=500 locport=500
outintf="port13" cookies="30820aa390687e39/886e72bf5461fb8d" user="N/A" group="N/A"
xauthuser="N/A" xauthgroup="N/A" assignip=11.11.11.1 vpntunnel="to_HQ" tunnelip=N/A
tunnelid=1530910786 tunneltype="ipsec" duration=6425 sentbyte=504 rcvdbyte=152 nextstat=0
```

IPsec phase1 sa deleted

```
logid="0101037134" type="event" subtype="vpn" level="notice" vd="root" eventtime=1544132571
logdesc="IPsec phase 1 SA deleted" msg="delete IPsec phase 1 SA" action="delete_phase1_sa"
remip=11.101.1.1 locip=173.1.1.1 remport=500 locport=500 outintf="port13"
cookies="30820aa390687e39/886e72bf5461fb8d" user="N/A" group="N/A" xauthuser="N/A"
xauthgroup="N/A" assignip=11.11.11.1 vpntunnel="to_HQ"
```

IPsec related diagnose commands

This section provides IPsec related diagnose commands.

- Daemon IKE summary information list: `diagnose vpn ike status`

  ```
  connection: 2/50
  IKE SA: created 2/51  established 2/9  times 0/13/40 ms
  IPsec SA: created 1/13  established 1/7  times 0/8/30 ms
  ```

- IPsec phase1 interface status: `diagnose vpn ike gateway list`

  ```
  vd: root/0
  name: tofgtc
  version: 1
  interface: port13 42
  addr: 173.1.1.1:500 -> 172.16.200.3:500
  created: 4313s ago
  IKE SA: created 1/1  established 1/1  time 10/10/10 ms
  IPsec SA: created 0/0

    id/spi: 92 5639f7f8a5dc54c0/809a6c9bbd266a4b
    direction: initiator
    status: established 4313-4313s ago = 10ms
  ```

```
  proposal: aes128-sha256
  key: 74aa3d63d88e10ea-8a1c73b296b06578
  lifetime/rekey: 86400/81786
  DPD sent/recv: 00000000/00000000

vd: root/0
name: to_HQ
version: 1
interface: port13 42
addr: 173.1.1.1:500 -> 11.101.1.1:500
created: 1013s ago
assigned IPv4 address: 11.11.11.1/255.255.255.252
IKE SA: created 1/1  established 1/1  time 0/0/0 ms
IPsec SA: created 1/1  established 1/1  time 0/0/0 ms

  id/spi: 95 255791bd30c749f4/c2505db65210258b
  direction: initiator
  status: established 1013-1013s ago = 0ms
  proposal: aes128-sha256
  key: bb101b9127ed5844-1582fd614d5a8a33
  lifetime/rekey: 86400/85086
  DPD sent/recv: 00000000/00000010
```

- IPsec phase2 tunnel status: `diagnose vpn tunnel list`

```
list all ipsec tunnel in vd 0
----
nname=L2tpoIPsec ver=1 serial=6 172.16.200.4:0->0.0.0.0:0 tun_id=0.0.0.0
bound_if=4 lgwy=static/1 tun=intf/0 mode=dialup/2 encap=none/24 options[0018]=npu
create_dev
proxyid_num=0 child_num=0 refcnt=10 ilast=13544 olast=13544 ad=/0
stat: rxp=0 txp=0 rxb=0 txb=0
dpd: mode=on-idle on=0 idle=60000ms retry=3 count=0 seqno=0
natt: mode=none draft=0 interval=0 remote_port=0
run_tally=0
----
name=to_HQ ver=1 serial=7 173.1.1.1:0->11.101.1.1:0 tun_id=11.101.1.1
bound_if=42 lgwy=static/1 tun=intf/0 mode=auto/1 encap=none/8 options[0008]=npu
proxyid_num=1 child_num=0 refcnt=13 ilast=10 olast=1112 ad=/0
stat: rxp=1 txp=4 rxb=152 txb=336
dpd: mode=on-demand on=1 idle=20000ms retry=3 count=0 seqno=5
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=to_HQ proto=0 sa=1 ref=2 serial=1
  src: 0:0.0.0.0/0.0.0.0:0
  dst: 0:0.0.0.0/0.0.0.0:0
  SA:  ref=6 options=10226 type=00 soft=0 mtu=1438 expire=41773/0B replaywin=2048
       seqno=5 esn=0 replaywin_lastseq=00000002 itn=0
  life: type=01 bytes=0/0 timeout=42900/43200
  dec: spi=ca64644a esp=aes key=16 6cc873fdef91337a6cf9b6948972c90f
       ah=sha1 key=20 e576dbe3ff92605931e5670ad57763c50c7dc73a
  enc: spi=747c10c8 esp=aes key=16 5060ad8d0da6824204e3596c0bd762f4
       ah=sha1 key=20 52965cbd5b6ad95212fc825929d26c0401948abe
  dec:pkts/bytes=1/84, enc:pkts/bytes=4/608
  npu_flag=03 npu_rgwy=11.101.1.1 npu_lgwy=173.1.1.1 npu_selid=5 dec_npuid=2 enc_npuid=2
```

- Packets encrypted/decrypted counter: `diagnose vpn ipsec status`

```
All ipsec crypto devices in use:
NP6_0:
    Encryption (encrypted/decrypted)
        null             : 0                 1.
        des              : 0                 1.
        3des             : 0                 1.
        aes              : 0                 1.
        aes-gcm          : 0                 1.
        aria             : 0                 1.
        seed             : 0                 1.
        chacha20poly1305 : 0                 1.
    Integrity (generated/validated)
        null             : 0                 1.
        md5              : 0                 1.
        sha1             : 0                 1.
        sha256           : 0                 1.
        sha384           : 0                 1.
        sha512           : 0                 1.

NP6_1:
    Encryption (encrypted/decrypted)
        null             : 0                 1.
        des              : 0                 1.
        3des             : 0                 1.
        aes              : 337152           46069
        aes-gcm          : 0                 1.
        aria             : 0                 1.
        seed             : 0                 1.
        chacha20poly1305 : 0                 1.
    Integrity (generated/validated)
        null             : 0                 1.
        md5              : 0                 1.
        sha1             : 337152           46069
        sha256           : 0                 1.
        sha384           : 0                 1.
        sha512           : 0                 1.

NPU Host Offloading:
    Encryption (encrypted/decrypted)
        null             : 0                 1.
        des              : 0                 1.
        3des             : 0                 1.
        aes              : 38                1.
        aes-gcm          : 0                 1.
        aria             : 0                 1.
        seed             : 0                 1.
        chacha20poly1305 : 0                 1.
    Integrity (generated/validated)
        null             : 0                 1.
        md5              : 0                 1.
        sha1             : 38                1.
        sha256           : 0                 1.
        sha384           : 0                 1.
        sha512           : 0                 1.
```

```
CP8:
    Encryption (encrypted/decrypted)
        null             : 0                 1.
        des              : 0                 1.
        3des             : 1337             1582
        aes              : 71               11426
        aes-gcm          : 0                 1.
        aria             : 0                 1.
        seed             : 0                 1.
        chacha20poly1305 : 0                 1.
    Integrity (generated/validated)
        null             : 0                 1.
        md5              : 48               28
        sha1             : 1360             12980
        sha256           : 0                 1.
        sha384           : 0                 1.
        sha512           : 0                 1.

SOFTWARE:
    Encryption (encrypted/decrypted)
        null             : 0                 1.
        des              : 0                 1.
        3des             : 0                 1.
        aes              : 0                 1.
        aes-gcm          : 0                 1.
        aria             : 0                 1.
        seed             : 0                 1.
        chacha20poly1305 : 0                 1.
    Integrity (generated/validated)
        null             : 0                 1.
        md5              : 0                 1.
        sha1             : 0                 1.
        sha256           : 0                 1.
        sha384           : 0                 1.
        sha512           : 0                 1.
```

- `diagnose debug application ike -1`
  - `diagnose vpn ike log-filter dst-addr4 11.101.1.1`
  - `diagnose vpn ike log-filter src-addr4 173.1.1.1`

```
# ike 0:to_HQ:101: initiator: aggressive mode is sending 1st message...
ike 0:to_HQ:101: cookie dff03f1d4820222a/0000000000000000
ike 0:to_HQ:101: sent IKE msg (agg_i1send): 173.1.1.1:500->11.101.1.1:500, len=912,
id=dff03f1d4820222a/0000000000000000
ike 0: comes 11.101.1.1:500->173.1.1.1:500,ifindex=42....
ike 0: IKEv1 exchange=Aggressive id=dff03f1d4820222a/6c2caf4dcf5bab75 len=624
ike 0:to_HQ:101: initiator: aggressive mode get 1st response...
ike 0:to_HQ:101: VID RFC 3947 4A131C81070358455C5728F20E95452F
ike 0:to_HQ:101: VID DPD AFCAD71368A1F1C96B8696FC77570100
ike 0:to_HQ:101: DPD negotiated
ike 0:to_HQ:101: VID draft-ietf-ipsra-isakmp-xauth-06.txt 09002689DFD6B712
ike 0:to_HQ:101: VID CISCO-UNITY 12F5F28C457168A9702D9FE274CC0204
ike 0:to_HQ:101: peer supports UNITY
ike 0:to_HQ:101: VID FORTIGATE 8299031757A36082C6A621DE00000000
ike 0:to_HQ:101: peer is [[QualityAssurance62/FortiGate]]/FortiOS (v0 b0)
ike 0:to_HQ:101: VID FRAGMENTATION 4048B7D56EBCE88525E7DE7F00D6C2D3
```

```
ike 0:to_HQ:101: VID FRAGMENTATION 4048B7D56EBCE88525E7DE7F00D6C2D3C0000000
ike 0:to_HQ:101: peer identifier IPV4_ADDR 11.101.1.1
ike 0:to_HQ:101: negotiation result
ike 0:to_HQ:101: proposal id = 1:
ike 0:to_HQ:101:   protocol id = ISAKMP:
ike 0:to_HQ:101:      trans_id = KEY_IKE.
ike 0:to_HQ:101:      encapsulation = IKE/none
ike 0:to_HQ:101:         type=OAKLEY_ENCRYPT_ALG, val=AES_CBC, key-len=128
ike 0:to_HQ:101:         type=OAKLEY_HASH_ALG, val=SHA2_256.
ike 0:to_HQ:101:         type=AUTH_METHOD, val=PRESHARED_KEY_XAUTH_I.
ike 0:to_HQ:101:         type=OAKLEY_GROUP, val=MODP2048.
ike 0:to_HQ:101: ISAKMP SA lifetime=86400
ike 0:to_HQ:101: received NAT-D payload type 20
ike 0:to_HQ:101: received NAT-D payload type 20
ike 0:to_HQ:101: selected NAT-T version: RFC 3947
ike 0:to_HQ:101: NAT not detected
ike 0:to_HQ:101: ISAKMP SA dff03f1d4820222a/6c2caf4dcf5bab75 key
16:D81CAE6B2500435BFF195491E80148F3
ike 0:to_HQ:101: PSK authentication succeeded
ike 0:to_HQ:101: authentication OK
ike 0:to_HQ:101: add INITIAL-CONTACT
ike 0:to_HQ:101: sent IKE msg (agg_i2send): 173.1.1.1:500->11.101.1.1:500, len=172,
id=dff03f1d4820222a/6c2caf4dcf5bab75
ike 0:to_HQ:101: established IKE SA dff03f1d4820222a/6c2caf4dcf5bab75
ike 0: comes 11.101.1.1:500->173.1.1.1:500,ifindex=42....
ike 0: IKEv1 exchange=Mode config id=dff03f1d4820222a/6c2caf4dcf5bab75:97d88fb4 len=92
ike 0:to_HQ:101: mode-cfg type 16521 request 0:
ike 0:to_HQ:101: mode-cfg type 16522 request 0:
ike 0:to_HQ:101: sent IKE msg (cfg_send): 173.1.1.1:500->11.101.1.1:500, len=108,
id=dff03f1d4820222a/6c2caf4dcf5bab75:97d88fb4
ike 0: comes 11.101.1.1:500->173.1.1.1:500,ifindex=42....
ike 0: IKEv1 exchange=Mode config id=dff03f1d4820222a/6c2caf4dcf5bab75:3724f295 len=92
ike 0:to_HQ:101: sent IKE msg (cfg_send): 173.1.1.1:500->11.101.1.1:500, len=92,
id=dff03f1d4820222a/6c2caf4dcf5bab75:3724f295
ike 0:to_HQ:101: initiating mode-cfg pull from peer
ike 0:to_HQ:101: mode-cfg request APPLICATION_VERSION
ike 0:to_HQ:101: mode-cfg request INTERNAL_IP4_ADDRESS
ike 0:to_HQ:101: mode-cfg request INTERNAL_IP4_NETMASK
ike 0:to_HQ:101: mode-cfg request UNITY_SPLIT_INCLUDE
ike 0:to_HQ:101: mode-cfg request UNITY_PFS
ike 0:to_HQ:101: sent IKE msg (cfg_send): 173.1.1.1:500->11.101.1.1:500, len=140,
id=dff03f1d4820222a/6c2caf4dcf5bab75:3bca961f
ike 0: comes 11.101.1.1:500->173.1.1.1:500,ifindex=42....
ike 0: IKEv1 exchange=Mode config id=dff03f1d4820222a/6c2caf4dcf5bab75:3bca961f len=172
ike 0:to_HQ:101: mode-cfg type 1 response 4:0B0B0B01
ike 0:to_HQ:101: mode-cfg received INTERNAL_IP4_ADDRESS 11.11.11.1
ike 0:to_HQ:101: mode-cfg type 2 response 4:FFFFFFFC
ike 0:to_HQ:101: mode-cfg received INTERNAL_IP4_NETMASK 255.255.255.252
ike 0:to_HQ:101: mode-cfg received UNITY_PFS 1
ike 0:to_HQ:101: mode-cfg type 28676 response
28:0A016400FFFFFF0000000000000000A016500FFFFFF00000000000000
ike 0:to_HQ:101: mode-cfg received UNITY_SPLIT_INCLUDE 0 10.1.100.0/255.255.255.0:0
local port 0
ike 0:to_HQ:101: mode-cfg received UNITY_SPLIT_INCLUDE 0 10.1.101.0/255.255.255.0:0
local port 0
```

```
ike 0:to_HQ:101: mode-cfg received APPLICATION_VERSION 'FortiGate-100D
v6.0.3,build0200,181009 (GA)'
ike 0:to_HQ: mode-cfg add 11.11.11.1/255.255.255.252 to 'to_HQ'/58
ike 0:to_HQ: set oper up
ike 0:to_HQ: schedule auto-negotiate
ike 0:to_HQ:101: no pending Quick-Mode negotiations
ike shrank heap by 159744 bytes
ike 0:to_HQ:to_HQ: IPsec SA connect 42 173.1.1.1->11.101.1.1:0
ike 0:to_HQ:to_HQ: using existing connection
```

```
# ike 0:to_HQ:to_HQ: config found
ike 0:to_HQ:to_HQ: IPsec SA connect 42 173.1.1.1->11.101.1.1:500 negotiating
ike 0:to_HQ:101: cookie dff03f1d4820222a/6c2caf4dcf5bab75:32f4cc01
ike 0:to_HQ:101:to_HQ:259: initiator selectors 0 0:0.0.0.0/0.0.0.0:0:0-
>0:0.0.0.0/0.0.0.0:0:0
ike 0:to_HQ:101: sent IKE msg (quick_i1send): 173.1.1.1:500->11.101.1.1:500, len=620,
id=dff03f1d4820222a/6c2caf4dcf5bab75:32f4cc01
ike 0: comes 11.101.1.1:500->173.1.1.1:500,ifindex=42....
ike 0: IKEv1 exchange=Quick id=dff03f1d4820222a/6c2caf4dcf5bab75:32f4cc01 len=444
ike 0:to_HQ:101:to_HQ:259: responder selectors 0:0.0.0.0/0.0.0.0:0->0:0.0.0.0/0.0.0.0:0
ike 0:to_HQ:101:to_HQ:259: my proposal:
ike 0:to_HQ:101:to_HQ:259: proposal id = 1:
ike 0:to_HQ:101:to_HQ:259:   protocol id = IPSEC_ESP:
ike 0:to_HQ:101:to_HQ:259:   PFS DH group = 14
ike 0:to_HQ:101:to_HQ:259:      trans_id = ESP_AES_CBC (key_len = 128)
ike 0:to_HQ:101:to_HQ:259:      encapsulation = ENCAPSULATION_MODE_TUNNEL
ike 0:to_HQ:101:to_HQ:259:         type = AUTH_ALG, val=SHA1
ike 0:to_HQ:101:to_HQ:259:      trans_id = ESP_AES_CBC (key_len = 256)
ike 0:to_HQ:101:to_HQ:259:      encapsulation = ENCAPSULATION_MODE_TUNNEL
ike 0:to_HQ:101:to_HQ:259:         type = AUTH_ALG, val=SHA1
ike 0:to_HQ:101:to_HQ:259:      trans_id = ESP_AES_CBC (key_len = 128)
ike 0:to_HQ:101:to_HQ:259:      encapsulation = ENCAPSULATION_MODE_TUNNEL
ike 0:to_HQ:101:to_HQ:259:         type = AUTH_ALG, val=SHA2_256
ike 0:to_HQ:101:to_HQ:259:      trans_id = ESP_AES_CBC (key_len = 256)
ike 0:to_HQ:101:to_HQ:259:      encapsulation = ENCAPSULATION_MODE_TUNNEL
ike 0:to_HQ:101:to_HQ:259:         type = AUTH_ALG, val=SHA2_256
ike 0:to_HQ:101:to_HQ:259:      trans_id = ESP_AES_GCM_16 (key_len = 128)
ike 0:to_HQ:101:to_HQ:259:      encapsulation = ENCAPSULATION_MODE_TUNNEL
ike 0:to_HQ:101:to_HQ:259:         type = AUTH_ALG, val=NULL
ike 0:to_HQ:101:to_HQ:259:      trans_id = ESP_AES_GCM_16 (key_len = 256)
ike 0:to_HQ:101:to_HQ:259:      encapsulation = ENCAPSULATION_MODE_TUNNEL
ike 0:to_HQ:101:to_HQ:259:         type = AUTH_ALG, val=NULL
ike 0:to_HQ:101:to_HQ:259:      trans_id = ESP_CHACHA20_POLY1305 (key_len = 256)
ike 0:to_HQ:101:to_HQ:259:      encapsulation = ENCAPSULATION_MODE_TUNNEL
ike 0:to_HQ:101:to_HQ:259:         type = AUTH_ALG, val=NULL
ike 0:to_HQ:101:to_HQ:259: incoming proposal:
ike 0:to_HQ:101:to_HQ:259: proposal id = 1:
ike 0:to_HQ:101:to_HQ:259:   protocol id = IPSEC_ESP:
ike 0:to_HQ:101:to_HQ:259:   PFS DH group = 14
ike 0:to_HQ:101:to_HQ:259:      trans_id = ESP_AES_CBC (key_len = 128)
ike 0:to_HQ:101:to_HQ:259:      encapsulation = ENCAPSULATION_MODE_TUNNEL
ike 0:to_HQ:101:to_HQ:259:         type = AUTH_ALG, val=SHA1
ike 0:to_HQ: schedule auto-negotiate
ike 0:to_HQ:101:to_HQ:259: replay protection enabled
ike 0:to_HQ:101:to_HQ:259: SA life soft seconds=42902.
ike 0:to_HQ:101:to_HQ:259: SA life hard seconds=43200.
ike 0:to_HQ:101:to_HQ:259: IPsec SA selectors #src=1 #dst=1
```

```
ike 0:to_HQ:101:to_HQ:259: src 0 4 0:0.0.0.0/0.0.0.0:0
ike 0:to_HQ:101:to_HQ:259: dst 0 4 0:0.0.0.0/0.0.0.0:0
ike 0:to_HQ:101:to_HQ:259: add IPsec SA: SPIs=ca64644b/747c10c9
ike 0:to_HQ:101:to_HQ:259: IPsec SA dec spi ca64644b key
16:D5C60F1A3951B288CE4DEC7E04D2119D auth 20:F872A7A26964208A9AA368A31AEFA3DB3F3780BC
ike 0:to_HQ:101:to_HQ:259: IPsec SA enc spi 747c10c9 key
16:97952E1594F718128D9D7B09400856EA auth 20:4D5E5BC45A9D5A9A4631E911932F5650A4639A37
ike 0:to_HQ:101:to_HQ:259: added IPsec SA: SPIs=ca64644b/747c10c9
ike 0:to_HQ:101:to_HQ:259: sending SNMP tunnel UP trap
ike 0:to_HQ:101: sent IKE msg (quick_i2send): 173.1.1.1:500->11.101.1.1:500, len=76,
id=dff03f1d4820222a/6c2caf4dcf5bab75:32f4cc01
```

SSL VPN

The following topics provide information about SSL VPN in FortiOS 7.2.0.

SSL VPN best practices

Securing remote access to network resources is a critical part of security operations. SSL VPN allows administrators to configure, administer, and deploy a remote access strategy for their remote workers. When not in use, SSL VPN can be disabled.

Choosing the correct mode of operation and applying the proper levels of security are integral to providing optimal performance and user experience, and keeping your user data safe.

The below guidelines outline selecting the correct SSL VPN mode for your deployment and employing best practices to ensure that your data are protected.

Information about SSL VPN throughput and maximum concurrent users is available on your device's datasheet; see Next-Generation Firewalls Models and Specifications.

Tunnel mode

In tunnel mode, the SSL VPN client encrypts all traffic from the remote client computer and sends it to the FortiGate through an SSL VPN tunnel over the HTTPS link between the user and the FortiGate.

The FortiGate establishes a tunnel with the client, and assigns a virtual IP (VIP) address to the client from a range reserved addresses. While the underlying protocols are different, the outcome is very similar to a IPsec VPN tunnel. All client traffic is encrypted, allowing the users and networks to exchange a wide range of traffic, regardless of the application or protocols.

Use this mode if you require:

- A wide range of applications and protocols to be accessed by the remote client.
- No proxying is done by the FortiGate.
- Straightforward configuration and administration, as traffic is controlled by firewall policies.
- A transparent experience for the end user. For example, a user that needs to RDP to their server only requires a tunnel connection; they can then use the usual client application, like Windows Remote Desktop, to connect.

Full tunneling forces all traffic to pass through the FortiGate (see SSL VPN full tunnel for remote user on page 1627). Split tunneling only routes traffic to the designated network through the FortiGate (see SSL VPN split tunnel for remote user on page 1620).

Limitations

Tunnel mode requires that the FortiClient VPN client be installed on the remote end. The standalone FortiClient VPN client is free to use, and can accommodate SSL VPN and IPsec VPN tunnels. For supported operating systems, see the FortiClient Technical Specifications.

Web mode

Web-only mode provides clientless network access using a web browser with built-in SSL encryption. Users authenticate to FortiGate's SSL VPN Web Portal, which provides access to network services and resources, including HTTP/HTTPS, Telnet, FTP, SMB/CIFS, VNC, RDP, and SSH. When a user starts a connection to a server from the web portal, FortiOS proxies this communication with the server. All communication between the FortiGate and the user continues to be over HTTPS, regardless of the service that is being accessed.

The clipboard can be disabled for SSL VPN web mode RDP/VNC connections, see Disable the clipboard in SSL VPN web mode RDP connections on page 1771.

Use this mode if you require:

- A clientless solution in which all remote services are access through a web portal.
- Tight control over the contents of the web portal.
- Limited services provided to the remote users.

Limitations

- Multiple applications and protocols are not supported.
- VNC and RDP access might have limitations, such as certain shortcut keys not being supported.
- In some configurations RDP can consume a significant amount of memory and CPU time.
- Firewall performance might decrease as remote usage increases.
- Highly customized web pages might not render correctly.

Security best practices

Integrate with authentication servers

For networks with many users, integrate your user configuration with existing authentication servers through LDAP, RADIUS, or FortiAuthenticator.

By integrating with existing authentication servers, such as Windows AD, there is a lower change of making mistakes when configuring local users and user groups. Your administration effort is also reduces.

See SSL VPN with LDAP user authentication on page 1647 for more information.

Use a non-factory SSL certificate for the SSL VPN portal

Your certificate should identify your domain so that a remote user can recognize the identity of the server or portal that they are accessing through a trusted CA.

The default Fortinet factory self-signed certificates are provided to simplify initial installation and testing. If you use these certificates you are vulnerable to man-in-the-middle attacks, where an attacker spoofs your certificate, compromises your connection, and steals your personal information. It is highly recommended that you purchase a server certificate from a trusted CA to allow remote users to connect to SSL VPN with confidence. See Procuring and importing a signed SSL certificate on page 2173 for more information.

Enabling the *Do not Warn Invalid Server Certificate* option on the client disables the certificate warning message, potentially allowing users to accidentally connect to untrusted servers. Disabling invalid server certificate warnings is not recommended.

Use multi-factor authentication

Multi-factor authentication (MFA) ensures that the end-user is who they claim to be by requiring at least two factors - a piece of information that the user knows (password), and an asset that the user has (OTP). A third factor, something a user is (fingerprint or face), may be enabled as well. FortiToken Mobile is typically used for MFA.

FortiGate comes with two free FortiTokens, and more can be purchased from the FortiToken Mobile iOS app or through Fortinet partners.

See SSL VPN with FortiToken mobile push authentication on page 1676 for more information.

2FA, a subset of MFA, can also be set up with email tokens. See Email Two-Factor Authentication on FortiGate for information.

Deploy user certificates for remote SSL VPN users

This method of 2FA uses a user certificate as the second authentication factor. This is more secure, as it identifies the end user using a certificate. The configuration and administration of this solution is significantly more complicated, and requires administrators with advanced knowledge of the FortiGate and certificate deployment.

See SSL VPN with certificate authentication on page 1658 for more information.

Define your minimum supported TLS version and cipher suites

Minimum and maximum supported TLS version can be configured in the FortiGate CLI. The cipher algorithm can also be customized.

See How to control the SSL version and cipher suite for SSL VPN for more information.

Properly administer firewall policies and profiles against only the access level required for the remote user

Users do not all require the same access. Access should only be granted after careful considerations. Typically, users are placed in groups, and each group is allowed access to limited resources.

Using SSL VPN realms simplifies defining the control structure for mapping users and groups to the appropriate resources.

See SSL VPN multi-realm on page 1723 for more information.

Disable SSL VPN

After the SSL VPN settings have been configured, SSL VPN can be disabled when not in use.

To disable SSL VPN in the GUI:

1. Go to *VPN > SSL-VPN Settings*.
2. Disable *Enable SSL-VPN*.
3. Click *Apply*.

To disable SSL VPN in the CLI:

```
config vpn ssl settings
    set status disable
end
```

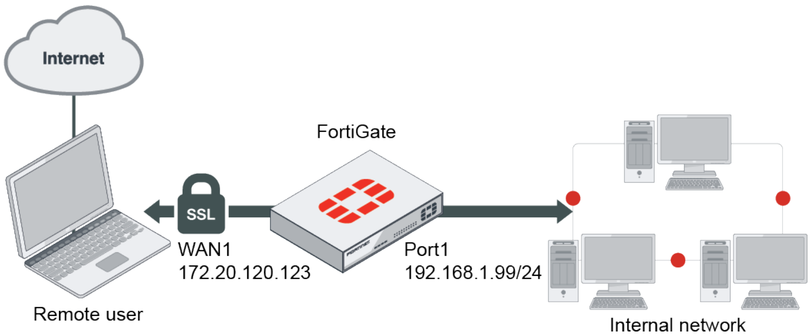
SSL VPN quick start

The following topics provide introductory instructions on configuring SSL VPN:

SSL VPN split tunnel for remote user

This is a sample configuration of remote users accessing the corporate network and internet through an SSL VPN by tunnel mode using FortiClient but accessing the Internet without going through the SSL VPN tunnel.

Sample topology



Sample configuration

WAN interface is the interface connected to ISP. This example shows static mode. You can also use DHCP or PPPoE mode. The SSL VPN connection is established over the WAN interface.

The split tunneling routing address cannot explicitly use an FQDN or an address group that includes an FQDN. To use an FQDN, leave the routing address blank and apply the FQDN as the destination address of the firewall policy.

To configure SSL VPN using the GUI:

1. Configure the interface and firewall address. The port1 interface connects to the internal network.
   a. Go to *Network > Interfaces* and edit the *wan1* interface.
   b. Set *IP/Network Mask* to *172.20.120.123/255.255.255.0*.
   c. Edit *port1* interface and set *IP/Network Mask* to *192.168.1.99/255.255.255.0*.
   d. Click *OK*.
   e. Go to *Policy & Objects > Address* and create an address for internal subnet *192.168.1.0*.
2. Configure user and user group.
   a. Go to *User & Authentication > User Definition* to create a local user *sslvpnuser1*.
   b. Go to *User & Authentication > User Groups* to create a group *sslvpngroup* with the member *sslvpnuser1*.
3. Configure SSL VPN web portal.
   a. Go to *VPN > SSL-VPN Portals* to create a tunnel mode only portal *my-split-tunnel-portal*.
   b. Enable *Split Tunneling*.
   c. Select *Routing Address* to define the destination network that will be routed through the tunnel. Leave undefined to use the destination in the respective firewall policies.
4. Configure SSL VPN settings.
   a. Go to *VPN > SSL-VPN Settings*.
   b. For *Listen on Interface(s)*, select *wan1*.
   c. Set *Listen on Port* to *10443*.
   d. Choose a certificate for *Server Certificate*. The default is *Fortinet_Factory*.
   e. In *Authentication/Portal Mapping All Other Users/Groups*, set the *Portal* to *tunnel-access*.
   f. Create new *Authentication/Portal Mapping* for group *sslvpngroup* mapping portal *my-split-tunnel-portal*.

5. Configure SSL VPN firewall policy.
   a. Go to *Policy & Objects* > *Firewall Policy*.
   b. Fill in the firewall policy name. In this example, *sslvpn split tunnel access*.
   c. Incoming interface must be *SSL-VPN tunnel interface(ssl.root)*.
   d. Choose an *Outgoing Interface*. In this example, *port1*.
   e. Set the *Source* to *all* and group to *sslvpngroup*.
   f. In this example, the *Destination* is *all*.
   g. Set *Schedule* to *always*, *Service* to *ALL*, and *Action* to *Accept*.
   h. Click *OK*.

To configure SSL VPN using the CLI:

1. Configure the interface and firewall address.

```
config system interface
    edit "wan1"
        set vdom "root"
        set ip 172.20.120.123 255.255.255.0
    next
end
```

2. Configure internal interface and protected subnet, then connect the port1 interface to the internal network.

```
config system interface
    edit "port1"
        set vdom "root"
        set ip 192.168.1.99 255.255.255.0
    next
end
```

```
config firewall address
    edit "192.168.1.0"
        set subnet 192.168.1.0 255.255.255.0
    next
end
```

3. Configure user and user group.

```
config user local
    edit "sslvpnuser1"
        set type password
        set passwd your-password
    next
end
```

```
config user group
    edit "sslvpngroup"
        set member "sslvpnuser1"
    next
end
```

4. Configure SSL VPN web portal.

```
config vpn ssl web portal
    edit "my-split-tunnel-portal"
        set tunnel-mode enable
        set split-tunneling  enable
```

```
        set split-tunneling-routing-address "192.168.1.0"
        set ip-pools "SSLVPN_TUNNEL_ADDR1"
    next
end
```

5. Configure SSL VPN settings.

```
config vpn ssl settings
    set servercert "Fortinet_Factory"
    set tunnel-ip-pools "SSLVPN_TUNNEL_ADDR1"
    set tunnel-ipv6-pools "SSLVPN_TUNNEL_IPv6_ADDR1"
    set source-interface "wan1"
    set source-address "all"
    set source-address6 "all"
    set default-portal "full-access"
    config authentication-rule
        edit 1
            set groups "sslvpngroup"
            set portal "my-split-tunnel-portal"
        next
    next
end
```

6. Configure one SSL VPN firewall policy to allow remote user to access the internal network. Traffic is dropped from internal to remote client.

```
config firewall policy
    edit 1
        set name "sslvpn split tunnel access"
        set srcintf "ssl.root"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "192.168.1.0"
        set groups “sslvpngroup”
        set action accept
        set schedule "always"
        set service "ALL"
    next
end
```

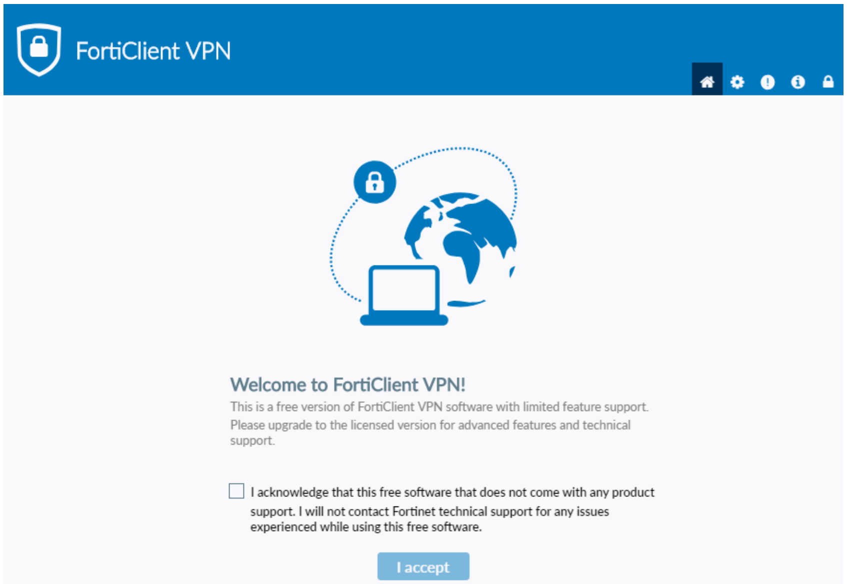
Connecting from FortiClient VPN client

For FortiGate administrators, a free version of FortiClient VPN is available which supports basic IPsec and SSL VPN and does not require registration with EMS. This version does not include central management, technical support, or some advanced features.

Downloading and installing the standalone FortiCient VPN client

You can download the free VPN client from FNDN or FortiClient.com.

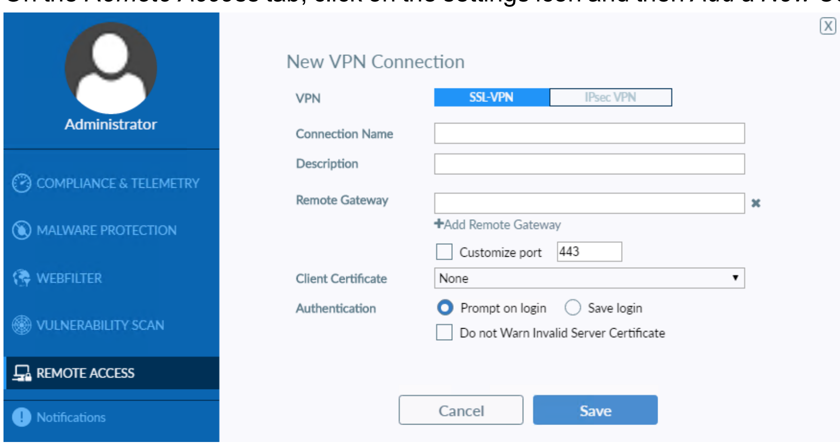
When the free VPN client is run for the first time, it displays a disclaimer. You cannot configure or create a VPN connection until you accept the disclaimer and click *I accept*:



Configuring an SSL VPN connection

To configure an SSL VPN connection:

1. On the *Remote Access* tab, click on the settings icon and then *Add a New Connection*.



2. Select *SSL-VPN*, then configure the following settings:

Connection Name	SSLVPNtoHQ
Description	(Optional)
Remote Gateway	172.20.120.123
Customize port	10443
Client Certificate	Select *Prompt on connect* or the certificate from the dropdown list.
Authentication	Select *Prompt on login* for a prompt on the connection screen

3. Click *Save* to save the VPN connection.

Connecting to SSL VPN

To connect to SSL VPN:

1. On the *Remote Access* tab, select the VPN connection from the dropdown list.
   Optionally, you can right-click the FortiTray icon in the system tray and select a VPN configuration to connect.
2. Enter your username and password.
3. Click the *Connect* button.
4. After connecting, you can now browse your remote network. Traffic to 192.168.1.0 goes through the tunnel, while other traffic goes through the local gateway. FortiClient displays the connection status, duration, and other relevant information.
5. Click the *Disconnect* button when you are ready to terminate the VPN session.

Checking the SSL VPN connection

To check the SSL VPN connection using the GUI:

1. On the FortiGate, go to *VPN > Monitor > SSL-VPN Monitor* to verify the list of SSL users.
2. On the FortiGate, go to *Log & Report > Forward Traffic* to view the details of the SSL entry.

To check the tunnel log in using the CLI:

```
get vpn ssl monitor
SSL VPN Login Users:
 Index    User           Auth Type     Timeout    From           HTTP in/out   HTTPS in/out
 0        sslvpnuser1    1(1)          291        10.1.100.254   0/0           0/0

SSL VPN sessions:
 Index    User           Source IP      Duration    I/O Bytes      Tunnel/Dest IP
 0        sslvpnuser1    10.1.100.254   9           22099/43228    10.212.134.200
```

Set up FortiToken multi-factor authentication

This configuration adds multi-factor authentication (MFA) to the split tunnel configuration (SSL VPN split tunnel for remote user on page 1620). It uses one of the two free mobile FortiTokens that is already installed on the FortiGate.

To configure MFA using the GUI:

1. Configure a user and user group:
   a. Go to *User & Authentication > User Definition* and edit local user *sslvpnuser1*.
   b. Enable *Two-factor Authentication*.
   c. For *Authentication Type*, click *FortiToken* and select one mobile *Token* from the list.
   d. Enter the user's *Email Address*.
   e. Enable *Send Activation Code* and select *Email*.
   f. Click *Next* and click *Submit*.
2. Activate the mobile token.
   When a FortiToken is added to user *sslvpnuser1*, an email is sent to the user's email address. Follow the instructions to install your FortiToken mobile application on your device and activate your token.

To configure MFA using the CLI:

1. Configure a user and user group:

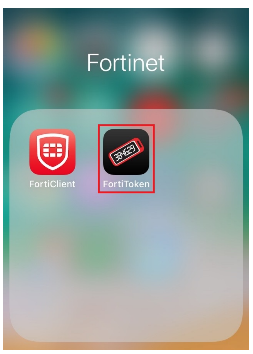
```
config user local
    edit "sslvpnuser1"
        set type password
        set two-factor fortitoken
        set fortitoken <select mobile token for the option list>
        set email-to <user's email address>
        set passwd <user's password>
    next
end
config user group
    edit "sslvpngroup"
        set member "sslvpnuser1"
    next
end
```

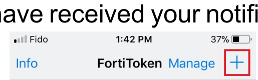
2. Activate the mobile token.
   When a FortiToken is added to user *sslvpnuser1*, an email is sent to the user's email address. Follow the instructions to install your FortiToken mobile application on your device and activate your token.

Connecting from FortiClient with FortiToken

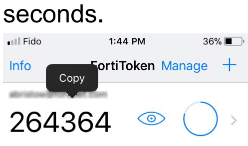
To activate your FortiToken:

1. On your device, open FortiToken Mobile. If this is your first time opening the application, it may prompt you to create a PIN for secure access to the application and tokens.



2. You should have received your notification via email, select + and use the device camera to scan the token QR code in your email. 

3. FortiToken Mobile provisions and activates your token and generates token codes immediately. To view the OTP's digits, select the eye icon. After you open the application, FortiToken Mobile generates a new six-digit OTP every 30 seconds.



To connect to SSL VPN:

1. On the *Remote Access* tab, select the VPN connection from the dropdown list.
   Optionally, you can right-click the FortiTray icon in the system tray and select a VPN configuration to connect.
2. Enter your username and password.
3. Click the *Connect* button.
4. A Token field will appear, prompting you for the FortiToken code. Enter the FortiToken code from your Mobile device.
5. After connecting, you can now browse your remote network. Traffic to 192.168.1.0 goes through the tunnel, while other traffic goes through the local gateway. FortiClient displays the connection status, duration, and other relevant information.
6. Click the *Disconnect* button when you are ready to terminate the VPN session.

SSL VPN tunnel mode

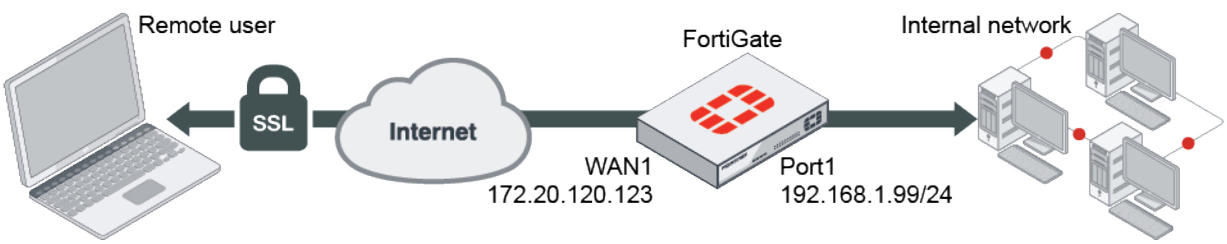
The following topics provide instructions on configuring SSL VPN tunnel mode:

- SSL VPN full tunnel for remote user
- SSL VPN tunnel mode host check
- SSL VPN split DNS on page 1633

SSL VPN full tunnel for remote user

This is a sample configuration of remote users accessing the corporate network and internet through an SSL VPN by tunnel mode using FortiClient.

Sample topology



Sample configuration

WAN interface is the interface connected to ISP. This example shows static mode. You can also use DHCP or PPPoE mode. The SSL VPN connection is established over the WAN interface.

To configure SSL VPN using the GUI:

1. Configure the interface and firewall address:
   a. Go to *Network > Interfaces* and edit the *wan1* interface.
   b. Set *IP/Network Mask* to *172.20.120.123/255.255.255.0*.
   c. Edit *port1* interface and set *IP/Network Mask* to *192.168.1.99/255.255.255.0*.
   d. Click *OK*.

2. Configure user and user group:
   a. Go to *User & Authentication* > *User Definition* to create a local user *sslvpnuser1*.
   b. Go to *User & Authentication* > *User Groups* to create a group *sslvpngroup* with the member *sslvpnuser1*.
3. Configure SSL VPN web portal:
   a. Go to *VPN* > *SSL-VPN Portals* to create a tunnel mode only portal *my-full-tunnel-portal*.
   b. Disable *Split Tunneling*.
4. Configure SSL VPN settings:
   a. Go to *VPN* > *SSL-VPN Settings*.
   b. For *Listen on Interface(s)*, select *wan1*.
   c. Set *Listen on Port* to *10443*.
   d. Choose a certificate for *Server Certificate*. The default is *Fortinet_Factory*.
   e. In *Authentication/Portal Mapping All Other Users/Groups*, set the *Portal* to *tunnel-access*.
   f. Create new *Authentication/Portal Mapping* for group *sslvpngroup* mapping portal *my-full-tunnel-portal*.
5. Configure SSL VPN firewall policies to allow remote user to access the internal network:
   a. Go to *Policy & Objects* > *Firewall Policy* and click *Create New*.
   b. Set *Name* to *sslvpn tunnel mode access*.
   c. Set *Incoming Interface* to *SSL-VPN tunnel interface(ssl.root)*.
   d. Set *Outgoing Interface* to *port1*.
   e. Set the *Source* Address to *all* and User to *sslvpngroup*.
   f. Set *Destination* to *all*, *Schedule* to *always*, *Service* to *ALL*, and *Action* to *Accept*.
   g. Click *OK*.
   h. Click *Create New*.
   i. Set *Name* to *sslvpn tunnel mode outgoing*.
   j. Configure the same settings as the previous policy, except set *Outgoing Interface* to *wan1*.
   k. Click *OK*.

To configure SSL VPN using the CLI:

1. Configure the interface and firewall address.

```
config system interface
    edit "wan1"
        set vdom "root"
        set ip 172.20.120.123 255.255.255.0
    next
end
```

2. Configure the internal interface and protected subnet, then connect the port1 interface to the internal network.

```
config system interface
    edit "port1"
        set vdom "root"
        set ip 192.168.1.99 255.255.255.0
    next
end
```

3. Configure user and user group.

```
config user local
    edit "sslvpnuser1"
        set type password
```

```
        set passwd your-password
    next
end
```

```
config user group
    edit "sslvpngroup"
        set member "sslvpnuser1"
    next
end
```

4. Configure SSL VPN web portal and predefine RDP bookmark for windows server.

```
config vpn ssl web portal
    edit "my-full-tunnel-portal"
        set tunnel-mode enable
        set split-tunneling disable
        set ip-pools "SSLVPN_TUNNEL_ADDR1"
    next
end
```

5. Configure SSL VPN settings.

```
config vpn ssl settings
    set servercert "Fortinet_Factory"
    set tunnel-ip-pools "SSLVPN_TUNNEL_ADDR1"
    set tunnel-ipv6-pools "SSLVPN_TUNNEL_IPv6_ADDR1"
    set source-interface "wan1"
    set source-address "all"
    set source-address6 "all"
    set default-portal "full-access"
    config authentication-rule
        edit 1
            set groups "sslvpngroup"
            set portal "my-full-tunnel-portal"
        next
    end
end
```

6. Configure SSL VPN firewall policies to allow remote user to access the internal network. Traffic is dropped from internal to remote client.

```
config firewall policy
    edit 1
        set name "sslvpn tunnel mode access"
        set srcintf "ssl.root"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set groups "sslvpngroup"
        set action accept
        set schedule "always"
        set service "ALL"
    next
    edit 2
        set name "sslvpn tunnel mode outgoing"
        set srcintf "ssl.root"
        set dstintf "wan1"
        set srcaddr "all"
```

```
        set dstaddr "all"
        set groups "sslvpngroup"
        set action accept
        set schedule "always"
        set service "ALL"
    next
end
```

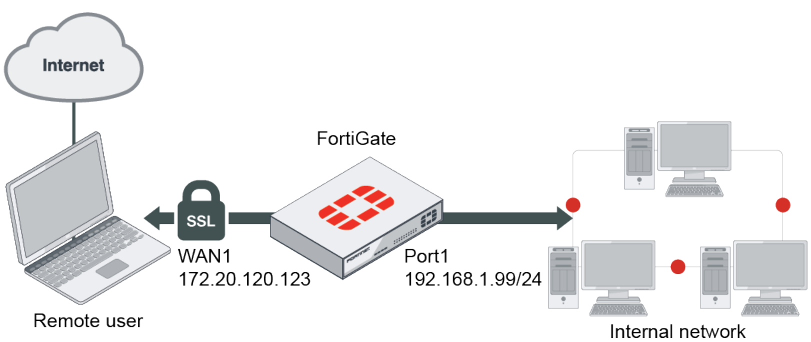
To see the results:

1. Download FortiClient from www.forticlient.com.
2. Open the FortiClient Console and go to *Remote Access*.
3. Add a new connection:
   - Set *VPN Type* to *SSL VPN*.
   - Set *Remote Gateway* to the IP of the listening FortiGate interface, in this example, *172.20.120.123*.
4. Select *Customize Port* and set it to *10443*.
5. Save your settings.
6. Use the credentials you've set up to connect to the SSL VPN tunnel.
7. After connection, all traffic except the local subnet will go through the tunnel *FGT*.
8. Go to *VPN > Monitor > SSL-VPN Monitor* to verify the list of SSL users.
9. On the FortiGate, go to *Log & Report > Forward Traffic* and view the details for the SSL entry.

SSL VPN tunnel mode host check

This is a sample configuration of remote users accessing the corporate network through an SSL VPN by tunnel mode using FortiClient with AV host check.

Sample topology



Sample configuration

WAN interface is the interface connected to ISP. This example shows static mode. You can also use DHCP or PPPoE mode. The SSL VPN connection is established over the WAN interface.

The split tunneling routing address cannot explicitly use an FQDN or an address group that includes an FQDN. To use an FQDN, leave the routing address blank and apply the FQDN as the destination address of the firewall policy.

To configure SSL VPN using the GUI:

1. Configure the interface and firewall address. The port1 interface connects to the internal network.
   a. Go to *Network* > *Interfaces* and edit the *wan1* interface.
   b. Set *IP/Network Mask* to *172.20.120.123/255.255.255.0*.
   c. Edit *port1* interface and set *IP/Network Mask* to *192.168.1.99/255.255.255.0*.
   d. Click *OK*.
   e. Go to *Policy & Objects* > *Address* and create an address for internet subnet *192.168.1.0*.
2. Configure user and user group.
   a. Go to *User & Authentication* > *User Definition* to create a local user *sslvpnuser1*.
   b. Go to *User & Authentication* > *User Groups* to create a group *sslvpngroup* with the member *sslvpnuser1*.
3. Configure SSL VPN web portal.
   a. Go to *VPN* > *SSL-VPN Portals* to create a tunnel mode only portal *my-split-tunnel-portal*.
   b. Enable *Tunnel Mode* and *Enable Split Tunneling*.
   c. Select *Routing Address*.
4. Configure SSL VPN settings.
   a. Go to *VPN* > *SSL-VPN Settings*.
   b. For *Listen on Interface(s)*, select *wan1*.
   c. Set *Listen on Port* to *10443*.
   d. Choose a certificate for *Server Certificate*.

   It is **HIGHLY** recommended that you acquire a signed certificate for your installation. Please review the SSL VPN best practices on page 1617 and learn how to Procuring and importing a signed SSL certificate on page 2173.

   e. In *Authentication/Portal Mapping All Other Users/Groups*, set the *Portal* to *tunnel-access*.
   f. Create new *Authentication/Portal Mapping* for group *sslvpngroup* mapping portal *my-split-tunnel-portal*.
5. Configure SSL VPN firewall policy.
   a. Go to *Policy & Objects* > *Firewall Policy*.
   b. Fill in the firewall policy name. In this example, *sslvpn tunnel access with av check*.
   c. Incoming interface must be *SSL-VPN tunnel interface(ssl.root)*.
   d. Choose an *Outgoing Interface*. In this example, *port1*.
   e. Set the *Source* to *all* and group to *sslvpngroup*.
   f. In this example, the *Destination* is *all*.
   g. Set *Schedule* to *always*, *Service* to *ALL*, and *Action* to *Accept*.
   h. Click *OK*.
6. Use CLI to configure SSL VPN web portal to enable the host to check for compliant antivirus software on the user's computer.

```
config vpn ssl web portal
    edit my-split-tunnel-access
        set host-check av
    next
end
```

To configure SSL VPN using the CLI:

1. Configure the interface and firewall address.

```
config system interface
    edit "wan1"
        set vdom "root"
        set ip 172.20.120.123 255.255.255.0
    next
end
```

2. Configure internal interface and protected subnet, then connect the port1 interface to the internal network.

```
config system interface
    edit "port1"
        set vdom "root"
        set ip 192.168.1.99 255.255.255.0
    next
end
```

```
config firewall address
    edit "192.168.1.0"
        set subnet 192.168.1.0 255.255.255.0
    next
end
```

3. Configure user and user group.

```
config user local
    edit "sslvpnuser1"
        set type password
        set passwd your-password
    next
end
```

```
config user group
    edit "sslvpngroup"
        set member "vpnuser1"
    next
end
```

4. Configure SSL VPN web portal.

```
config vpn ssl web portal
    edit "my-split-tunnel-portal"
        set tunnel-mode enable
        set split-tunneling  enable
        set split-tunneling-routing-address "192.168.1.0"
        set ip-pools "SSLVPN_TUNNEL_ADDR1"
    next
end
```

5. Configure SSL VPN settings.

```
config vpn ssl settings
    set servercert "Fortinet_Factory"
    set tunnel-ip-pools "SSLVPN_TUNNEL_ADDR1"
    set tunnel-ipv6-pools "SSLVPN_TUNNEL_IPv6_ADDR1"
    set source-interface "wan1"
    set source-address "all"
```

```
        set source-address6 "all"
        set default-portal "full-access"
        config authentication-rule
            edit 1
                set groups "sslvpngroup"
                set portal "my-split-tunnel-portal"
            next
        end
    end
```

6. Configure one SSL VPN firewall policy to allow remote user to access the internal network. Traffic is dropped from internal to remote client.

```
config firewall policy
    edit 1
        set name "sslvpn web mode access"
        set srcintf "ssl.root"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "192.168.1.0"
        set groups "sslvpngroup"
        set action accept
        set schedule "always"
        set service "ALL"
    next
end
```

7. Configure SSL VPN web portal to enable the host to check for compliant antivirus software on the user's computer:

```
config vpn ssl web portal
    edit my-split-tunnel-access
        set host-check av
    next
end
```

To see the results:

1. Download FortiClient from www.forticlient.com.
2. Open the FortiClient Console and go to *Remote Access*.
3. Add a new connection:
   - Set *VPN Type* to *SSL VPN*.
   - Set *Remote Gateway* to the IP of the listening FortiGate interface, in this example, *172.20.120.123*.
4. Select *Customize Port* and set it to *10443*.
5. Save your settings.
6. Use the credentials you've set up to connect to the SSL VPN tunnel.
   If the user's computer has antivirus software, a connection is established; otherwise FortiClient shows a compliance warning.
7. After connection, traffic to *192.168.1.0* goes through the tunnel. Other traffic goes through local gateway.
8. On the FortiGate, go to *VPN > Monitor > SSL-VPN Monitor* to verify the list of SSL users.
9. On the FortiGate, go to *Log & Report > Forward Traffic* and view the details for the SSL entry.

SSL VPN split DNS

SSL VPN clients in tunnel mode can enable the following settings to split DNS traffic:

- Resolve DNS requests for a specific domain, or suffix, using specific DNS servers.
- Resolve all other DNS requests using a DNS server configured in the SSL VPN settings. This DNS server can be the same as the client system DNS server, or another DNS server.

Administrators typically configure SSL VPN clients to use DNS servers that are behind the FortiGate on the internal network. This will require DNS traffic to traverse the SSL VPN tunnel.

Configuring SSL VPN DNS servers to use DNS suffixes

The `dns-suffix` setting under `config vpn ssl settings` is used to specify domains for SSL VPN DNS servers in the tunnel mode configuration. This setting can only be configured in the CLI.

The DNS servers and suffixes configured under `config vpn ssl settings` have a global scope, and apply only to SSL VPN portals that do not have their own DNS server configuration.

To configure DNS servers for all SSL VPN portals:

```
config vpn ssl settings
    set dns-suffix domain1.com
    set dns-server1 10.10.10.10
    set dns-server2 10.10.10.11
end
```

SSL VPN portals configured with their own DNS servers and suffixes under `config vpn ssl web portal` override the settings configured under `config vpn ssl settings`.

To configure DNS servers for a specific SSL VPN portal in split tunnel mode:

```
config vpn ssl web portal
    edit "full-access"
        set dns-suffix domain2.com
        set dns-server1 10.10.10.12
        set dns-server2 10.10.10.13
        set split-tunneling enable
    next
end
```

Only DNS requests that match DNS suffixes use the DNS servers configured in the VPN. Due to iOS limitations, the DNS suffixes are not used for searching as in Windows. Using short (non-FQDN) names may not be possible.

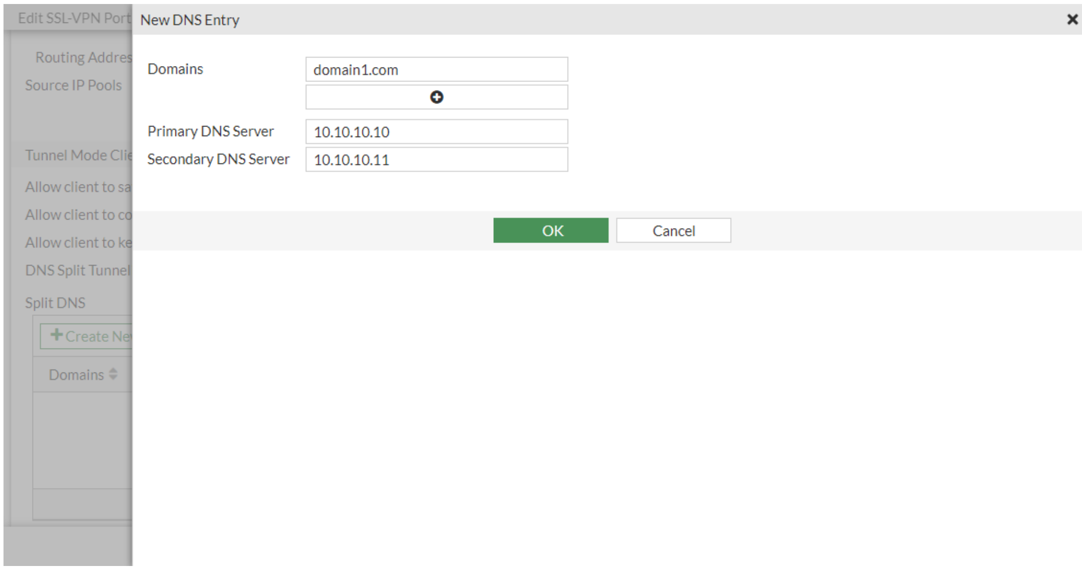
Configuring SSL VPN DNS servers for tunnel mode using DNS split tunneling

The DNS split tunneling setting can be used to configure domains that apply to a specific SSL VPN portal by specifying primary and secondary DNS servers to be used to resolve specific suffixes. This setting can be configured in the GUI and CLI. In the following example, DNS split tunneling is configured on the default tunnel-access portal with two DNS entries.

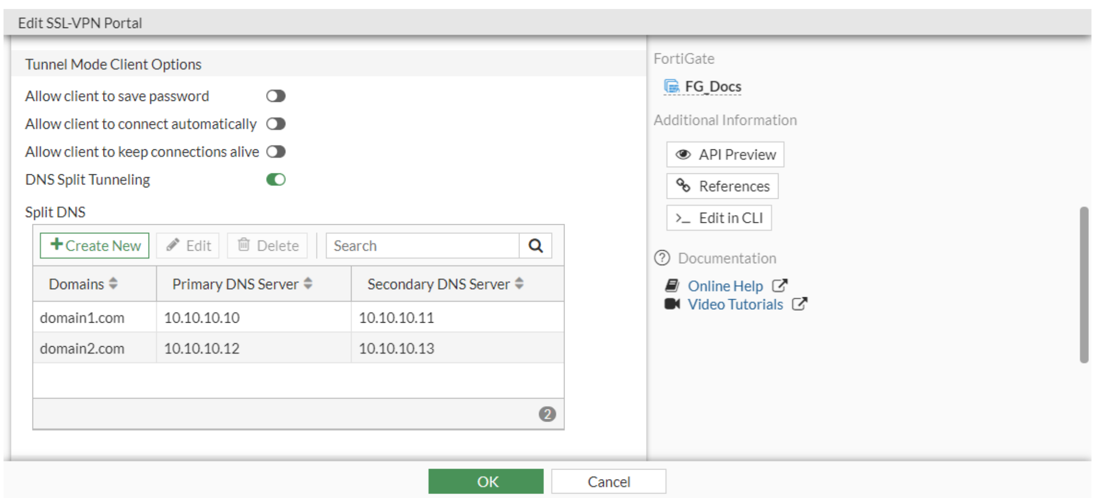
To configure DNS split tunneling in the GUI:

1. Go to *VPN > SSL-VPN Portals* and double-click *tunnel-access* to edit the portal.
2. In the *Tunnel Mode Client Options* section, enable *DNS Split Tunneling*.

3. In the *Split DNS* table, click *Create New*. The *New DNS Entry* pane opens.
4. Configure the first DNS entry:
   a. For *Domains*, enter *domain1.com*.
   b. Set the *Primary DNS Server* to *10.10.10.10*.
   c. Set the *Secondary DNS Server* to *10.10.10.11*.



   d. Click *OK*.
5. Configure the second DNS entry:
   a. Click *Create New*.
   b. For *Domains*, enter *domain2.com*.
   c. Set the *Primary DNS Server* to *10.10.10.12*.
   d. Set the *Secondary DNS Server* to *10.10.10.13*.
   e. Click *OK*.



6. Click *OK* to save the portal settings.

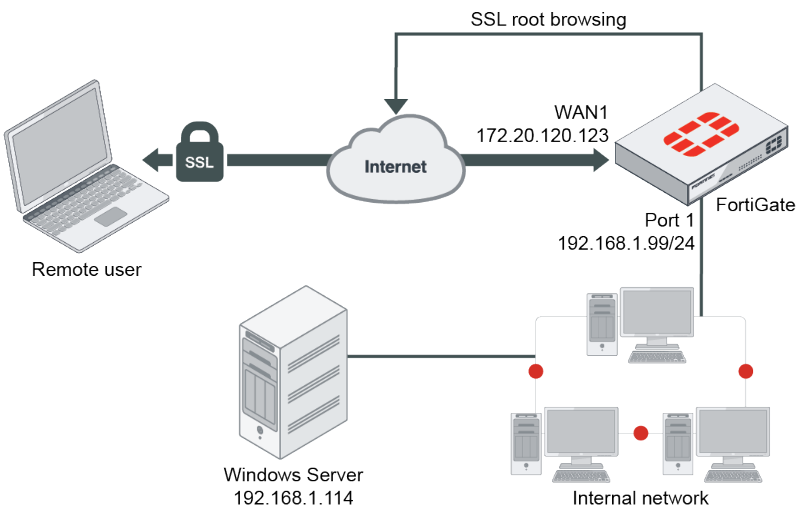
To configure DNS split tunneling in the CLI:

```
config vpn ssl web portal
    edit "tunnel-access"
        set dns-suffix "domain0.com"
        set dns-server1 10.10.10.8
        set dns-server2 10.10.10.9
        set split-tunneling enable
        config split-dns
            edit 1
                set domains "domain1.com"
                set dns-server1 10.10.10.10
                set dns-server2 10.10.10.11
            next
            edit 2
                set domains "domain2.com"
                set dns-server1 10.10.10.12
                set dns-server2 10.10.10.13
            next
        end
    next
end
```

SSL VPN web mode for remote user

This is a sample configuration of remote users accessing the corporate network through an SSL VPN by web mode using a web browser.

Sample topology



Sample configuration

WAN interface is the interface connected to ISP. This example shows static mode. You can also use DHCP or PPPoE mode. The SSL VPN connection is established over the WAN interface.

To configure SSL VPN using the GUI:

1. Configure the interface and firewall address. The port1 interface connects to the internal network.
   a. Go to *Network > Interfaces* and edit the *wan1* interface.
   b. Set *IP/Network Mask* to *172.20.120.123/255.255.255.0*.
   c. Edit *port1* interface and set *IP/Network Mask* to *192.168.1.99/255.255.255.0*.
   d. Click *OK*.
   e. Go to *Policy & Objects > Address* and create an address for internet subnet *192.168.1.0*.
2. Configure user and user group.
   a. Go to *User & Authentication > User Definition* to create a local user *sslvpnuser1*.
   b. Go to *User & Authentication > User Groups* to create a group *sslvpngroup* with the member *sslvpnuser1*.
3. Configure SSL VPN web portal.
   a. Go to *VPN > SSL-VPN Portals* to create a web mode only portal *my-web-portal*.
   b. Set *Predefined Bookmarks for Windows server* to type *RDP*.
4. Configure SSL VPN settings.
   a. Go to *VPN > SSL-VPN Settings*.
   b. For *Listen on Interface(s)*, select *wan1*.
   c. Set *Listen on Port* to *10443*.
   d. Choose a certificate for *Server Certificate*.

   It is **HIGHLY** recommended that you acquire a signed certificate for your installation. Please review the SSL VPN best practices on page 1617 and learn how to Procuring and importing a signed SSL certificate on page 2173.

   e. In *Authentication/Portal Mapping All Other Users/Groups*, set the *Portal* to *web-access*.
   f. Create new *Authentication/Portal Mapping* for group *sslvpngroup* mapping portal *my-Web-portal*.
5. Configure SSL VPN firewall policy.
   a. Go to *Policy & Objects > Firewall Policy*.
   b. Fill in the firewall policy name. In this example, *sslvpn web mode access*.
   c. Incoming interface must be *SSL-VPN tunnel interface(ssl.root)*.
   d. Choose an *Outgoing Interface*. In this example, *port1*.
   e. Set the *Source* to *all* and group to *sslvpngroup*.
   f. In this example, the *Destination* is the internal protected subnet *192.168.1.0*.
   g. Set *Schedule* to *always*, *Service* to *ALL*, and *Action* to *Accept*.
   h. Click *OK*.

To configure SSL VPN using the CLI:

1. Configure the interface and firewall address.

```
config system interface
    edit "wan1"
        set vdom "root"
        set ip 172.20.120.123 255.255.255.0
    next
end
```

2. Configure the internal interface and protected subnet, then connect the port1 interface to the internal network.

```
config system interface
    edit "port1"
        set vdom "root"
        set ip 192.168.1.99 255.255.255.0
    next
end
```

```
config firewall address
    edit "192.168.1.0"
        set subnet 192.168.1.0 255.255.255.0
    next
end
```

3. Configure user and user group.

```
config user local
    edit "sslvpnuser1"
        set type password
        set passwd your-password
    next
end
```

```
config user group
    edit "sslvpngroup"
        set member "vpnuser1"
    next
end
```

4. Configure SSL VPN web portal and predefine RDP bookmark for windows server.

```
config vpn ssl web portal
    edit "my-web-portal"
        set web-mode enable
        config bookmark-group
            edit "gui-bookmarks"
                config bookmarks
                    edit "Windows Server"
                        set apptype rdp
                        set host "192.168.1.114"
                        set port 3389
                        set logon-user "your-windows-server-user-name"
                        set logon-password your-windows-server-password
                    next
                end
            next
        end
    next
end
```

5. Configure SSL VPN settings.

```
config vpn ssl settings
    set servercert "Fortinet_Factory"
    set tunnel-ip-pools "SSLVPN_TUNNEL_ADDR1"
    set tunnel-ipv6-pools "SSLVPN_TUNNEL_IPv6_ADDR1"
    set source-interface "wan1"
    set source-address "all"
    set source-address6 "all"
```

```
        set default-portal "full-access"
        config authentication-rule
            edit 1
                set groups "sslvpngroup"
                set portal "my-web-portal"
            next
        end
    end
```

6. Configure one SSL VPN firewall policy to allow remote user to access the internal network. Traffic is dropped from internal to remote client

```
config firewall policy
    edit 1
        set name "sslvpn web mode access"
        set srcintf "ssl.root"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "192.168.1.0"
        set groups “sslvpngroup”
        set action accept
        set schedule "always"
        set service "ALL"
    next
end
```

To see the results:

1. In a web browser, log into the portal *https://172.20.120.123:10443* using the credentials you've set up.
2. In the portal with the predefined bookmark, select the bookmark to begin an RDP session. If there are no predefined bookmarks, the Quick Connection tool can be used; see Quick Connection tool on page 1639 for more information.
3. Go to *VPN > Monitor > SSL-VPN Monitor* to verify the list of SSL users.
4. Go to *Log & Report > Forward Traffic* to view the details for the SSL entry.

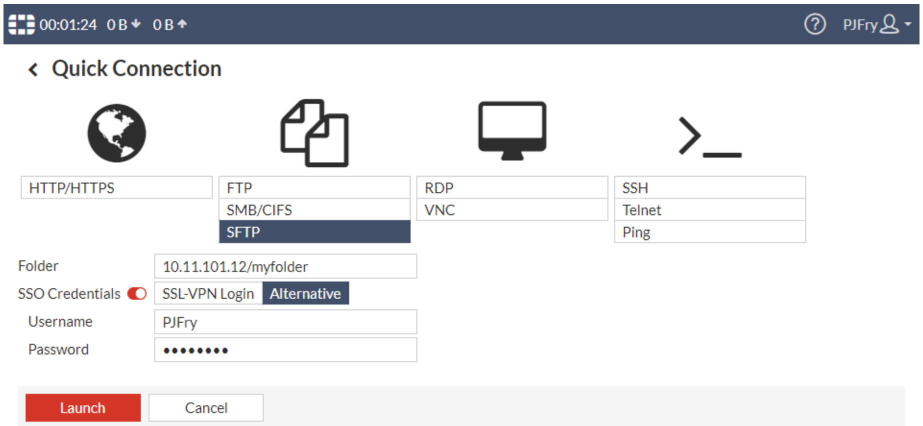
Quick Connection tool

The Quick Connection tool allows a user to connect to a resource when it is not a predefined bookmark. The tool allows the user to specify the type of server and the URL or IP address of the host.

To connect to a resource:

1. Select the connection type.
2. Enter the required information, such as the IP address or URL of the host.

3. Click *Launch*.



In a VNC session, to send Ctrl+Alt+Del, press *F8* then select *Send Ctrl-Alt-Delete*.

RDP sessions

Some Windows servers require that a specific security be set for RDP sessions, as opposed to the standard RDP encryption security. For example, Windows 10 requires that TLS be used.

You can specify a location option if the remote computer does not use the same keyboard layout as your computer by appending it to the *Host* field using the following format: `<IP address> -m <locale>`

The available options are:

ar	Arabic	fr-be	Belgian French	no	Norwegian
da	Danish	fr-ca	Canadian French	pl	Polish
de	German	fr-ch	Swiss French	pt	Portuguese
de-ch	Swiss German	hr	Croatian	pt-br	Brazilian Portuguese
en-gb	British English	hu	Hungarian	ru	Russian
en-uk	UK English	it	Italian	sl	Slovenian
en-us	US English	ja	Japanese	sv	Sudanese
es	Spanish	lt	Lithuanian	tk	Turkmen
fi	Finnish	lv	Latvian	tr	Turkish
fr	French	mk	Macedonian		

Customizing the RDP display size

The RDP display size (width and height settings) can be customized for SSL VPN web mode when creating a new connection or bookmark. Administrators can also specify the display size when preconfiguring bookmarks.

To configure the default window dimensions in an RDP web portal:

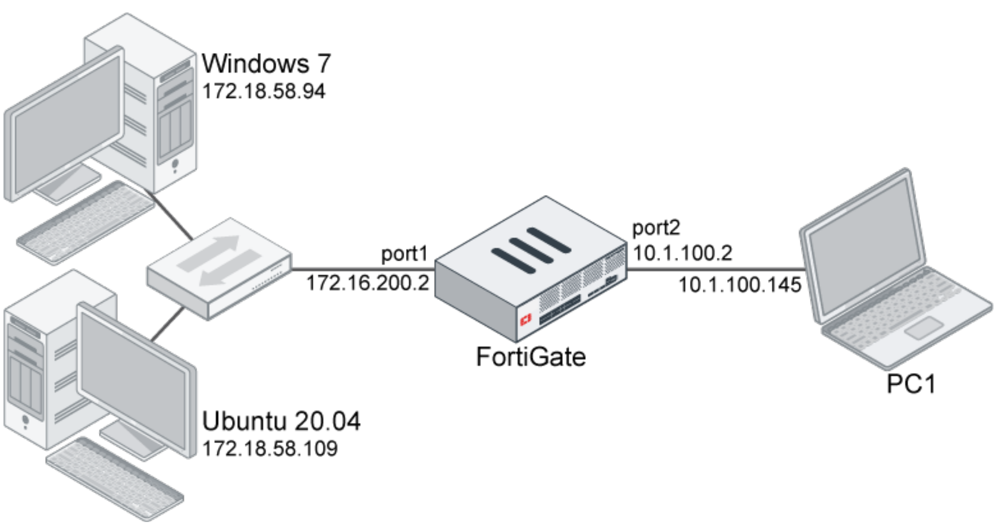
```
config vpn ssl web portal
    edit <name>
        set default-window-width <integer>
        set default-window-height <integer>
    next
end
```

`default-window-width <integer>`	Set the default RDP screen width, in pixels (0 - 65535, default = 1024).
`default-window-height <integer>`	Set the default RDP screen height, in pixels (0 - 65535, default = 768).

Example

In this example, a user has a monitor with a resolution of 1920 × 1080. The user creates two bookmarks for RDP servers with different resolutions:

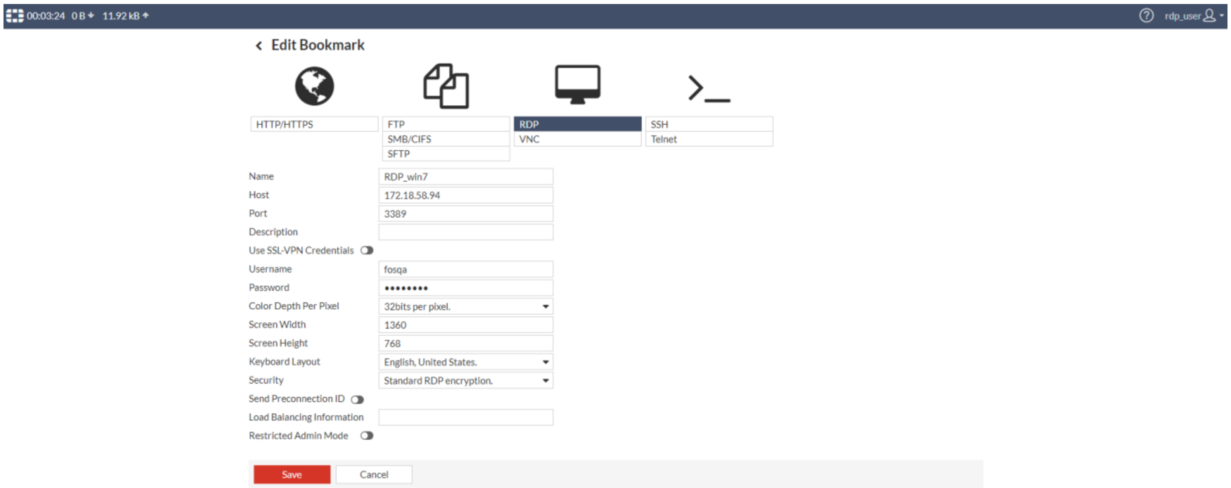
- Windows 7: 1360 × 768
- Ubuntu 20.04: 800 × 600



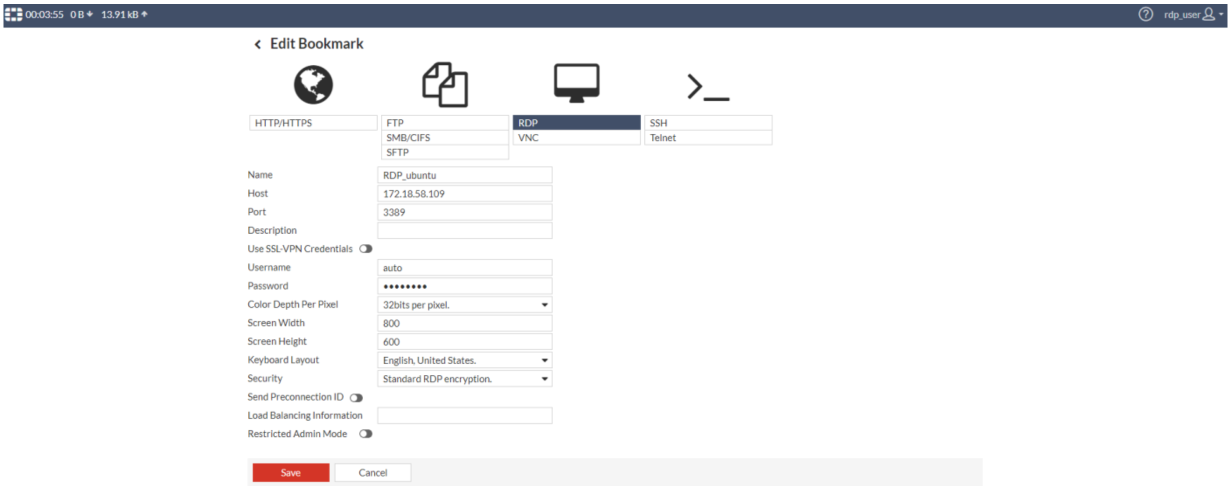
To customize the RDP bookmark display size:

1. Log in to the SSL VPN web portal.
2. Create a new RDP bookmark (+ *New Bookmark*), or hover over an existing bookmark and click the edit (pencil) icon.

3. Set the *Screen Width* and *Screen Height* fields as required.
   a. Windows 7: 1360 width and 768 height.



   b. Ubuntu 20.04: 800 width and 600 height.

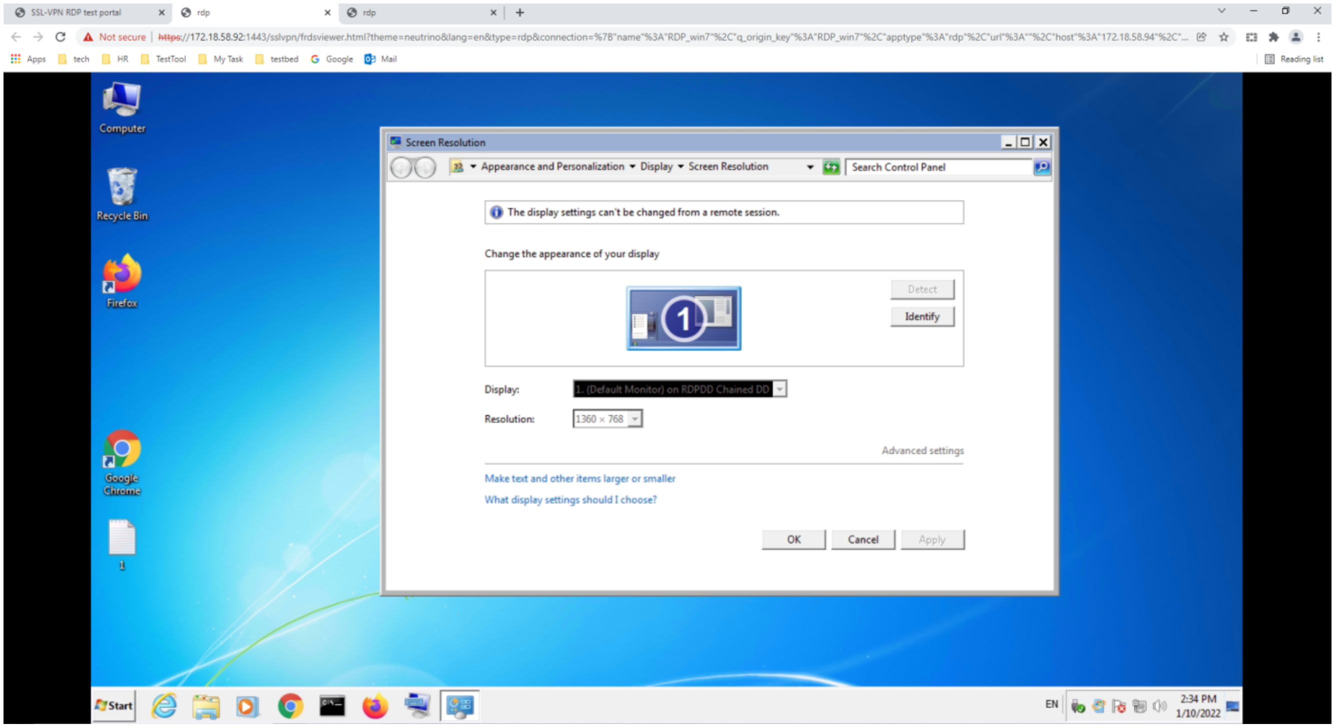


4. Click *Save*.

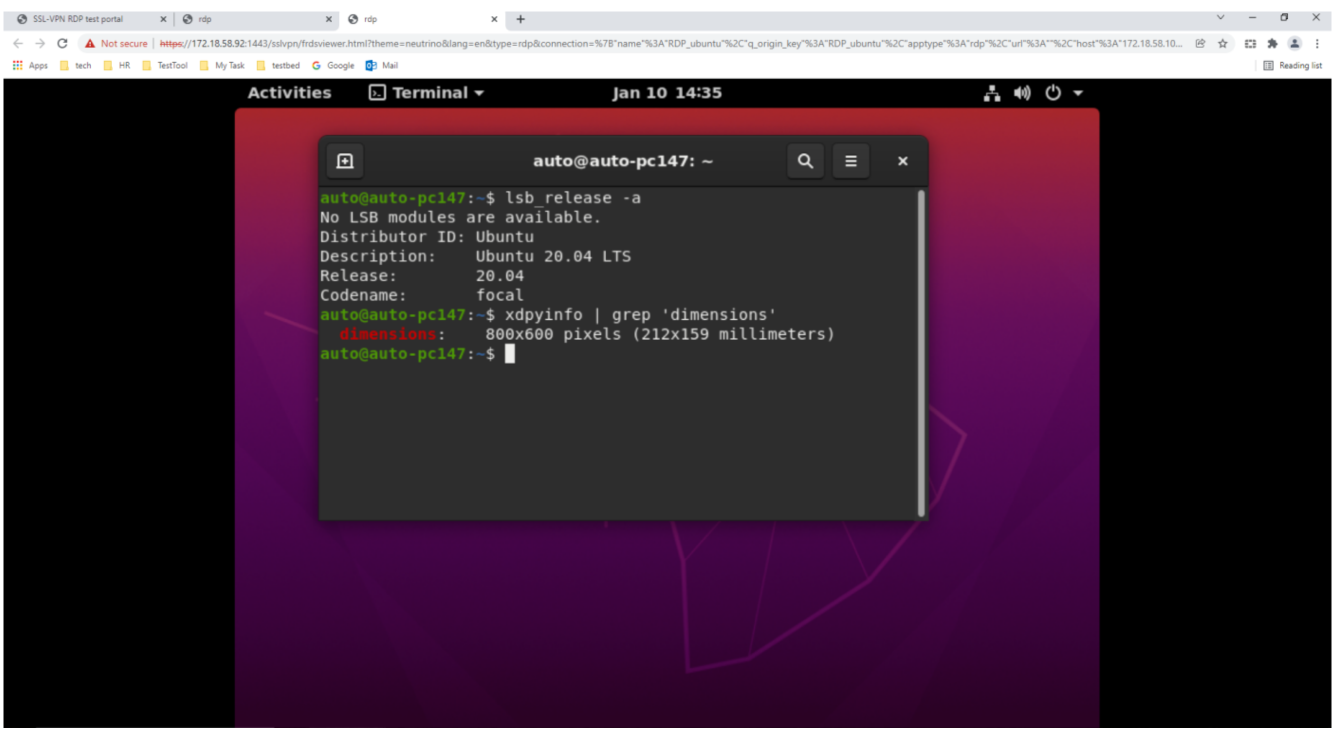
Verification:

When the user connects to the RDP servers using the bookmarks, the customized screen resolutions are applied regardless of the client PC's screen resolution (1920 × 1080).

Windows 7:



Ubuntu 20.04:



To view the bookmarks created by the user:

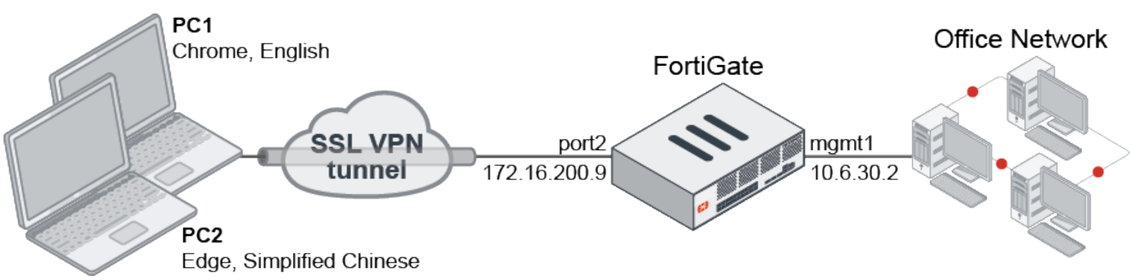
```
show vpn ssl web user-bookmark
config vpn ssl web user-bookmark
```

```
    edit "rdp_user#"
        config bookmarks
            edit "RDP_win7"
                set apptype rdp
                set host "172.18.58.94"
                set port 3389
                set logon-user "fosqa"
                set logon-password ********
                set color-depth 32
                set width 1360
                set height 768
            next
            edit "RDP_ubuntu"
                set apptype rdp
                set host "172.18.58.109"
                set port 3389
                set logon-user "auto"
                set logon-password ********
                set color-depth 32
                set width 800
                set height 600
            next
        end
    next
end
```

Showing the SSL VPN portal login page in the browser's language

By default, the browser's language preference is automatically detected and used by the SSL VPN portal login page. The system language can still be used by changing the settings on the *SSL-VPN Settings* page of the GUI, or disabling `browser-language-detection` in the CLI:

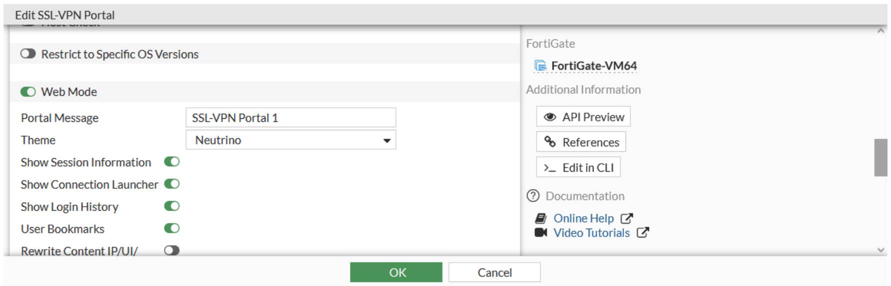
```
config vpn ssl settings
    set browser-language-detection disable
end
```



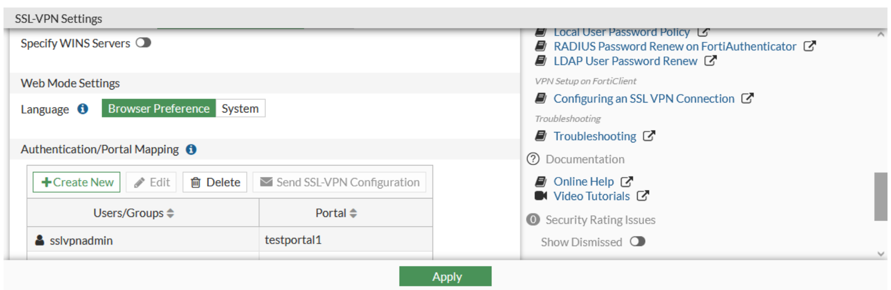
In this example, the *sslvpnadmin* user account is used for SSL VPN connections on the *testportal1* SSL VPN portal. The account is shared by users from different countries that use different browsers and different languages in their browsers. The user on PC1 uses Chrome in English, and the user on PC2 uses Edge in Simplified Chinese. When a user logs in to the SSL VPN web portal, all of the pages are shown in the same language as their browser.

To configure the SSL VPN portal to use the client's browser language:

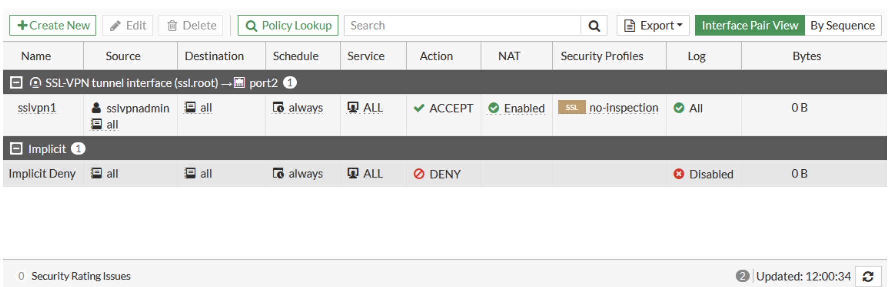
1. Configure the SSL VPN portal:
   a. Go to *VPN > SSL-VPN Portals* and edit the SSL VPN portal.
      For information about configuring SSL VPN portals, see SSL VPN on page 1617.
   b. Enable *Web Mode*.



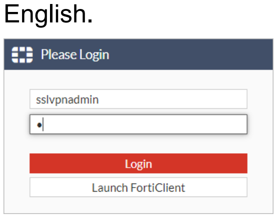
   c. Click *OK*.
2. Set the language preference:
   a. Go to *VPN > SSL-VPN Settings*.
   b. Under *Web Mode Settings*, set *Language* to *Browser Preference*.

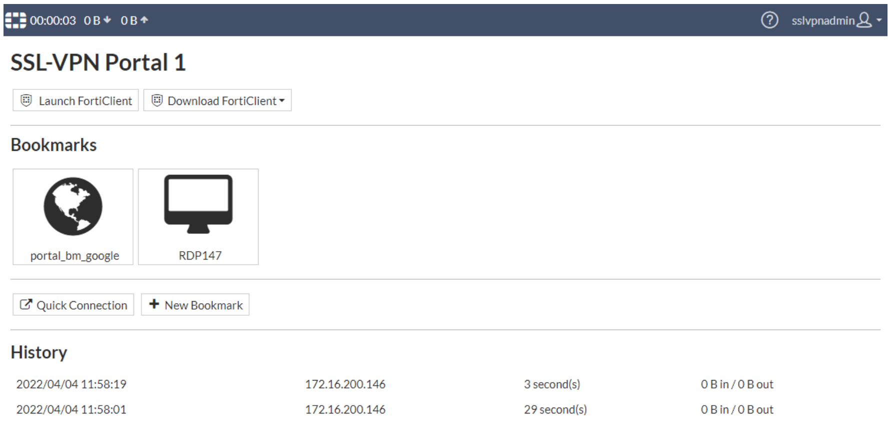


   c. Click *Apply*.
3. Add the *sslvpnadmin* user to the policy used by the SSL VPN portal.

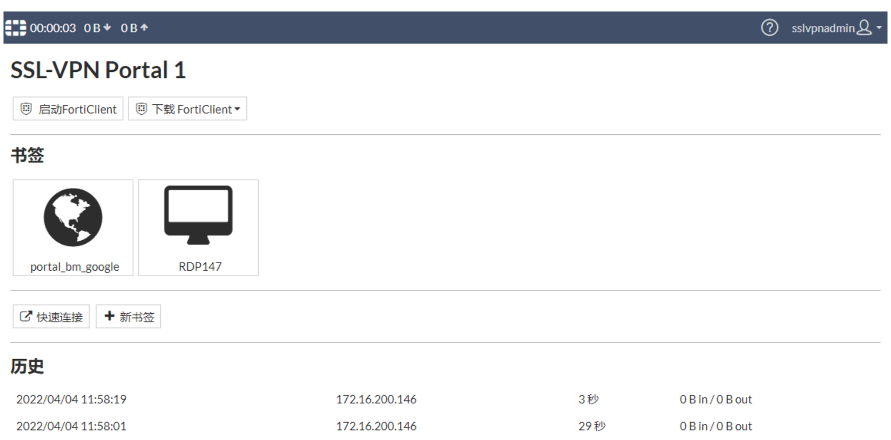


4. Confirm that the configuration works:
   - When the user on PC1 logs in to the SSL VPN portal using Chrome in English, all of the pages are shown in English.





- When the user on PC2 logs in to the SSL VPN portal using Edge in Simplified Chinese, all of the pages are shown in Simplified Chinese.



SSL VPN authentication

The following topics provide instructions on configuring SSL VPN authentication:

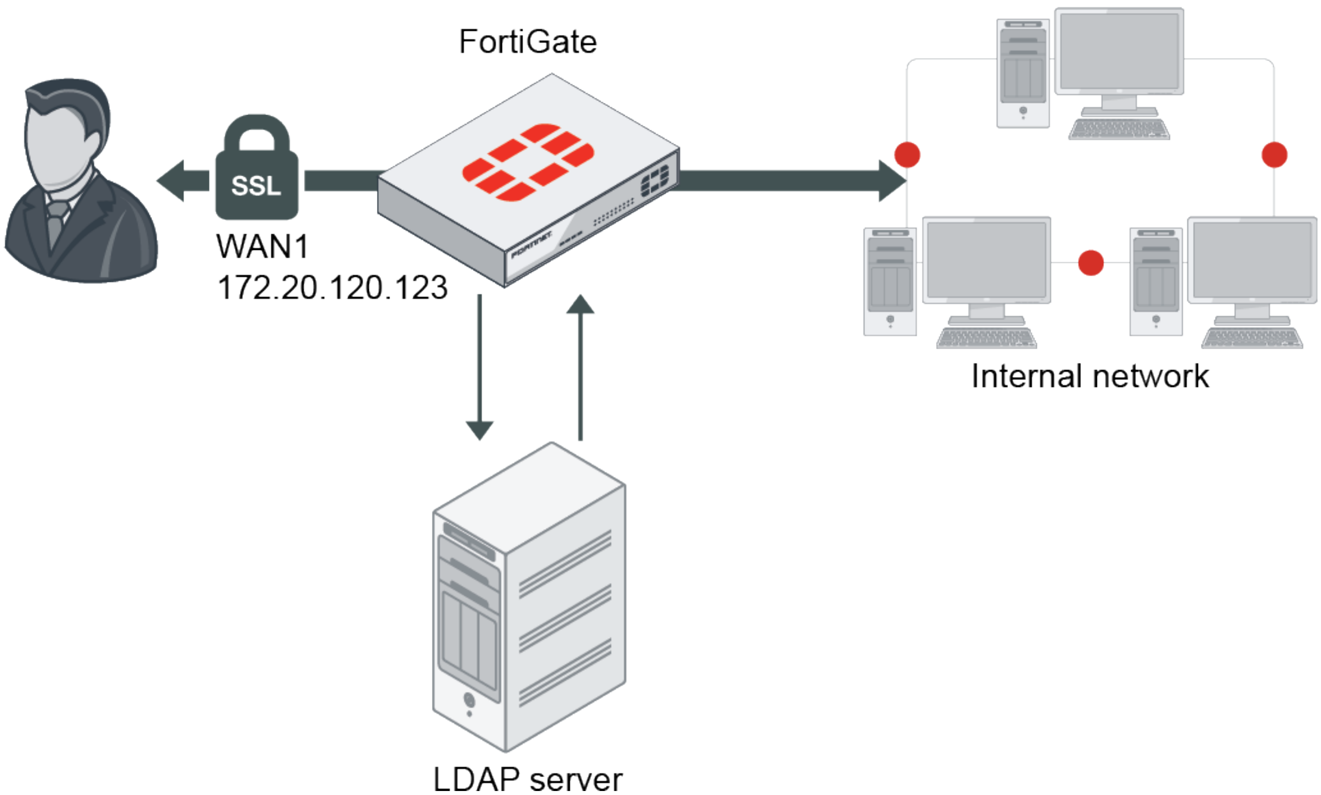
- SSL VPN with LDAP user authentication on page 1647
- SSL VPN with LDAP user password renew on page 1652

SSL VPN with LDAP user authentication

This is a sample configuration of SSL VPN for LDAP users. In this example, the LDAP server is a Windows 2012 AD server. A user *ldu1* is configured on Windows 2012 AD server.

You must have generated and exported a CA certificate from the AD server and then have imported it as an external CA certificate into the FortiGate.

Sample topology



Sample configuration

WAN interface is the interface connected to ISP. This example shows static mode. You can also use DHCP or PPPoE mode. The SSL VPN connection is established over the WAN interface.

To configure SSL VPN using the GUI:

1. Configure the interface and firewall address. The port1 interface connects to the internal network:
   a. Go to *Network* > *Interfaces* and edit the *wan1* interface.
   b. Set *IP/Network Mask* to *172.20.120.123/255.255.255.0*.
   c. Edit *port1* interface and set *IP/Network Mask* to *192.168.1.99/255.255.255.0*.
   d. Click *OK*.
   e. Go to *Policy & Objects* > *Address* and create an address for internet subnet *192.168.1.0*.
2. Import CA certificate into FortiGate:
   a. Go to *System* > *Features Visibility* and ensure *Certificates* is enabled.
   b. Go to *System* > *Certificates* and select *Import* > *CA Certificate*.
   c. Select *Local PC* and then select the certificate file.
      The CA certificate now appears in the list of *External CA Certificates*. In this example, it is called *CA_Cert_1*.
   d. If you want, you can use CLI commands to rename the system-generated *CA_Cert_1* to be more descriptive:

```
config vpn certificate ca
    rename CA_Cert_1 to LDAPS-CA
end
```

3. Configure the LDAP user:
   a. Go to *User & Authentication > LDAP Servers* and click *Create New*.
   b. Specify *Name* and *Server IP/Name*.
   c. Specify *Common Name Identifier* and *Distinguished Name*.
   d. Set *Bind Type* to *Regular*.
   e. Specify *Username* and *Password*.
   f. Enable *Secure Connection* and set *Protocol* to *LDAPS*.
   g. For *Certificate*, select *LDAP server CA LDAPS-CA* from the list.
4. Configure user group:
   a. Go to *User & Authentication > User Groups* to create a user group.
   b. Enter a *Name*.
   c. In *Remote Groups*, click *Add* to add *ldaps-server*.
5. Configure SSL VPN web portal:
   a. Go to *VPN > SSL-VPN Portals* to edit the *full-access* portal.
      This portal supports both web and tunnel mode.
   b. Disable *Enable Split Tunneling* so that all SSL VPN traffic goes through the FortiGate.
6. Configure SSL VPN settings:
   a. Go to *VPN > SSL-VPN Settings*.
   b. Select the *Listen on Interface(s)*, in this example, *wan1*.
   c. Set *Listen on Port* to *10443*.
   d. Set *Server Certificate* to the authentication certificate.
   e. Under *Authentication/Portal Mapping*, set default Portal *web-access* for *All Other Users/Groups*.
   f. Create new *Authentication/Portal Mapping* for group *ldaps-group* mapping portal *full-access*.
7. Configure SSL VPN firewall policy:
   a. Go to *Policy & Objects > Firewall Policy*.
   b. Fill in the firewall policy name, in this example, *sslvpn certificate auth*.
   c. Incoming interface must be *SSL-VPN tunnel interface(ssl.root)*.
   d. Set the *Source Address* to *all* and *Source User* to *ldaps-group*.
   e. Set the *Outgoing Interface* to the local network interface so that the remote user can access the internal network, in this example, *port1*.
   f. Set *Destination Address* to the internal protected subnet *192.168.1.0*.
   g. Set *Schedule* to *always*, *Service* to *ALL*, and *Action* to *Accept*.
   h. Enable *NAT*.
   i. Configure any remaining firewall and security options as desired.
   j. Click *OK*.

To configure SSL VPN using the CLI:

1. Configure the interface and firewall address:

```
config system interface
    edit "wan1"
        set vdom "root"
        set ip 172.20.120.123 255.255.255.0
    next
end
```

2. Configure internal interface and protected subnet, then connect the port1 interface to the internal network:

```
config system interface
    edit "port1"
        set vdom "root"
        set ip 192.168.1.99 255.255.255.0
    next
end
```

```
config firewall address
    edit "192.168.1.0"
        set subnet 192.168.1.0 255.255.255.0
    next
end
```

3. Import CA certificate into FortiGate:
   a. Go to *System > Features Visibility* and ensure *Certificates* is enabled.
   b. Go to *System > Certificates* and select *Import > CA Certificate*.
   c. Select *Local PC* and then select the certificate file.
      The CA certificate now appears in the list of *External CA Certificates*. In the example, it is called *CA_Cert_1*.
   d. If you want, you can use CLI commands to rename the system-generated *CA_Cert_1* to be more descriptive:

   ```
   config vpn certificate ca
       rename CA_Cert_1 to LDAPS-CA
   end
   ```

4. Configure the LDAP server:

```
config user ldap
    edit "ldaps-server"
        set server "172.20.120.161"
        set cnid "cn"
        set dn "cn=Users,dc=qa,dc=fortinet,dc=com"
        set type regular
        set username "CN=Administrator,cn=users,DC=qa,DC=fortinet,DC=com"
        set password **********
        set group-member-check group-object
        set secure ldaps
        set ca-cert "LDAPS-CA"
        set port 636
    next
end
```

5. Configure user group:

```
config user group
    edit "ldaps-group"
        set member "ldaps-server"
    next
end
```

6. Configure SSL VPN web portal:

```
config vpn ssl web portal
    edit "full-access"
        set tunnel-mode enable
        set web-mode enable
        set ip-pools "SSLVPN_TUNNEL_ADDR1"
        set split-tunneling disable
```

```
        next
    end
```

7. Configure SSL VPN settings:

```
config vpn ssl settings
    set servercert "server_certificate"
    set tunnel-ip-pools "SSLVPN_TUNNEL_ADDR1"
    set source-interface "wan1"
    set source-address "all"
    set default-portal "web-access"
    config authentication-rule
        edit 1
            set groups "ldaps-group"
            set portal "full-access"
        next
    end
end
```

8. Configure one SSL VPN firewall policy to allow remote user to access the internal network:

```
config firewall policy
    edit 1
        set name "sslvpn web mode access"
        set srcintf "ssl.root"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "192.168.1.0"
        set groups “ldaps-group”
        set action accept
        set schedule "always"
        set service "ALL"
        set nat enable
    next
end
```

To see the results of web portal:

1. From a remote device, use a web browser to log into the SSL VPN web portal *http://172.20.120.123:10443*.
2. Enter the *ldu1* user credentials, then click *Login*.
3. Go to *Dashboard* > *Network* and expand the *SSL-VPN* widget to verify the user’s connection.

To see the results of tunnel connection:

1. Download FortiClient from www.forticlient.com.
2. Open the FortiClient Console and go to *Remote Access* > *Configure VPN*.
3. Add a new connection:
   a. Set the connection name.
   b. Set *Remote Gateway* to the IP of the listening FortiGate interface, in this example, *172.20.120.123*.
   c. Select *Customize Port* and set it to *10443*.
4. Save your settings.
5. Log in using the *ldu1* credentials.

To check the SSL VPN connection using the GUI:

1. Go to *Dashboard > Network* and expand the *SSL-VPN* widget to verify the user's connection.
2. Go to *Log & Report > System Events* and select the *VPN Events* card to view the details of the SSL VPN connection event log.
3. Go to *Log & Report > Forward Traffic* to view the details of the SSL VPN traffic.

To check the web portal login using the CLI:

```
# get vpn ssl monitor
SSL VPN Login Users:
 Index    User     Auth Type     Timeout     From           HTTP in/out  HTTPS in/out
 0        ldu1     1(1)          229         10.1.100.254   0/0          0/0

SSL VPN sessions:
 Index    User     Source IP      Duration        I/O Bytes       Tunnel/Dest IP
```

To check the tunnel login using the CLI:

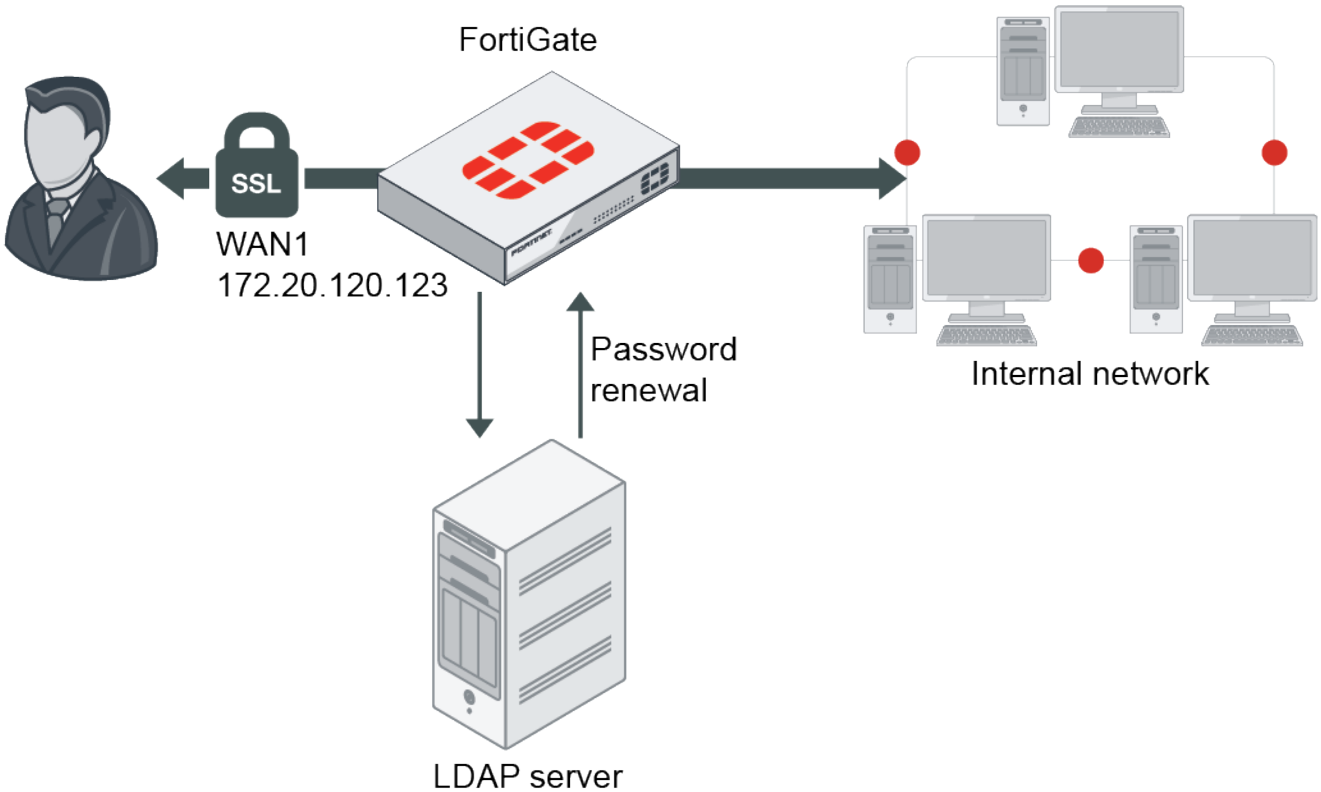
```
# get vpn ssl monitor
SSL VPN Login Users:
 Index    User     Auth Type     Timeout     From           HTTP in/out  HTTPS in/out
 0        ldu1     1(1)          291         10.1.100.254   0/0          0/0

SSL VPN sessions:
 Index    User     Source IP      Duration    I/O Bytes      Tunnel/Dest IP
 0        ldu1     10.1.100.254   9           22099/43228    10.212.134.200
```

SSL VPN with LDAP user password renew

This is a sample configuration of SSL VPN for LDAP users with *Force Password Change on next logon*. In this example, the LDAP server is a Windows 2012 AD server. A user *ldu1* is configured on Windows 2012 AD server with *Force password change on next logon*.

You must have generated and exported a CA certificate from the AD server and then have imported it as an external CA certificate into the FortiGate.

Sample topology



Sample configuration

WAN interface is the interface connected to ISP. This example shows static mode. You can also use DHCP or PPPoE mode. The SSL VPN connection is established over the WAN interface.

To configure SSL VPN using the GUI:

1. Configure the interface and firewall address. The port1 interface connects to the internal network.
   a. Go to *Network > Interfaces* and edit the *wan1* interface.
   b. Set *IP/Network Mask* to *172.20.120.123/255.255.255.0*.
   c. Edit *port1* interface and set *IP/Network Mask* to *192.168.1.99/255.255.255.0*.
   d. Click *OK*.
   e. Go to *Policy & Objects > Address* and create an address for internet subnet *192.168.1.0*.
2. Import CA certificate into FortiGate:
   a. Go to *System > Features Visibility* and ensure *Certificates* is enabled.
   b. Go to *System > Certificates* and select *Import > CA Certificate*.
   c. Select *Local PC* and then select the certificate file.
      The CA certificate now appears in the list of *External CA Certificates*. In this example, it is called *CA_Cert_1*.
   d. If you want, you can use CLI commands to rename the system-generated *CA_Cert_1* to be more descriptive:

```
config vpn certificate ca
    rename CA_Cert_1 to LDAPS-CA
end
```

3. Configure the LDAP user:

> The LDAP user must either be an administrator, or have the proper permissions delegated to it, to be able to change passwords of other registered users on the LDAP server.

   a. Go to *User & Authentication > LDAP Servers* and click *Create New*.
   b. Specify *Name* and *Server IP/Name*.
   c. Specify *Common Name Identifier* and *Distinguished Name*.
   d. Set *Bind Type* to *Regular*.
   e. Specify *Username* and *Password*.
   f. Enable *Secure Connection* and set *Protocol* to *LDAPS*.
   g. For *Certificate*, select *LDAP server CA LDAPS-CA* from the list.
   h. To enable the `password-renew` option, use these CLI commands.

```
config user ldap
    edit "ldaps-server"
        set password-expiry-warning enable
        set password-renewal enable
    next
end
```

4. Configure user group:
   a. Go to *User & Authentication > User Groups* to create a user group.
   b. Enter a *Name*.
   c. In *Remote Groups*, click *Add* to add *ldaps-server*.
5. Configure SSL VPN web portal:
   a. Go to *VPN > SSL-VPN Portals* to edit the *full-access* portal.
   This portal supports both web and tunnel mode.
   b. Disable *Enable Split Tunneling* so that all SSL VPN traffic goes through the FortiGate.
6. Configure SSL VPN settings:
   a. Go to *VPN > SSL-VPN Settings*.
   b. Select the *Listen on Interface(s)*, in this example, *wan1*.
   c. Set *Listen on Port* to *10443*.
   d. Set *Server Certificate* to the authentication certificate.
   e. Under *Authentication/Portal Mapping*, set default Portal *web-access* for *All Other Users/Groups*.
   f. Create new *Authentication/Portal Mapping* for group *ldaps-group* mapping portal *full-access*.
7. Configure SSL VPN firewall policy:
   a. Go to *Policy & Objects > Firewall Policy*.
   b. Fill in the firewall policy name, in this example, *sslvpn certificate auth*.
   c. Incoming interface must be *SSL-VPN tunnel interface(ssl.root)*.
   d. Set the *Source Address* to *all* and *Source User* to *ldaps-group*.

e. Set the *Outgoing Interface* to the local network interface so that the remote user can access the internal network, in this example, *port1*.
f. Set *Destination Address* to the internal protected subnet *192.168.1.0*.
g. Set *Schedule* to *always*, *Service* to *ALL*, and *Action* to *Accept*.
h. Enable *NAT*.
i. Configure any remaining firewall and security options as desired.
j. Click *OK*.

To configure SSL VPN using the CLI:

1. Configure the interface and firewall address:

```
config system interface
    edit "wan1"
        set vdom "root"
        set ip 172.20.120.123 255.255.255.0
    next
end
```

2. Configure internal interface and protected subnet, then connect the port1 interface to the internal network:

```
config system interface
    edit "port1"
        set vdom "root"
        set ip 192.168.1.99 255.255.255.0
    next
end
```

```
config firewall address
    edit "192.168.1.0"
        set subnet 192.168.1.0 255.255.255.0
    next
end
```

3. Import CA certificate into FortiGate:
   a. Go to *System > Features Visibility* and ensure *Certificates* is enabled.
   b. Go to *System > Certificates* and select *Import > CA Certificate*.
   c. Select *Local PC* and then select the certificate file.
      The CA certificate now appears in the list of *External CA Certificates*. In the example, it is called *CA_Cert_1*.
   d. If you want, you can use CLI commands to rename the system-generated *CA_Cert_1* to be more descriptive:

```
config vpn certificate ca
    rename CA_Cert_1 to LDAPS-CA
end
```

4. Configure the LDAP server:

The LDAP user must either be an administrator, or have the proper permissions delegated to it, to be able to change passwords of other registered users on the LDAP server.

```
config user ldap
    edit "ldaps-server"
        set server "172.20.120.161"
```

```
        set cnid "cn"
        set dn "cn=Users,dc=qa,dc=fortinet,dc=com"
        set type regular
        set username "CN=Administrator,cn=users,DC=qa,DC=fortinet,DC=com"
        set password **********
        set group-member-check group-object
        set secure ldaps
        set ca-cert "LDAPS-CA"
        set port 636
        set password-expiry-warning enable
        set password-renewal enable
    next
end
```

5. Configure user group:

```
config user group
    edit "ldaps-group"
        set member "ldaps-server"
    next
end
```

6. Configure SSL VPN web portal:

```
config vpn ssl web portal
    edit "full-access"
        set tunnel-mode enable
        set web-mode enable
        set ip-pools "SSLVPN_TUNNEL_ADDR1"
        set split-tunneling disable
     next
end
```

7. Configure SSL VPN settings:

```
config vpn ssl settings
    set servercert "server_certificate"
    set tunnel-ip-pools "SSLVPN_TUNNEL_ADDR1"
    set source-interface "wan1"
    set source-address "all"
    set default-portal "web-access"
    config authentication-rule
        edit 1
            set groups "ldaps-group"
            set portal "full-access"
        next
    end
end
```

8. Configure one SSL VPN firewall policy to allow remote user to access the internal network:

```
config firewall policy
    edit 1
        set name "sslvpn web mode access"
        set srcintf "ssl.root"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "192.168.1.0"
        set groups “ldaps-group”
```

```
            set action accept
            set schedule "always"
            set service "ALL"
            set nat enable
        next
    end
```

To see the results of web portal:

1. From a remote device, use a web browser to log into the SSL VPN web portal *http://172.20.120.123:10443*.
2. Log in using the *ldu1* credentials.
   Use a user that is configured on FortiAuthenticator with *Force password change on next logon*.
3. Click *Login*. You are prompted to enter a new password. The prompt will timeout after 90 seconds.
4. Go to *Dashboard > Network* and expand the *SSL-VPN* widget to verify the user's connection.

To see the results of tunnel connection:

1. Download FortiClient from www.forticlient.com.
2. Open the FortiClient Console and go to *Remote Access > Configure VPN*.
3. Add a new connection:
   a. Set the connection name.
   b. Set *Remote Gateway* to the IP of the listening FortiGate interface, in this example, *172.20.120.123*.
   c. Select *Customize Port* and set it to *10443*.
4. Save your settings.
5. Log in using the *ldu1* credentials.
   You are prompted to enter a new password. The prompt will timeout after 90 seconds.

To check the SSL VPN connection using the GUI:

1. Go to *Dashboard > Network* and expand the *SSL-VPN* widget to verify the user's connection.
2. Go to *Log & Report > System Events* and select the *VPN Events* card to view the details of the SSL VPN connection event log.
3. Go to *Log & Report > Forward Traffic* to view the details of the SSL VPN traffic.

To check the web portal login using the CLI:

```
# get vpn ssl monitor
SSL VPN Login Users:
 Index    User     Auth Type     Timeout     From            HTTP in/out   HTTPS in/out
 0        ldu1     1(1)          229         10.1.100.254    0/0           0/0

SSL VPN sessions:
 Index    User     Source IP      Duration        I/O Bytes       Tunnel/Dest IP
```

To check the tunnel login using the CLI:

```
# get vpn ssl monitor
SSL VPN Login Users:
 Index    User     Auth Type     Timeout     From            HTTP in/out   HTTPS in/out
 0        ldu1     1(1)          291         10.1.100.254    0/0           0/0
```

```
SSL VPN sessions:
 Index    User     Source IP       Duration      I/O Bytes       Tunnel/Dest IP
 0        ldu1     10.1.100.254    9             22099/43228     10.212.134.200
```

SSL VPN with certificate authentication

This is an example configuration of SSL VPN that requires users to authenticate using a client certificate. The client certificate is issued by the company Certificate Authority (CA). Each user is issued a certificate with their username in the subject.

There are two ways to configure certificate authentication:

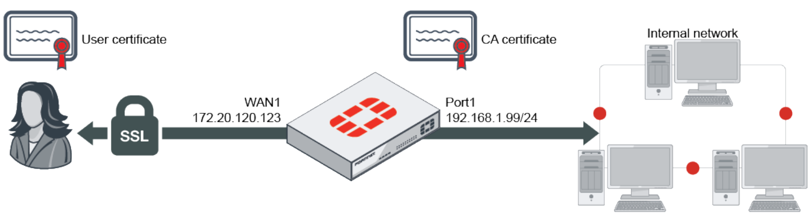
1. Using PKI users
2. Configuring the SSL VPN settings to require a client certificate

In this example, the server and client certificates are signed by the same Certificate Authority (CA).

Self-signed certificates are provided by default to simplify initial installation and testing. It is **HIGHLY** recommended that you acquire a signed certificate for your installation.

Continuing to use these certificates can result in your connection being compromised, allowing attackers to steal your information, such as credit card details.

For more information, please review the Use a non-factory SSL certificate for the SSL VPN portal on page 1619 and learn how to Procuring and importing a signed SSL certificate on page 2173.



Using PKI users

When using PKI users, the FortiGate authenticates the user based on there identity in the subject or the common name on the certificate. The certificate must be signed by a CA that is known by the FortiGate, either through the default CA certificates or through importing a CA certificate.

The user can either match a static subject or common name defined in the PKI user settings, or match an LDAP user in the LDAP server defined in the PKI user settings. Multi-factor authentication can also be enabled with the password as the second factor.

Configuring the SSL VPN settings to require a client certificate

Using this method, the user is authenticated based on their regular username and password, but SSL VPN will still require an additional certificate check. The client certificate only needs to be signed by a known CA in order to pass authentication.

This method can be configured by enabling *Require Client Certificate* (`reqclientcert`) in the SSL-VPN settings.

Configuration

In the following example, SSL VPN users are authenticated using the first method. A PKI user is configured with multi-factor authentication

Pre-requisites:

- The CA has already issued a client certificate to the user.
- The CA has issued a server certificate for the FortiGate's SSL VPN portal.
- The CA certificate is available to be imported on the FortiGate.

To configure SSL VPN in the GUI:

1. Install the server certificate. The server certificate allows the clients to authenticate the server and to encrypt the SSL VPN traffic.
   a. Go to *System > Feature Visibility* and ensure *Certificates* is enabled.
   b. Go to *System > Certificates* and select *Import > Local Certificate*.
      - Set *Type* to *Certificate*.
      - Choose the *Certificate file* and the *Key file* for your certificate, and enter the *Password*.
      - If required, you can change the *Certificate Name*.

   The server certificate now appears in the list of *Certificates*.
2. Install the CA certificate.

   The CA certificate is the certificate that signed both the server certificate and the user certificate. In this example, it is used to authenticate SSL VPN users.
   a. Go to *System > Certificates* and select *Import > CA Certificate*.
   b. Select *Local PC* and then select the certificate file.

   The CA certificate now appears in the list of *External CA Certificates*. In this example, it is called *CA_Cert_1*.
3. Configure PKI users and a user group.

   To use certificate authentication, use the CLI to create PKI users.

```
config user peer
    edit pki01
        set ca CA_Cert_1
        set subject "CN=User01"
    next
end
```

   Ensure that the subject matches the name of the user certificate. In this example, *User01*.
4. After you have create a PKI user, a new menu is added to the GUI:
   a. Go to *User & Authentication > PKI* to see the new user.
   b. Edit the user account.
   c. Enable *Two-factor authentication* and set a password for the account.
   d. Go to *User & Authentication > User Groups* and create a group called *sslvpngroup*.
   e. Add the PKI user *pki01* to the group.
5. Configure SSL VPN web portal.
   a. Go to *VPN > SSL-VPN Portals* to edit the *full-access* portal.

      This portal supports both web and tunnel mode.
   b. Disable *Enable Split Tunneling* so that all SSL VPN traffic goes through the FortiGate.

6. Configure SSL VPN settings.
   a. Go to *VPN > SSL-VPN Settings* and enable SSL-VPN.
   b. Set the *Listen on Interface(s)* to *wan1*.
   c. Set *Listen on Port* to *10443*.
   d. Set *Server Certificate* to the local certificate that was imported.
   e. Under *Authentication/Portal Mapping*, set default Portal *web-access* for *All Other Users/Groups*.
   f. Create new *Authentication/Portal Mapping* for group *sslvpngroup* mapping portal *full-access*.
7. Configure SSL VPN firewall policy.
   a. Go to *Policy & Objects > Firewall Policy*.
   b. Fill in the firewall policy name. In this example, *sslvpn certificate auth*.
   c. Incoming interface must be *SSL-VPN tunnel interface(ssl.root)*.
   d. Set the *Source Address* to *all* and *Source User* to *sslvpngroup*.
   e. Set the *Outgoing Interface* to the local network interface so that the remote user can access the internal network. In this example, *port1*.
   f. Set *Destination Address* to the internal protected subnet *192.168.1.0*.
   g. Set *Schedule* to *always*, *Service* to *ALL*, and *Action* to *Accept*.
   h. Enable *NAT*.
   i. Configure any remaining firewall and security options as needed.
   j. Click *OK*.

To configure SSL VPN in the CLI:

1. Configure the protected subnet:

```
config firewall address
    edit "192.168.1.0"
        set subnet 192.168.1.0 255.255.255.0
    next
end
```

2. Install the server certificate:

   The server certificate allows the clients to authenticate the server and to encrypt the SSL VPN traffic. While it is easier to install the server certificate in the GUI, the CLI can be used to import a p12 certificate from a TFTP server.

   To import a p12 certificate, put the certificate *server_certificate.p12* on your TFTP server, then run following command on the FortiGate:

```
execute vpn certificate local import tftp server_certificate.p12 <your tftp_server> p12
<your password for PKCS12 file>
```

   To check that the server certificate is installed:

```
show vpn certificate local server_certificate
```

3. Install the CA certificate:

   The CA certificate is the certificate that signed both the server certificate and the user certificate. In this example, it is used to authenticate SSL VPN users. While it is easier to install the CA certificate from GUI, the CLI can be used to import a CA certificates from a TFTP server.

   To import a CA certificate, put the CA certificate on your TFTP server, then run following command on the FortiGate:

```
execute vpn certificate ca import tftp <your CA certificate name> <your tftp server>
```

To check that a new CA certificate is installed:

```
show vpn certificate ca
```

4. Configure PKI users and a user group:

```
config user peer
    edit pki01
        set ca CA_Cert_1
        set subject "CN=User01"
        set two-factor enable
        set passwd **********
    next
end
```

```
config user group
    edit "sslvpngroup"
        set member "pki01"
    next
end
```

5. Configure SSL VPN web portal:

```
config vpn ssl web portal
    edit "full-access"
        set tunnel-mode enable
        set web-mode enable
        set ip-pools "SSLVPN_TUNNEL_ADDR1"
        set split-tunneling disable
     next
end
```

6. Configure SSL VPN settings:

```
config vpn ssl settings
    set servercert "server_certificate"
    set tunnel-ip-pools "SSLVPN_TUNNEL_ADDR1"
    set source-interface "wan1"
    set source-address "all"
    set default-portal "web-access"
    config authentication-rule
        edit 1
            set groups "sslvpngroup"
            set portal "full-access"
        next
    end
end
```

7. Configure one SSL VPN firewall policy to allow remote user to access the internal network:

```
config firewall policy
    edit 1
        set name "sslvpn web mode access"
        set srcintf "ssl.root"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "192.168.1.0"
        set groups “sslvpngroup”
        set action accept
        set schedule "always"
```

```
            set service "ALL"
            set nat enable
        next
    end
```

Installation

To use the user certificate, you must first install it on the user's PC. When the user tries to authenticate, the user certificate is checked against the CA certificate to verify that they match.

Every user should have a unique user certificate. This allows you to distinguish each user and revoke a specific user's certificate, such as if a user no longer has VPN access.

To install the user certificate on Windows 7, 8, and 10:

1. Double-click the certificate file to open the *Import Wizard*.
2. Use the *Import Wizard* to import the certificate into the *Personal store* of the current user.

To install the user certificate on Mac OS X:

1. Open the certificate file, to open *Keychain Access*.
2. Double-click the certificate.
3. Expand *Trust* and select *Always Trust*.

To see the results of tunnel connection:

1. Download FortiClient from www.forticlient.com.
2. Open the FortiClient Console and go to *Remote Access > Configure VPN*.
3. Add a new connection.
   - Set *VPN Type* to *SSL VPN*.
   - Set *Remote Gateway* to the IP of the listening FortiGate interface, in this example, *172.20.120.123*.
4. Select *Customize Port* and set it to *10443*.
5. Enable *Client Certificate* and select the authentication certificate.
6. Save your settings.
7. Use the credentials you've set up to connect to the SSL VPN tunnel.
   If the certificate is correct, you can connect.

To see the results of web portal:

1. In a web browser, log into the portal *http://172.20.120.123:10443*.
   A message requests a certificate for authentication.
2. Select the user certificate.
3. Enter your user credentials.
   If the certificate is correct, you can connect to the SSL VPN web portal.

To check the SSL VPN connection using the GUI:

1. Go to *VPN > Monitor > SSL-VPN Monitor* to verify the list of SSL users.
2. Go to *Log & Report > System Events* and select the *VPN Events* card to view the details for the SSL connection log.

To check the SSL VPN connection using the CLI:

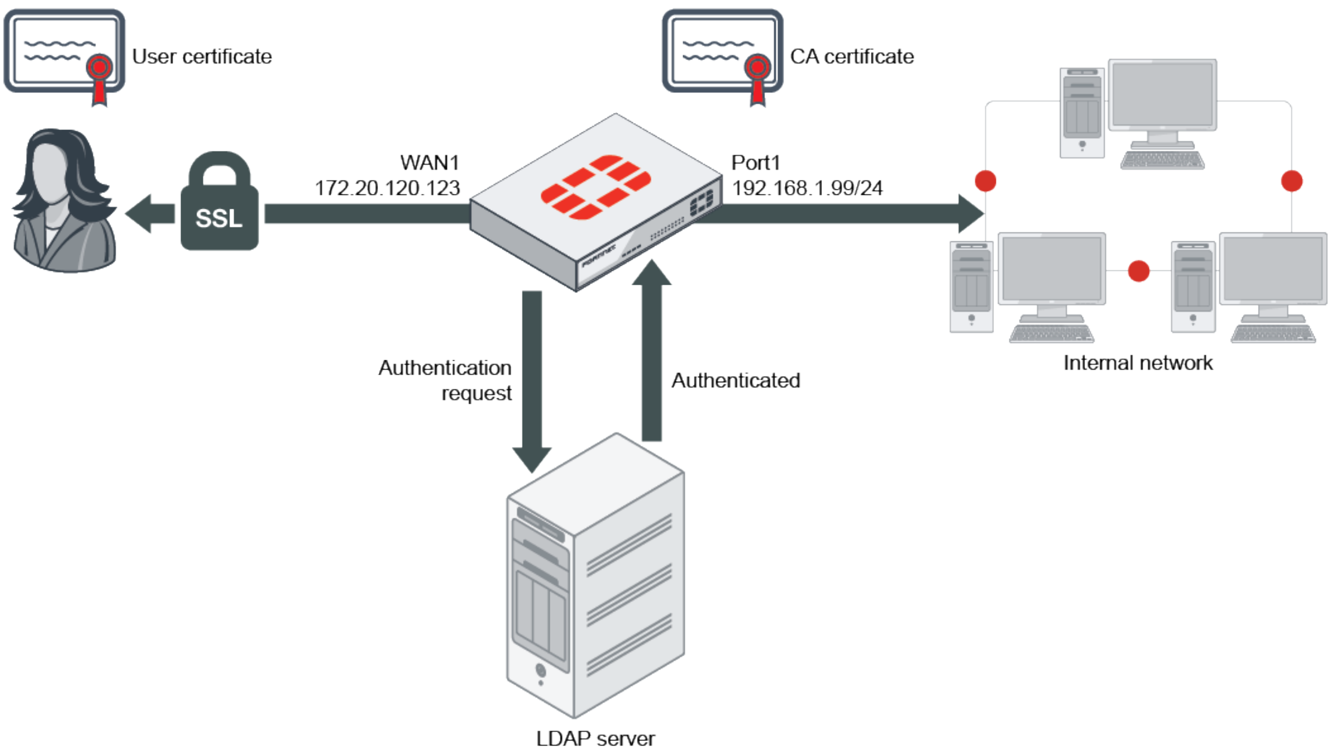
```
get vpn ssl monitor
SSL VPN Login Users:
 Index    User     Auth Type       Timeout          From      HTTP in/out    HTTPS in/out
 0        pki01,cn=User01            1(1)             229      10.1.100.254    0/0      0/0
 1        pki01,cn=User01            1(1)             291      10.1.100.254    0/0      0/0

SSL VPN sessions:
 Index    User     Source IP       Duration         I/O Bytes        Tunnel/Dest IP
 0        pki01,cn=User01            10.1.100.254     9        22099/43228     10.212.134.200
```

SSL VPN with LDAP-integrated certificate authentication

This is a sample configuration of SSL VPN that requires users to authenticate using a certificate with LDAP `UserPrincipalName` checking.

This sample uses Windows 2012R2 Active Directory acting as both the user certificate issuer, the certificate authority, and the LDAP server.

Sample topology



Sample configuration

WAN interface is the interface connected to ISP. This example shows static mode. You can also use DHCP or PPPoE mode. The SSL VPN connection is established over the WAN interface.

In this sample, the *User Principal Name* is included in the subject name of the issued certificate. This is the user field we use to search LDAP in the connection attempt.

To use the user certificate, you must first install it on the user's PC. When the user tries to authenticate, the user certificate is checked against the CA certificate to verify that they match.

Every user should have a unique user certificate. This allows you to distinguish each user and revoke a specific user's certificate, such as if a user no longer has VPN access.

To install the server certificate:

The server certificate is used for authentication and for encrypting SSL VPN traffic.

1. Go to *System > Feature Visibility* and ensure *Certificates* is enabled.
2. Go to *System > Certificates* and select *Import > Local Certificate*.
3. Set *Type* to *Certificate*.
4. Choose the *Certificate file* and the *Key file* for your certificate, and enter the *Password*.
5. If required, change the *Certificate Name*.
   The server certificate now appears in the list of *Certificates*.

To install the CA certificate:

The CA certificate is the certificate that signed both the server certificate and the user certificate. In this example, it is used to authenticate SSL VPN users.

1. Go to *System > Certificates* and select *Import > CA Certificate*.
2. Select *Local PC* and then select the certificate file.
   The CA certificate now appears in the list of *External CA Certificates*. In this example, it is called *CA_Cert_1*.

To configure SSL VPN using the GUI:

1. Configure the interface and firewall address. The port1 interface connects to the internal network.
   a. Go to *Network > Interfaces* and edit the *wan1* interface.
   b. Set *IP/Network Mask* to *172.20.120.123/255.255.255.0*.
   c. Edit *port1* interface and set *IP/Network Mask* to *192.168.1.99/255.255.255.0*.
   d. Click *OK*.
   e. Go to *Policy & Objects > Address* and create an address for internet subnet *192.168.1.0*.
2. Configure the LDAP server:
   a. Go to *User & Authentication > LDAP Servers* and click *Create New*.
   b. Specify *Name* and *Server IP/Name*.
   c. Set *Distinguished Name* to *dc=fortinet-fsso,dc=com*.
   d. Set *Bind Type* to *Regular*.
   e. Set *Username* to *cn=admin,ou=testing,dc=fortinet-fsso,dc=com*.
   f. Set *Password*.
   g. Click *OK*.
3. Configure PKI users and a user group:
   To use certificate authentication, use the CLI to create PKI users.

```
config user peer
    edit user1
```

```
        set ca CA_Cert_1
        set ldap-server "ldap-AD"
        set ldap-mode principal-name
    next
end
```

When you have create a PKI user, a new menu is added to the GUI:

a. Go to *User & Authentication* > *PKI* to see the new user.
b. Go to *User & Authentication* > *User* > *User Groups* and create a group *sslvpn-group*.
c. Add the PKI peer object you created as a local member of the group.
d. Add a remote group on the LDAP server and select the group of interest.
   You need these users to be members using the LDAP browser window.

4. Configure SSL VPN web portal:
   a. Go to *VPN* > *SSL-VPN Portals* to edit the *full-access* portal.
      This portal supports both web and tunnel mode.
   b. Disable *Enable Split Tunneling* so that all SSL VPN traffic goes through the FortiGate.
5. Configure SSL VPN settings:
   a. Go to *VPN* > *SSL-VPN Settings*.
   b. Select the *Listen on Interface(s)*, in this example, *wan1*.
   c. Set *Listen on Port* to *10443*.
   d. Set *Server Certificate* to the authentication certificate.
   e. Under *Authentication/Portal Mapping*, set default Portal *web-access* for *All Other Users/Groups*.
   f. Create new *Authentication/Portal Mapping* for group *sslvpn-group* mapping portal *full-access*.
6. Configure SSL VPN firewall policy:
   a. Go to *Policy & Objects* > *Firewall Policy*.
   b. Fill in the firewall policy name. In this example, *sslvpn certificate auth*.
   c. Incoming interface must be *SSL-VPN tunnel interface(ssl.root)*.
   d. Set the *Source Address* to *all* and *Source User* to *sslvpn-group*.
   e. Set the *Outgoing Interface* to the local network interface so that the remote user can access the internal network. In this example, *port1*.
   f. Set *Destination Address* to the internal protected subnet *192.168.1.0*.
   g. Set *Schedule* to *always*, *Service* to *ALL*, and *Action* to *Accept*.
   h. Enable *NAT*.
   i. Configure any remaining firewall and security options as desired.
   j. Click *OK*.

To configure SSL VPN using the CLI:

1. Configure the interface and firewall address:

```
config system interface
    edit "wan1"
        set vdom "root"
        set ip 172.20.120.123 255.255.255.0
    next
end
```

2. Configure internal interface and protected subnet, then connect the port1 interface to the internal network:

```
config system interface
    edit "port1"
        set vdom "root"
        set ip 192.168.1.99 255.255.255.0
    next
end
```

```
config firewall address
    edit "192.168.1.0"
        set subnet 192.168.1.0 255.255.255.0
    next
end
```

3. Configure the LDAP server:

```
config user ldap
    edit "ldap-AD"
        set server "172.18.60.206"
        set cnid "cn"
        set dn "dc=fortinet-fsso,dc=com"
        set type regular
        set username "cn=admin,ou=testing,dc=fortinet-fsso,dc=com"
        set password ldap-server-password
    next
end
```

4. Configure PKI users and a user group:

```
config user peer
    edit user1
        set ca CA_Cert_1
        set ldap-server "ldap-AD"
        set ldap-mode principal-name
    next
end
```

```
config user group
    edit "sslvpn-group"
        set member "ldap-AD" "user1"
        config match
            edit 1
                set server-name "ldap-AD"
                set group-name "CN=group3,OU=Testing,DC=Fortinet-FSSO,DC=COM"
            next
        end
    next
end
```

5. Configure SSL VPN web portal:

```
config vpn ssl web portal
    edit "full-access"
        set tunnel-mode enable
        set web-mode enable
        set ip-pools "SSLVPN_TUNNEL_ADDR1"
        set split-tunneling disable
    next
end
```

6. Configure SSL VPN settings:

```
config vpn ssl settings
    set servercert "server_certificate"
    set tunnel-ip-pools "SSLVPN_TUNNEL_ADDR1"
    set source-interface "wan1"
    set source-address "all"
    set default-portal "web-access"
    config authentication-rule
        edit 1
            set groups "sslvpn-group"
            set portal "full-access"
        next
    end
end
```

7. Configure one SSL VPN firewall policy to allow remote user to access the internal network:

```
config firewall policy
    edit 1
        set name "sslvpn web mode access"
        set srcintf "ssl.root"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "192.168.1.0"
        set groups “sslvpn-group”
        set action accept
        set schedule "always"
        set service "ALL"
        set nat enable
    next
end
```

To see the results of tunnel connection:

1. Download FortiClient from www.forticlient.com.
2. Open the FortiClient Console and go to *Remote Access* > *Configure VPN*.
3. Add a new connection.
   a. Set the connection name.
   b. Set *Remote Gateway* to the IP of the listening FortiGate interface, in this example, *172.20.120.123*.
   c. Select *Customize Port* and set it to *10443*.
   d. Enable *Client Certificate* and select the authentication certificate.
4. Save your settings.
   Connecting to the VPN only requires the user's certificate. It does not require username or password.

To see the results of web portal:

1. In a web browser, log into the portal *http://172.20.120.123:10443*.
   A message requests a certificate for authentication.
2. Select the user certificate.
   You can connect to the SSL VPN web portal.

To check the SSL VPN connection using the GUI:

1. Go to *Dashboard > Network* and expand the *SSL-VPN* widget to verify the user's connection.
2. Go to *Log & Report > VPN Events* to view the details of the SSL VPN connection event log.
3. Go to *Log & Report > Forward Traffic* to view the details of the SSL VPN traffic.

To check the SSL VPN connection using the CLI:

Below is a sample output of `diagnose debug application fnbamd -1` while the user connects. This is a shortened output sample of a few locations to show the important parts. This sample shows lookups to find the group memberships (three groups total) of the user and that the correct group being found results in a match.

```
[1148] fnbamd_ldap_recv-Response len: 16, svr: 172.18.60.206
[829] fnbamd_ldap_parse_response-Got one MESSAGE. ID:4, type:search-result
[864] fnbamd_ldap_parse_response-ret=0
[1386] __fnbamd_ldap_primary_grp_next-Auth accepted
[910] __ldap_rxtx-Change state to 'Done'
[843] __ldap_rxtx-state 23(Done)
[925] fnbamd_ldap_send-sending 7 bytes to 172.18.60.206
[937] fnbamd_ldap_send-Request is sent. ID 5
[753] __ldap_stop-svr 'ldap-AD'
[53] ldap_dn_list_del_all-Del CN=test3,OU=Testing,DC=Fortinet-FSSO,DC=COM
[399] ldap_copy_grp_list-copied CN=group3,OU=Testing,DC=Fortinet-FSSO,DC=COM
[399] ldap_copy_grp_list-copied CN=Domain Users,CN=Users,DC=Fortinet-FSSO,DC=COM
[2088] fnbamd_auth_cert_check-Matching group 'sslvpn-group'
[2007] __match_ldap_group-Matching server 'ldap-AD' - 'ldap-AD'
[2015] __match_ldap_group-Matching group 'CN=group3,OU=Testing,DC=Fortinet-FSSO,DC=COM' -
'CN=group3,OU=Testing,DC=Fortinet-FSSO,DC=COM'
[2091] fnbamd_auth_cert_check-Group 'sslvpn-group' matched
[2120] fnbamd_auth_cert_result-Result for ldap svr[0] 'ldap-AD' is SUCCESS
[2126] fnbamd_auth_cert_result-matched user 'test3', matched group 'sslvpn-group'
```

You can also use `diagnose firewall auth list` to validate that a firewall user entry exists for the SSL VPN user and is part of the right groups.

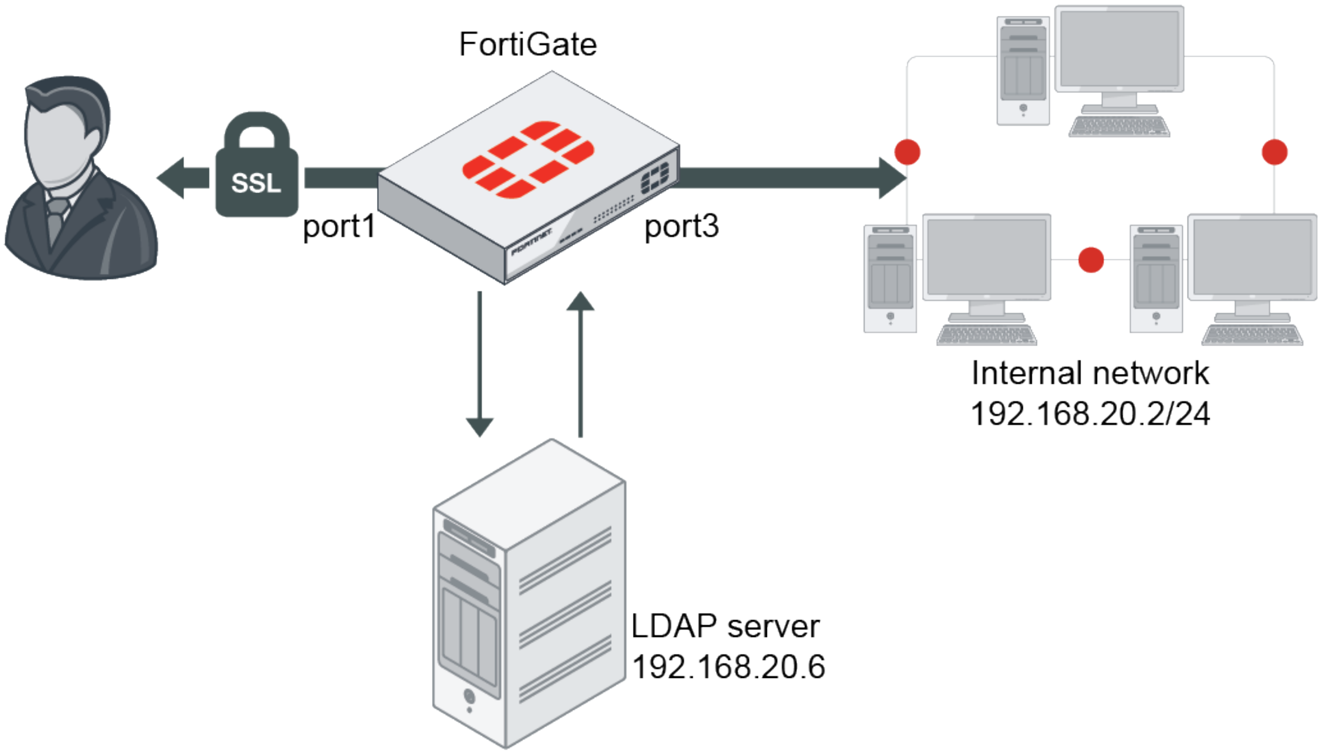
SSL VPN for remote users with MFA and user sensitivity

By default, remote LDAP and RADIUS user names are case sensitive. When a remote user object is applied to SSL VPN authentication, the user must type the exact case that is used in the user definition on the FortiGate.

Case sensitivity and accents can be ignored by disabling the `username-sensitivity` CLI command, allowing the remote user object to match any case or accents that the end user types in.

In this example, a remote user is configured with multi-factor authentication (MFA). The user group includes the LDAP user and server, and is applied to SSL VPN authentication and the policy.

Topology



Example configuration

To configure the LDAP server:

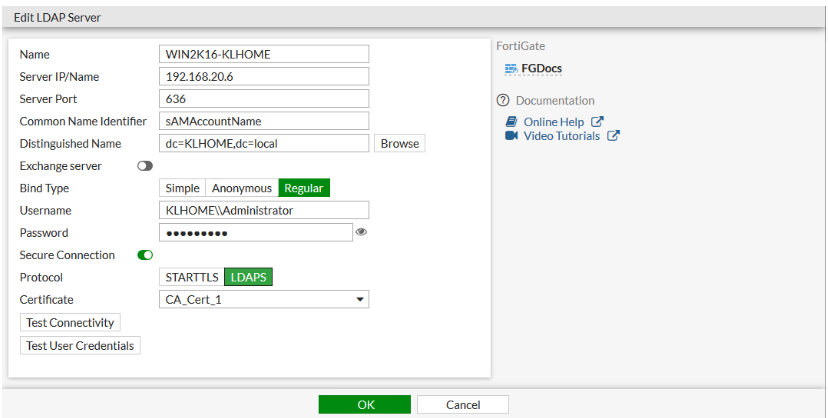
1. Generate and export a CA certificate from the AD server .
2. Import the CA certificate into FortiGate:
   a. Go to *System* > *Features Visibility* and ensure *Certificates* is enabled.
   b. Go to *System* > *Certificates* and select *Import* > *CA Certificate*.
   c. Select *Local PC* and then select the certificate file.
   The CA certificate now appears in the list of *External CA Certificates*. In this example, it is called *CA_Cert_1*.
   d. If you want, you can use CLI commands to rename the system-generated *CA_Cert_1* to be more descriptive:

```
config vpn certificate ca
    rename CA_Cert_1 to LDAPS-CA
end
```

3. Configure the LDAP user:
   a. Go to *User & Authentication* > *LDAP Servers* and click *Create New*.
   b. Configure the following options for this example:

Name	WIN2K16-KLHOME
Server IP/Name	192.168.20.6
Server Port	636

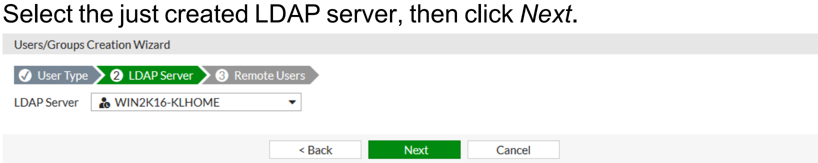
Common Name Identifier	sAMAccountName
Distinguished Name	dc=KLHOME,dc=local
Bind Type	Regular
Username	KLHOME\\Administrator
Password	*********
Secure Connection	Enable
Protocol	LDAPS
Certificate	CA_Cert_1 This is the CA certificate that you imported in step 2.



c. Click *OK*.

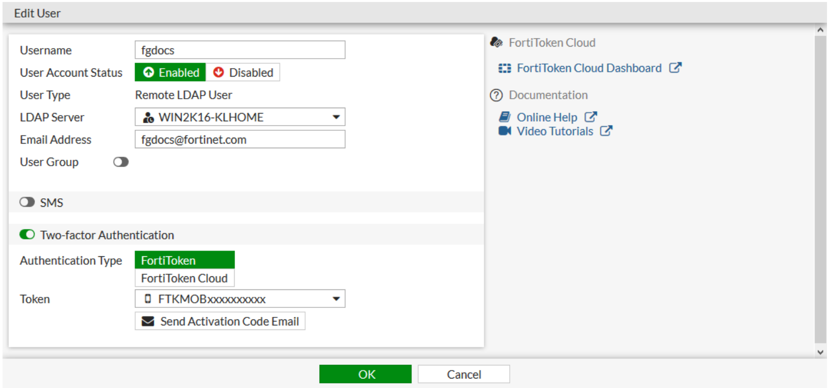
To configure an LDAP user with MFA:

1. Go to *User & Authentication > User Definition* and click *Create New*.
2. Select *Remote LDAP User*, then click *Next*.
3. Select the just created LDAP server, then click *Next*.



4. Right click to add the selected user, then click *Submit*.
5. Edit the user that you just created.
   The username will be pulled from the LDAP server with the same case as it has on the server.
6. Set the *Email Address* to the address that FortiGate will send the FortiToken to.
7. Enable *Two-factor Authentication*.
8. Set *Authentication Type* to *FortiToken*.

9. Set *Token* to a FortiToken device. See for more information.



10. Click *OK*.

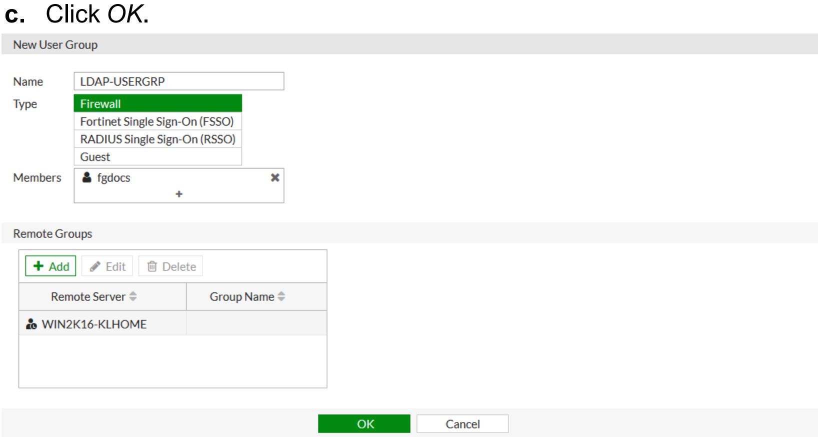
To disable case and accent sensitivity on the remote user:

This can only be configured in the CLI.

```
config user local
    edit "fgdocs"
        set type ldap
        set two-factor fortitoken
        set fortitoken "FTKMOBxxxxxxxxxx"
        set email-to "fgdocs@fortinet.com"
        set username-sensitivity disable
        set ldap-server "WIN2K16-KLHOME"
    next
end
```

To configure a user group with the remote user and the LDAP server:

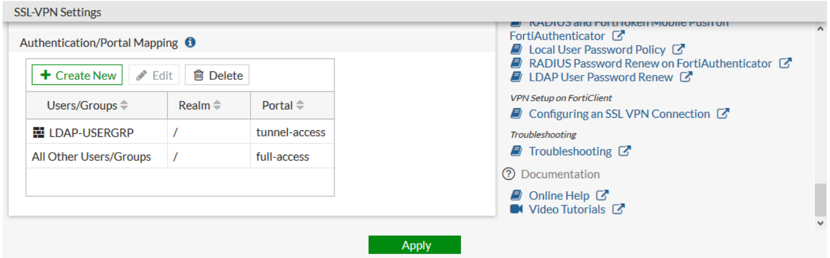
1. Go to *User & Authentication > User Groups* and click *Create New*.
2. Set the *Name* to *LDAP-USERGRP*.
3. Set *Members* to the just created remote user.
4. In the *Remote Groups* table, click *Add*:
   a. Set *Remote Server* to the LDAP server.
   b. Set the group or groups that apply, and right click to add them.
   c. Click *OK*.



5. Click *OK*.

To apply the user group to the SSL VPN portal:

1. Go to *VPN > SSL-VPN Settings*.
2. In the *Authentication/Portal Mapping* table, click *Create New*.
   a. Set *Users/Groups* to the just created user group.
   b. Configure the remaining settings as required.
   c. Click *OK*.

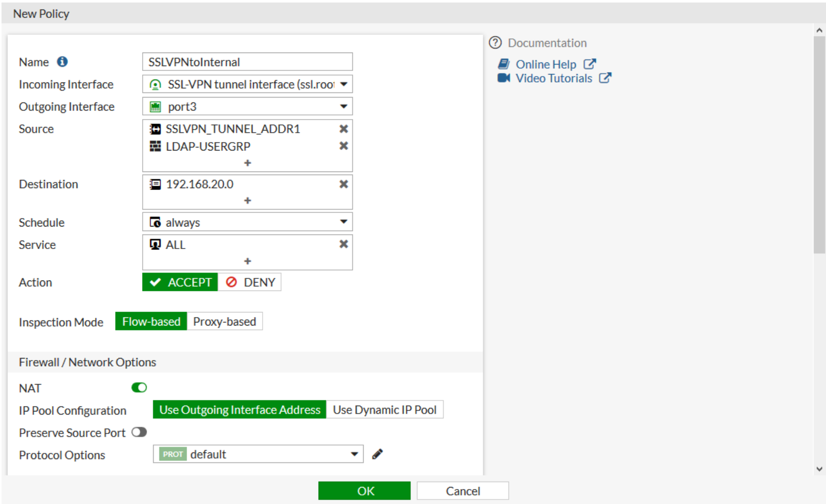


3. Click *Apply*.

To apply the user group to a firewall policy:

1. Go to *Policy & Objects > Firewall Policy* and click *Create New*.
2. Configure the following:

Name	SSLVPNtoInteral
Incoming Interface	SSL-VPN tunnel interface (ssl.root)
Outgoing Interface	port3
Source	Address - SSLVPN_TUNNEL_ADDR1 User - LDAP-USERGRP
Destination	The address of the internal network. In this case: 192.168.20.0.
Schedule	always
Service	ALL
Action	ACCEPT
NAT	Enabled



3. Configuring the remaining settings as required.
4. Click *OK*.

To configure this example in the CLI:

1. Configure the LDAP server:

```
config user ldap
    edit "WIN2K16-KLHOME"
        set server "192.168.20.6"
        set cnid "sAMAccountName"
        set dn "dc=KLHOME,dc=local"
        set type regular
        set username "KLHOME\\Administrator"
        set password *********
        set secure ldaps
        set ca-cert "CA_Cert_1"
        set port 636
    next
end
```

2. Configure an LDAP user with MFA and disable case and accent sensitivity on the remote user:

```
config user local
    edit "fgdocs"
        set type ldap
        set two-factor fortitoken
        set fortitoken "FTKMOBxxxxxxxxxx"
        set email-to "fgdocs@fortinet.com"
        set username-sensitivity disable
        set ldap-server "WIN2K16-KLHOME"
    next
end
```

3. Configure a user group with the remote user and the LDAP server:

```
config user group
    edit "LDAP-USERGRP"
        set member "fgdocs" "WIN2K16-KLHOME"
```

```
    next
end
```

4. Apply the user group to the SSL VPN portal:

```
config vpn ssl settings
    set servercert <server certificate>
    set tunnel-ip-pools "SSLVPN_TUNNEL_ADDR1"
    set source-interface "port1"
    set source-address "all"
    set default-portal "web-access"
    config authentication-rule
        edit 1
            set groups "LDAP-USERGRP"
            set portal "full-access"
        next
    end
end
```

5. Apply the user group to a firewall policy:

```
config firewall policy
    edit 5
        set name "SSLVPNtoInternal"
        set srcintf "ssl.root"
        set dstintf "port3"
        set srcaddr "SSLVPN_TUNNEL_ADDR1"
        set dstaddr "192.168.20.0"
        set action accept
        set schedule "always"
        set service "ALL"
        set groups "LDAP-USERGRP"
        set nat enable
    next
end
```

Verification

To setup the VPN connection:

1. Download FortiClient from www.forticlient.com.
2. Open the FortiClient Console and go to *Remote Access*.
3. Add a new connection:
   a. Set the connection name.
   b. Set *Remote Gateway* to the IP of the listening FortiGate interface.
   c. If required, set the *Customize Port*.
4. Save your settings.

To test the connection with case sensitivity disabled:

1. Connect to the VPN:
   a. Log in to the tunnel with the username, using the same case that it is on the FortiGate.
   b. When prompted, enter your FortiToken code.
      You should now be connected.

2. Check the web portal log in using the CLI:

```
# get vpn ssl monitor
SSL VPN Login Users:
 Index   User    Group   Auth Type      Timeout         From     HTTP in/out   HTTPS
in/out
 0       fgdocs          LDAP-USERGRP    16(1)           289     192.168.2.202 0/0
0/0

SSL VPN sessions:
 Index   User    Group   Source IP      Duration        I/O Bytes       Tunnel/Dest IP
 0       fgdocs          LDAP-USERGRP    192.168.2.202   45      99883/5572
10.212.134.200
```

3. Disconnect from the VPN connection.
4. Reconnect to the VPN:
   a. Log in to the tunnel with the username, using a different case than on the FortiGate.
   b. When prompted, enter your FortiToken code.
      You should now be connected.
5. Check the web portal log in using the CLI:

```
# get vpn ssl monitor
SSL VPN Login Users:
 Index   User    Group   Auth Type      Timeout         From     HTTP in/out   HTTPS
in/out
 0       FGDOCS          LDAP-USERGRP    16(1)           289     192.168.2.202 0/0
0/0

SSL VPN sessions:
 Index   User    Group   Source IP      Duration        I/O Bytes       Tunnel/Dest IP
 0       FGDOCS          LDAP-USERGRP    192.168.2.202   45      99883/5572
10.212.134.200
```

In both cases, the remote user is matched against the remote LDAP user object and prompted for multi-factor authentication.

To test the connection with case and accent sensitivity enabled:

1. Enable case and accent sensitivity for the user:

```
config user local
    edit "fgdocs"
        set username-sensitivity enable
    next
end
```

2. Connect to the VPN
   a. Log in to the tunnel with the username, using the same case that it is on the FortiGate.
   b. When prompted, enter your FortiToken code.
      You should now be connected.
3. Check the web portal log in using the CLI:

```
# get vpn ssl monitor
SSL VPN Login Users:
 Index   User    Group   Auth Type      Timeout         From     HTTP in/out   HTTPS
in/out
```

```
 0       fgdocs         LDAP-USERGRP    16(1)          289     192.168.2.202 0/0
0/0

SSL VPN sessions:
 Index    User    Group    Source IP      Duration        I/O Bytes       Tunnel/Dest IP
 0       fgdocs         LDAP-USERGRP    192.168.2.202   45      99883/5572
10.212.134.200
```

1. Disconnect from the VPN connection.
2. Reconnect to the VPN:
   a. Log in to the tunnel with the username, using a different case than on the FortiGate.
      You will not be prompted for your FortiToken code. You should now be connected.
3. Check the web portal log in using the CLI:

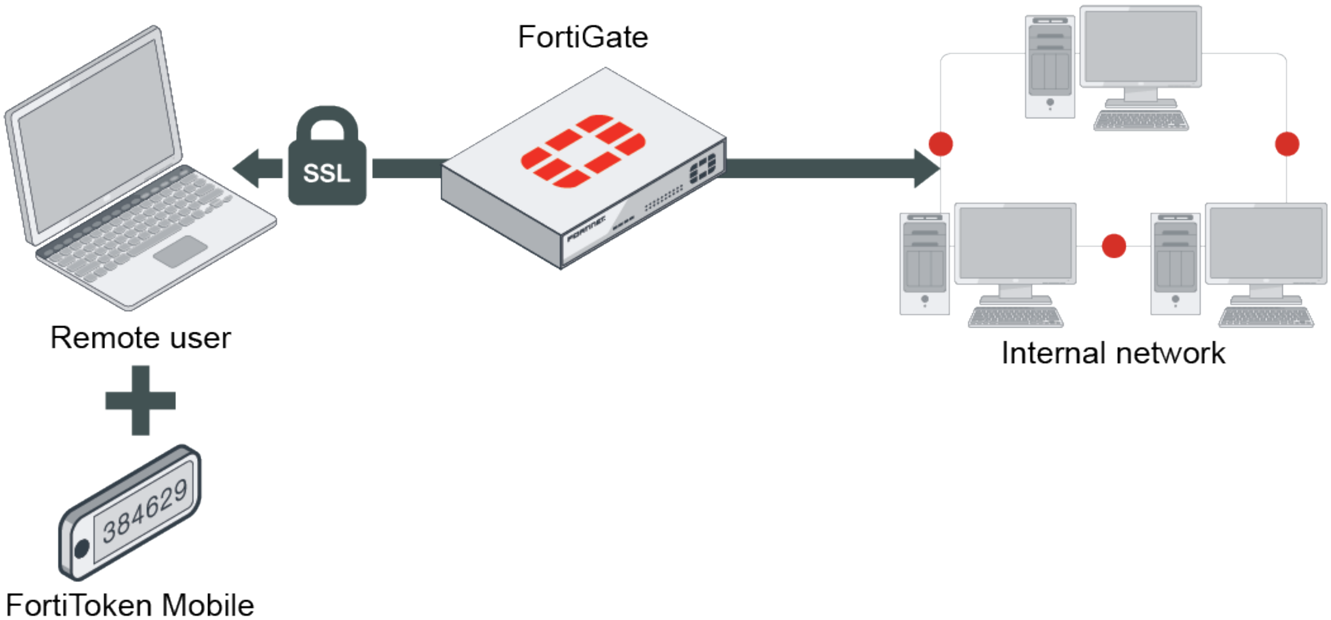
```
# get vpn ssl monitor
SSL VPN Login Users:
 Index    User    Group    Auth Type      Timeout         From     HTTP in/out   HTTPS
in/out
 0       FGdocs         LDAP-USERGRP    16(1)          289     192.168.2.202 0/0
0/0

SSL VPN sessions:
 Index    User    Group    Source IP      Duration        I/O Bytes       Tunnel/Dest IP
 0       FGdocs         LDAP-USERGRP    192.168.2.202   45      99883/5572
10.212.134.200
```

In this case, the user is allowed to log in without a FortiToken code because the entered user name did not match the name defined on the remote LDAP user object. Authentication continues to be evaluated against the LDAP server though, which is not case sensitive.

SSL VPN with FortiToken mobile push authentication

This is a sample configuration of SSL VPN that uses FortiToken mobile push two-factor authentication. If you enable push notifications, users can accept or deny the authentication request.

Sample topology



Sample configuration

WAN interface is the interface connected to ISP. This example shows static mode. You can also use DHCP or PPPoE mode. The SSL VPN connection is established over the WAN interface.

To configure SSL VPN using the GUI:

1. Configure the interface and firewall address. The port1 interface connects to the internal network.
   a. Go to *Network > Interfaces* and edit the *wan1* interface.
   b. Set *IP/Network Mask* to *172.20.120.123/255.255.255.0*.
   c. Edit *port1* interface and set *IP/Network Mask* to *192.168.1.99/255.255.255.0*.
   d. Click *OK*.
   e. Go to *Policy & Objects > Address* and create an address for internet subnet *192.168.1.0*.
2. Register FortiGate for FortiCare Support:
   To add or download a mobile token on FortiGate, FortiGate must be registered for FortiCare Support. If your FortiGate is registered, skip this step.
   a. Go to *Dashboard > Licenses*.
   b. Hover the pointer on *FortiCare Support* to check if FortiCare registered. If not, click it and select *Register*.
3. Add FortiToken mobile to FortiGate:
   If your FortiGate has FortiToken installed, skip this step.
   a. Go to *User & Authentication > FortiTokens* and click *Create New*.
   b. Select *Mobile Token* and type in *Activation Code*.
   c. Every FortiGate has two free mobile tokens. Go to *User & Authentication > FortiTokens* and click *Import Free Trial Tokens*.
4. Enable FortiToken mobile push:
   To use FTM-push authentication, use CLI to enable FTM-Push on the FortiGate.

a. Ensure `server-ip` is reachable from the Internet and enter the following CLI commands:

```
config system ftm-push
    set server-ip 172.20.120.123
    set status enable
end
```

b. Go to *Network > Interfaces*.
c. Edit the *wan1* interface.
d. Under *Administrative Access > IPv4*, select *FTM*.
e. Click *OK*.

5. Configure user and user group:
   a. Go to *User & Authentication > User Definition* to create a local user *sslvpnuser1*.
   b. Enter the user's *Email Address*.
   c. Enable *Two-factor Authentication* and select one mobile *Token* from the list,
   d. Enable *Send Activation Code* and select *Email*.
   e. Click *Next* and click *Submit*.
   f. Go to *User & Authentication > User Groups* to create a group *sslvpngroup* with the member *sslvpnuser1*.
6. Activate the mobile token:
   a. When the user *sslvpnuser1* is created, an email is sent to the user's email address. Follow the instructions to install your FortiToken mobile application on your device and activate your token.
7. Configure SSL VPN web portal:
   a. Go to *VPN > SSL-VPN Portals* to edit the *full-access* portal.
      This portal supports both web and tunnel mode.
   b. Disable *Enable Split Tunneling* so that all SSL VPN traffic goes through the FortiGate.
8. Configure SSL VPN settings:
   a. Go to *VPN > SSL-VPN Settings*.
   b. Select the *Listen on Interface(s)*, in this example, *wan1*.
   c. Set *Listen on Port* to *10443*.
   d. Set *Server Certificate* to the authentication certificate.
   e. Under *Authentication/Portal Mapping*, set default Portal *web-access* for *All Other Users/Groups*.
   f. Create new *Authentication/Portal Mapping* for group *sslvpngroup* mapping portal *full-access*.
9. Configure SSL VPN firewall policy:
   a. Go to *Policy & Objects > Firewall Policy*.
   b. Fill in the firewall policy name. In this example, *sslvpn certificate auth*.
   c. Incoming interface must be *SSL-VPN tunnel interface(ssl.root)*.
   d. Set the *Source Address* to *all* and *Source User* to *sslvpngroup*.
   e. Set the *Outgoing Interface* to the local network interface so that the remote user can access the internal network. In this example, *port1*.
   f. Set *Destination Address* to the internal protected subnet *192.168.1.0*.
   g. Set *Schedule* to *always*, *Service* to *ALL*, and *Action* to *Accept*.
   h. Enable *NAT*.
   i. Configure any remaining firewall and security options as desired.
   j. Click *OK*.

To configure SSL VPN using the CLI:

1. Configure the interface and firewall address.

```
config system interface
    edit "wan1"
        set vdom "root"
        set ip 172.20.120.123 255.255.255.0
    next
end
```

2. Configure internal interface and protected subnet, then connect the port1 interface to the internal network.

```
config system interface
    edit "port1"
        set vdom "root"
        set ip 192.168.1.99 255.255.255.0
    next
end
```

```
config firewall address
    edit "192.168.1.0"
        set subnet 192.168.1.0 255.255.255.0
    next
end
```

3. Register FortiGate for FortiCare Support.
To add or download a mobile token on FortiGate, FortiGate must be registered for FortiCare Support. If your FortiGate is registered, skip this step.

```
diagnose forticare direct-registration product-registration -a "your account@xxx.com" -p
"your password" -T "Your Country/Region" -R "Your Reseller" -e 1
```

4. Add FortiToken mobile to FortiGate:

```
execute fortitoken-mobile import <your FTM code>
```

If your FortiGate has FortiToken installed, skip this step.
Every FortiGate has two free mobile Tokens. You can download the free token.

```
execute fortitoken-mobile import 0000-0000-0000-0000-0000
```

5. Enable FortiToken mobile push:
   a. To use FTM-push authentication, ensure `server-ip` is reachable from the Internet and enable FTM-push in the FortiGate:

   ```
   config system ftm-push
       set server-ip 172.20.120.123
       set status enable
   end
   ```

   b. Enable FTM service on WAN interface:

   ```
   config system interface
       edit "wan1"
           append allowaccess ftm
       next
   end
   ```

6. Configure user and user group:

```
config user local
    edit "sslvpnuser1"
        set type password
        set two-factor fortitoken
        set fortitoken <select mobile token for the option list>
        set email-to <user's email address>
        set passwd <user's password>
    next
end
config user group
    edit "sslvpngroup"
        set member "sslvpnuser1"
    next
end
```

7. Activate the mobile token.
When the user *sslvpnuser1* is created, an email is sent to the user's email address. Follow the instructions to install your FortiToken mobile application on your device and activate your token.
8. Configure SSL VPN web portal:

```
config vpn ssl web portal
    edit "full-access"
        set tunnel-mode enable
        set web-mode enable
        set ip-pools "SSLVPN_TUNNEL_ADDR1"
        set split-tunneling disable
    next
end
```

9. Configure SSL VPN settings:

```
config vpn ssl settings
    set servercert "server_certificate"
    set tunnel-ip-pools "SSLVPN_TUNNEL_ADDR1"
    set source-interface "wan1"
    set source-address "all"
    set default-portal "web-access"
    config authentication-rule
        edit 1
            set groups "sslvpngroup"
            set portal "full-access"
        next
    end
end
```

10. Configure one SSL VPN firewall policy to allow remote user to access the internal network:

```
config firewall policy
    edit 1
        set name "sslvpn web mode access"
        set srcintf "ssl.root"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "192.168.1.0"
        set groups “sslvpngroup”
        set action accept
```

```
        set schedule "always"
        set service "ALL"
        set nat enable
    next
end
```

To see the results of web portal:

1. From a remote device, use a web browser to log into the SSL VPN web portal *http://172.20.120.123:10443*.
2. Log in using the *sslvpnuser1* credentials.
   The FortiGate pushes a login request notification through the FortiToken mobile application.
3. Check your mobile device and select *Approve*.
   When the authentication is approved, *sslvpnuser1* is logged into the SSL VPN portal.
4. On the FortiGate, go to *Dashboard > Network* and expand the *SSL-VPN* widget to verify the user's connection.

To see the results of tunnel connection:

1. Download FortiClient from www.forticlient.com.
2. Open the FortiClient Console and go to *Remote Access > Configure VPN*.
3. Add a new connection:
   a. Set the connection name.
   b. Set *Remote Gateway* to the IP of the listening FortiGate interface, in this example, *172.20.120.123*.
   c. Select *Customize Port* and set it to *10443*.
4. Save your settings.
5. Log in using the *sslvpnuser1* credentials and click *FTM Push*.
   The FortiGate pushes a login request notification through the FortiToken mobile application.
6. Check your mobile device and select *Approve*.
   When the authentication is approved, *sslvpnuser1* is logged into the SSL VPN tunnel.

To check the SSL VPN connection using the GUI:

1. Go to *Dashboard > Network* and expand the *SSL-VPN* widget to verify the user's connection.
2. Go to *Log & Report > Forward Traffic* to view the details of the SSL VPN traffic.

To check the web portal login using the CLI:

```
get vpn ssl monitor
SSL VPN Login Users:
 Index    User           Auth Type    Timeout    From           HTTP in/out    HTTPS in/out
 0        sslvpnuser1    1(1)         229        10.1.100.254   0/0            0/0

SSL VPN sessions:
 Index    User    Source IP      Duration        I/O Bytes       Tunnel/Dest IP
```

To check the tunnel login using the CLI:

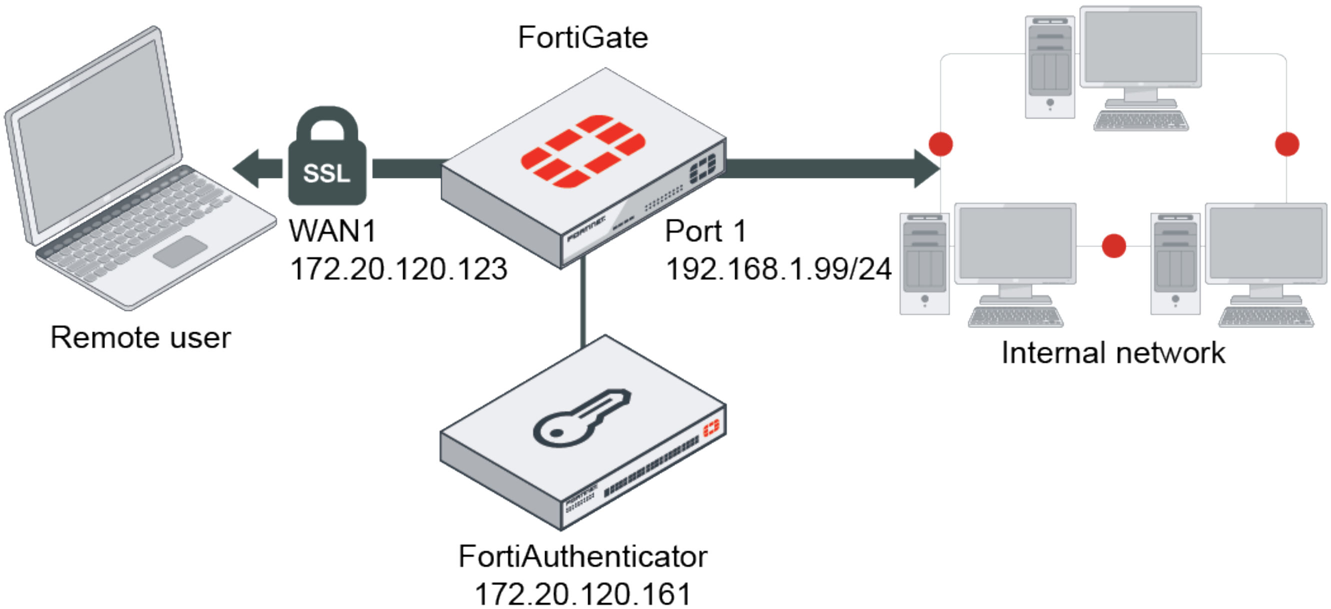
```
get vpn ssl monitor
SSL VPN Login Users:
 Index    User          Auth Type    Timeout    From           HTTP in/out  HTTPS in/out
 0        sslvpnuser1   1(1)         291        10.1.100.254   0/0          0/0
```

```
SSL VPN sessions:
 Index    User            Source IP       Duration    I/O Bytes       Tunnel/Dest IP
 0        sslvpnuser1     10.1.100.254    9           22099/43228     10.212.134.200
```

SSL VPN with RADIUS on FortiAuthenticator

This is a sample configuration of SSL VPN that uses FortiAuthenticator as a RADIUS authentication server.

Sample topology



Sample configuration

WAN interface is the interface connected to ISP. This example shows static mode. You can also use DHCP or PPPoE mode. The SSL VPN connection is established over the WAN interface.

To configure FortiAuthenticator using the GUI:

1. Create a user on the FortiAuthenticator.
   a. On the FortiAuthenticator, go to *Authentication* > *User Management* > *Local Users* to create a user *sslvpnuser1*.
   b. Enable *Allow RADIUS authentication* and click *OK* to access additional settings.
   c. Go to *Authentication* > *User Management* > *User Groups* to create a group *sslvpngroup*.
   d. Add *sslvpnuser1* to the group by moving the user from *Available users* to *Selected users*.
2. Create the RADIUS client (FortiGate) on the FortiAuthenticator.
   a. On the FortiAuthenticator, go to *Authentication* > *RADIUS Service* > *Clients* to add the FortiGate as a RADIUS client *OfficeServer*).
   b. Enter the FortiGate IP address and set a *Secret*.
      The secret is a pre-shared secure password that the FortiGate uses to authenticate to the FortiAuthenticator.
   c. Set *Realms* to *local* | *Local users*.

To configure SSL VPN using the GUI:

1. Configure the interface and firewall address. The port1 interface connects to the internal network.
   a. Go to *Network > Interfaces* and edit the *wan1* interface.
   b. Set *IP/Network Mask* to *172.20.120.123/255.255.255.0*.
   c. Edit *port1* interface and set *IP/Network Mask* to *192.168.1.99/255.255.255.0*.
   d. Click *OK*.
   e. Go to *Policy & Objects > Addresses* and create an address for internal subnet *192.168.1.0*.
2. Create a RADIUS user and user group .
   a. On the FortiGate, go to *User & Authentication > RADIUS Servers* to create a user to connect to the RADIUS server (FortiAuthenticator).
   b. For *Name*, use *FAC-RADIUS*.
   c. Enter the IP address of the FortiAuthenticator, and enter the *Secret* created above.
   d. Click *Test Connectivity* to ensure you can connect to the RADIUS server.
   e. Select *Test User Credentials* and enter the credentials for *sslvpnuser1*.
      The FortiGate can now connect to the FortiAuthenticator as the RADIUS client.
   f. Go to *User & Authentication > User Groups* and click *Create New* to map authenticated remote users to a user group on the FortiGate.
   g. For *Name*, use *SSLVPNGroup*.
   h. In *Remote Groups*, click *Add*.
   i. In the *Remote Server* dropdown list, select *FAC-RADIUS*.
   j. Leave the *Groups* field blank.
3. Configure SSL VPN web portal.
   a. Go to *VPN > SSL-VPN Portals* to edit the *full-access* portal.
      This portal supports both web and tunnel mode.
   b. Disable *Enable Split Tunneling* so that all SSL VPN traffic goes through the FortiGate.
4. Configure SSL VPN settings.
   a. Go to *VPN > SSL-VPN Settings*.
   b. Select the *Listen on Interface(s)*, in this example, *wan1*.
   c. Set *Listen on Port* to *10443*.
   d. Set *Server Certificate* to the authentication certificate.
   e. Under *Authentication/Portal Mapping*, set default Portal *web-access* for *All Other Users/Groups*.
   f. Create new *Authentication/Portal Mapping* for group *sslvpngroup* mapping portal *full-access*.
5. Configure SSL VPN firewall policy.
   a. Go to *Policy & Objects > Firewall Policy*.
   b. Fill in the firewall policy name. In this example, *sslvpn certificate auth*.
   c. *Incoming Interface* must be *SSL-VPN tunnel interface(ssl.root)*.
   d. Set the *Outgoing Interface* to the local network interface so that the remote user can access the internal network. In this example: *port1*.
   e. Set the *Source > Address* to *all* and *Source > User* to *sslvpngroup*.
   f. Set *Destination > Address* to the internal protected subnet *192.168.1.0*.
   g. Set *Schedule* to *always*, *Service* to *ALL*, and *Action* to *Accept*.
   h. Enable *NAT*.
   i. Configure the remaining options as required.
   j. Click *OK*.

To configure SSL VPN using the CLI:

1. Configure the interface and firewall address.

```
config system interface
    edit "wan1"
        set vdom "root"
        set ip 172.20.120.123 255.255.255.0
    next
end
```

2. Configure internal interface and protected subnet, then connect the port1 interface to the internal network.

```
config system interface
    edit "port1"
        set vdom "root"
        set ip 192.168.1.99 255.255.255.0
    next
end
```

```
config firewall address
    edit "192.168.1.0"
        set subnet 192.168.1.0 255.255.255.0
    next
end
```

3. Create a RADIUS user and user group.

```
config user radius
    edit "FAC-RADIUS"
        set server "172.20.120.161"
        set secret <FAC client secret>
    next
end
```

```
config user group
    edit "sslvpngroup"
        set member "FAC-RADIUS"
    next
end
```

4. Configure SSL VPN web portal.

```
config vpn ssl web portal
    edit "full-access"
        set tunnel-mode enable
        set web-mode enable
        set ip-pools "SSLVPN_TUNNEL_ADDR1"
        set split-tunneling disable
    next
end
```

5. Configure SSL VPN settings.

```
config vpn ssl settings
    set servercert "server_certificate"
    set tunnel-ip-pools "SSLVPN_TUNNEL_ADDR1"
    set source-interface "wan1"
    set source-address "all"
    set default-portal "web-access"
```

```
    config authentication-rule
        edit 1
            set groups "sslvpngroup"
            set portal "full-access"
        next
    end
end
```

6. Configure one SSL VPN firewall policy to allow remote user to access the internal network.

```
config firewall policy
    edit 1
        set name "sslvpn web mode access"
        set srcintf "ssl.root"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "192.168.1.0"
        set groups "sslvpngroup"
        set action accept
        set schedule "always"
        set service "ALL"
        set nat enable
    next
end
```

To see the results of web portal:

1. From a remote device, use a web browser to log into the SSL VPN web portal *http://172.20.120.123:10443*.
2. Log in using the *sslvpnuser1* credentials.
3. On the FortiGate, go to *Dashboard* > *Network* and expand the *SSL-VPN* widget to verify the user's connection.

To see the results of tunnel connection:

1. Download FortiClient from www.forticlient.com.
2. Open the FortiClient Console and go to *Remote Access* > *Configure VPN*.
3. Add a new connection.
   - Set the connection name.
   - Set *Remote Gateway* to *172.20.120.123*.
4. Select *Customize Port* and set it to *10443*.
5. Save your settings.
6. Log in using the *sslvpnuser1* credentials and check that you are logged into the SSL VPN tunnel.

To check the SSL VPN connection using the GUI:

1. Go to *Dashboard* > *Network* and expand the *SSL-VPN* widget to verify the user's connection.
2. Go to *Log & Report* > *Forward Traffic* to view the details of the SSL VPN traffic.

To check the web portal login using the CLI:

```
get vpn ssl monitor
SSL VPN Login Users:
 Index   User          Auth Type    Timeout    From           HTTP in/out  HTTPS in/out
 0       sslvpnuser1   1(1)         229        10.1.100.254   0/0          0/0
```

```
SSL VPN sessions:
 Index    User     Source IP       Duration         I/O Bytes        Tunnel/Dest IP
```

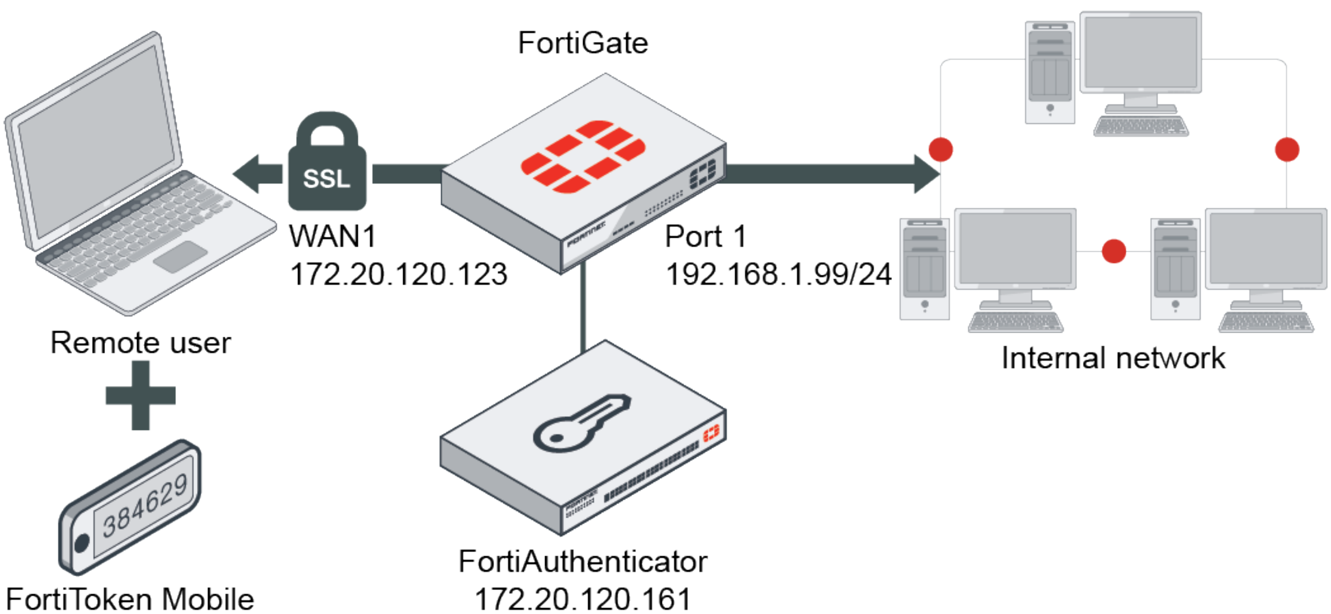
To check the tunnel login using the CLI:

```
get vpn ssl monitor
SSL VPN Login Users:
 Index    User           Auth Type    Timeout     From           HTTP in/out   HTTPS in/out
 0        sslvpnuser1    1(1)         291         10.1.100.254   0/0           0/0

SSL VPN sessions:
 Index    User           Source IP       Duration      I/O Bytes       Tunnel/Dest IP
 0        sslvpnuser1    10.1.100.254    9             22099/43228     10.212.134.200
```

SSL VPN with RADIUS and FortiToken mobile push on FortiAuthenticator

This is a sample configuration of SSL VPN that uses FortiAuthenticator as a RADIUS authentication server and FortiToken mobile push two-factor authentication. If you enable push notifications, users can accept or deny the authentication request.

Sample topology



Sample configuration

WAN interface is the interface connected to ISP. This example shows static mode. You can also use DHCP or PPPoE mode. The SSL VPN connection is established over the WAN interface.

To configure FortiAuthenticator using the GUI:

1. On the FortiAuthenticator, go to *System > Administration > System Access* and configure a *Public IP/FQDN for FortiToken Mobile*. If the FortiAuthenticator is behind a firewall, the public IP/FQDN will be an IP/port forwarding rule directed to one of the FortiAuthenticator interfaces. The interface that receives the approve/deny FTM push responses must have the *FortiToken Mobile API* service enabled.
2. Add a FortiToken mobile license on the FortiAuthenticator:
   a. Go to *Authentication > User Management > FortiTokens*.
   b. Click *Create New*.
   c. Set *Token type* to *FortiToken Mobile* and enter the *FortiToken Activation codes*.
3. Create the RADIUS client (FortiGate) on the FortiAuthenticator:
   a. Go to *Authentication > RADIUS Service > Clients* to add the FortiGate as a RADIUS client *OfficeServer*).
   b. Enter the FortiGate IP address and set a *Secret*.
      The secret is a pre-shared secure password that the FortiGate uses to authenticate to the FortiAuthenticator.
   c. Set *Authentication method* to *Enforce two-factor authentication*.
   d. Select *Enable FortiToken Mobile push notifications authentication*.
   e. Set *Realms* to *local | Local users*.
4. Create a user and assign FortiToken mobile to the user on the FortiAuthenticator:
   a. Go to *Authentication > User Management > Local Users* to create a user *sslvpnuser1*.
   b. Enable *Allow RADIUS authentication* and click *OK* to access additional settings.
   c. Enable *Token-based authentication* and select to deliver the token code by *FortiToken*.
   d. Select the FortiToken added from the FortiToken Mobile dropdown menu.
   e. Set *Delivery method* to *Email* and fill in the *User Information* section.
   f. Go to *Authentication > User Management > User Groups* to create a group *sslvpngroup*.
   g. Add *sslvpnuser1* to the group by moving the user from *Available users* to *Selected users*.
5. Install the FortiToken mobile application on your Android or iOS smartphone.
   The FortiAuthenticator sends the FortiToken mobile activation to the user’s email address.
6. Activate the FortiToken mobile through the FortiToken mobile application by entering the activation code or scanning the QR code.

To configure SSL VPN using the GUI:

1. Configure the interface and firewall address. The port1 interface connects to the internal network.
   a. Go to *Network > Interfaces* and edit the *wan1* interface.
   b. Set *IP/Network Mask* to *172.20.120.123/255.255.255.0*.
   c. Edit *port1* interface and set *IP/Network Mask* to *192.168.1.99/255.255.255.0*.
   d. Click *OK*.
   e. Go to *Policy & Objects > Address* and create an address for internet subnet *192.168.1.0*.
2. Create a RADIUS user and user group:
   a. On the FortiGate, go to *User & Authentication > RADIUS Servers* to create a user to connect to the RADIUS server (FortiAuthenticator).
   b. For *Name*, use *FAC-RADIUS*.
   c. Enter the IP address of the FortiAuthenticator, and enter the *Secret* created above.
   d. Click *Test Connectivity* to ensure you can connect to the RADIUS server.
   e. Select *Test User Credentials* and enter the credentials for *sslvpnuser1*.
      The FortiGate can now connect to the FortiAuthenticator as the RADIUS client.

f. Go to *User & Authentication > User Groups* and click *Create New* to map authenticated remote users to a user group on the FortiGate.
g. For *Name*, use *SSLVPNGroup*.
h. In *Remote Groups*, click *Add*.
i. In the *Remote Server* dropdown list, select *FAC-RADIUS*.
j. Leave the *Groups* field blank.

3. Configure SSL VPN web portal:
   a. Go to *VPN > SSL-VPN Portals* to edit the *full-access* portal.
      This portal supports both web and tunnel mode.
   b. Disable *Enable Split Tunneling* so that all SSL VPN traffic goes through the FortiGate.
4. Configure SSL VPN settings:
   a. Go to *VPN > SSL-VPN Settings*.
   b. Select the *Listen on Interface(s)*, in this example, *wan1*.
   c. Set *Listen on Port* to *10443*.
   d. Set *Server Certificate* to the authentication certificate.
   e. Under *Authentication/Portal Mapping*, set default Portal *web-access* for *All Other Users/Groups*.
   f. Create new *Authentication/Portal Mapping* for group *sslvpngroup* mapping portal *full-access*.
5. Configure SSL VPN firewall policy:
   a. Go to *Policy & Objects > Firewall Policy*.
   b. Fill in the firewall policy name. In this example, *sslvpn certificate auth*.
   c. Incoming interface must be *SSL-VPN tunnel interface(ssl.root)*.
   d. Set the *Source Address* to *all* and *Source User* to *sslvpngroup*.
   e. Set the *Outgoing Interface* to the local network interface so that the remote user can access the internal network. In this example: *port1*.
   f. Set *Destination Address* to the internal protected subnet *192.168.1.0*.
   g. Set *Schedule* to *always*, *Service* to *ALL*, and *Action* to *Accept*.
   h. Enable *NAT*.
   i. Configure any remaining firewall and security options as desired.
   j. Click *OK*.

To configure SSL VPN using the CLI:

1. Configure the interface and firewall address:

```
config system interface
    edit "wan1"
        set vdom "root"
        set ip 172.20.120.123 255.255.255.0
    next
end
```

2. Configure internal interface and protected subnet, then connect the port1 interface to the internal network:

```
config system interface
    edit "port1"
        set vdom "root"
        set ip 192.168.1.99 255.255.255.0
    next
end
```

```
config firewall address
    edit "192.168.1.0"
        set subnet 192.168.1.0 255.255.255.0
    next
end
```

3. Create a RADIUS user and user group:

```
config user radius
    edit "FAC-RADIUS"
        set server "172.20.120.161"
        set secret <FAC client secret>
    next
end
```

```
config user group
    edit "sslvpngroup"
        set member "FAC-RADIUS"
    next
end
```

4. Configure SSL VPN web portal:

```
config vpn ssl web portal
    edit "full-access"
        set tunnel-mode enable
        set web-mode enable
        set ip-pools "SSLVPN_TUNNEL_ADDR1"
        set split-tunneling disable
     next
end
```

5. Configure SSL VPN settings:

```
config vpn ssl settings
    set servercert "server_certificate"
    set tunnel-ip-pools "SSLVPN_TUNNEL_ADDR1"
    set source-interface "wan1"
    set source-address "all"
    set default-portal "web-access"
    config authentication-rule
        edit 1
            set groups "sslvpngroup"
            set portal "full-access"
        next
    end
end
```

6. Configure one SSL VPN firewall policy to allow remote user to access the internal network:

```
config firewall policy
    edit 1
        set name "sslvpn web mode access"
        set srcintf "ssl.root"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "192.168.1.0"
        set groups “sslvpngroup”
        set action accept
```

```
            set schedule "always"
            set service "ALL"
            set nat enable
        next
    end
```

To see the results of web portal:

1. From a remote device, use a web browser to log into the SSL VPN web portal *http://172.20.120.123:10443*.
2. Log in using the *sslvpnuser1* credentials.
   The FortiAuthenticator pushes a login request notification through the FortiToken Mobile application.
3. Check your mobile device and select *Approve*.
   When the authentication is approved, *sslvpnuser1* is logged into the SSL VPN portal.
4. On the FortiGate, go to *Dashboard > Network* and expand the *SSL-VPN* widget to verify the user's connection.

To see the results of tunnel connection:

1. Download FortiClient from www.forticlient.com.
2. Open the FortiClient Console and go to *Remote Access > Configure VPN*.
3. Add a new connection:
   a. Set the connection name.
   b. Set *Remote Gateway* to the IP of the listening FortiGate interface, in this example: *172.20.120.123*.
   c. Select *Customize Port* and set it to *10443*.
4. Save your settings.
5. Log in using the *sslvpnuser1* credentials and click *FTM Push*.
   The FortiAuthenticator pushes a login request notification through the FortiToken Mobile application.
6. Check your mobile device and select *Approve*.
   When the authentication is approved, *sslvpnuser1* is logged into the SSL VPN tunnel.

To check the SSL VPN connection using the GUI:

1. Go to *Dashboard > Network* and expand the *SSL-VPN* widget to verify the user's connection.
2. Go to *Log & Report > Forward Traffic* to view the details of the SSL VPN traffic.

To check the web portal login using the CLI:

```
get vpn ssl monitor
SSL VPN Login Users:
 Index    User          Auth Type    Timeout    From            HTTP in/out    HTTPS in/out
 0        sslvpnuser1   1(1)         229        10.1.100.254    0/0            0/0

SSL VPN sessions:
 Index    User    Source IP      Duration         I/O Bytes       Tunnel/Dest IP
```

To check the tunnel login on CLI:

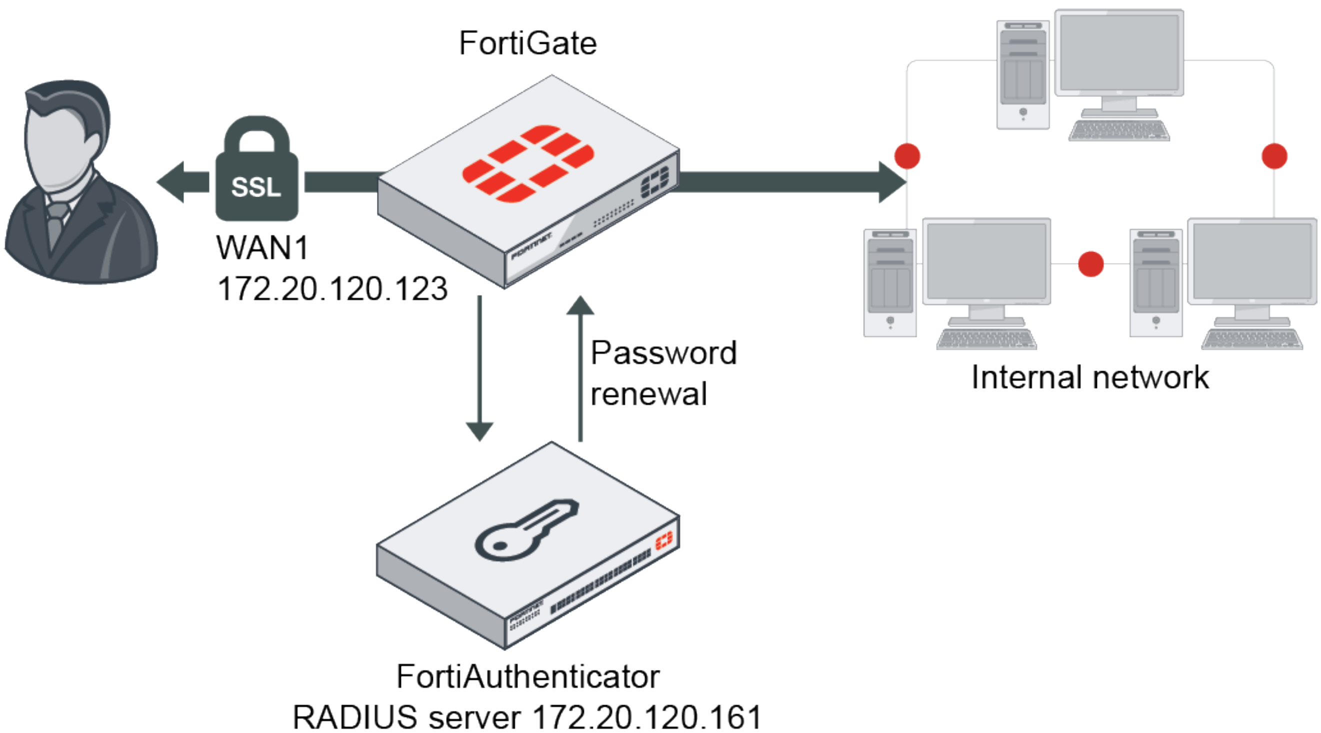
```
get vpn ssl monitor
SSL VPN Login Users:
 Index    User          Auth Type     Timeout    From            HTTP in/out   HTTPS in/out
 0        sslvpnuser1   1(1)          291        10.1.100.254    0/0           0/0
```

```
SSL VPN sessions:
 Index    User           Source IP      Duration     I/O Bytes      Tunnel/Dest IP
 0        sslvpnuser1    10.1.100.254   9            22099/43228    10.212.134.200
```

SSL VPN with RADIUS password renew on FortiAuthenticator

This is a sample configuration of SSL VPN for RADIUS users with *Force Password Change on next logon*. In this example, the RADIUS server is a FortiAuthenticator. A user *test1* is configured on FortiAuthenticator with *Force password change on next logon*.

Sample topology



Sample configuration

WAN interface is the interface connected to ISP. This example shows static mode. You can also use DHCP or PPPoE mode. The SSL VPN connection is established over the WAN interface.

To configure SSL VPN using the GUI:

1. Configure the interface and firewall address. The port1 interface connects to the internal network.
   a. Go to *Network > Interfaces* and edit the *wan1* interface.
   b. Set *IP/Network Mask* to *172.20.120.123/255.255.255.0*.
   c. Edit *port1* interface and set *IP/Network Mask* to *192.168.1.99/255.255.255.0*.
   d. Click *OK*.
   e. Go to *Policy & Objects > Address* and create an address for internet subnet *192.168.1.0*.

2. Create a RADIUS user.
   a. Go to *User & Authentication > RADIUS Servers* to create a user.
   b. Set *Authentication method* to *MS-CHAP-v2*.
   c. Enter the *IP/Name* and *Secret*.
   d. Click *Create*.
      Password renewal only works with the MS-CHAP-v2 authentication method.
   e. To enable the `password-renew` option, use these CLI commands.

```
config user radius
    edit "fac"
        set server "172.20.120.161"
        set secret <fac radius password>
        set auth-type ms_chap_v2
        set password-renewal enable
    next
end
```

3. Configure user group.
   a. Go to *User & Authentication > User Groups* to create a user group.
   b. For the *Name*, enter *fac-group*.
   c. In *Remote Groups*, click *Add* to add *Remote Server* you just created.
4. Configure SSL VPN web portal.
   a. Go to *VPN > SSL-VPN Portals* to edit the *full-access* portal.
      This portal supports both web and tunnel mode.
   b. Disable *Enable Split Tunneling* so that all SSL VPN traffic goes through the FortiGate.
5. Configure SSL VPN settings.
   a. Go to *VPN > SSL-VPN Settings*.
   b. Select the *Listen on Interface(s)*, in this example, *wan1*.
   c. Set *Listen on Port* to *10443*.
   d. Set *Server Certificate* to the authentication certificate.
   e. Under *Authentication/Portal Mapping*, set default Portal *web-access* for *All Other Users/Groups*.
   f. Create new *Authentication/Portal Mapping* for group *fac-group* mapping portal *full-access*.
6. Configure SSL VPN firewall policy.
   a. Go to *Policy & Objects > Firewall Policy*.
   b. Fill in the firewall policy name, in this example, *sslvpn certificate auth*.
   c. Incoming interface must be *SSL-VPN tunnel interface(ssl.root)*.
   d. Set the *Source Address* to *all* and *Source User* to *fac-group*.
   e. Set the *Outgoing Interface* to the local network interface so that the remote user can access the internal network, in this example, *port1*.
   f. Set *Destination Address* to the internal protected subnet *192.168.1.0*.
   g. Set *Schedule* to *always*, *Service* to *ALL*, and *Action* to *Accept*.
   h. Enable *NAT*.
   i. Configure any remaining firewall and security options as desired.
   j. Click *OK*.

To configure SSL VPN using the CLI:

1. Configure the interface and firewall address.

```
config system interface
    edit "wan1"
        set vdom "root"
        set ip 172.20.120.123 255.255.255.0
    next
end
```

2. Configure internal interface and protected subnet, then connect the port1 interface to the internal network.

```
config system interface
    edit "port1"
        set vdom "root"
        set ip 192.168.1.99 255.255.255.0
    next
end
```

```
config firewall address
    edit "192.168.1.0"
        set subnet 192.168.1.0 255.255.255.0
    next
end
```

3. Configure the RADIUS server.

```
config user radius
    edit "fac"
        set server "172.18.58.107"
        set secret <fac radius password>
        set auth-type ms_chap_v2
        set password-renewal enable
    next
end
```

4. Configure user group.

```
config user group
    edit "fac-group"
        set member "fac"
    next
end
```

5. Configure SSL VPN web portal.

```
config vpn ssl web portal
    edit "full-access"
        set tunnel-mode enable
        set web-mode enable
        set ip-pools "SSLVPN_TUNNEL_ADDR1"
        set split-tunneling disable
    next
end
```

6. Configure SSL VPN settings.

```
config vpn ssl settings
    set servercert "server_certificate"
    set tunnel-ip-pools "SSLVPN_TUNNEL_ADDR1"
```

```
        set source-interface "wan1"
        set source-address "all"
        set default-portal "web-access"
        config authentication-rule
            edit 1
                set groups "fac-group"
                set portal "full-access"
            next
        end
    end
```

7. Configure one SSL VPN firewall policy to allow remote user to access the internal network.

```
config firewall policy
    edit 1
        set name "sslvpn web mode access"
        set srcintf "ssl.root"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "192.168.1.0"
        set groups "fac-group"
        set action accept
        set schedule "always"
        set service "ALL"
        set nat enable
    next
end
```

To see the results of web portal:

1. From a remote device, use a web browser to log into the SSL VPN web portal *http://172.20.120.123:10443*.
2. Log in using the *test1* credentials.
   Use a user which is configured on FortiAuthenticator with *Force password change on next logon*.
3. Click *Login*. You are prompted to enter a new password.
4. On the FortiGate, go to *Dashboard* > *Network* and expand the *SSL-VPN* widget to verify the user's connection.

To see the results of tunnel connection:

1. Download FortiClient from www.forticlient.com.
2. Open the FortiClient Console and go to *Remote Access* > *Configure VPN*.
3. Add a new connection.
   - Set the connection name.
   - Set *Remote Gateway* to the IP of the listening FortiGate interface, in this example, *172.20.120.123*.
4. Select *Customize Port* and set it to *10443*.
5. Save your settings.
6. Log in using the *test1* credentials.
   You are prompted to enter a new password.

To check the SSL VPN connection using the GUI:

1. Go to *Dashboard* > *Network* and expand the *SSL-VPN* widget to verify the user's connection.
2. Go to *Log & Report* > *System Events* and select the *VPN Events* card to view the details of the SSL VPN connection

event log.

3. Go to *Log & Report > Forward Traffic* to view the details of the SSL VPN traffic.

To check the web portal login using the CLI:

```
get vpn ssl monitor
SSL VPN Login Users:
 Index    User     Auth Type     Timeout     From            HTTP in/out    HTTPS in/out
 0        test1    1(1)          229         10.1.100.254    0/0            0/0

SSL VPN sessions:
 Index    User     Source IP      Duration        I/O Bytes       Tunnel/Dest IP
```

To check the tunnel login using the CLI:

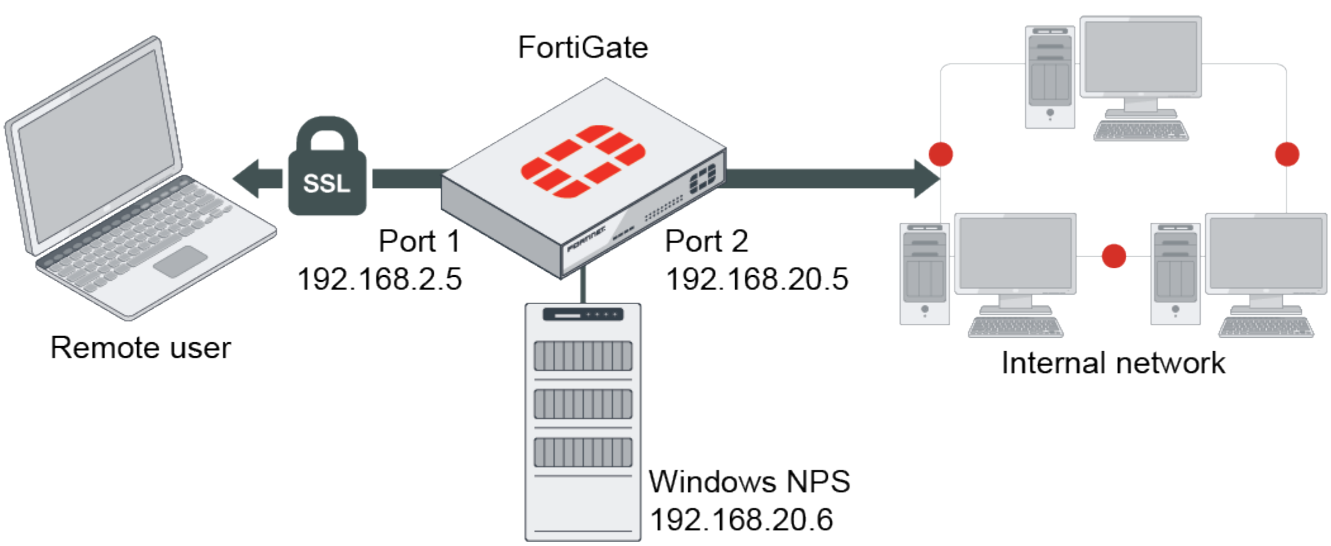
```
get vpn ssl monitor
SSL VPN Login Users:
 Index    User     Auth Type     Timeout    From            HTTP in/out   HTTPS in/out
 0        test1    1(1)          291        10.1.100.254    0/0           0/0

SSL VPN sessions:
 Index    User     Source IP      Duration    I/O Bytes      Tunnel/Dest IP
 0        test1    10.1.100.254   9           22099/43228    10.212.134.200
```

SSL VPN with RADIUS on Windows NPS

This is an example configuration of SSL VPN that uses Windows Network Policy Server (NPS) as a RADIUS authentication server.

The NPS must already be configured to accept the FortiGate as a RADIUS client and the choice of authentication method, such as MS-CHAPv2. A shared key must also have been created.

Example



The user is connecting from their PC to the FortiGate's port1 interface. RADIUS authentication occurs between the FortiGate and the Windows NPS, and the SSL-VPN connection is established once the authentication is successful.

Configure SSL-VPN with RADIUS on Windows NPS in the GUI

To configure the internal and external interfaces:

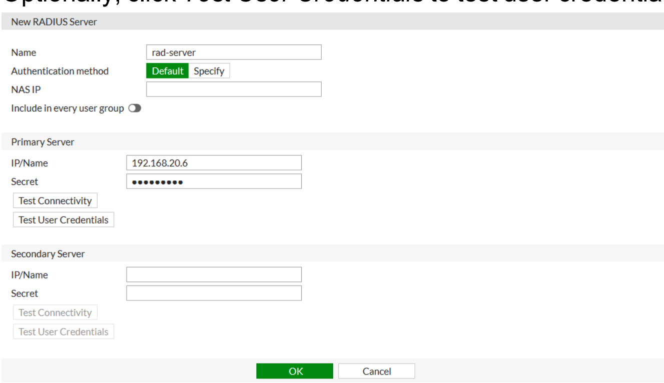
1. Go to *Network > Interfaces*
2. Edit the *port1* interface and set *IP/Network Mask* to *192.168.2.5/24*.
3. Edit the *port2* interface and set *IP/Network Mask* to *192.168.20.5/24*.
4. Click *OK*.

To create a firewall address:

1. Go to *Policy & Objects > Addresses* and click *Create New > Address*.
2. Set *Name* to *192.168.20.0*.
3. Leave *Type* as *Subnet*
4. Set *IP/Netmask* to *192.168.20.0/24*.
5. Click *OK*.

To add the RADIUS server:

1. Go to *User & Authentication > RADIUS Servers* and click *Create New*.
2. Set *Name* to *rad-server*.
3. Leave *Authentication method* set to *Default*. The PAP, MS-CHAPv2, and CHAP methods will be tried in order.
4. Under *Primary Server*, set *IP/Name* to *192.168.20.6* and *Secret* to the shared secret configured on the RADIUS server.
5. Click *Test Connectivity* to test the connection to the server, and ensure that *Connection status* is *Successful*.
6. Optionally, click *Test User Credentials* to test user credentials. Testing from the GUI is limited to PAP.

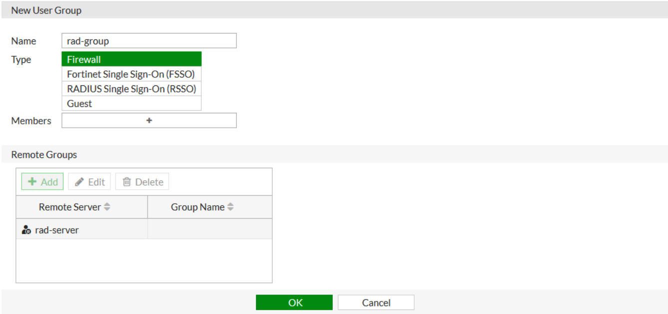


7. Click *OK*.

To configure a user group:

1. Go to *User & Authentication > User Groups* and click *Create New*.
2. Set *Name* to *rad-group*.

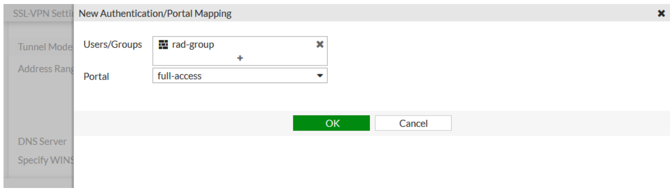
3. Under *Remote Groups*, click *Add* and add the *rad-server*.



4. Click *OK*.

To configure SSL VPN settings:

1. Go to *VPN > SSL-VPN Settings*.
2. Select the *Listen on Interface(s)*, in this example, *port1*.
3. Set *Listen on Port* to *10443*.
4. If you have a server certificate, set *Server Certificate* to the authentication certificate.
5. Under *Authentication/Portal Mapping*:
   a. Edit *All Other Users/Groups* and set *Portal* to *web-access*.
   b. Click *Create New* and create a mapping for the *rad-group* user group with *Portal* set to *full-access*.

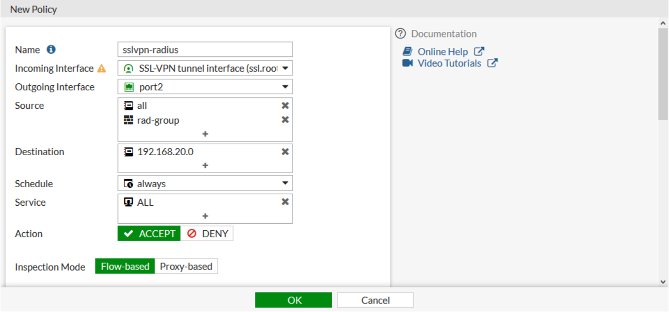


   c. Click *OK*.
6. Click *Apply*.

To configure an SSL VPN firewall policy:

1. Go to *Policy & Objects > Firewall Policy* and click *Create New*.
2. Set the policy name, in this example, *sslvpn-radius*.
3. Set *Incoming Interface* to *SSL-VPN tunnel interface(ssl.root)*.
4. Set *Outgoing Interface* to the local network interface so that the remote user can access the internal network. In this example, *port2*.
5. Set the *Source > Address* to *all* and *Source > User* to *rad-group*.
6. Set *Destination > Address* to the internal protected subnet *192.168.20.0*.
7. Set *Schedule* to *always*, *Service* to *ALL*, and *Action* to *Accept*.

8. Enable *NAT*.



9. Configure the remaining options as required.
10. Click *OK*.

Configure SSL-VPN with RADIUS on Windows NPS in the CLI

To configure SSL VPN using the CLI:

1. Configure the internal and external interfaces:

```
config system interface
    edit "port1"
        set vdom "root"
        set ip 192.168.2.5 255.255.255.0
        set alias internal
    next
    edit "port2"
        set vdom "root"
        set ip 192.168.20.5 255.255.255.0
        set alias external
    next
end
```

2. Configure the firewall address:

```
config firewall address
    edit "192.168.20.0"
        set subnet 192.168.20.0 255.255.255.0
    next
end
```

3. Add the RADIUS server:

```
config user radius
    edit "rad-server"
        set server "192.168.20.6"
        set secret *********
    next
end
```

4. Create a user group and add the RADIUS server to it:.

```
config user group
    edit "rad-group"
        set member "rad-server"
    next
end
```

5. Configure SSL VPN settings:

```
config vpn ssl settings
    set servercert "server_certificate"
    set tunnel-ip-pools "SSLVPN_TUNNEL_ADDR1"
    set source-interface "port1"
    set source-address "all"
    set default-portal "web-access"
    config authentication-rule
        edit 1
            set groups "rad-group"
            set portal "full-access"
        next
    end
end
```

6. Configure an SSL VPN firewall policy to allow remote user to access the internal network.

```
config firewall policy
    edit 1
        set name "sslvpn-radius"
        set srcintf "ssl.root"
        set dstintf "port2"
        set srcaddr "all"
        set dstaddr "192.168.20.0"
        set groups “rad-group”
        set action accept
        set schedule "always"
        set service "ALL"
        set nat enable
    next
end
```

Results

To connect with FortiClient in tunnel mode:

1. Download FortiClient from www.forticlient.com.
2. Open the FortiClient Console and go to *Remote Access > Configure VPN*.
3. Add a new connection:
   a. Set the connection name.
   b. Set *Remote Gateway* to *192.168.2.5*.
   c. Select *Customize Port* and set it to *10443*.
4. Save your settings.
5. Log in using the RADIUS user credentials.

To check the SSL VPN connection using the GUI:

1. Go to *Dashboard > Network* and expand the *SSL-VPN* widget to verify the user’s connection.
2. Go to *Log & Report > System Events* and select the *VPN Events* card to view the details of the SSL VPN connection event log.
3. Go to *Log & Report > Forward Traffic* to view the details of the SSL VPN traffic.

To check the login using the CLI:

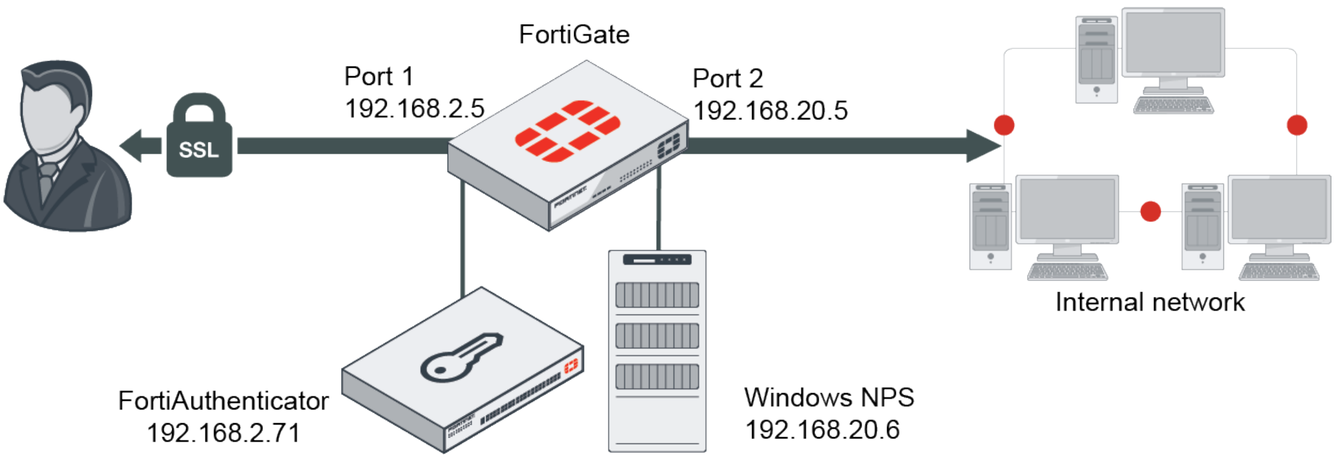
```
# get vpn ssl monitor
SSL VPN Login Users:
 Index    User     Group    Auth Type       Timeout         From     HTTP in/out    HTTPS in/out
 0        radkeith          rad-group         2(1)            295      192.168.2.202  0/0      0/0

SSL VPN sessions:
 Index    User     Group    Source IP       Duration        I/O Bytes        Tunnel/Dest IP
 0        radkeith          rad-group         192.168.2.202   18       28502/4966
10.212.134.200
```

SSL VPN with multiple RADIUS servers

When configuring two or more RADIUS servers, you can configure a Primary and Secondary server within the same RADIUS server configurations for backup purposes. You can also configure multiple RADIUS servers within the same User Group to service the access request at the same time.

A tertiary server can be configured in the CLI.

Sample topology



Sample configurations

- Configure a Primary and Secondary server for backup on page 1700
- Authenticating to two RADIUS servers concurrently on page 1704

Configure a Primary and Secondary server for backup

When you define a Primary and Secondary RADIUS server, the access request will always be sent to the Primary server first. If the request is denied with an Access-Reject, then the user authentication fails. However, if there is no response from the Primary server after another attempt, the access request will be sent to the Secondary server.

In this example, you will use a Windows NPS server as the Primary server and a FortiAuthenticator as the Secondary server. It is assumed that users are synchronized between the two servers.

To configure the internal and external interfaces:

1. Go to *Network > Interfaces*.
2. Edit the *port1* interface and set *IP/Network Mask* to *192.168.2.5/24*.
3. Edit the *port2* interface and set *IP/Network Mask* to *192.168.20.5/24*.
4. Click *OK*.

To create a firewall address:

1. Go to *Policy & Objects > Addresses* and click *Create New > Address*.
2. Set *Name* to *192.168.20.0*.
3. Leave *Type* as *Subnet*
4. Set IP/*Netmask* to *192.168.20.0/24*.
5. Click *OK*.

To add the RADIUS servers:

1. Go to *User & Authentication > RADIUS Servers* and click *Create New*.
2. Set *Name* to *PrimarySecondary*.
3. Leave *Authentication method* set to *Default*. The PAP, MS-CHAPv2, and CHAP methods will be tried in order.
4. Under *Primary Server*, set *IP/Name* to *192.168.20.6* and *Secret* to the shared secret configured on the RADIUS server.
5. Click *Test Connectivity* to test the connection to the server, and ensure that *Connection status* is *Successful*.
6. Under *Secondary Server*, set *IP/Name* to *192.168.2.71* and *Secret* to the shared secret configured on the RADIUS server.
7. Click *Test Connectivity* to test the connection to the server, and ensure that *Connection status* is *Successful*.
8. *Click OK.*

To configure the user group:

1. Go to *User & Authentication > User Groups* and click *Create New*.
2. In the *Name* field, enter *PrimarySecondaryGroup*.
3. In the *Remote Groups* area, click *Add*, and from the *Remote Server* dropdown, select *PrimarySecondary*.
4. Click *OK*, and then click *OK* again.

To configure the SSL VPN settings:

1. Go to *VPN > SSL-VPN Settings*.
2. From the *Listen on Interface(s)* dropdown select *port1*.
3. In the *Listen on Port* field enter *10443*.
4. Optionally, from the *Server Certificate* dropdown, select the authentication certificate if you have one for this SSL VPN portal.
5. Under *Authentication/Portal Mapping*, set the default portal web-access.
   a. Select *All Other Users/Groups* and click *Edit*.
   b. From the *Portal* dropdown, select *web-access*.

   c. Click *OK*.
6. Create a web portal for *PrimarySecondaryGroup*.
   a. Under *Authentication/Portal Mapping*, click *Create New*.
   b. Click *Users/Groups* and select *PrimarySecondaryGroup*.
   c. From the *Portal* dropdown, select *full-access*.
   d. Click *OK*.

To configure SSL VPN firewall policy:

1. Go to *Policy & Objects > Firewall Policy*.
2. Click *Create New* to create a new policy, or double-click an existing policy to edit it and configure the following settings:

Name	Enter a name for the policy.
Incoming Interface	*SSL-VPN tunnel interface (ssl.root)*
Outgoing interface	Set to the local network interface so that the remote user can access the internal network. For this example, select *port3*.
Source	In the *Address* tab, select *SSLVPN_TUNNEL_ADDR1* In the *User* tab, select *PrimarySecondaryGroup*
Destination	Select the internal protected subnet *192.168.20.0*.
Schedule	*always*
Service	*All*
Action	*Accept*
NAT	*Enable*

3. Configure any remaining firewall and security options as required.
4. Click *OK*.

To configure SSL VPN using the CLI:

1. Configure the internal interface and firewall address:
```
config system interface
   edit "port3"
      set vdom "root"
      set ip 192.168.20.5 255.255.255.0
      set alias "internal"
   next
end
config firewall address
   edit "192.168.20.0"
      set uuid cc41eec2-9645-51ea-d481-5c5317f865d0
      set subnet 192.168.20.0 255.255.255.0
   next
end
```
2. Configure the RADIUS server:
```
config user radius
```

```
   edit "PrimarySecondary"
      set server "192.168.20.6"
      set secret <secret>
      set secondary-server "192.168.2.71"
      set secondary-secret <secret>
   next
end
```

3. Add the RADIUS user to the user group:

```
config user group
   edit "PrimarySecondaryGroup"
      set member "PrimarySecondary "
   next
end
```

4. Configure SSL VPN settings:

```
config vpn ssl settings
   set servercert "server_certificate"
   set tunnel-ip-pools "SSLVPN_TUNNEL_ADDR1"
   set source-interface "port1"
   set source-address "all"
   set default-portal "web-access"
   config authentication-rule
      edit 1
         set groups "PrimarySecondaryGroup "
         set portal "full-access"
      next
   end
end
```

5. Configure one SSL VPN firewall policy to allow remote users to access the internal network:

```
config firewall policy
   edit 1
      set name "sslvpn-radius"
      set srcintf "ssl.root"
      set dstintf "port3"
      set srcaddr "all"
      set dstaddr "192.168.20.0"
      set groups "PrimarySecondaryGroup"
      set action accept
      set schedule "always"
      set service "ALL"
      set nat enable
   next
end
```

To verify the connection:

User *radkeith* is a member of both the NPS server and the FAC server.

When the Primary server is up, it will connect to the SSL VPN tunnel using FortiClient.

```
# diagnose sniffer packet any 'port 1812' 4 0 l
interfaces=[any]
filters=[port 1812]
2020-05-15 16:26:50.838453 port3 out 192.168.20.5.2374 -> 192.168.20.6.1812: udp 118
2020-05-15 16:26:50.883166 port3 in 192.168.20.6.1812 -> 192.168.20.5.2374: udp 20
2020-05-15 16:26:50.883374 port3 out 192.168.20.5.2374 -> 192.168.20.6.1812: udp 182
2020-05-15 16:26:50.884683 port3 in 192.168.20.6.1812 -> 192.168.20.5.2374: udp 228
```

The access request is sent to the Primary NPS server 192.168.20.6, and the connection is successful.

```
# get vpn ssl monitor
SSL VPN Login Users:

Index   User            Group                    Auth Type      Timeout         From      HTTP
in/out    HTTPS in/out

0       radkeith       PrimarySecondaryGroup    2(1)            285      192.168.2.202
 0/0     0/0

SSL VPN sessions:

Index   User       Group                   Source IP      Duration        I/O Bytes
Tunnel/Dest IP

0       radkeith   PrimarySecondaryGroup   192.168.2.202   62              132477/4966
10.212.134.200
```

When the Primary server is down, and the Secondary server is up, the connection is made to the SSLVPN tunnel again:

```
# diagnose sniffer packet any 'port 1812' 4 0 l
interfaces=[any]
filters=[port 1812]
2020-05-15 16:31:23.016875 port3 out 192.168.20.5.7989 -> 192.168.20.6.1812: udp 118
2020-05-15 16:31:28.019470 port3 out 192.168.20.5.7989 -> 192.168.20.6.1812: udp 118
2020-05-15 16:31:30.011874 port1 out 192.168.2.5.23848 -> 192.168.2.71.1812: udp 118
2020-05-15 16:31:30.087564 port1 in 192.168.2.71.1812 -> 192.168.2.5.23848: udp 20
```

Access request is sent to the Primary NPS server 192.168.20.6, but there was no response. RADIUS authentication falls through to the Secondary FortiAuthenticator 192.168.2.71, and the authentication was accepted. The VPN connection is established.

```
# get vpn ssl monitor
SSL VPN Login Users:

Index   User            Group                   Auth Type      Timeout         From      HTTP
in/out    HTTPS in/out

0       radkeith        PrimarySecondaryGroup   2(1)            287      192.168.2.202
0/0         0/0

SSL VPN sessions:

Index   User            Group                     Source IP      Duration        I/O Bytes
   Tunnel/Dest IP

0       radkeith        PrimarySecondaryGroup   192.168.2.202   48              53544/4966
  10.212.134.200
```

Authenticating to two RADIUS servers concurrently

There are times where users are located on separate RADIUS servers. This may be the case when migrating from an old server to a new one for example. In this scenario, a Windows NPS server and a FortiAuthenticator are configured in the same User Group. The access-request is sent to both servers concurrently. If FortiGate receives an access-accept from either server, authentication is successful.

To configure the internal and external interfaces:

1. Go to *Network > Interfaces*.
2. Edit the *port1* interface and set *IP/Network Mask* to *192.168.2.5/24*.

3. Edit the *port2* interface and set *IP/Network Mask* to *192.168.20.5/24*.
4. Click *OK*.

To create a firewall address:

1. Go to *Policy & Objects > Addresses* and click *Create New > Address*.
2. Set *Name* to *192.168.20.0*.
3. Leave *Type* as *Subnet*
4. Set IP/*Netmask* to *192.168.20.0/24*.
5. Click *OK*.

To configure the first RADIUS server:

1. Go to *User & Authentication > RADIUS Servers* and click *Create New*.
2. Set *Name* to *win2k16*.
3. Leave *Authentication method* set to *Default*. The PAP, MS-CHAPv2, and CHAP methods will be tried in order.
4. Under *Primary Server*, set *IP/Name* to *192.168.20.6* and *Secret* to the shared secret configured on the RADIUS server.
5. Click *Test Connectivity* to test the connection to the server, and ensure that *Connection status* is *Successful*.
6. *Click OK.*

To configure the second RADIUS server:

1. Go to *User & Authentication > RADIUS Servers* and click *Create New*.
2. Set *Name* to *fac*.
3. Leave *Authentication method* set to *Default*. The PAP, MS-CHAPv2, and CHAP methods will be tried in order.
4. Under *Primary Server*, set *IP/Name* to *192.168.2.71* and *Secret* to the shared secret configured on the RADIUS server.
5. Click *Test Connectivity* to test the connection to the server, and ensure that *Connection status* is *Successful*.
6. *Click OK.*

To configure the user group:

1. Go to *User & Authentication > User Groups* and click *Create New*.
2. In the *Name* field, enter *dualPrimaryGroup.*.
3. In the *Remote Groups* area, click *Add*, and from the *Remote Server* dropdown, select *fac*.
4. Click *Add* again. From the *Remote Server* dropdown select *win2k16* and click *OK*.
5. Click *OK*, and then click *OK* again.

To configure the SSL VPN settings:

1. Go to *VPN > SSL-VPN Settings*.
2. From the *Listen on Interface(s)* dropdown select *port1*.
3. In the *Listen on Port* field enter *10443*.
4. Optionally, from the *Server Certificate* dropdown, select the authentication certificate if you have one for this SSL VPN portal.

5. Under *Authentication/Portal Mapping*, set the default portal web-access.
   a. Select *All Other Users/Groups* and click Edit.
   b. From the *Portal* dropdown, select *web-access*.
   c. Click *OK*.
6. Create a web portal for *PrimarySecondaryGroup*.
   a. Under *Authentication/Portal Mapping*, click *Create New*.
   b. Click *Users/Groups* and select *dualPrimaryGroup*.
   c. From the *Portal* dropdown, select *full-access*.
   d. Click *OK*.

To configure SSL VPN firewall policy:

1. Go to *Policy & Objects > Firewall Policy*.
2. Click *Create New* to create a new policy, or double-click an existing policy to edit it.

Name	Enter a name for the policy.
Incoming Interface	*SSL-VPN tunnel interface (ssl.root)*
Outgoing interface	Set to the local network interface so that the remote user can access the internal network. For this example, select *port3*.
Source	In the *Address* tab, select *SSLVPN_TUNNEL_ADDR1* In the *User* tab, select *dualPrimaryGroup*
Destination	Select the internal protected subnet *192.168.20.0*.
Schedule	*always*
Service	*All*
Action	*Accept*
NAT	*Enable*

3. Configure any remaining firewall and security options as required.
4. Click *OK*.

To configure SSL VPN using the CLI:

1. Configure the internal interface and firewall address:

```
config system interface
   edit "port3"
      set vdom "root"
      set ip 192.168.20.5 255.255.255.0
      set alias "internal"
   next
end
config firewall address
   edit "192.168.20.0"
      set uuid cc41eec2-9645-51ea-d481-5c5317f865d0
      set subnet 192.168.20.0 255.255.255.0
   next
end
```

2. Configure the RADIUS server:

```
config user radius
    edit "win2k16"
        set server "192.168.20.6"
        set secret <secret>
    next
    edit "fac"
        set server "192.168.2.71"
        set secret <secret>
    next
end
```

3. Add the RADIUS user to the user group:

```
config user group
    edit "dualPrimaryGroup"
        set member "win2k16" “fac”
    next
end
```

4. Configure SSL VPN settings:

```
config vpn ssl settings
    set servercert "server_certificate"
    set tunnel-ip-pools "SSLVPN_TUNNEL_ADDR1"
    set source-interface "port1"
    set source-address "all"
    set default-portal "web-access"
    config authentication-rule
        edit 1
            set groups "dualPrimaryGroup"
            set portal "full-access"
        next
    end
end
```

5. Configure one SSL VPN firewall policy to allow remote users to access the internal network:

```
config firewall policy
    edit 1
        set name "sslvpn-radius"
        set srcintf "ssl.root"
        set dstintf "port3"
        set srcaddr "all"
        set dstaddr "192.168.20.0"
        set groups “dualPrimaryGroup”
        set action accept
        set schedule "always"
        set service "ALL"
        set nat enable
    next
end
```

To verify the connection:

User *fackeith* is a member of the FortiAuthenticator server only.

User *radkeith* is a member of both the NPS server and the FortiAuthenticator server, but has different passwords on each server.

Case 1: Connect to the SSLVPN tunnel using FortiClient with user FacAdmin:

```
# diagnose sniffer packet any 'port 1812' 4 0 l
interfaces=[any]
filters=[port 1812]
2020-05-15 17:21:31.217985 port3 out 192.168.20.5.11490 -> 192.168.20.6.1812: udp 118
2020-05-15 17:21:31.218091 port1 out 192.168.2.5.11490 -> 192.168.2.71.1812: udp 118
2020-05-15 17:21:31.219314 port3 in 192.168.20.6.1812 -> 192.168.20.5.11490: udp 20 <--
     access-reject
2020-05-15 17:21:31.219519 port3 out 192.168.20.5.11490 -> 192.168.20.6.1812: udp 182
2020-05-15 17:21:31.220219 port3 in 192.168.20.6.1812 -> 192.168.20.5.11490: udp 42
2020-05-15 17:21:31.220325 port3 out 192.168.20.5.11490 -> 192.168.20.6.1812: udp 119
2020-05-15 17:21:31.220801 port3 in 192.168.20.6.1812 -> 192.168.20.5.11490: udp 20
2020-05-15 17:21:31.236009 port1 in 192.168.2.71.1812 -> 192.168.2.5.11490: udp 20 <--
     access-accept
```

Access is denied by the NPS server because the user does not exist. However, access is accepted by FortiAuthenticator. The end result is the authentication is successful.

```
# get vpn ssl monitor
SSL VPN Login Users:

Index   User       Group             Auth Type      Timeout         From       HTTP
in/out    HTTPS in/out

0       fackeith  dualPrimaryGroup   2(1)            292       192.168.2.202      0/0
      0/0

SSL VPN sessions:

Index   User       Group                  Source IP      Duration        I/O Bytes
Tunnel/Dest IP

0       fackeith  dualPrimaryGroup       192.168.2.202   149              70236/4966
 10.212.134.200
```

Case 2: Connect to the SSLVPN tunnel using FortiClient with user radkeith:

```
# diagnose sniffer packet any 'port 1812' 4 0 l
interfaces=[any]
filters=[port 1812]
2020-05-15 17:26:07.335791 port1 out 192.168.2.5.17988 -> 192.168.2.71.1812: udp 118
2020-05-15 17:26:07.335911 port3 out 192.168.20.5.17988 -> 192.168.20.6.1812: udp 118
2020-05-15 17:26:07.337659 port3 in 192.168.20.6.1812 -> 192.168.20.5.17988: udp 20 <--
     access-accept
2020-05-15 17:26:07.337914 port3 out 192.168.20.5.17988 -> 192.168.20.6.1812: udp 182
2020-05-15 17:26:07.339451 port3 in 192.168.20.6.1812 -> 192.168.20.5.17988: udp 228
2020-05-15 17:26:08.352597 port1 in 192.168.2.71.1812 -> 192.168.2.5.17988: udp 20 <--
     access-reject
```

There is a password mismatch for this user on the Secondary RADIUS server. However, even though the authentication was rejected by FortiAuthenticator, it was accepted by Windows NPS. Therefore, the end result is authentication successful.

```
# get vpn ssl monitor
SSL VPN Login Users:

Index   User       Group             Auth Type      Timeout         From         HTTP
in/out    HTTPS in/out

0       radkeith  dualPrimaryGroup   2(1)            290       192.168.2.202    0/0
      0/0
```

```
SSL VPN sessions:
Index   User      Group                Source IP      Duration        I/O Bytes
Tunnel/Dest IP
0       radkeith dualPrimaryGroup   192.168.2.202   142              64875/4966
10.212.134.200
```

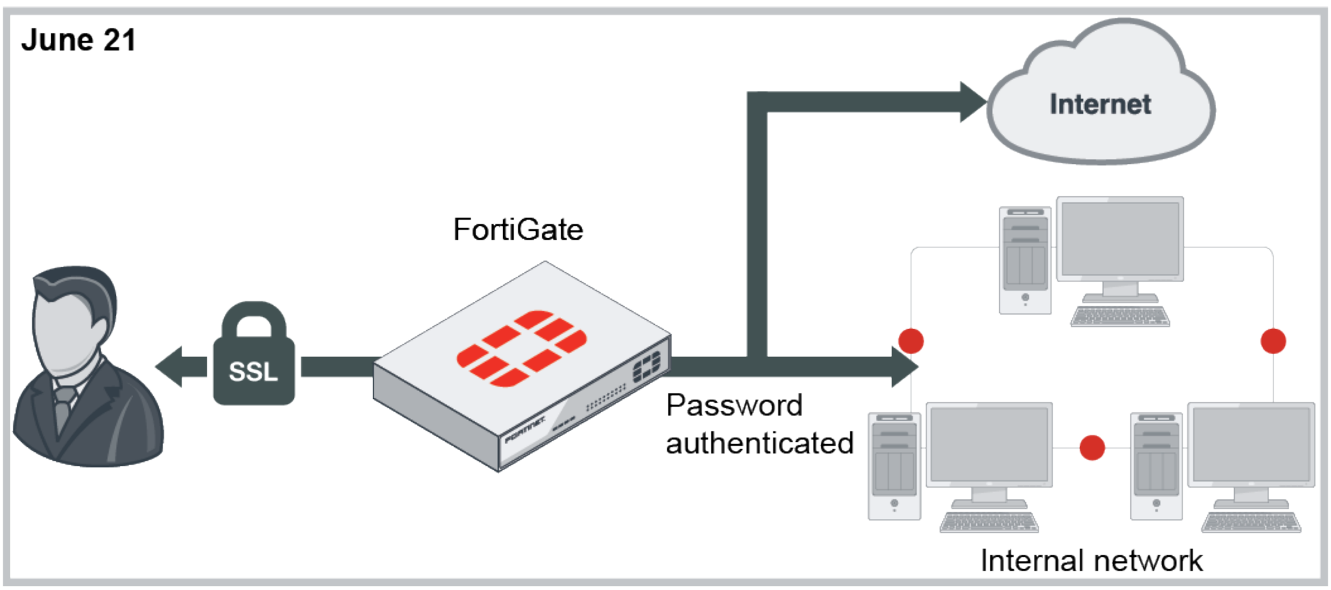
SSL VPN with local user password policy

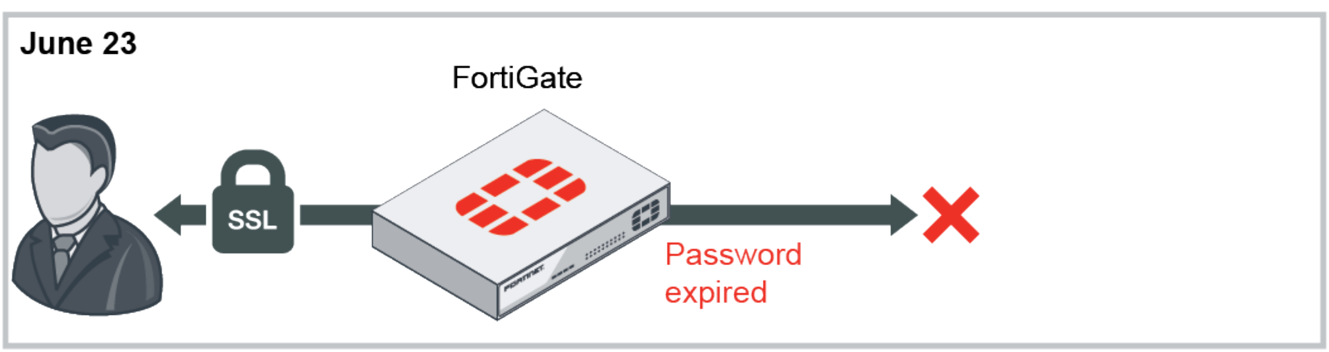
This is a sample configuration of SSL VPN for users with passwords that expire after two days. Users are warned after one day about the password expiring. The password policy can be applied to any local user password. The password policy cannot be applied to a user group or a local remote user such as LDAP/RADIUS/TACACS+.

In FortiOS 6.2, users are warned after one day about the password expiring and have one day to renew it. If the password expires, the user cannot renew the password and must contact the administrator for assistance.

In FortiOS 6.0/5.6, users are warned after one day about the password expiring and have to renew it. If the password expires, the user can still renew the password.

Sample topology





Sample configuration

WAN interface is the interface connected to ISP. This example shows static mode. You can also use DHCP or PPPoE mode. The SSL VPN connection is established over the WAN interface.

To configure SSL VPN using the GUI:

1. Configure the interface and firewall address. The port1 interface connects to the internal network.
   a. Go to *Network > Interfaces* and edit the *wan1* interface.
   b. Set *IP/Network Mask* to *172.20.120.123/255.255.255.0*.
   c. Edit *port1* interface and set *IP/Network Mask* to *192.168.1.99/255.255.255.0*.
   d. Click *OK*.
   e. Go to *Policy & Objects > Address* and create an address for internet subnet *192.168.1.0*.
2. Configure user and user group.
   a. Go to *User & Authentication > User Definition* to create a local user.
   b. Go to *User & Authentication > User Groups* to create a user group and add that local user to it.
3. Configure and assign the password policy using the CLI.
   a. Configure a password policy that includes an expiry date and warning time. The default start time for the password is the time the user was created.

```
config user password-policy
    edit "pwpolicy1"
        set expire-days 2
        set warn-days 1
    next
end
```

   b. Assign the password policy to the user you just created.

```
config user local
    edit "sslvpnuser1"
        set type password
        set passwd-policy "pwpolicy1"
    next
end
```

4. Configure SSL VPN web portal.
   a. Go to *VPN > SSL-VPN Portals* to edit the *full-access* portal.
      This portal supports both web and tunnel mode.
   b. Disable *Enable Split Tunneling* so that all SSL VPN traffic goes through the FortiGate.
5. Configure SSL VPN settings.
   a. Go to *VPN > SSL-VPN Settings*.
   b. Select the *Listen on Interface(s)*, in this example, *wan1*.
   c. Set *Listen on Port* to *10443*.
   d. Set *Server Certificate* to the authentication certificate.
   e. Under *Authentication/Portal Mapping*, set default Portal *web-access* for *All Other Users/Groups*.
   f. Create new *Authentication/Portal Mapping* for group *sslvpngroup* mapping portal *full-access*.
6. Configure SSL VPN firewall policy.
   a. Go to *Policy & Objects > Firewall Policy*.
   b. Fill in the firewall policy name. In this example, *sslvpn certificate auth*.
   c. Incoming interface must be *SSL-VPN tunnel interface(ssl.root)*.

d. Set the *Source Address* to *all* and *Source User* to *sslvpngroup*.
e. Set the *Outgoing Interface* to the local network interface so that the remote user can access the internal network. In this example, *port1*.
f. Set *Destination Address* to the internal protected subnet *192.168.1.0*.
g. Set *Schedule* to *always*, *Service* to *ALL*, and *Action* to *Accept*.
h. Enable *NAT*.
i. Configure any remaining firewall and security options as desired.
j. Click *OK*.

To configure SSL VPN using the CLI:

1. Configure the interface and firewall address.

```
config system interface
    edit "wan1"
        set vdom "root"
        set ip 172.20.120.123 255.255.255.0
    next
end
```

2. Configure internal interface and protected subnet, then connect the port1 interface to the internal network.

```
config system interface
    edit "port1"
        set vdom "root"
        set ip 192.168.1.99 255.255.255.0
    next
end
```

```
config firewall address
    edit "192.168.1.0"
        set subnet 192.168.1.0 255.255.255.0
    next
end
```

3. Configure user and user group.

```
config user local
    edit "sslvpnuser1"
        set type password
        set passwd your-password
    next
end
config user group
    edit "sslvpngroup"
        set member "vpnuser1"
    next
end
```

4. Configure and assign the password policy.
   a. Configure a password policy that includes an expiry date and warning time. The default start time for the password is the time the user was created.

```
config user password-policy
    edit "pwpolicy1"
        set expire-days 2
        set warn-days 1
```

```
    next
end
```

b. Assign the password policy to the user you just created.

```
config user local
    edit "sslvpnuser1"
        set type password
        set passwd-policy "pwpolicy1"
    next
end
```

5. Configure SSL VPN web portal.

```
config vpn ssl web portal
    edit "full-access"
        set tunnel-mode enable
        set web-mode enable
        set ip-pools "SSLVPN_TUNNEL_ADDR1"
        set split-tunneling disable
     next
end
```

6. Configure SSL VPN settings.

```
config vpn ssl settings
    set servercert "server_certificate"
    set tunnel-ip-pools "SSLVPN_TUNNEL_ADDR1"
    set source-interface "wan1"
    set source-address "all"
    set default-portal "web-access"
    config authentication-rule
        edit 1
            set groups "sslvpngroup"
            set portal "full-access"
        next
    end
end
```

7. Configure one SSL VPN firewall policy to allow remote user to access the internal network.

```
config firewall policy
    edit 1
        set name "sslvpn web mode access"
        set srcintf "ssl.root"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "192.168.1.0"
        set groups “sslvpngroup”
        set action accept
        set schedule "always"
        set service "ALL"
        set nat enable
    next
end
```

To see the results of web portal:

1. From a remote device, use a web browser to log into the SSL VPN web portal *http://172.20.120.123:10443*.
2. Log in using the *sslvpnuser1* credentials.
   When the warning time is reached, the user is prompted to enter a new password.
   In FortiOS 6.2, when the password expires, the user cannot renew the password and must contact the administrator.
   In FortiOS 6.0/5.6, when the password expires, the user can still renew the password.
3. On the FortiGate, go to *Dashboard > Network* and expand the *SSL-VPN* widget to verify the user's connection.

To see the results of tunnel connection:

1. Download FortiClient from www.forticlient.com.
2. Open the FortiClient Console and go to *Remote Access > Configure VPN*.
3. Add a new connection.
   - Set the connection name.
   - Set *Remote Gateway* to the IP of the listening FortiGate interface, in this example, *172.20.120.123*.
4. Select *Customize Port* and set it to *10443*.
5. Save your settings.
6. Log in using the *sslvpnuser1* credentials.
   When the warning time is reached, the user is prompted to enter a new password.

To check the SSL VPN connection using the GUI:

1. Go to *Dashboard > Network* and expand the *SSL-VPN* widget to verify the user's connection.
2. Go to *Log & Report > Forward Traffic* to view the details of the SSL VPN traffic.

To check that login failed due to password expired on GUI:

1. Go to *Log & Report > System Events* and select the *VPN Events* card to see the SSL VPN alert labeled `ssl-login-fail`.
2. Click *Details* to see the log details about the *Reason* `sslvpn_login_password_expired`.

To check the web portal login using the CLI:

```
get vpn ssl monitor
SSL VPN Login Users:
 Index    User         Auth Type    Timeout     From           HTTP in/out    HTTPS in/out
 0        sslvpnuser1  1(1)         229         10.1.100.254   0/0            0/0

SSL VPN sessions:
 Index    User     Source IP      Duration         I/O Bytes        Tunnel/Dest IP
```

To check the tunnel login using the CLI:

```
get vpn ssl monitor
SSL VPN Login Users:
 Index    User          Auth Type    Timeout     From           HTTP in/out    HTTPS in/out
 0        sslvpnuser1   1(1)         291         10.1.100.254   0/0            0/0
```

```
SSL VPN sessions:
 Index    User           Source IP      Duration    I/O Bytes      Tunnel/Dest IP
 0        sslvpnuser1    10.1.100.254   9           22099/43228    10.212.134.200
```

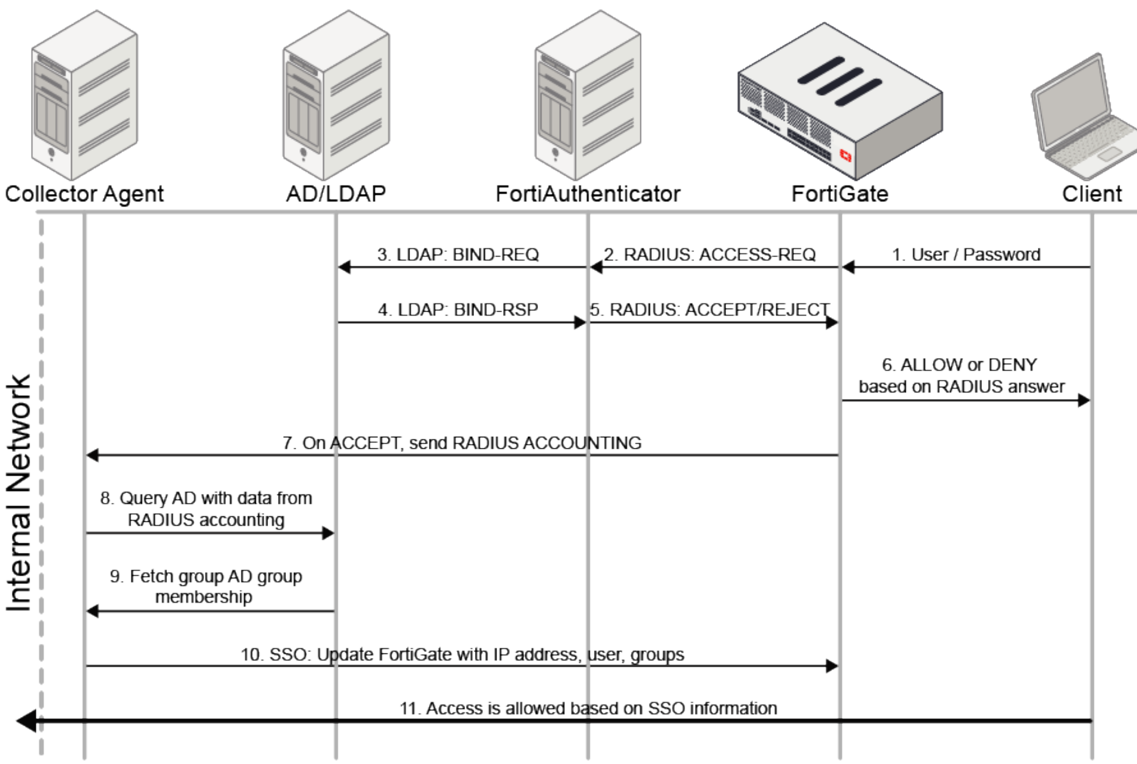
To check the FortiOS 6.2 login password expired event log:

```
FG201E4Q17901354 # execute log  filter category event

FG201E4Q17901354 # execute log  filter field subtype vpn

FG201E4Q17901354 # execute log  filter field action ssl-login-fail

FG201E4Q17901354 # execute log  display
1: date=2019-02-15 time=10:57:56 logid="0101039426" type="event" subtype="vpn" level="alert"
vd="root" eventtime=1550257076 logdesc="SSL VPN login fail" action="ssl-login-fail"
tunneltype="ssl-web" tunnelid=0 remip=10.1.100.254 user="u1" group="g1" dst_host="N/A"
reason="sslvpn_login_password_expired" msg="SSL user failed to logged in"
```

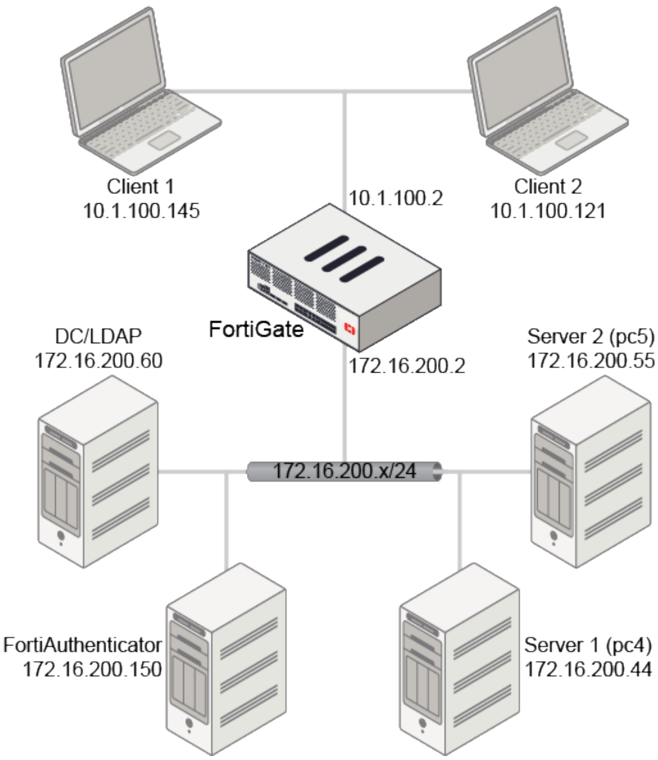
Dynamic address support for SSL VPN policies

Dynamic SSO user groups can be used in place of address objects when configuring SSL VPN policies. This allows dynamic IP addresses to be used in SSL VPN policies. A remote user group can be used for authentication while an FSSO group is separately used for authorization. Using a dummy policy for remote user authentication and a policy for FSSO group authorization, FSSO can be used with SSL VPN tunnels.

This image shows the authentication and authorization flow:



In this example, FortiAuthenticator is used as a RADIUS server. It uses a remote AD/LDAP server for authentication, then returns the authentication results to the FortiGate. This allows the client to have a dynamic IP address after successful authentication.



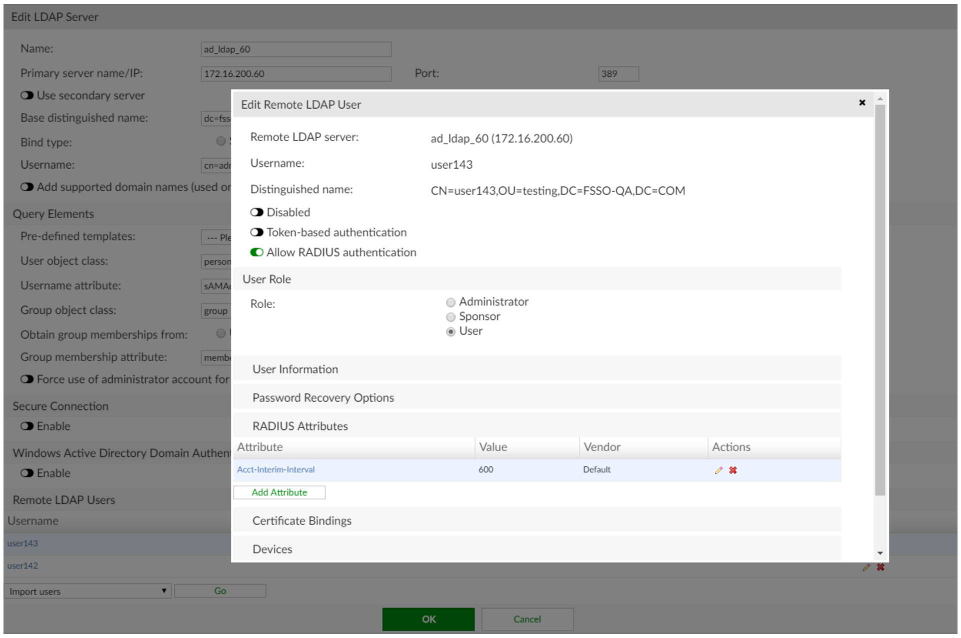
First, on the LDAP server, create two users each in their own group, *user142* in group *pc_group1*, and *user143* in group *pc_group2*.

Configure the FortiAuthenticator

To add a remote LDAP server and users on the FortiAuthenticator:

1. Go to *Authentication > Remote Auth. Servers > LDAP*.
2. Click *Create New*.
3. Set the following:
   - *Name*: *ad_ldap_60*
   - *Primary server name/IP*: *172.16.200.60*
   - *Base distinguished name*: *dc=fsso-qa,dc=com*
   - *Bind type*: Regular
   - *Username*: *cn=administrator,cn=User*
   - *Password*: <enter a password>
4. Click *OK*.
5. Edit the new LDAP server.
6. Import the remote LDAP users.
7. Edit each user to confirm that they have the RADIUS attribute *Acct-Interim-Interval*. This attribute is used by

FortiGate to send interim update account messages to the RADIUS server.



To create a RADIUS client for FortiGate as a remote authentication server:

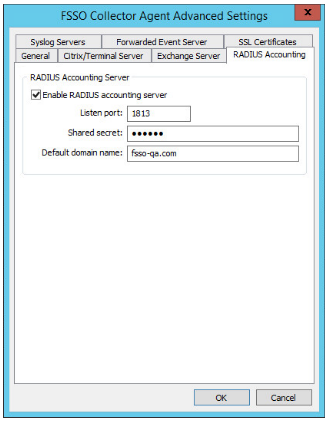
1. Go to *Authentication > RADIUS Service > Clients*.
2. Click *Create New*.
3. Set the following:
   - *Name*: *fsso_ldap*
   - *Client address*: *Range 172.16.200.1~172.16.200.10*
   - *Secret*: <enter a password>
4. In the *Realms* table, set the realm to the LDAP server that was just added: *ad_ldap_60*.
5. Click *OK*.
   FortiAuthenticator can now be used as a RADIUS server, and the authentication credentials all come from the DC/LDAP server.

Fortinet Single Sign-On Collector Agent

To configure the Fortinet Single Sign-On Collector Agent:

1. Select *Require authenticated connection from FortiGate* and enter a *Password*.
2. Click *Advanced Settings*.
3. Select the *RADIUS Accounting* tab.

4. Select *Enable RADIUS accounting server* and set the *Shared secret*.



5. Click *OK*, then click *Save&close*.
The collector agent can now accept accounting requests from FortiGate, and retrieve the IP addresses and usernames of SSL VPN client from the FortiGate with accounting request messages.

Configure the FortiGate

To configure the FortiGate in the CLI:

1. Create a Fortinet Single Sign-On Agent fabric connector:

```
config user fsso
    edit "AD_CollectAgent"
        set server "172.16.200.60"
        set password 123456
    next
end
```

2. Add the RADIUS server:

```
config user radius
    edit "rad150"
        set server "172.16.200.150"
        set secret 123456
        set acct-interim-interval 600
        config accounting-server
            edit 1
                set status enable
                set server "172.16.200.60"
                set secret 123456
            next
        end
    next
end
```

3. Create a user group for the RADIUS server:

```
config user group
    edit "rad_group"
        set member "rad150"
    next
end
```

4. Create user groups for each of the FSSO groups:

```
config user group
    edit "fsso_group1"
        set group-type fsso-service
        set member "CN=PC_GROUP1,OU=TESTING,DC=FSSO-QA,DC=COM"
    next
    edit "fsso_group2"
        set group-type fsso-service
        set member "CN=PC_GROUP2,OU=TESTING,DC=FSSO-QA,DC=COM"
    next
end
```

5. Create an SSL VPN portal and assign the RADIUS user group to it:

```
config vpn ssl web portal
    edit "testportal"
        set tunnel-mode enable
        set ipv6-tunnel-mode enable
        set web-mode enable
        ...
    next
end
config vpn ssl settings
    ...
    set default-portal "full-access"
    config authentication-rule
        edit 1
            set groups "rad_group"
            set portal "testportal"
        next
    end
end
```

6. Create firewall addresses:

```
config firewall address
    edit "none"
        set subnet 0.0.0.0 255.255.255.255
    next
    edit "pc4"
        set subnet 172.16.200.44 255.255.255.255
    next
    edit "pc5"
        set subnet 172.16.200.55 255.255.255.255
    next
end
```

7. Create one dummy policy for authentication only, and two normal policies for authorization:

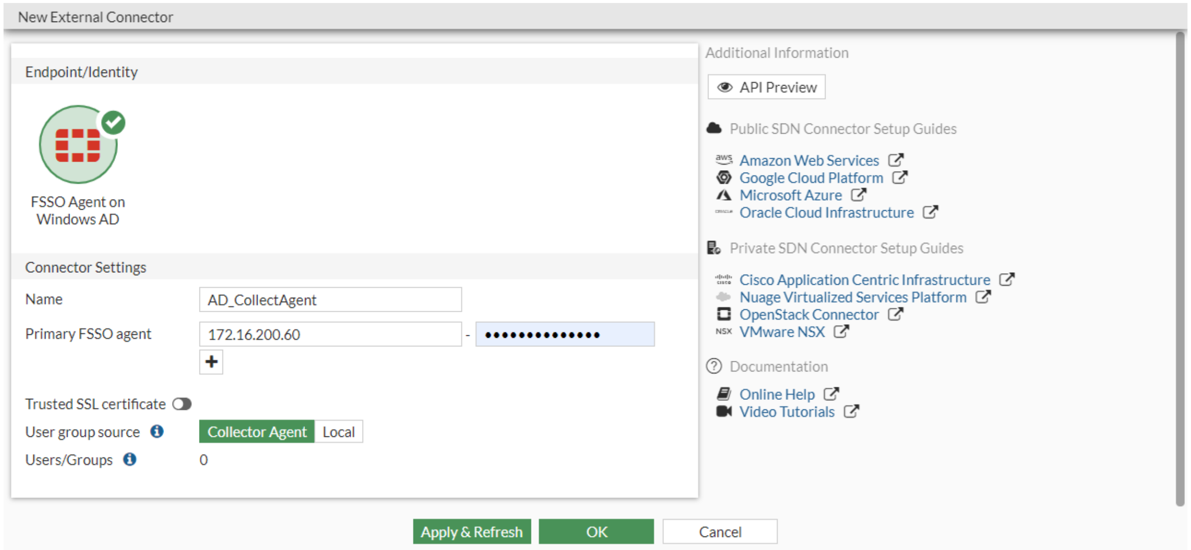
```
config firewall policy
    edit 1
        set name "sslvpn_authentication"
        set srcintf "ssl.vdom1"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "none"
        set action accept
        set schedule "always"
```

```
            set service "ALL"
            set logtraffic all
            set groups "rad_group"
            set nat enable
        next
        edit 3
            set name "sslvpn_authorization1"
            set srcintf "ssl.vdom1"
            set dstintf "port1"
            set srcaddr "all"
            set dstaddr "pc4"
            set action accept
            set schedule "always"
            set service "ALL"
            set logtraffic all
            set groups "fsso_group1"
            set nat enable
        next
        edit 4
            set name "sslvpn_authorization2"
            set srcintf "ssl.vdom1"
            set dstintf "port1"
            set srcaddr "all"
            set dstaddr "pc5"
            set action accept
            set schedule "always"
            set service "ALL"
            set logtraffic all
            set groups "fsso_group2"
            set nat enable
        next
    end
```

To create an FSSO agent fabric connector in the GUI:

1. Go to *Security Fabric > External Connectors*.
2. Click *Create New*.
3. Click *FSSO Agent on Windows AD*.

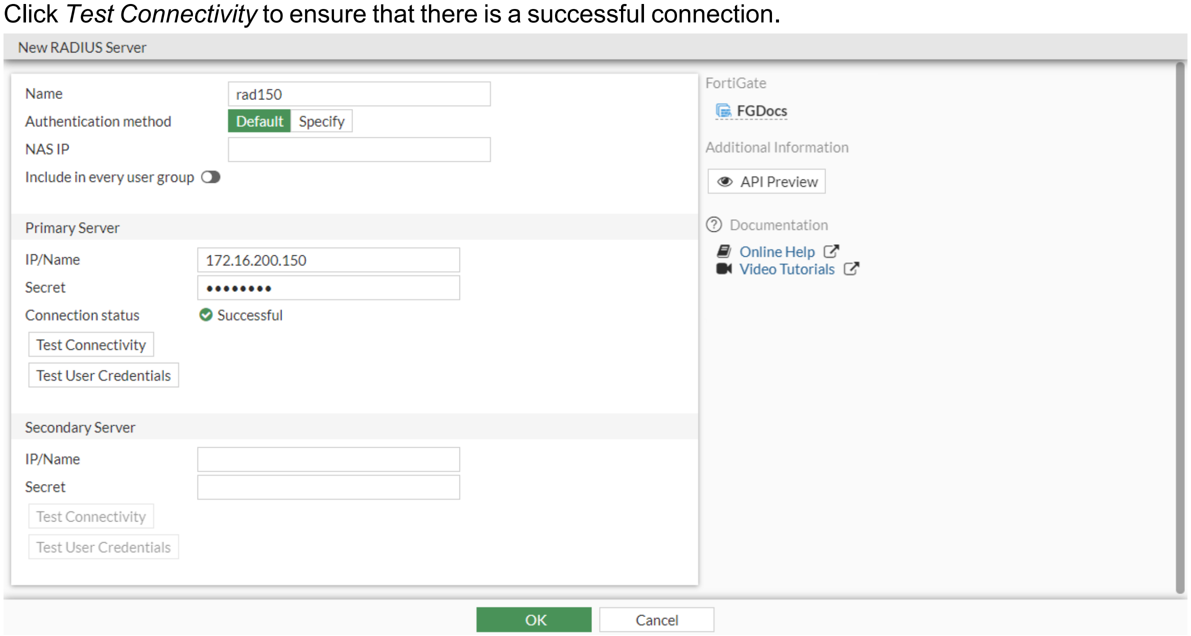
4. Enter the name and *Primary FSSO agent* information.



5. Click *Apply & Refresh*.
   The FSSO groups are retrieved from the collector agent.

To add the RADIUS server in the GUI:

1. Go to *User & Authentication > RADIUS Servers*.
2. Click *Create New*.
3. Enter a name for the server.
4. Enter the *IP/Name* and *Secret* for the primary server.
5. Click *Test Connectivity* to ensure that there is a successful connection.



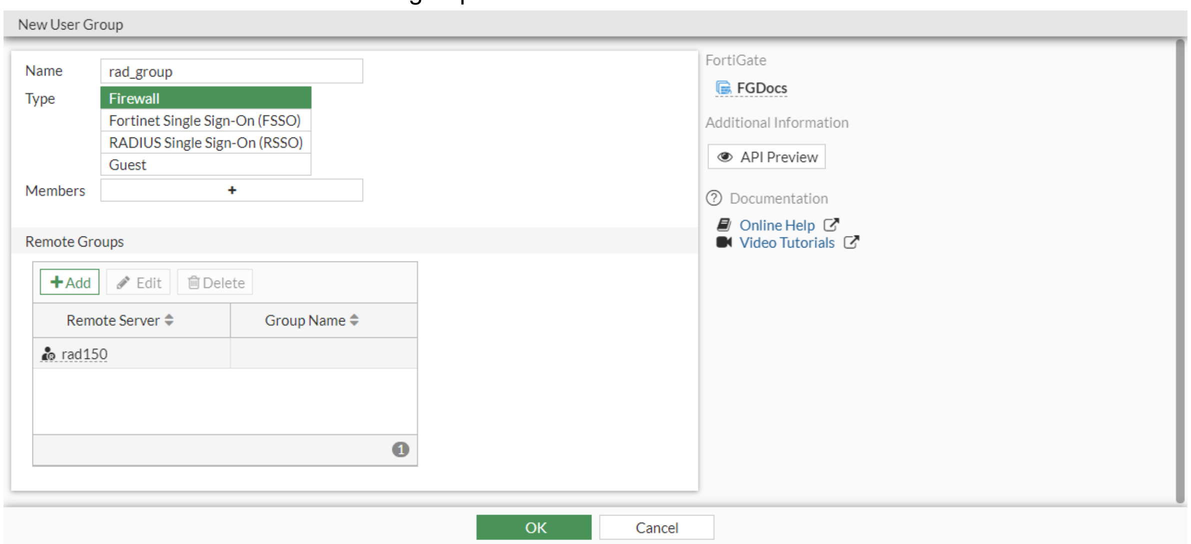
6. Click *OK*.

7. Configure an accounting server with the following CLI command:

```
config user radius
    edit rad150
        set acct-interim-interval 600
        config accounting-server
            edit 1
                set status enable
                set server 172.16.200.60
                set secret *********
            next
        end
    next
end
```

To create a user group for the RADIUS server in the GUI:

1. Go to *User & Authentication > User Groups*.
2. Click *Create New*.
3. Enter a name for the group and set the *Type* to *Firewall*.
4. Add the RADIUS server as a remote group.



5. Click *OK*.

To create user groups for each of the FSSO groups in the GUI:

1. Go to *User & Authentication > User Groups*.
2. Click *Create New*.
3. Enter a name for the group and set the *Type* to *Fortinet Single Sign-On (FSSO)*.
4. Add PC_GROUP1 as a member:
   `CN=PC_GROUP1,OU=TESTING,DC=FSSO-QA,DC=COM`
5. Click *OK*.
6. Add a second user group with PC_GROUP2 as a member:
   `CN=PC_GROUP1,OU=TESTING,DC=FSSO-QA,DC=COM`
7. Click *OK*.

To create an SSL VPN portal and assign the RADIUS user group to it in the GUI:

1. Go to *VPN > SSL VPN Portals*.
2. Click *Create New*.
3. Configure the portal, then click *OK*.
4. Go to *VPN > SSL VPN Settings*.
5. Configure the required settings.
6. Create an *Authentication/Portal Mapping* table entry:
   a. Click *Create New*.
   b. Set *User/Groups* to *rad_group*.
   c. Set *Portal* to *testportal*.
   d. Click *OK*.
7. Click *OK*.

To create policies for authentication and authorization in the GUI:

1. Go to *Policy & Objects > Firewall Policy*.
2. Configure a dummy policy for authentication. Set the destination to *none* so that traffic is not allowed through the FortiGate, and add *rad_group* as a source.
3. Configure two authorization policies, with the FSSO groups as sources.

Confirmation

On *Client 1*, log in to FortiClient using *user142*. Traffic can go to *pc4* (172.16.200.44), but cannot go to *pc5* (172.16.200.55).

On *Client 2*, log in to FortiClient using *user143*. Traffic can go to *pc5* (172.16.200.55), but cannot go to *pc4* (172.16.200.44).

On the FortiGate, check the authenticated users list and the SSL VPN status:

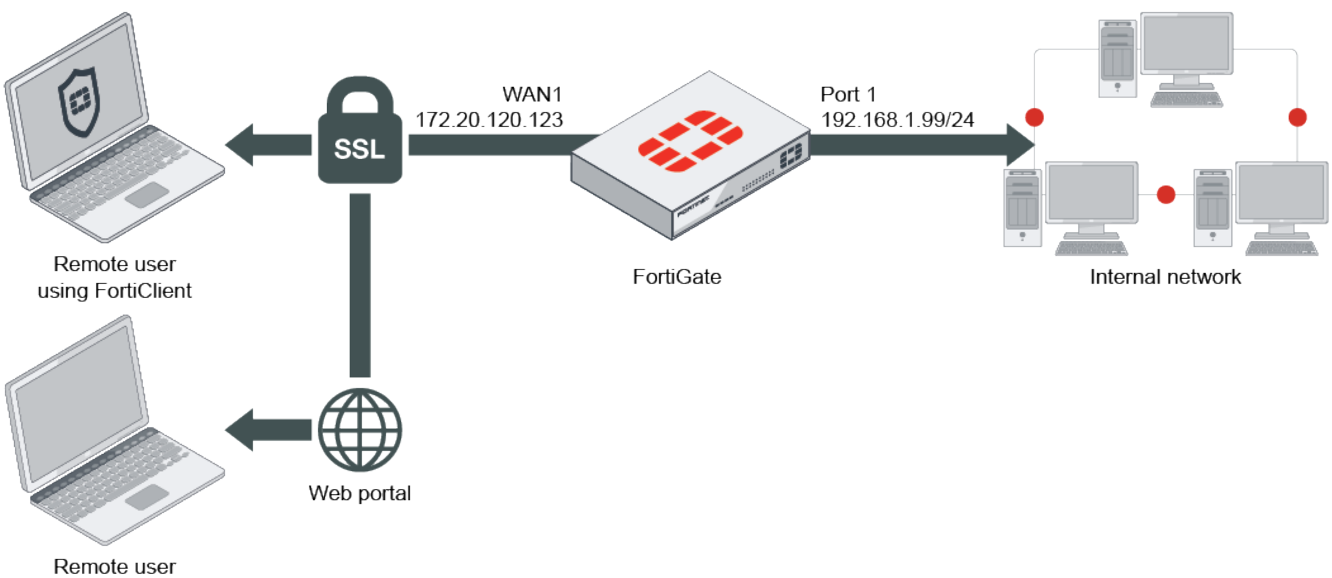
```
# diagnose firewall auth list

10.212.134.200, USER142
        type: fsso, id: 0, duration: 173, idled: 173
        server: AD_CollectAgent
        packets: in 0 out 0, bytes: in 0 out 0
        user_id: 16777229
        group_id: 3 33554434
        group_name: fsso_group1 CN=PC_GROUP1,OU=TESTING,DC=FSSO-QA,DC=COM

10.212.134.200, user142
        type: fw, id: 0, duration: 174, idled: 174
        expire: 259026, allow-idle: 259200
        flag(80): sslvpn
        server: rad150
        packets: in 0 out 0, bytes: in 0 out 0
        group_id: 4
        group_name: rad_group

10.212.134.201, USER143
        type: fsso, id: 0, duration: 78, idled: 78
        server: AD_CollectAgent
```

```
        packets: in 0 out 0, bytes: in 0 out 0
        group_id: 1 33554435
        group_name: fsso_group2 CN=PC_GROUP2,OU=TESTING,DC=FSSO-QA,DC=COM

10.212.134.201, user143
        type: fw, id: 0, duration: 79, idled: 79
        expire: 259121, allow-idle: 259200
        flag(80): sslvpn
        server: rad150
        packets: in 0 out 0, bytes: in 0 out 0
        group_id: 4
        group_name: rad_group

----- 4 listed, 0 filtered ------

# get vpn ssl monitor
SSL VPN Login Users:
 Index   User    Auth Type      Timeout         From     HTTP in/out    HTTPS in/out
 0       user142         2(1)           600      10.1.100.145   0/0      0/0
 1       user143         2(1)           592      10.1.100.254   0/0      0/0

SSL VPN sessions:
 Index   User    Source IP      Duration        I/O Bytes       Tunnel/Dest IP
 0       user142         10.1.100.145    104      32190/16480    10.212.134.200
 1       user143         10.1.100.254    11       4007/4966      10.212.134.201
```

SSL VPN multi-realm

This sample shows how to create a multi-realm SSL VPN that provides different portals for different user groups.

Sample topology



Sample configuration

WAN interface is the interface connected to ISP. This example shows static mode. You can also use DHCP or PPPoE mode. The SSL VPN connection is established over the WAN interface.

The split tunneling routing address cannot explicitly use an FQDN or an address group that includes an FQDN. To use an FQDN, leave the routing address blank and apply the FQDN as the destination address of the firewall policy.

To configure SSL VPN using the GUI:

1. Configure the interface and firewall address. The port1 interface connects to the internal network.
   a. Go to *Network > Interfaces* and edit the *wan1* interface.
   b. Set *IP/Network Mask* to *172.20.120.123/255.255.255.0*.
   c. Edit *port1* interface and set *IP/Network Mask* to *192.168.1.99/255.255.255.0*.
   d. Click *OK*.
   e. Go to *Policy & Objects > Address* and create an address for internet *QA_subnet* with subnet *192.168.1.0/24* and *HR_subnet* with subnet *10.1.100.0/24*.
2. Configure user and user group.
   a. Go to *User & Authentication > User Definition* to create local users *qa-user1* and *hr-user1*.
   b. Go to *User & Authentication > User Groups* to create separate user groups for web-only and full-access portals:
      - *QA_group* with member *qa-user1*.
      - *HR_group* with the member *hr-user1*.
3. Configure SSL VPN web portal.
   a. Go to *VPN > SSL-VPN Portals* to create portal *qa-tunnel*.
   b. Enable *Tunnel Mode*.
   c. Create a portal *hr-web* with *Web Mode* enabled.
4. Configure SSL VPN realms.
   a. Go to *System > Feature Visibility* to enable *SSL-VPN Realms*.
   b. Go to *VPN > SSL-VPN Realms* to create realms for *qa* and *hr*.
   c. (Optional) To access each realm with FQDN instead of the default URLs *https://172.20.120.123:10443/hr* and *https://172.20.120.123:10443/qa*, you can configure a virtual-host for the realm in the CLI.
      ```
      config vpn ssl web realm
         edit hr
            set virtual-host hr.mydomain.com
         next
         edit qa
            set virtual-host qa.mydomain.com
         next
      end
      ```
      Where `mydomain.com` is the name of your domain. Ensure FQDN resolves to the FortiGate wan1 interface and that your certificate is a wildcard certificate.
5. Configure SSL VPN settings.
   a. Go to *VPN > SSL-VPN Settings*.
   b. For *Listen on Interface(s)*, select *wan1*.
   c. Set *Listen on Port* to *10443*.
   d. Choose a certificate for *Server Certificate*. The default is *Fortinet_Factory*.

e. In *Authentication/Portal Mapping All Other Users/Groups*, set the *Portal* to *web-access*.
f. Create new *Authentication/Portal Mapping* for group *QA_group* mapping portal *qa-tunnel*.
g. Specify the realm *qa*.
h. Add another entry for group *HR_group* mapping portal *hr-web*.
i. Specify the realm *hr*.

6. Configure SSL VPN firewall policy.
   a. Go to *Policy & Objects > Firewall Policy*.
   b. Create a firewall policy for QA access.
   c. Fill in the firewall policy name. In this example, *QA sslvpn tunnel mode access*.
   d. Incoming interface must be *SSL-VPN tunnel interface(ssl.root)*.
   e. Choose an *Outgoing Interface*. In this example, *port1*.
   f. Set the *Source* to *all* and group to *QA_group*.
   g. In this example, the *Destination* is the internal protected subnet *QA_subnet*.
   h. Set *Schedule* to *always*, *Service* to *ALL*, and *Action* to *Accept*.
   i. Click *OK*.
   j. Create a firewall policy for HR access.
   k. Fill in the firewall policy name. In this example, *HR sslvpn web mode access*.
   l. Incoming interface must be *SSL-VPN tunnel interface(ssl.root)*.
   m. Choose an *Outgoing Interface*. In this example, *port1*.
   n. Set the *Source* to *all* and group to *HR_group*.
   o. In this example, the *Destination* is the internal protected subnet *HR_subnet*.
   p. Set *Schedule* to *always*, *Service* to *ALL*, and *Action* to *Accept*.
   q. Click *OK*.

To configure SSL VPN using the CLI:

1. Configure the interface and firewall address.

```
config system interface
    edit "wan1"
        set vdom "root"
        set ip 172.20.120.123 255.255.255.0
    next
end
```

2. Configure internal interface and protected subnet, then connect the port1 interface to the internal network.

```
config system interface
    edit "port1"
        set vdom "root"
        set ip 192.168.1.99 255.255.255.0
    next
end
```

```
config firewall address
    edit "QA_subnet"
        set subnet 192.168.1.0 255.255.255.0
    next
    edit "HR_subnet"
        set subnet 10.1.100.0 255.255.255.0
```

```
    next
end
```

3. Configure user and user group.

```
config user local
    edit "qa_user1"
        set type password
        set passwd your-password
    next
end
config user group
    edit "QA_group"
        set member "qa_user1"
    next
end
```

```
config user local
    edit "hr_user1"
        set type password
        set passwd your-password
    next
end
config user group
    edit "HR_group"
        set member "hr_user1"
    next
end
```

4. Configure SSL VPN web portal.

```
config vpn ssl web portal
    edit "qa-tunnel"
        set tunnel-mode enable
        set ip-pools "SSLVPN_TUNNEL_ADDR1"
        set split-tunneling enable
        set split-tunneling-routing-address "QA_subnet"
    next
end
```

```
config vpn ssl web portal
    edit "hr-web"
        set web-mode enable
    next
end
```

5. Configure SSL VPN realms.

```
config vpn ssl web realm
   edit hr
     set virtual-host hr.mydomain.com
   next
   edit qa
        set virtual-host qa.mydomain.com
   next
end
```

The `set virtual-host` setting is optional. For example:

```
config vpn ssl web realm
   edit hr
   next
```

```
        edit qa
        next
    end
```

6. Configure SSL VPN settings.

```
config vpn ssl settings
    set servercert "Fortinet_Factory"
    set tunnel-ip-pools "SSLVPN_TUNNEL_ADDR1"
    set tunnel-ipv6-pools "SSLVPN_TUNNEL_IPv6_ADDR1"
    set source-interface "wan1"
    set source-address "all"
    set source-address6 "all"
    set default-portal "full-access"
    config authentication-rule
        edit 1
            set groups "QA_group"
            set portal "qa-tunnel"
            set realm qa
        next
        edit 2
            set groups "HR_group"
            set portal "hr-web"
            set realm hr
        next
    end
end
```

7. Configure two SSL VPN firewall policies to allow remote QA user to access internal QA network and HR user to access HR network.

```
config firewall policy
    edit 1
        set name "QA sslvnpn tunnel access"
        set srcintf "ssl.root"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "QA_subnet"
        set groups “QA_group”
        set action accept
        set schedule "always"
        set service "ALL"
    next
    edit 2
        set name "HR sslvpn web access"
        set srcintf "ssl.root"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "HR_subnet"
        set groups “HR_group”
        set action accept
        set schedule "always"
        set service "ALL"
    next
end
```

To see the results for QA user:

1. Download FortiClient from www.forticlient.com.
2. Open the FortiClient Console and go to *Remote Access*.
3. Add a new connection.
   - Set *VPN Type* to *SSL VPN*.
   - Set *Remote Gateway* to *https://172.20.120.123:10443/qa*..
   - If a virtual-host is specified, use the FQDN defined for the realm (*qa.mydomain.com*).
4. Select *Customize Port* and set it to *10443*.
5. Save your settings.
6. Use the credentials you've set up to connect to the SSL VPN tunnel.
   If the user's computer has antivirus software, a connection is established; otherwise FortiClient shows a compliance warning.
7. After connection, traffic to subnet *192.168.1.0* goes through the tunnel.
8. On the FortiGate, go to *Dashboard* > *Network* and expand the *SSL-VPN* widget to verify the list of SSL users.
9. On the FortiGate, go to *VPN* > *Monitor* > *SSL-VPN Monitor* to verify the list of SSL users.
10. On the FortiGate, go to *Log & Report* > *Forward Traffic* and view the details of the traffic.

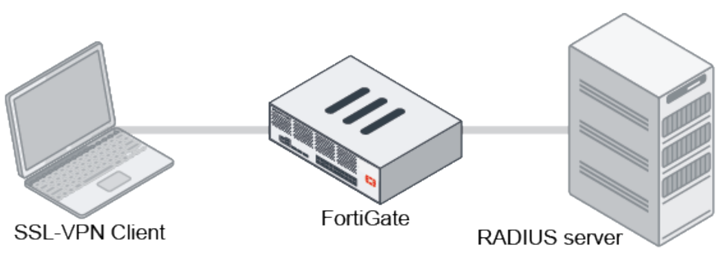
To see the results for HR user:

1. In a web browser, log into the portal *https://172.20.120.123:10443/hr* using the credentials you've set up.
2. Alternatively, if a virtual-host is specified, use the FQDN defined for the realm (*hr.mydomain.com*).
3. On the FortiGate, go to *Dashboard* > *Network* and expand the *SSL-VPN* widget to verify the list of SSL users.
4. Go to *Log & Report* > *Forward Traffic* and view the details of the traffic.

NAS-IP support per SSL-VPN realm

For RADIUS authentication and authorization, the RADIUS client (the FortiGate) passes the username, password, and NAS-IP to the RADIUS server in its access request. The RADIUS server authenticates and authorizes based on this information. Each RADIUS server can be configured with multiple NAS-IPs for authenticating different groups and NAS clients.

On the FortiGate, configuring the NAS-IP in the realm settings overrides the RADIUS server setting, allowing multiple NAS-IPs to be mapped to the same RADIUS server.



In this example, the user wants to present one FortiGate VDOM with different NAS-IPs to a single RADIUS server based on specific rules.

To configure the SSL-VPN to use the NAS-IP in the realm settings:

1. Configure a RADIUS user and add it to a group:

```
config user radius
    edit "fac150"
        set server "172.16.200.150"
        set secret ********
        set nas-ip 172.16.200.2
        config accounting-server
            edit 1
                set status enable
                set server "172.16.200.150"
                set secret ********
            next
        end
    next
end
config user group
    edit "radgrp"
        set member "fac150"
    next
end
```

2. Configure a realm for the user with a different NAS-IP:

```
config vpn ssl web realm
    edit "realm1"
        set login-page '.......'
        set radius-server "fac150"
        set nas-ip 10.1.100.2
    next
end
```

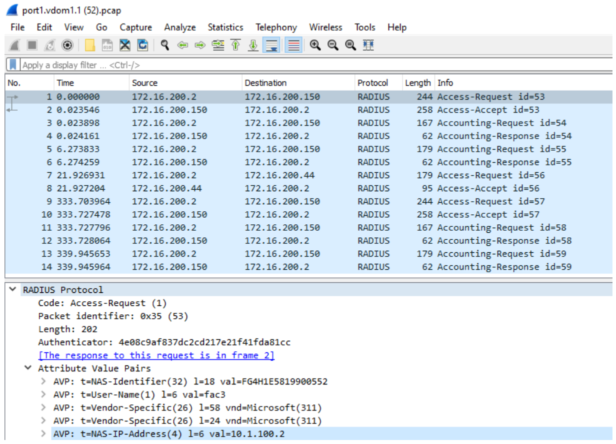
3. Configure SSL-VPN with an authentication rule that includes the user group and the realm:

```
config vpn ssl settings
    ...
    config authentication-rule
        edit 1
            set groupd "radgrp"
            set portal "testportal1"
            set realm "realm1"
        next
    end
end
```

4. Create a firewall policy:

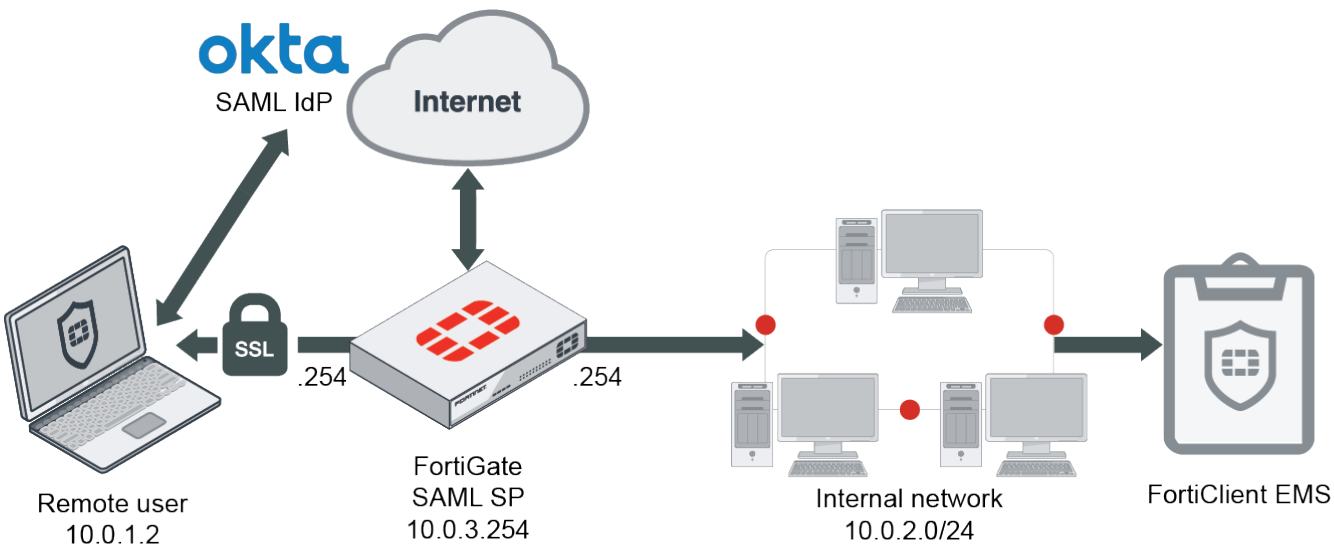
```
config firewall policy
    edit 1
        set name "sslvpn1"
        ...
        set srcintf "ssl.vdom1"
        set groups "radgrp"
    next
end
```

Because the RADIUS server and NAS-IP are specified in realm1, its NAS-IP is used for authentication.



SSL VPN with Okta as SAML IdP

In this configuration, the FortiGate acts as a SAML service provider (SP) requesting authentication from Okta, which acts as a SAML identity provider (IdP). The following shows the topology in this configuration:



The authentication process is as follows in this deployment:

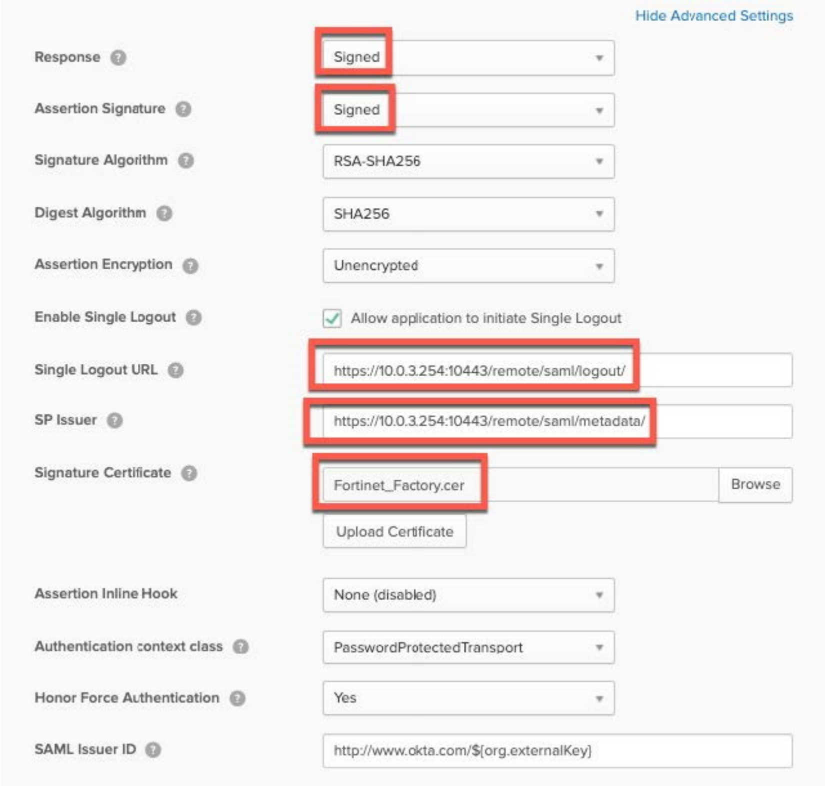
1. The user initiates an SSL VPN request to the FortiGate.
2. The FortiGate sends the browser POST redirect to FortiClient.
3. FortiClient redirects the SAML authentication request to Okta.
4. The user authenticates with Okta using their credentials.
5. Okta sends a SAML assertion that contains the user and group authentication in a POST redirect to the SSL VPN login page.
6. FortiClient sends the redirected Okta request that contains the SAML assertion to the FortiGate.
7. The FortiGate consumes the assertion and provides the user with access to resources based on the defined firewall security policy.

The example assumes that you already have an Okta account. This example uses users locally defined within the Okta directory and does not include LDAP mapping. The instructions describe the steps that you take if using the free Okta developer edition.

To configure Okta for SSL VPN with FortiOS:

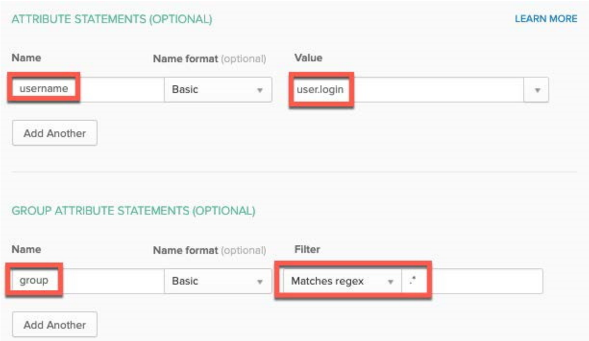
1. Log in to the Okta portal as the registered admin user.
2. Add the FortiGate application:
   a. Go to *Applications*.
   b. Click *Applications*, then click *Create App Integration*.
   c. Click *SAML 2.0*, then *Next*.
   d. Configure SAML settings:
      i. Proceed through the application creation wizard. In the *Single sign on URL* field, enter https://<FortiGate IP address>:<port>/remote/saml/login/. In this example, it is https://10.0.3.254:10443/remote/saml/login/.
      ii. Enable *Use this for Recipient URL and Destination URL*.
      iii. In the *Audience URI (SP Entity ID)* field, enter the https://<FortiGate IP address>:<port>/remote/saml/metadata/. In this example, it is https://10.0.3.254:10443/remote/saml/metadata/.
      iv. Click *Download Okta Certificate* to download the Okta certificate to your machine. You provide this certificate to the FortiGate.
      v. Click *Show Advanced Settings*. From the *Response* dropdown list, select *Signed*.
      vi. From the *Assertion Signature* dropdown list, select *Signed*.
      vii. In the *Single Logout URL* field, enter https://<FortiGate IP address>:<port>/remote/saml/logout/. In this example, it is https://10.0.3.254:10443/remote/saml/logout/.
      viii. In the *SP Issuer* field, enter https://<FortiGate IP address>:<port>/remote/saml/metadata/. In this example, it is https://10.0.3.254:10443/remote/saml/metadata/.
      ix. In the *Signature Certificate* field, first download the Fortinet_Factory certificate by logging into FortiOS, going to *System > Local Certificate*, then browsing to and uploading the FortiGate certificate. Okta uses

this to authenticate the SAML SP.



e. Under *ATTRIBUTE STATEMENTS* and *GROUP ATTRIBUTE STATEMENTS*, define attribute mappings for Okta to use in SAML assertion. In this example, the following is entered as a attribute statement and a group attribute statement, respectively:

- username, with value user.login
- group, with *Matches regex* filter



f. On the *Feedback* step, select *I'm an Okta customer adding an internal app*.

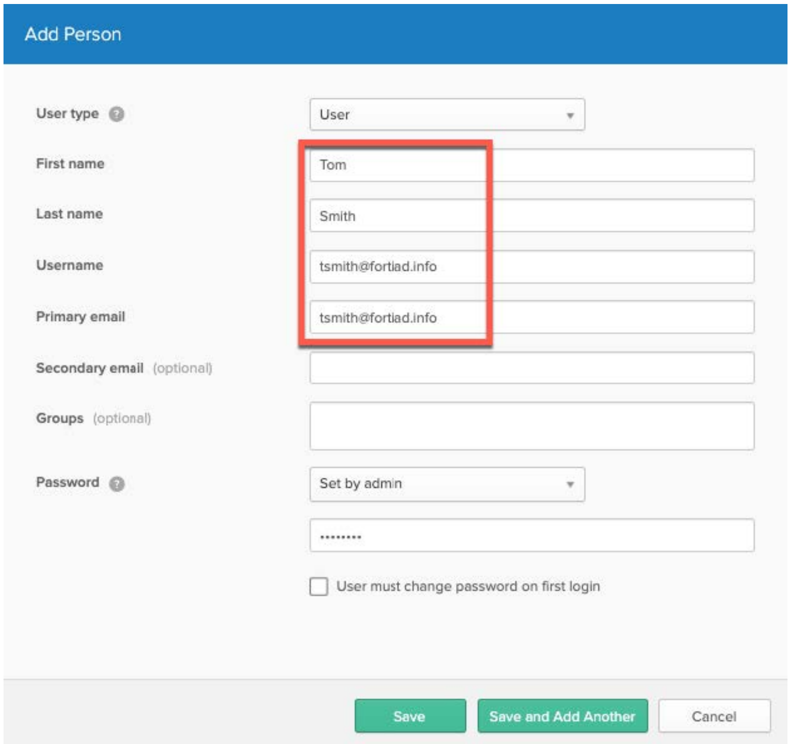
g. Select *This is an internal app that we have created*.

h. Click *Finish*.

3. Go to *Directory* > *People*.

4. Click *Add Person*.

5. Enter the person's details as desired. Click *Save*.



6. Add a group:
   a. Go to *Directory* > *Groups*.
   b. Click *Add Group*.
   c. Enter the desired name, then click *Add Group*. In this example, the name is corporate-saml.
   d. Select the newly added group, then click *Assign People*.
   e. Add the person that you created as a member of the new group. Click *Save*.
7. Assign the group to the FortiGate application:
   a. Go to *Applications* > *FortiGate application* > *Assignments*.
   b. From the *Assign* dropdown list, select *Assign to Groups*.
   c. Assign the group that you created to the FortiGate application.
8. To view the SAML setup instructions, do the following:
   a. Click the newly created application's name.
   b. Click *Sign On*.
   c. Go to *View SAML Setup Instructions*. Note down the *Identity Provider Single Sign-On URL*, *Identity Provider Single Logout URL*, and *Identity Provider Issuer* values.
9. Download the Okta certificate and upload it to FortiOS:
   a. From *View SAML Setup Instructions*, download the certificate.
   b. In FortiOS, go to *System* > *Certificates*.
   c. From the *Create/Import* dropdown list, select *Remote Certificate*.
   d. Click *Upload* and upload the downloaded Okta certificate.

To configure the FortiGate:

1. Configure the FortiGate SP to be a SAML user:

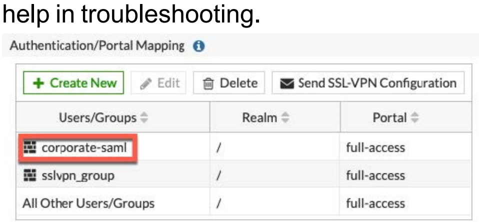
```
config user saml
```

```
    edit "okta-idp"
        set cert "Fortinet_Factory"
        set entity-id "https://10.0.3.254:10443/remote/saml/metadata/"
        set single-sign-on-url "https://10.0.3.254:10443/remote/saml/login"
        set single-logout-url "https://10.0.3.254:10443/remote/saml/logout"
        set idp-entity-id "http://www.okta.com/exk103foxaa8gk5qy4x7"
        set idp-single-sign-on-url "https://fortinet01.okta.com/app/fortinetorg878484_
            fortigate_1/exk103foxaa8gk5qy4x7/sso/saml"
        set idp-single-logout-url "https://fortinet01.okta.com/app/fortinetorg878484_
            fortigate_1/exk103foxaa8gk5qy4x7/slo/saml"
        set idp-cert "Okta-IDP_Certificate"
        set user-name "username"
        set group-name "group"
    next
end
```

2. Configure user group assertion on Okta as part of the SAML assertion attributes. It is important that the group attribute value received is locally matched with the `group-name` value:

```
config user group
    edit "corporate-saml"
        set member "okta-idp"
        config match
            edit 1
                set server-name "okta-idp"
                set group-name "corporate-saml"
            next
        end
    next
end
```

3. Go to *VPN* > *SSL-VPN Settings*. Configure VPN settings as desired. When testing the VPN solution, starting with a web-based configuration, then moving to a tunnel-based configuration is recommended. Web-based testing can help in troubleshooting.



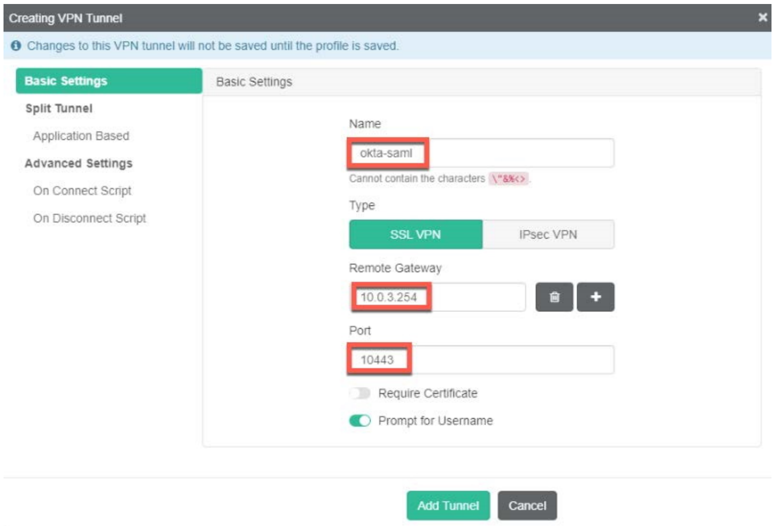
4. Configure a local or RADIUS user as a backup. This setting also provides a login web user with a choice of local or SSO login.
5. Go to *Policy and Objects* > *Firewall Policies*. Configure a policy as desired.
6. Increase the global authentication timeout period to allow users to fill in their credentials in time. The default timeout is five seconds:

```
config system global
    set remoteauthtimeout 60
end
```

To configure EMS:

1. In EMS, go to *Endpoint Profiles* > *Manage Profiles*. Edit a VPN profile.
2. Under *VPN Tunnels*, click *Add Tunnel*.
3. In the *Remote Gateway* field, enter the FortiGate IP address. In this example, it is 10.0.3.254.

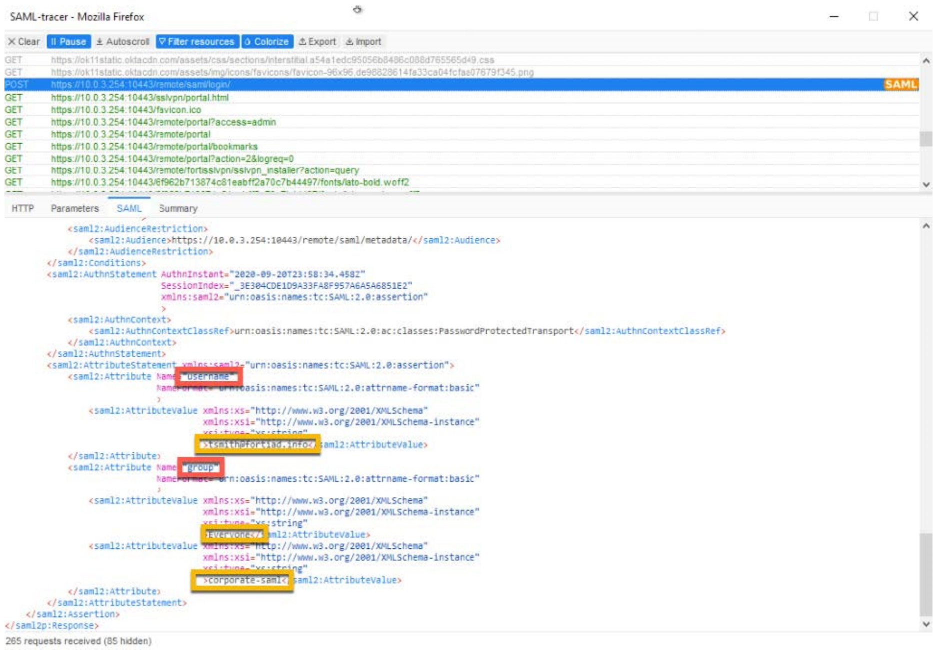
4. In the *Port* field, enter the port number. In this example, it is 10443.



5. In *Advanced Settings*, enable *Enable SAML Login*.
6. Click *Add Tunnel*.
7. Save the profile.
8. After the policy synchronizes to the endpoint, the *SAML Login* button is visible on the Remote Access tab in FortiClient.

To test the configuration:

1. Test web-based SSL VPN authentication using Firefox with the SAML tracer plugin enabled. Install the SAML-tracer plugin to Firefox.
2. In Firefox, go to the FortiOS SSL VPN login page. In this example, this is https://10.0.3.254:10443.
3. Open the SAML tracer.
4. The browser redirects to the Okta SAML login page. Enter the Okta credentials, then click *Sign in*.
5. Upon successful authentication, the browser redirects to the authenticated SSL VPN page. If authentication does not succeed, review the SAML tracer to confirm the SAML assertion attributes that are passed during the authentication session. Select the POST message with the SAML information. On the SAML tab, confirm the username and group attributes.

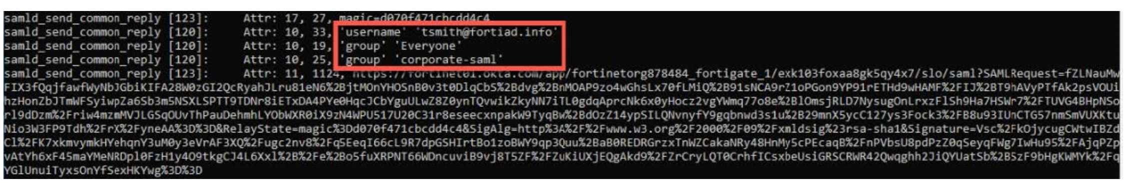


6. To test tunnel mode, go to the *Remote Access* tab in FortiClient. Click the *SAML Login* button.
7. A FortiAuthenticator web login page opens within FortiClient. Enter the Okta credentials, then log in to connect to the VPN tunnel.

To troubleshoot the configuration:

You can view FortiOS event logs in *Log & Report > Events* to verify successful authentication and user group allocation.

You can also run the `diagnose debug application samld -1` command to verify that the SAML IdP sent the correct information. The following shows example output for this scenario:



SSL VPN with Azure AD SSO integration

You can use SAML single sign on to authenticate against Azure Active Directory with SSL VPN SAML user via tunnel and web modes. See:

- Configuring SAML SSO login for SSL VPN with Azure AD acting as SAML IdP
- Tutorial: Azure AD SSO integration with FortiGate SSL VPN

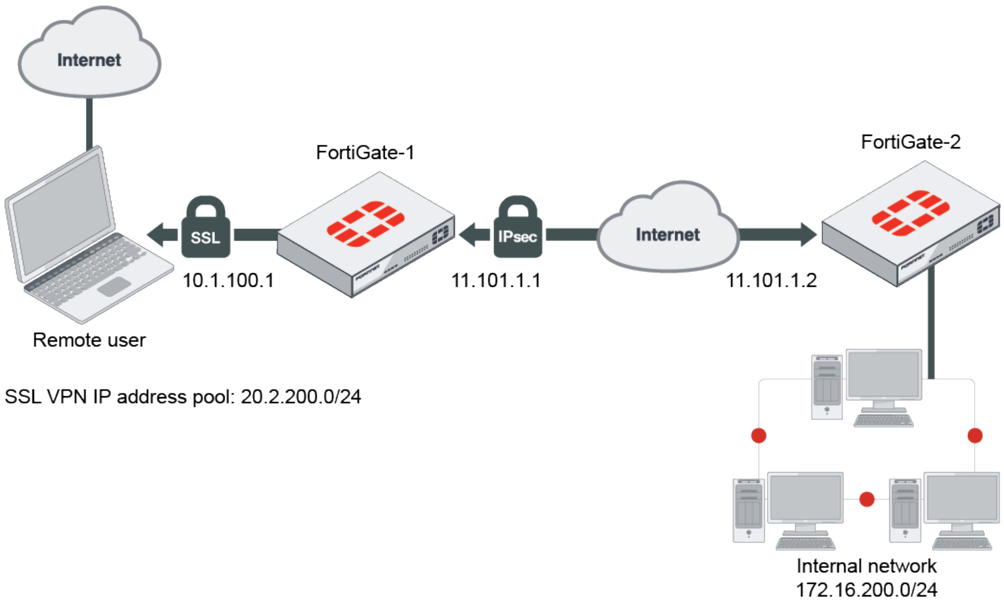
SSL VPN to IPsec VPN

This is a sample configuration of site-to-site IPsec VPN that allows access to the remote endpoint via SSL VPN.

This example uses a pre-existing user group, a tunnel mode SSL VPN with split tunneling, and a route-based IPsec VPN between two FortiGates. All sessions must start from the SSL VPN interface.

If you want sessions to start from the FGT_2 subnet, you need more policies. Also, if the remote subnet is beyond FGT_2 (if there are multiple hops), you need to include the SSL VPN subnet in those routers as well.

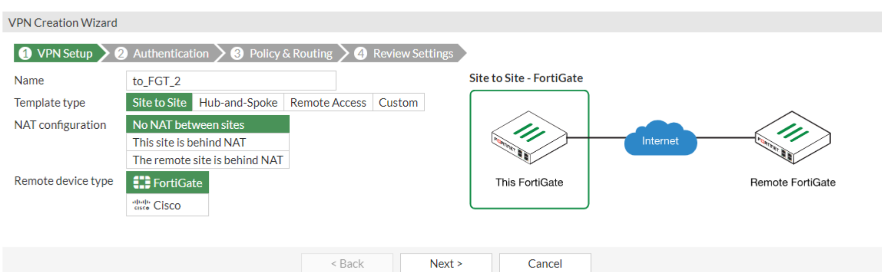
Sample topology



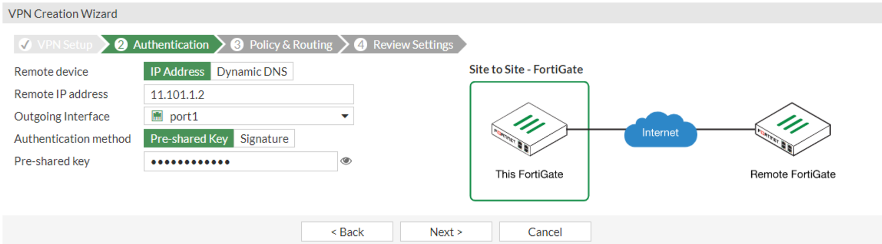
Sample configuration

To configure the site-to-site IPsec VPN on FGT_1:

1. Go to *VPN > IPsec Wizard*.
2. In the *VPN Setup* pane:
   a. Specify the VPN connection *Name* as *to_FGT_2*.
   b. Select *Site to Site*.

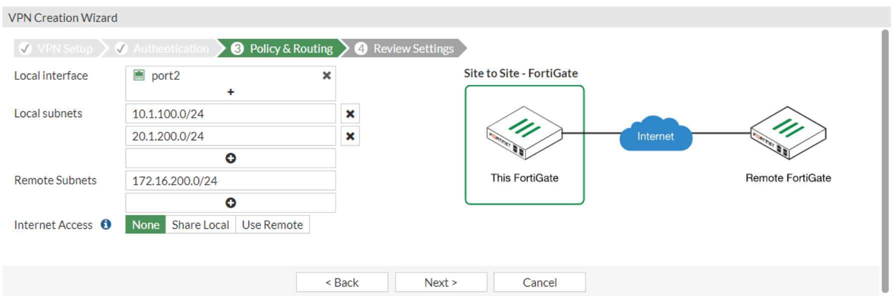


   c. Click *Next*.
3. In the *Authentication* pane:
   a. Enter the *IP Address* to the Internet-facing interface.
   b. For *Authentication Method*, click *Pre-shared Key* and enter the *Pre-shared Key*.

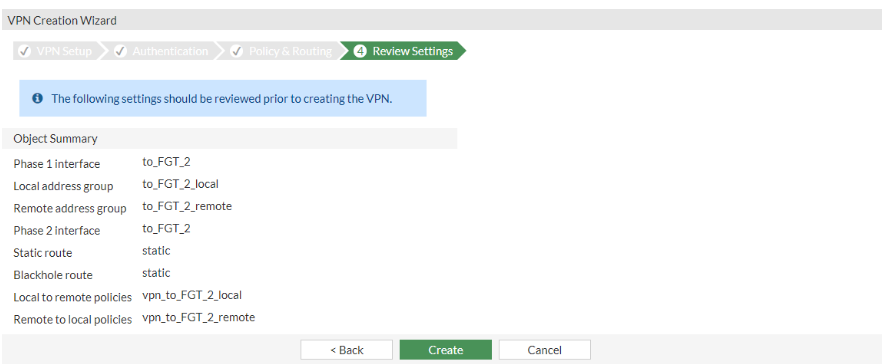


c. Click *Next*.

4. In the *Policy & Routing* pane:
   a. Set the *Local Interface* to the internal interface.
   b. Set the *Local Subnets* to include the internal and SSL VPN subnets for FGT_1.
   c. Set *Remote Subnets* to include the internal subnet for FGT_2.

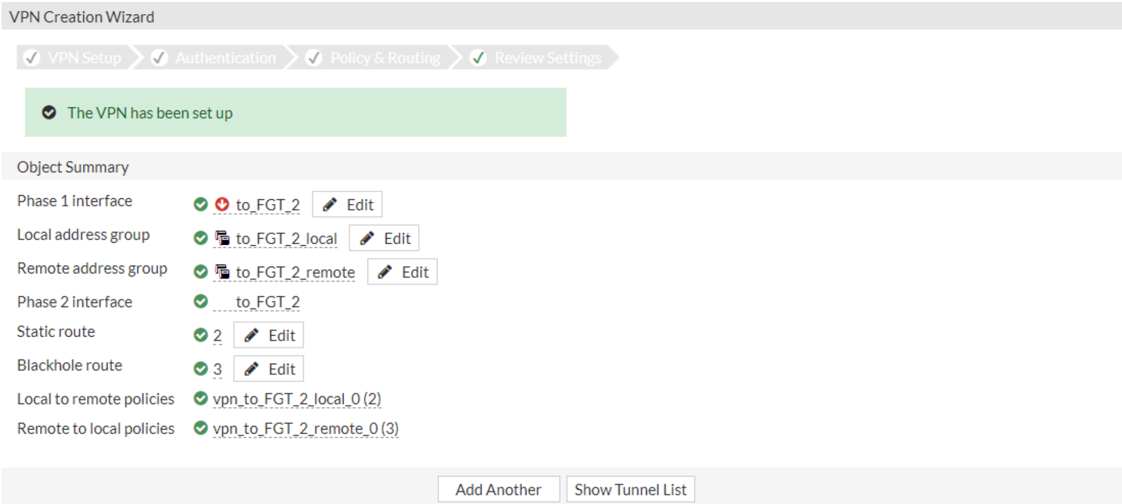


   d. Click *Next*.



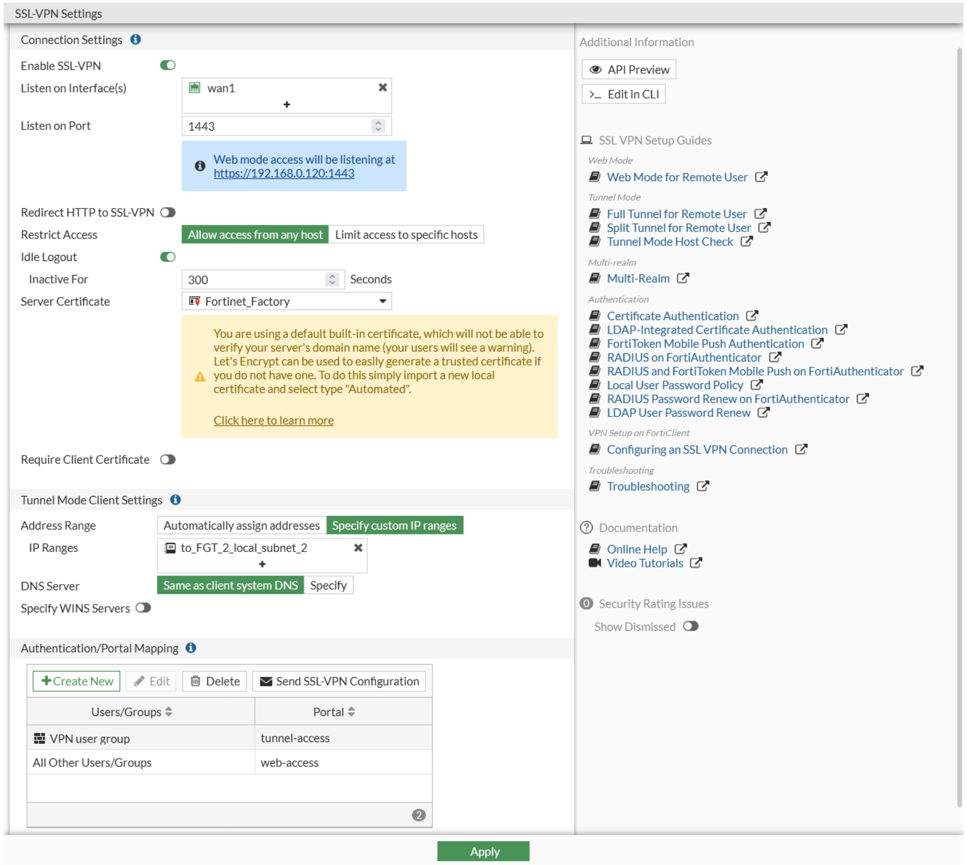
5. Review the VPN settings and click *Create*.

A confirmation screen shows a summary of the configuration including the firewall address groups for both the local and remote subnets, static routes, and security policies.



To configure SSL VPN settings:

1. Go to *VPN > SSL-VPN Settings*.
2. Set *Listen on Interface(s)* to *wan1*.
3. To avoid port conflicts, set *Listen on Port* to *10443*.
4. Set *Restrict Access* to *Allow access from any host*.
5. In the Tunnel Mode Client Settings section, select *Specify custom IP ranges* and include the SSL VPN subnet range created by the *IPsec Wizard*.
6. In the *Authentication/Portal Mapping* section, add the *VPN user group* to the *tunnel-access Portal*. Set *All Other Users/Groups* to the *web-access Portal*.

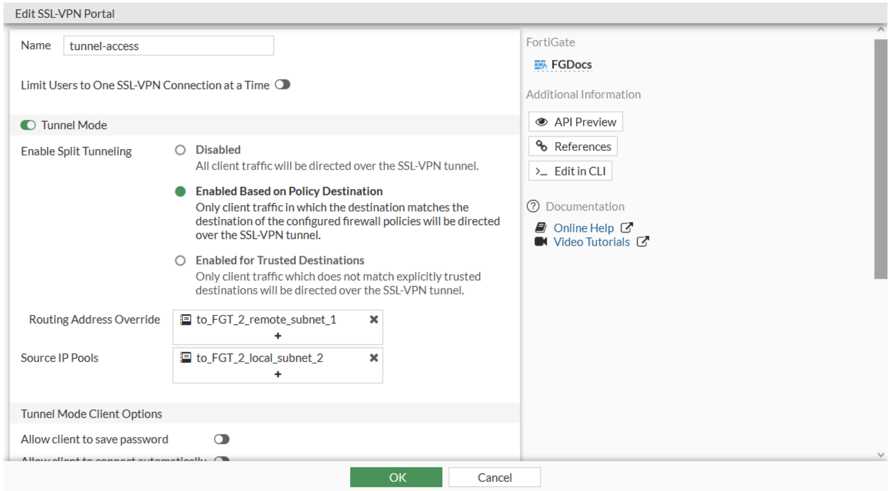


It is **HIGHLY** recommended that you acquire a signed certificate for your installation. Please review the SSL VPN best practices on page 1617 and learn how to Procuring and importing a signed SSL certificate on page 2173.

7. Click *Apply*.

To configure SSL VPN portal:

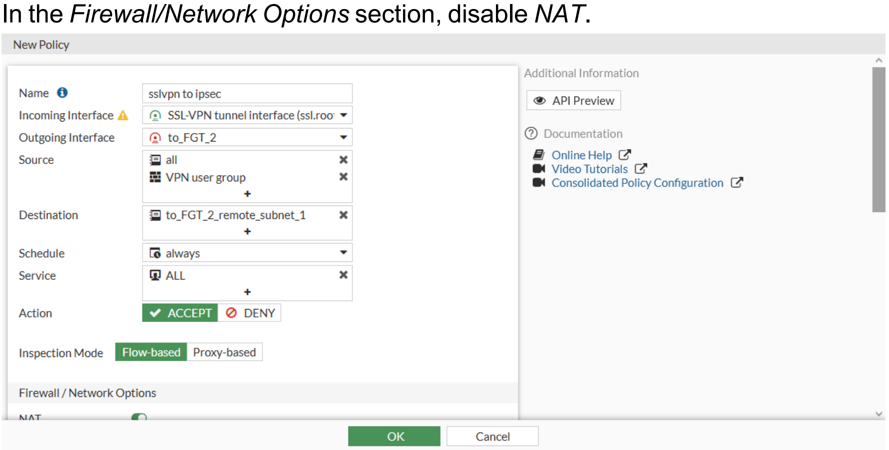
1. Go to *VPN > SSL-VPN Portals*.
2. Select *tunnel-access* and click *Edit*.
3. Turn on *Enable Split Tunneling* so that only traffic intended for the local or remote networks flow through FGT_1 and follows corporate security profiles.
4. For *Routing Address*, add the local and remote IPsec VPN subnets created by the *IPsec Wizard*.
5. For *Source IP Pools*, add the SSL VPN subnet range created by the *IPsec Wizard*.



6. Click *OK*.

To add policies to FGT_1:

1. Go to *Policy & Objects > Firewall Policy*.
2. Click *Create New* to create a policy that allows SSL VPN users access to the IPsec VPN tunnel.
3. For *Incoming Interface*, select *ssl.root*.
4. For *Outgoing Interface*, select the IPsec tunnel interface *to_FGT_2*.
5. Set the *Source* to *all* and the *VPN user group*.
6. Set *Destination* to the remote IPsec VPN subnet.
7. Specify the *Schedule*.
8. Set the *Service* to *ALL*.
9. In the *Firewall/Network Options* section, disable *NAT*.



10. Click *OK*.

To configure the site-to-site IPsec VPN on FGT_2:

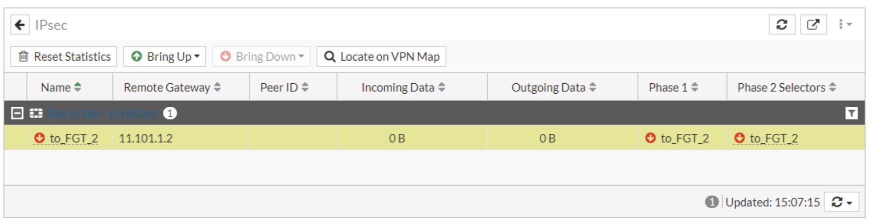
1. Go to *VPN > IPsec Wizard*.
2. In the *VPN Setup* pane:
   a. Specify the VPN connection *Name* as *to FGT_1*.
   b. Select *Site to Site*.

  c. Click *Next*.
3. In the *Authentication* pane:
  a. Enter the *IP Address* to the Internet-facing interface.
  b. For *Authentication Method*, click *Pre-shared Key* and enter the *Pre-shared Key* of the FGT_1.
  c. Click *Next*.
4. In the *Policy & Routing* pane:
  a. Set the *Local Interface* to the internal interface.
  b. Set the *Local Subnets* to include the internal and SSL VPN subnets for FGT_2.
  c. Set *Remote Subnets* to include the internal subnet for FGT_1.
  d. Click *Create*.

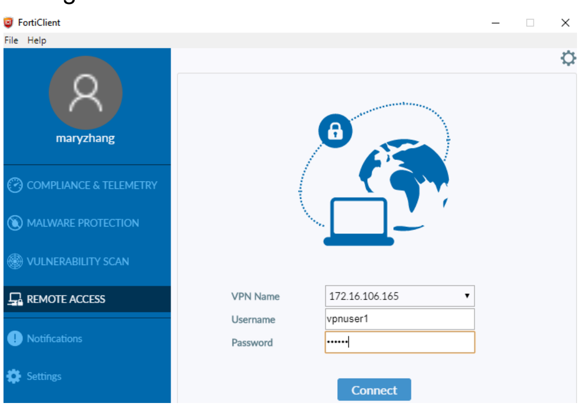
A confirmation screen shows a summary of the configuration including the firewall address groups for both the local and remote subnets, static routes, and security policies.

To check the results:

1. Go to *Dashboard > Network* and click the *IPsec* widget to expand to full screen view.
2. Select the tunnel and click *Bring Up*.



3. Verify that the *Status* changes to *Up*.
4. Configure the SSL VPN connection on the user's FortiClient and connect to the tunnel.



5. On the user's computer, send a ping though the tunnel to the remote endpoint to confirm access:

```
C:\>ping 172.16.200.55

Pinging 172.16.200.55 with 32 bytes of data:
Replay from 172.16.200.55: bytes=32 times=2ms TTL=62
Replay from 172.16.200.55: bytes=32 times=1ms TTL=62
Replay from 172.16.200.55: bytes=32 times=1ms TTL=62
Replay from 172.16.200.55: bytes=32 times=1ms TTL=62

Ping statistics for 172.16.200.55:
```

```
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip time in milli-seconds:
    Minimum = 1ms, Maximum = 2ms, Average = 1ms
```

6. In FortiOS, go to the following pages for further verification:
   a. Go to *Dashboard > Network* and click the *Routing* widget to verify the IPsec and SSL VPNs are added.
   b. Go to *VPN > SSL-VPN Clients* to verify the connected users.
   c. Go to *VPN > VPN Location Map* to view the connection activity.
   d. Go to *Log & Report > System Events* and select the *VPN Events* card to view tunnel statistics.
   e. Go to *Dashboard > FortiView Policies* to view the policy usage.

Troubleshooting

To troubleshoot on FGT_1, use the following CLI commands:

```
diagnose debug reset
diagnose debug flow show function-name enable
diagnose debug flow show iprope enable
diagnose debug flow filter addr 172.16.200.55
diagnose debug flow filter proto 1
diagnose debug flow trace start 2
diagnose debug enable
```

To troubleshoot using ping:

1. Send a ping through the SSL VPN tunnel to 172.16.200.55 and analyze the output of the debug.
2. Disable the debug output with: `diagnose debug disable`.

If traffic is entering the correct VPN tunnel on FGT_1, then run the same commands on FGT_2 to check whether the traffic is reaching the correct tunnel. If it is reaching the correct tunnel, confirm that the SSL VPN tunnel range is configured in the remote side quick mode selectors.

To troubleshoot using a sniffer command:

```
diagnose sniff packet any "host 172.16.200.44 and icmp" 4
```

To troubleshoot IPsec VPN issues, use the following commands on either FortiGate:

```
diagnose debug reset
diagnose vpn ike gateway clear
diagnose debug application ike -1
diagnose debug enable
```

SSL VPN protocols

The following topics provide information about SSL VPN protocols:

- TLS 1.3 support on page 1744
- SMBv2 support on page 1744

TLS 1.3 support

FortiOS supports TLS 1.3 for SSL VPN.

TLS 1.3 support requires IPS engine 4.205 or later and endpoints running FortiClient 6.2.0 or later.

To establish a client SSL VPN connection with TLS 1.3 to the FortiGate:

1. Enable TLS 1.3 support using the CLI:
```
config vpn ssl setting
   set ssl-max-proto-ver tls1-3
   set ssl-min-proto-ver tls1-3
end
```
2. Configure the SSL VPN and firewall policy:
   a. Configure the SSL VPN settings and firewall policy as needed.
3. For Linux clients, ensure OpenSSL 1.1.1a is installed:
   a. Run the following commands in the Linux client terminal:
   ```
   root@PC1:~/tools# openssl
   OpenSSL> version
   ```
   If OpenSSL 1.1.1a is installed, the system displays a response like the following:
   ```
   OpenSSL 1.1.1a 20 Nov 2018
   ```
4. For Linux clients, use OpenSSL with the TLS 1.3 option to connect to SSL VPN:
   a. Run the following command in the Linux client terminal:
   ```
   #openssl s_client -connect 10.1.100.10:10443 -tls1_3
   ```
5. Ensure the SSL VPN connection is established with TLS 1.3 using the CLI:
```
# diagnose debug application sslvpn -1
# diagnose debug enable
```
The system displays a response like the following:
```
[207:root:1d]SSL established: TLSv1.3 TLS_AES_256_GCM_SHA384
```

Deep inspection (flow-based)

FortiOS supports TLS 1.3 for policies that have the following security profiles applied:

- Web filter profile with flow-based inspection mode enabled.
- Deep inspection SSL/SSH inspection profile.

For example, when a client attempts to access a website that supports TLS 1.3, FortiOS sends the traffic to the IPS engine. The IPS engine then decodes TLS 1.3 and the client is able to access the website.

SMBv2 support

On all FortiGate models, SMBv2 is enabled by default for SSL VPN. Client PCs can access the SMBv2 server using SSL VPN web-only mode.

To configure SMBv2:

1. Set the minimum and maximum SMB versions.

```
config vpn ssl web portal
    edit portal-name
       set smb-min-version smbv2
       set smb-max-version smbv3
    next
end
```

2. Configure SSL VPN and firewall policies as usual.
3. Connect to the SSL VPN web portal and create an SMB bookmark for the SMBv2 server.
4. Click the bookmark to connect to the SMBv2 server.
5. On the FortiGate, use package capture to verify that SMBv2 works:

8	-440785802.3…	172.16.200.10	172.16.200.44	SMB2	252 Negotiate Protocol Request
9	-440785802.3…	172.16.200.44	172.16.200.10	SMB2	338 Negotiate Protocol Response

Configuring OS and host check

Beyond the basics of setting up the SSL VPN, you can configure a number of other options that can help to ensure your internal network is secure and can limit the possibility of attacks and viruses entering the network from an outside source. These include verifying OS and performing host checks on software running on the remote device.

Verifying remote user OS

To verify that remote users are using devices with up-to-date Operating Systems to connect to your network, you can configure a host check for Windows and Mac OS. You can configure an OS host check for specific OS versions, such as Windows 7, 8.1, 10, and 11.

To configure an OS host check for specific OS versions:

1. Go to *VPN > SSL-VPN*.
2. Click *Create New*.
3. Enable *Restrict to Specific OS Versions*.
4. Select an OS version and click *Edit* to change the action.
5. Select the action:
   - *Allow*: The selected OS version is allowed to connect. This is the default action.
   - *Block*: The selected OS version is not allowed to connect.
   - *Check up to date*: Specify a *Tolerance* and *Latest patch level* that is allowed for the selected OS version.
6. Click *OK*.
7. Configure other parameters as needed.
8. Click *OK*.

Host check

Host check verifies whether the client device has AntiVirus, firewall, both, or other custom security software enabled on their Windows device. Admins may also define their own custom host check software, which supports Windows and Mac

OS. See Creating a custom host check list on page 1747.

Host Check is only available for SSL VPN tunnel mode.

To configure host checking:

1. Go to *VPN > SSL-VPN Portal*.
2. Click *Create New*.
3. Enable *Host Check*.
4. Set the *Type*:
   - *Realtime AntiVirus*: Checks that AntiVirus software recognized by Windows Security Center is enabled.
   - *Firewall*: Checks that firewall software recognized by Windows Security Center is enabled.
   - *Enable both*: Checks that both *Realtime AntiVirus* and *Firewall* are enabled.
   - Custom: Not configurable from the GUI. See CLI settings below.
5. Configure other parameters as needed.
6. Click *OK*.

You can configure the full-access portal to perform a custom host check for FortiClient Host Security AV and firewall software.

To configure custom host checking:

```
config vpn ssl web portal
    edit full-access
        set host-check custom
        set host-check-policy FortiClient-AV FortiClient-FW
    next
end
```

Many other security software can also be configured. Use `set host-check-policy ?` to see a list of software.

Replacing the host check error message

You can add your own host security check error message using either the GUI or the CLI. The default message reads:

```
Your PC does not meet the host checking requirements set by the firewall. Please try again
in a few minutes. If the issue persists check that your OS version meets the minimum
requirements, that your antivirus and firewall applications are installed and running
properly, and that you have the correct network interface.
```

To replace the host check error message in the GUI:

1. Go to *System > Replacement Messages*.
2. Select *Extended View* in the upper right corner.

3. Scroll down to *SSL-VPN* and select *Hostcheck Error Message*.
4. Click *Edit*. The *Hostcheck Error Message* pane opens.
5. Edit the text in the right-hand column.
6. Click *Save*.

If you are unhappy with the new message, you can restore the message to its default by selecting *Restore Defaults* instead of *Save*.

MAC address check

Aside from OS and Host check, FortiGate can also perform a MAC address check on the remote host.

To configure a MAC address check on the remote host in the CLI:

```
config vpn ssl web portal
    edit <portal_name>
        set mac-addr-check enable
        config mac-addr-check-rule
            edit <rule_name>
                set mac-addr-list <address> [address]
                set mac-addr-mask <mask between 1-48>
            next
        end
        set set mac-addr-action {allow | deny}
    next
end
```

Creating a custom host check list

You can add your own software requirements to the host check list using the CLI. Host integrity checking is only possible with client computers running Microsoft Windows platforms.

To add software requirements to the host check list:

```
config vpn ssl web host-check-software
    edit <software_name>
        set os-type {windows | macos}
        set type {av | fw}
        set version <version_number>
        set guid <guid_value>
        config check-item-list
            edit <ID>
                set action {require | deny}
                set type {file | registry | process}
                set target <target string>
                set version <version string>
                set md5s <hext string>
            next
        end
```

```
    next
end
```

If known, enter the Globally Unique Identifier (GUID) for the host check application. Windows uses GUIDs to identify applications in the Windows Registry. The GUID can be found in the Windows registry in the `HKEY_CLASSES_ROOT` section.

To obtain the exact versioning, in Windows, right-click on the .EXE file of the application and select *Properties*, then select the *Version* tab.

Example: Tunnel Mode Host Check - Registry Key Check

The following example configuration checks if a required registry key is present on a Windows device.

```
config vpn ssl web host-check-software
    edit <computer_name>
        config check-item-list
            edit 1
                set target "HKEY_LOCAL_
MACHINE\\SYSTEM\\CurrentControlSet\\Control\\ComputerName\\ActiveComputerName:ComputerName=W
INXP32SP3B62"
                set type registry
            next
        end
    next
end
```

Example: Tunnel Mode Host Check - Application Running Check

The following example configuration checks if a required application is installed and/or running:

```
config vpn ssl web host-check-software
    edit "calc"
        config check-item-list
            edit 1
                set target "calc.exe"
                set type process
            next
        end
    next
end
```

Example: Mac OS host check and process check

The `os-type` option is available under `vpn ssl web host-check-software`; if `os-type` is `macos`, then `type`, `version` and `guid` are hidden. Furthermore, `type` in `check-item-list` can only be set to `file` or `process`.

```
config vpn ssl web portal
    edit <portal_name>
        set os-check enable
        config os-check-list macos-bigsur-11
            set action {allow | deny | check-up-to-date}
            set tolerance <value>
            set latest-patch-level <value>
        end
    next
```

```
end
config vpn ssl web host-check-software
    edit <name>
        set os-type macos
        config check-item-list
            edit <name>
                set type process
                set target <target process>
            next
        end
    next
end
```

Example: Configuring Windows OS Check with patch version

The Windows patch check enables you to define the minimum Windows version and patch level allowed when connecting to the SSL VPN portal. When the user attempts to connect to the web portal, FortiOS performs a query on the version of Windows the user has installed. If it does not match the minimum requirement, the connection is denied. The Windows patch check is configured in the CLI.

To specify the acceptable patch level, you set the `latest-patch-level` and the `tolerance`. The lowest acceptable patch level is `latest-patch-level` minus *tolerance*. In this case, *latest-patch-level* is three and *tolerance* is one, so two is the lowest acceptable patch level.

To configure OS check:

```
config vpn ssl web portal
    edit <portal_name>
        set os-check enable
        config os-check-list <windows OS version>
            set action {allow | check-up-to-date | deny}
            set latest-patch-level {disable | 0 - 255}
            set tolerance <tolerance_num>
        end
    next
end
```

Example: Host check for Windows firewall

The Windows built-in firewall does not have a GUID in root\securitycenter or root\securitycenter2, but you can use a registry value to detect the firewall status.

If Windows firewall is on, the following registry value will be set to one:

- **KeyName**: HKEY_LOCAL_ MACHINE\SYSTEM\CurrentControlSet\Services\SharedAccess\Parameters\FirewallPolicy\StandardProfile
- **ValueName**: EnableFirewall

In FortiOS, use the `registry-value-check` feature to define the Windows firewall software.

To define the Windows firewall software:

```
config vpn ssl web host-check-software
    edit "Microsoft-Windows-Firewall"
        set type fw
```

```
        config check-item-list
            edit 1
                set target
"HKLM\\SYSTEM\\CurrentControlSet\\Services\\SharedAccess\\Parameters\\FirewallPolicy\\Standa
rdProfile:EnableFirewall==1"
                set type registry
            next
            edit 2
                set target
"HKLM\\SYSTEM\\CurrentControlSet\\Services\\SharedAccess\\Parameters\\FirewallPolicy\\Public
Profile:EnableFirewall==1"
                set type registry
            next
            edit 3
                set target
"HKLM\\SYSTEM\\CurrentControlSet\\Services\\SharedAccess\\Parameters\\FirewallPolicy\\Domain
Profile:EnableFirewall==1"
                set type registry
            next
        end
    next
end
config vpn ssl web portal
    edit <portal_name>
        set host-check custom
        set host-check-policy Microsoft-Windows-Firewall
    next
end
```

Troubleshooting

To troubleshoot OS and host check, enable the following real-time debugs from the CLI:

```
# diagnose debug app sslvpn -1
```

```
# diagnose debug enable
```

From the remote client, connect to SSL VPN. Look for debug output similar to the following:

```
[263:root:3cca1]host check result:4 0100,10.0.19042,74:78:27:4d:81:93|84:1b:77:3a:95:84
```

To interpret the above output:

Field	Description
`host check result: 4`	This is the hex number of portal's host check value: • `0`: None • `1`: Check antivirus • `2`: Check firewall • `3`: Check antivirus and firewall • `4`: Custom check
`0100`	The 4 bytes shows the result of host check checking in the FortiGate Settings. Position counts from left to right, zero to three:

Field	Description
	• Position zero means result of third party firewall. • Position one means result of third party antivirus. • Position two means result of FortiClient firewall. • Position three means result of FortiClient antivirus. `0` means not in use. `1` means in use.
`10.0.19042`	This is the OS version.
`74:78:27:4d:81:93|84:1b:77:3a:95:84`	The MAC address of the client machine's network interface, that is used for the mac address check. Multiple MAC address are separately by '`|`'.

FortiGate as SSL VPN Client

The FortiGate can be configured as an SSL VPN client, using an *SSL-VPN Tunnel* interface type. When an SSL VPN client connection is established, the client dynamically adds a route to the subnets that are returned by the SSL VPN server. Policies can be defined to allow users that are behind the client to be tunneled through SSL VPN to destinations on the SSL VPN server.

FortiOS can be configured as an SSL VPN server that allows IP-level connectivity in tunnel mode, and can act as an SSL VPN client that uses the protocol used by the FortiOS SSL VPN server. This allows hub-and-spoke topologies to be configured with FortiGates as both the SSL VPN hub and spokes.

For an IP-level VPN between a device and a VPN server, this can be useful to avoid issues caused by intermediate devices, such as:

- ESP packets being blocked.
- UDP ports 500 or 4500 being blocked.
- Fragments being dropped, causing IKE negotiation that uses large certificates to fail if the peer does not support IKE fragmentation.

If the client specified destination is *all*, a default route is effectively dynamically created on the SSL VPN client, and the new default route is added to the existing default route in the form of ECMP. Some examples how to configure routing are:

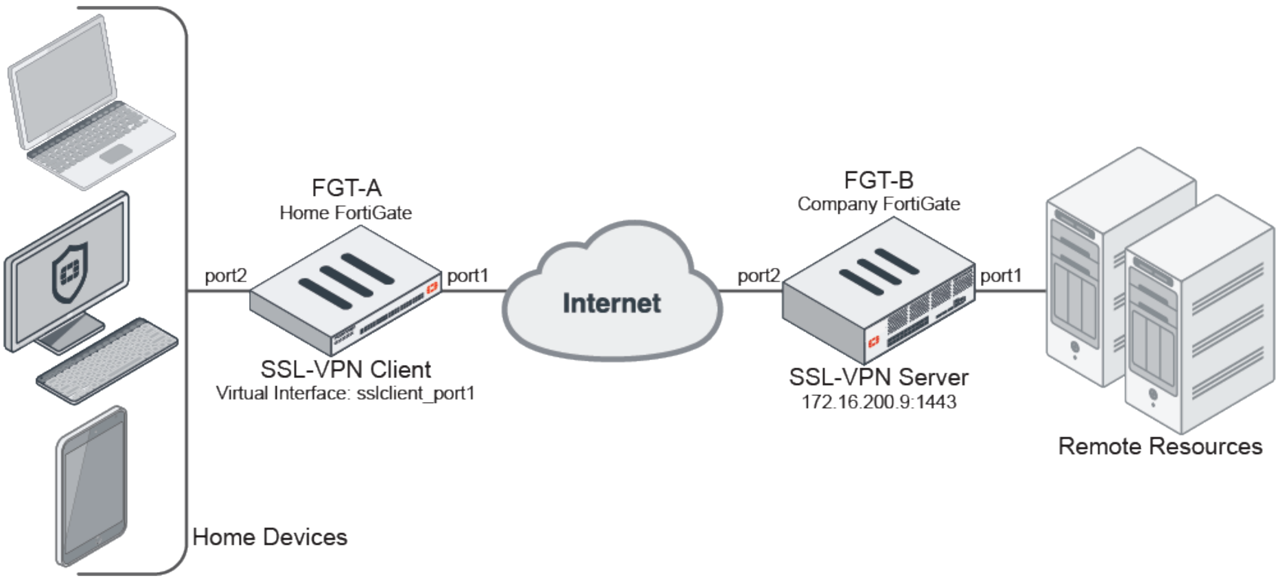
- To make all traffic default to the SSL VPN server and still have a route to the server's listening interface, on the SSL VPN client set a lower distance for the default route that is learned from the server.
- To include both default routes in the routing table, with the route learned from the SSL VPN server taking priority, on the SSL VPN client set a lower distance for the route learned from the server. If the distance is already zero, then increase the priority on the default route.
- To avoid a default being learned on the SSL VPN client, on the SSL VPN server define a specific destination.

Example

In this example, the home FortiGate (FGT-A) is configured as an SSL VPN client, and the company FortiGate (FGT-B) is configured as an SSL VPN server. After FGT-A connects to FGT-B, the devices that are connected to FGT-A can access the resources behind FGT-B.

The SSL VPN server has a custom server certificate defined, and the SSL VPN client user uses PSK and a PKI client certificate to authenticate. The FortiGates must have the proper CA certificate installed to verify the certificate chain to the root CA that signed the certificate.

Split tunneling is used so that only the destination addresses defined in the server's firewall policies are routed to the server, and all other traffic is connected directly to the internet.



Configure the SSL VPN server

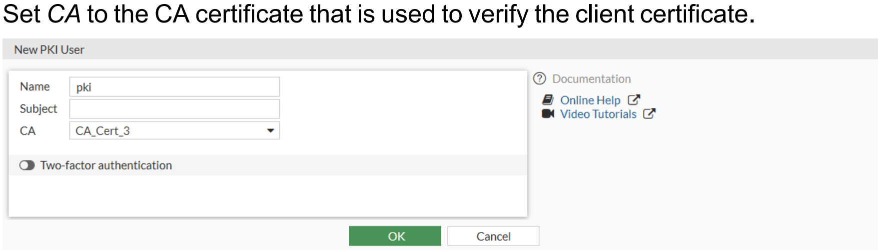
To create a local user in the GUI:

1. Go to *User & Authentication > User Definition* and click *Create New*.
2. Use the wizard to create a local user named *client2*.

To create a PKI user in the GUI:

The PKI menu is only available in the GUI after a PKI user has been created using the CLI, and a CN can only be configured in the CLI.

1. Go to *User & Authentication > PKI* and click *Create New*.
2. Set the *Name* to *pki*.
3. Set *CA* to the CA certificate that is used to verify the client certificate.



4. Click *OK*.

5. In the CLI, specify the CN that must be matched. If no CN is specified, then any certificate that is signed by the CA will be valid and matched.

```
config user peer
    edit "pki"
        set cn "*.fos.automation.com"
    next
end
```

To create an SSL VPN portal in the GUI:

1. Go to *VPN > SSL-VPN Portals* and click *Create New*.
2. Set the *Name* to *testportal2*.
3. Set *Enable Split Tunneling* to *Enabled Based on Policy Destination*.
4. Set *Source IP Pools* to *SSLVPN_TUNNEL_ADDR1*.
5. Click *OK*.

To configure SSL VPN settings in the GUI:

1. Go to *VPN > SSL-VPN Settings* and enable *Enable SSL-VPN*.
2. Set *Listen on Interface(s)* to *port2*.
3. Set *Listen on Port* to *1443*.
4. Set *Server Certificate* to *fgt_gui_automation*.
5. In the *Authentication/Portal Mapping* table click *Create New*:
    a. Set *Users/Groups* to *client2*.
    b. Set *Portal* to *testportal2*.
    c. Click *OK*.
6. Click *OK*.
7. In the CLI, enable SSL VPN client certificate restrictive and set the user peer to *pki*:

```
config vpn ssl settings
    config authentication-rule
        edit 1
            set client-cert enable
            set user-peer "pki"
        next
    end
end
```

To create a firewall address in the GUI:

1. Go to *Policy & Objects > Addresses* and click *Create New > Address*.
2. Set the *Name* to *bing.com*.
3. Set *Type* to *FQDN*.
4. Set *FQDN* to *www.bing.com*.
5. Click *OK*.

To create a firewall policy in the GUI:

1. Go to *Policy & Objects* > *Firewall Policy* and click *Create New*.
2. Configure the policy:

Name	*sslvpn2*
Incoming Interface	*SSL-VPN tunnel interface (ssl.root)*
Outgoing Interface	*port1*
Source	*Address*: *all* *User*: *client2*
Destination	*bing.com*: This FQDN resolves to 13.107.21.200 and 204.79.197.200. Traffic to these addresses is directed to the SSL VPN, while other traffic is routed to the remote devices' default adapters or interfaces. *mantis*
Schedule	*always*
Service	*ALL*
Action	*Accept*

3. Click *OK*.

To configure the SSL VPN server (FGT-B) in the CLI:

1. Create a local user:

```
config user local
    edit "client2"
        set passwd **********
    next
end
```

2. Create a PKI user:

```
config user peer
    edit "pki"
        set ca "CA_Cert_3"
        set cn "*.fos.automation.com"
    next
end
```

3. Create a new SSL VPN portal:

```
config vpn ssl web portal
    edit "testportal2"
        set tunnel-mode enable
        set ipv6-tunnel-mode enable
        set ip-pools "SSLVPN_TUNNEL_ADDR1"
        set split-tunneling enable
        set ipv6-pools "SSLVPN_TUNNEL_IPv6_ADDR1"
        set ipv6-split-tunneling enable
        ....
```

```
    next
end
```

4. Configure SSL VPN settings, including the authentication rule for user mapping:

```
config vpn ssl settings
    set ssl-min-proto-ver tls1-1
    set servercert "fgt_gui_automation"
    set auth-timeout 0
    set login-attempt-limit 10
    set login-timeout 180
    set tunnel-ip-pools "SSLVPN_TUNNEL_ADDR1"
    set tunnel-ipv6-pools "SSLVPN_TUNNEL_IPv6_ADDR1"
    set dns-suffix "sslvpn.com"
    set port 1443
    set source-interface "port2"
    set source-address "all"
    set source-address6 "all"
    set default-portal "testportal1"
    config authentication-rule
        edit 1
            set users "client2"
            set portal "testportal2"
            set client-cert enable
            set user-peer "pki"
        next
    end
end
```

5. Create a firewall address and policy. The destination addresses used in the policy are routed to the SSL VPN server.

```
config firewall address
    edit "bing.com"
        set type fqdn
        set fqdn "www.bing.com"
    next
end
```

```
config firewall policy
    edit 2
        set name "sslvpn2"
        set srcintf "ssl.root"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "mantis" "bing.com"
        set action accept
        set schedule "always"
        set service "ALL"
        set nat enable
        set users "client2"
    next
end
```

Configure the SSL VPN client

To create a PKI user in the GUI:

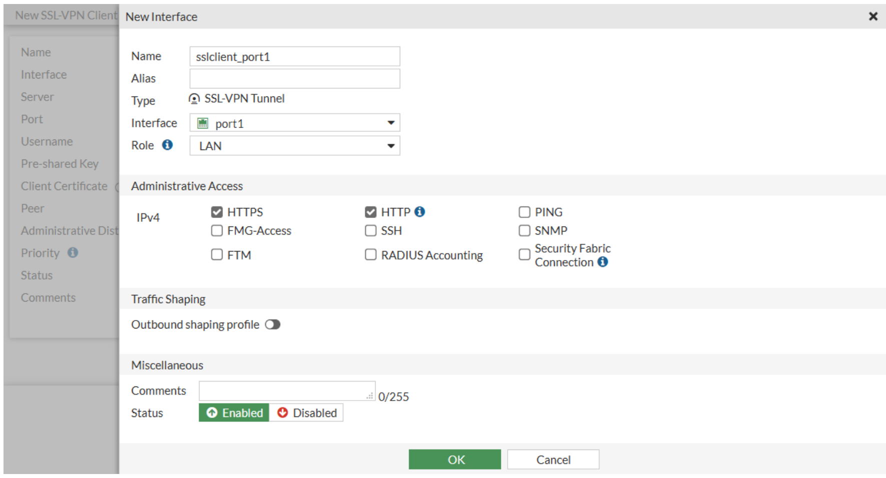
The PKI menu is only available in the GUI after a PKI user has been created using the CLI, and a CN can only be configured in the CLI.

1. Go to *User & Authentication > PKI* and click *Create New*.
2. Set the *Name* to *fgt_gui_automation*.
3. Set *CA* to the CA certificate. The CA certificate allows the FortiGate to complete the certificate chain and verify the server 's certificate, and is assumed to already be installed on the FortiGate.
4. Click *OK*.
5. In the CLI, specify the CN of the certificate on the SSL VPN server:

```
config user peer
    edit "fgt_gui_automation"
        set cn "*.fos.automation.com"
    next
end
```

To create an SSL VPN client and virtual interface in the GUI:

1. Go to *VPN > SSL-VPN Clients* and click *Create New*.
2. Expand the *Interface* drop down and click *Create* to create a new virtual interface:
   a. Set the *Name* to *sslclient_port1*.
   b. Set *Interface* to *port1*.
   c. Under *Administrative Access*, select *HTTPS* and *PING*.



   d. Click *OK*.

3. Configure the SSL VPN client:

Name	*sslclientTo9*
Interface	*sslclient_port1*
Server	*172.16.200.9*
Port	*1443*
Username	*client2*
Pre-shared Key	**********
Client Certificate	*fgtb_gui_automation* This is the local certificate that is used to identify this client, and is assumed to already be installed on the FortiGate. The SSL VPN server requires it for authentication.
Peer	*fgt_gui_automation*
Administrative Distance	Configure as needed.
Priority	Configure as needed.
Status	Enabled

4. Click *OK*.

To create a firewall policy in the GUI:

1. Go to *Policy & Objects* > *Firewall Policy* and click *Create New*.
2. Configure the policy:

Name	*policy_to_sslvpn_tunnel*
Incoming Interface	*port2*
Outgoing Interface	*sslclient_port1*
Source	*all*
Destination	*all*
Schedule	*always*
Service	*ALL*
Action	*Accept*

3. Click *OK*.

To configure the SSL VPN client (FGT-A) in the CLI:

1. Create the PKI user. Use the CA that signed the certificate *fgt_gui_automation*, and the CN of that certificate on the SSL VPN server.

```
config user peer
    edit "fgt_gui_automation"
```

```
            set ca "GUI_CA"
            set cn "*.fos.automation.com"
        next
    end
```

2. Create the SSL interface that is used for the SSL VPN connection:

```
config system interface
    edit "sslclient_port1"
        set vdom "vdom1"
        set allowaccess ping https
        set type ssl
        set role lan
        set snmp-index 46
        set interface "port1"
    next
end
```

3. Create the SSL VPN client to use the PKI user and the client certificate *fgtb_gui_automation*:

```
config vpn ssl client
    edit "sslclientTo9"
        set interface "sslclient_port1"
        set user "client2"
        set psk 123456
        set peer "fgt_gui_automation"
        set server "172.16.200.9"
        set port 1443
        set certificate "fgtb_gui_automation"
    next
end
```

4. Create a firewall policy:

```
config firewall policy
    edit 1
        set name "policy_to_sslvpn_tunnel"
        set srcintf "port2"
        set dstintf "sslclient_port1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set nat enable
    next
end
```

Verification

After the tunnel is established, the route to 13.107.21.200 and 204.79.197.200 on FGT-A connects through the SSL VPN virtual interface *sslclient_port1*.

To check the routing table details:

```
(vdom1) # get router info routing-table details
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
```

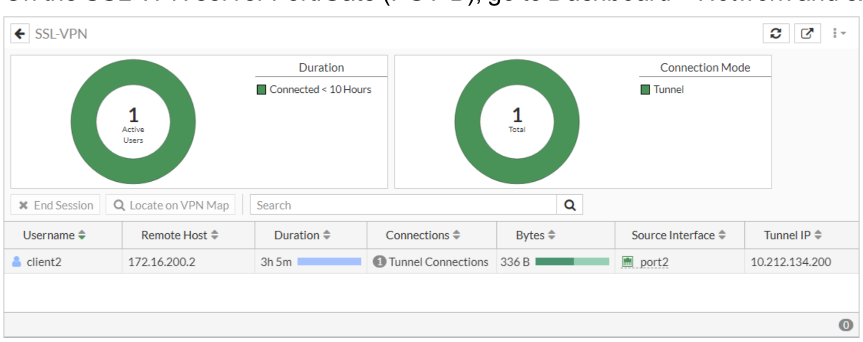
```
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default

Routing table for VRF=0
S*      0.0.0.0/0 [10/0] via 172.16.200.254, port1
C       10.0.1.0/24 is directly connected, link_11
C       10.1.100.0/24 is directly connected, port2
                      is directly connected, port2
C       10.212.134.200/32 is directly connected, sslclient_port1
S       13.107.21.200/32 [10/0] is directly connected, sslclient_port1
C       172.16.200.0/24 is directly connected, port1
S       192.168.100.126/32 [10/0] is directly connected, sslclient_port1
S       204.79.197.200/32 [10/0] is directly connected, sslclient_port1
```

To check the added routing for an IPv6 tunnel:

```
(vdom1) # get router info6 routing-table database
IPv6 Routing Table
Codes: K - kernel route, C - connected, S - static, R - RIP, O - OSPF,
       IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, B - BGP
       > - selected route, * - FIB route, p - stale info
Timers: Uptime

S     *> ::/0 [10/0] via 2000:172:16:200::254, port1, 00:00:01, [1024/0]
      *>      [10/0] via ::, sslclient_port1, 00:00:01, [1024/0]
C     *> ::1/128 via ::, vdom1, 03:26:35
C     *> 2000:10:0:1::/64 via ::, link_11, 03:26:35
C     *> 2000:10:1:100::/64 via ::, port2, 03:26:35
C     *> 2000:172:16:200::/64 via ::, port1, 03:26:35
C     *> 2001:1::1:100/128 via ::, sslclient_port1, 00:00:01
C     *> fe80::/64 via ::, port2, 03:26:35
```

To check the connection in the GUI:

1. On the SSL VPN server FortiGate (FGT-B), go to *Dashboard* > *Network* and expand the *SSL-VPN* widget.



2. On the SSL VPN client FortiGate (FGT-A), go to *VPN* > *SSL-VPN Clients* to see the tunnel list.

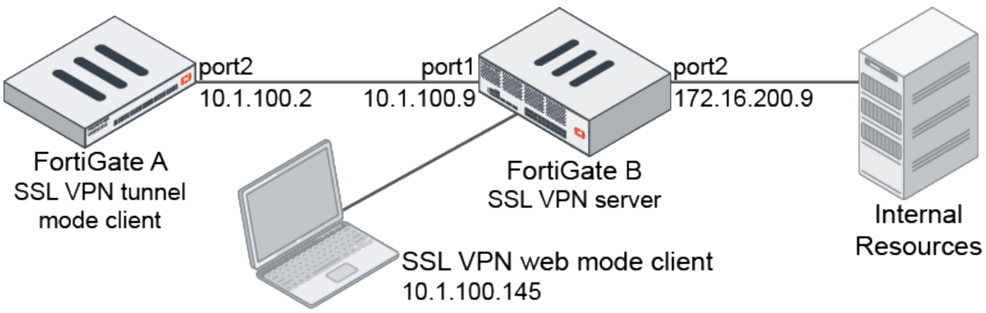
Dual stack IPv4 and IPv6 support for SSL VPN

Dual stack IPv4 and IPv6 support for SSL VPN servers and clients enables a client to establish a dual stack tunnel to allow both IPv4 and IPv6 traffic to pass through. FortiGate SSL VPN clients also support dual stack, which allows it to establish dual stack tunnels with other FortiGates.

Users connecting in web mode can connect to the web portal over IPv4 or IPv6. They can access bookmarks in either IPv4 or IPv6, depending on the preferred DNS setting of the web portal.

Example

In this example, FortiGate B works as an SSL VPN server with dual stack enabled. A test portal is configured to support tunnel mode and web mode SSL VPN.



FortiGate A is an SSL VPN client that connects to FortiGate B to establish an SSL VPN tunnel connection. It attempts to access www.bing.com and www.apple.com via separate IPv4 and IPv6 connections. Two addresses are configured on FortiGate B:

- *bing.com* uses IPv4 FQDN and resolves to 13.107.21.200 and 204.79.197.200.
- *apple_v6* uses IPv6 FQDN and resolves to 2600:140a:c000:385::1aca and 2600:140a:c000:398::1aca.

The server certificate used is fgt_gui_automation, and the CN is *.fos.automation.com.

A PC serves as a client to connect to FortiGate B in SSL VPN web mode. The PC can connect to the SSL VPN server over IPv4 or IPv6. Based on the preferred DNS setting, it will access the destination website over IPv4 or IPv6.

Dual stack tunnel mode support requires a supported client. In 7.0.0, a FortiGate in SSL VPN client mode can support dual stack tunnels. FortiClient 7.0.1 and later releases support dual stack.

To configure an SSL VPN server in tunnel and web mode with dual stack support in the GUI:

1. Create a local user:
   a. Go to *User & Authentication > User Definition* and click *Create New*. The *Users/Groups Creation Wizard* opens.
   b. Set the *User Type* to *Local User* and click *Next*.
   c. Enter the *Username* (*client2*) and password, then click *Next*.
   d. Optionally, configure the contact information and click *Next*.
   e. Click *Submit*.
2. Configure the addresses:

a. Go to *Policy & Objects* > *Addresses* and click *Create New* > *Address*.
b. Enter the following for the IPv4 address:

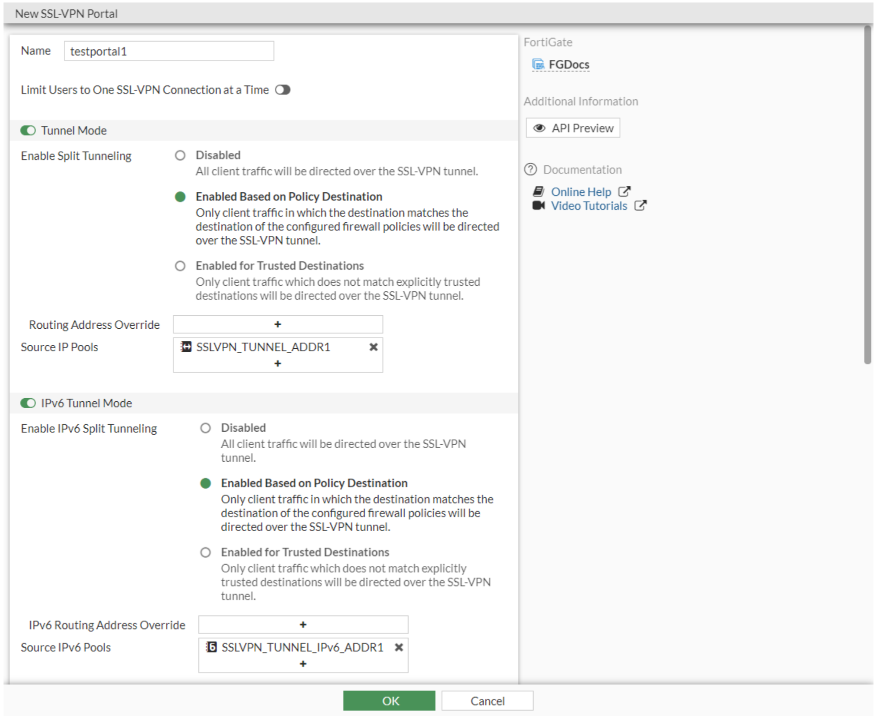
Category	Address
Name	bing.com
Type	FQDN
FQDN	www.bing.com

c. Click *OK*.
d. Click *Create New* > *Address* and enter the following for the IPv6 address:

Category	IPv6 Address
Name	apple_v6
Type	FQDN
FQDN	www.apple.com

e. Click *OK*.

3. Configure the SSL VPN portal:
   a. Go to *VPN* > *SSL-VPN Portals* and click *Create New*.
   b. Enter a name (*testportal1*).
   c. Enable *Tunnel Mode* and for *Enable Split Tunneling*, select *Enable Based on Policy Destination*.
   d. For *Source IP Pools*, add *SSLVPN_TUNNEL_ADDR1*.
   e. Enable *IPv6 Tunnel Mode* and for *Enable Split Tunneling*, select *Enable Based on Policy Destination*.
   f. For *Source IP Pools*, add *SSLVPN_TUNNEL_IPv6_ADDR1*.
   g. Enable *Enable Web Mode*.

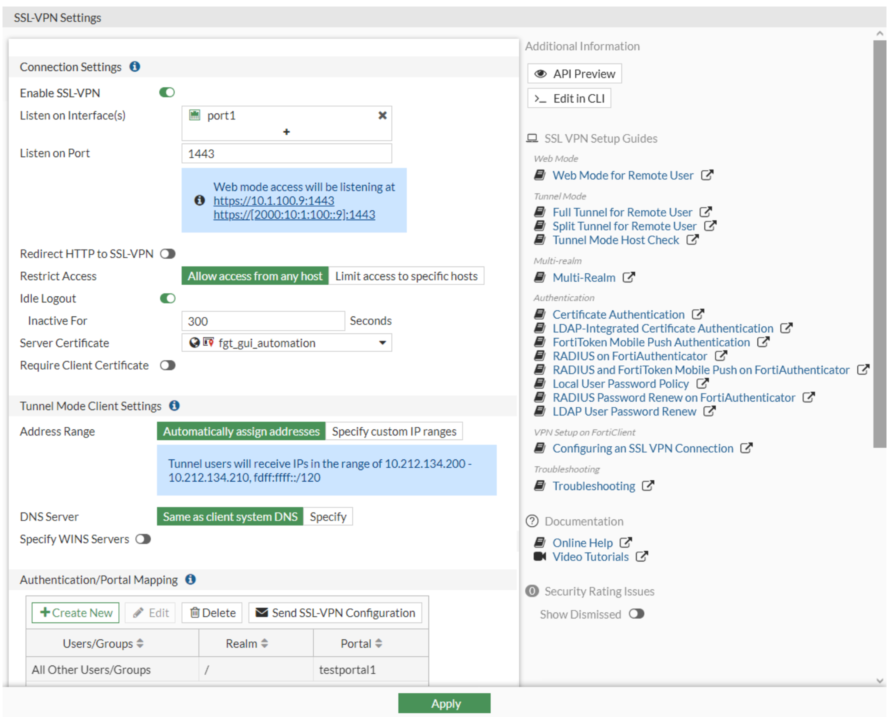


h. Click *OK*.

4. Configure the SSL VPN settings:

a. Go to *VPN > SSL-VPN Settings* and configure the following:

Listen on Interface(s)	port1
Listen on Port	1443
Restrict Access	Allow access from any host
Server Certificate	fgt_gui_automation
Address Range	Automatically assign addresses
DNS Server	Same as client system DNS
Authentication/Portal Mapping	Edit the *All Other Users/Groups* entry to use *testportal1*.



b. Click *Apply*.

c. Enable dual stack in the CLI:

```
config vpn ssl settings
    set dual-stack-mode enable
end
```

5. Configure the firewall policy:

a. Go to *Policy & Objects > Firewall Policy* and click *Create New*.

b. Enter the following:

Name	sslvpn
Incoming Interface	ssl.root

Outgoing Interface	port2
Source	all (IPv4), all (IPv6), client2
Destination	bing.com, apple_v6
Schedule	Always
Service	All
NAT	Enabled

c. Click *OK*.

To configure FortiGate A as an SSL VPN client in the GUI:

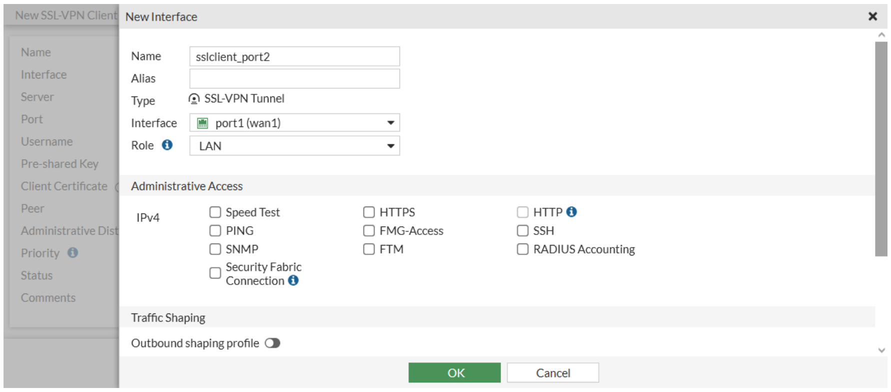
1. Create a peer to verify the server certificate:

> The PKI menu is only available in the GUI (*User & Authentication > PKI*) after a PKI user has been created using the CLI, and a CN can only be configured in the CLI.
> If the CA is not known or is public, import the CA that signed the server certificate.

   a. Go to *User & Authentication > PKI* and click *Create New*.
   b. Set the *Name* to *fgt_gui_automation*.
   c. Set *CA* to the CA certificate that is used to verify the server certificate.
   d. Click *OK*.
   e. In the CLI, specify the CN that must be matched:

```
config user peer
    edit "fgt_gui_automation"
        set ca "GUI_CA"
        set cn "*.fos.automation.com"
    next
end
```

2. Configure the SSL VPN client:
   a. Go to *VPN > SSL-VPN Clients* and click *Create New*.
   b. In the *Interface* dropdown, click *Create*.
      i. Enter a Name (*sslclient_port2*).
      ii. Set *Interface* to *port2*.
      iii. Set *Role* to *LAN*.



iv. Click *OK*.

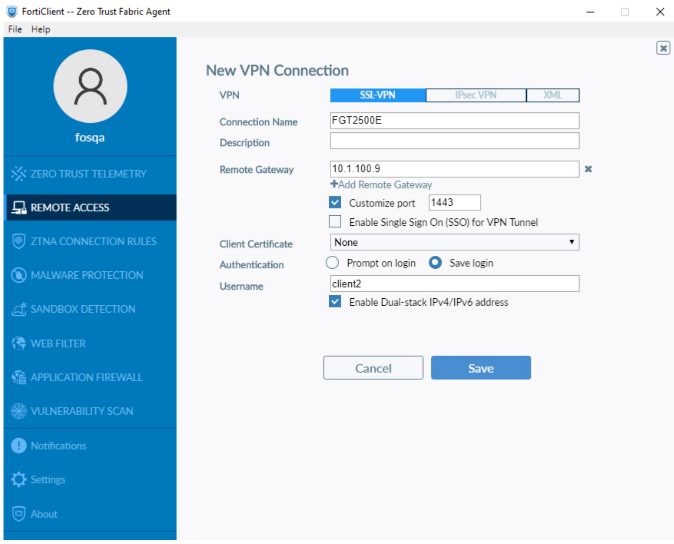
c. Configure the SSL VPN client:

Name	sslclientTo9
Interface	sslclient_port2
Server	Either IPv4 address *10.1.100.9* or IPv6 address *2000:10:1:100::9* can be used and will have the same results.
Port	1443
Username	client2
Pre-shared Key	******
Peer	fgt_gui_automation
Status	Enabled

d. Click *OK*.

To configure FortiClient and connect to the VPN:

1. On the *Remote Access* tab and click *Configure VPN*, or if other connections have already been configured, click the sandwich icon and select *Add a new connection*.
2. Set *Connection Name* to *FGT2500E*, and *Remote Gateway* to *10.1.100.9*.
3. Enable *Customize port* and enter the port number *1443*.
4. Set *Username* to *client2*.
5. Enable *Enable Dual-stack IPv6/IPv6 address*.



6. Click *Save*.
7. Enter the password, then click *Connect*.

To configure an SSL VPN server in tunnel and web mode with dual stack support in the CLI:

1. Create a local user:

```
config user local
    edit "client1"
        set type password
        set passwd ******
    next
end
```

2. Configure the addresses:

```
config firewall address
    edit "bing.com"
        set type fqdn
        set fqdn "www.bing.com"
    next
end
```

```
config firewall address6
    edit "apple_v6"
        set type fqdn
        set fqdn "www.apple.com"
    next
end
```

3. Configure the SSL VPN portal:

```
config vpn ssl web portal
    edit "testportal1"
        set tunnel-mode enable
        set ipv6-tunnel-mode enable
        set web-mode enable
        set ip-pools "SSLVPN_TUNNEL_ADDR1"
        set ipv6-pools "SSLVPN_TUNNEL_IPv6_ADDR1"
```

```
        set split-tunneling enable
        set ipv6-split-tunneling enable
    next
end
```

4. Configure the SSL VPN settings:

```
config vpn ssl settings
    set servercert "fgt_gui_automation"
    set tunnel-ip-pools "SSLVPN_TUNNEL_ADDR1"
    set tunnel-ipv6-pools "SSLVPN_TUNNEL_IPv6_ADDR1"
    set port 1443
    set source-interface "port1"
    set source-address "all"
    set source-address6 "all"
    set default-portal "testportal1"
    set dual-stack-mode enable
end
```

5. Configure the firewall policy:

```
config firewall policy
    edit 1
        set name "sslvpn"
        set srcintf "ssl.root"
        set dstintf "port2"
        set srcaddr "all"
        set dstaddr "bing.com"
        set srcaddr6 "all"
        set dstaddr6 "apple_v6"
        set action accept
        set schedule "always"
        set service "ALL"
        set nat enable
        set users "client2"
    next
end
```

To configure FortiGate A as an SSL VPN client in the CLI:

1. Create a peer to verify the server certificate:

```
config user peer
    edit "fgt_gui_automation"
        set ca "GUI_CA"
        set cn "*.fos.automation.com"
    next
end
```

2. Configure the interface:

```
config system interface
    edit "sslclient_port2"
        set vdom "vdom1"
        set type ssl
        set role lan
        set snmp-index 46
        set interface "port2"
```

```
    next
end
```

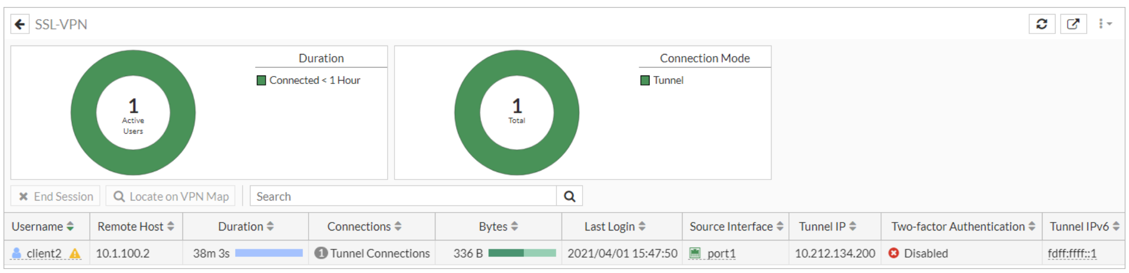
3. Configure the SSL VPN client. Either IPv4 address 10.1.100.9 or IPv6 address 2000:10:1:100::9 can be used and will have the same results:

```
config vpn ssl client
    edit "sslclientTo9"
        set interface "sslclient_port2"
        set user "client2"
        set psk ******
        set peer "fgt_gui_automation"
        set server {10.1.100.9 | 2000:10:1:100::9}
        set port 1443
    next
end
```

Testing dual stack with tunnel mode

To verify the SSL VPN tunnel connection in the GUI:

1. On FortiGate B, go to *Dashboard > Network*.
2. Expand the *SSL-VPN* widget.



To verify the SSL VPN tunnel connection in the CLI:

1. On FortiGate B, verify that the client is assigned with both IPv4 and IPv6 addresses:

```
(root) # get vpn ssl monitor
SSL VPN Login Users:
 Index    User     Group    Auth Type       Timeout          Auth-Timeout     From      HTTP
in/out     HTTPS in/out     Two-factor Auth
 0        client2                1(1)              292      2147483647        10.1.100.2
0/0      0/0      0

SSL VPN sessions:
 Index    User     Group    Source IP       Duration         I/O Bytes        Tunnel/Dest IP
 0        client2                10.1.100.2        5427     1756/1772
10.212.134.200,fdff:ffff::1
```

2. On FortiGate A, verify the routing tables.
   a. IPv4 with resolved addresses for www.bing.com:

```
(vdom1) # get router info routing-table database
...
Routing table for VRF=0
```

```
S    *> 0.0.0.0/0 [10/0] via 172.16.200.254, port1
C    *> 10.0.1.0/24 is directly connected, link_11
C    *> 10.1.100.0/24 is directly connected, port2
C    *> 10.212.134.200/32 is directly connected, sslclient_port2
S    *> 13.107.21.200/32 [10/0] is directly connected, sslclient_port2
C    *> 172.16.200.0/24 is directly connected, port1
S    *> 204.79.197.200/32 [10/0] is directly connected, sslclient_port2
```

b. IPv6 with resolved addresses for www.apple.com:

```
(vdom1) # get router info6 routing-table database
...
S    *> ::/0 [10/0] via 2000:172:16:200::254, port1, 01:57:23, [1024/0]
C    *> ::1/128 via ::, vdom1, 06:12:54
C    *> 2000:10:0:1::/64 via ::, link_11, 06:12:54
C    *> 2000:10:1:100::/64 via ::, port2, 06:12:54
C    *> 2000:172:16:200::/64 via ::, port1, 06:12:54
S    *> 2600:140a:c000:385::1aca/128 [10/0] via ::, sslclient_port2, 01:33:08,
[1024/0]
S    *> 2600:140a:c000:398::1aca/128 [10/0] via ::, sslclient_port2, 01:33:08,
[1024/0]
C    *> fdff:ffff::/120 via ::, sslclient_port2, 01:33:08
C    *> fe80::/64 via ::, port2, 06:12:54
```

To test the address connections using ping:

1. On FortiGate A, ping www.bing.com using IPv4 ping:

```
# execute ping www.bing.com
PING www-bing-com.dual-a-0001.a-msedge.net (13.107.21.200): 56 data bytes
64 bytes from 13.107.21.200: icmp_seq=0 ttl=117 time=1.8 ms
...
```

2. On FortiGate B, sniff for IPv4 ICMP packets and observe the results:

```
# diagnose sniffer packet any icmp 4
interfaces=[any]
filters=[icmp]
9.675101 ssl.root in 10.212.134.200 -> 13.107.21.200: icmp: echo request
9.675219 port2 out 172.16.200.9 -> 13.107.21.200: icmp: echo request
9.676698 port2 in 13.107.21.200 -> 172.16.200.9: icmp: echo reply
9.676708 ssl.root out 13.107.21.200 -> 10.212.134.200: icmp: echo reply
...
```

3. On FortiGate A, ping www.apple.com using IPv6 ping:

```
# execute ping6 www.apple.com
PING www.apple.com (2600:140a:c000:385::1aca): 56 data bytes
64 bytes from 2600:140a:c000:385::1aca: icmp_seq=1 ttl=52 time=1.88 ms
...
```

4. On FortiGate B, sniff for IPv6 ICMP packets and observe the results:

```
# diagnose sniffer packet any icmp6 4
interfaces=[any]
filters=[icmp6]
3.564296 ssl.root in fdff:fff::1 -> 2600:140a:c000:385::1aca: icmp6: echo request seq 1
3.564435 port2 out 2000:172:16:200::9 -> 2600:140a:c000:385::1aca: icmp6: echo request
seq 1
```

```
3.565929 port2 in 2600:140a:c000:385::1aca -> 2000:172:16:200::9: icmp6: echo reply seq
1 [flowlabel 0x1fdff]
3.565953 ssl.root out 2600:140a:c000:385::1aca -> fdff:fff::1: icmp6: echo reply seq 1
[flowlabel 0x1fdff]
...
```

Testing dual stack with web mode

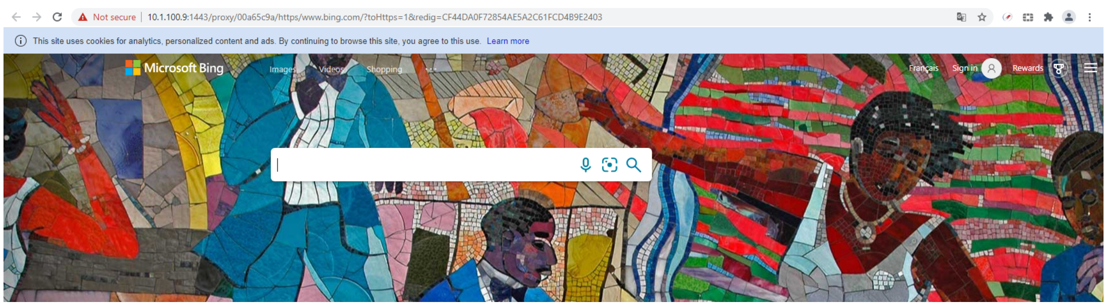
In SSL VPN web mode, users can access both IPv4 and IPv6 bookmarks in the portal. The attribute, `prefer-ipv6-dns` can be enabled to prefer querying IPv6 DNS first, or disabled to prefer querying IPv4.

To test an IPv4 connection to the web portal and access www.bing.com over IPv6:

1. On FortiGate B, prioritize resolving IPv6 addresses:

```
config vpn ssl web portal
    edit "testportal1"
        set prefer-ipv6-dns enable
    next
end
```

2. Log in to the web portal in the browser over the IPv4 address 10.1.100.9.
3. Create a new HTTP/HTTPS bookmark named *bing* for the URL www.bing.com.
4. Click the *bing* bookmark. The bing page will open over IPv6.



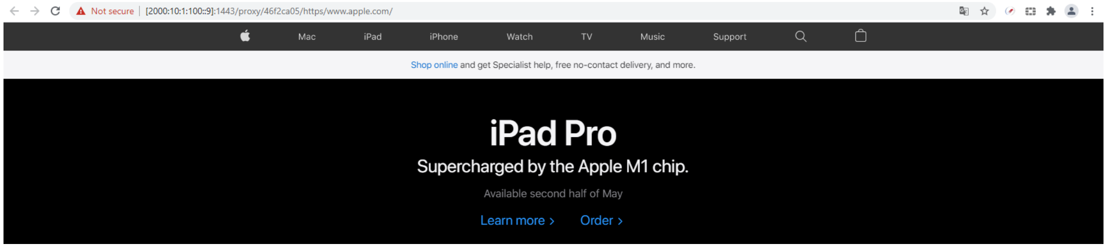
To test an IPv6 connection to the web portal and access www.apple.com over IPv4:

1. On FortiGate B, prioritize resolving IPv4 addresses:

```
config vpn ssl web portal
    edit "testportal1"
        set prefer-ipv6-dns disable
    next
end
```

2. Log in to the web portal in the browser over the IPv6 address [2000:10:1:100::9].
3. Create a new HTTP/HTTPS bookmark named *apple* for the URL www.apple.com.

4. Click the *apple* bookmark. The apple page will open over IPv4.

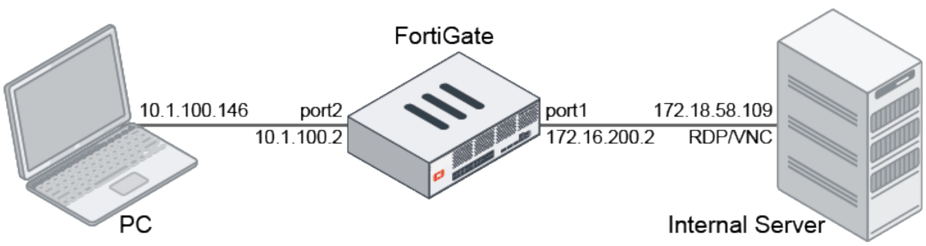


Disable the clipboard in SSL VPN web mode RDP connections

In web portal profiles, the clipboard can be disabled for SSL VPN web mode RDP/VNC connections. User will not be able to copy and paste content to or from the internal server.

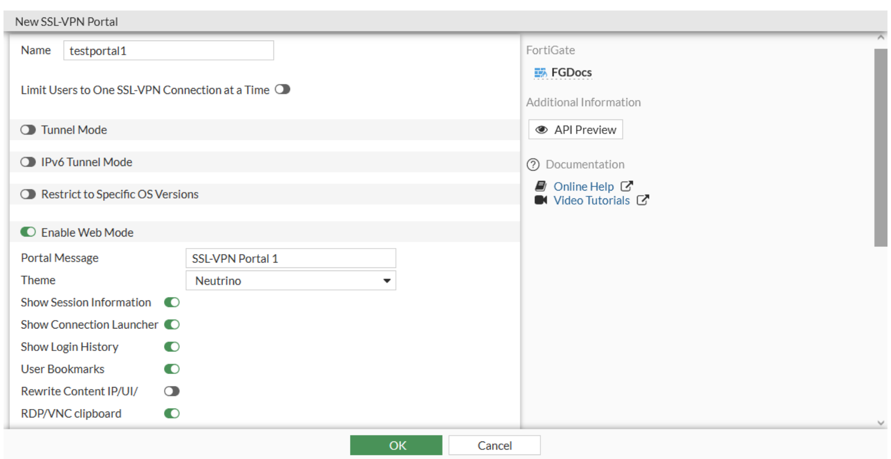
Example

In this example, two groups of users are using SSL VPN web mode to access internal servers with RDP/VNC. One group is allowed to copy and paste content to and from the internal server using the clipboard, while the other is not.



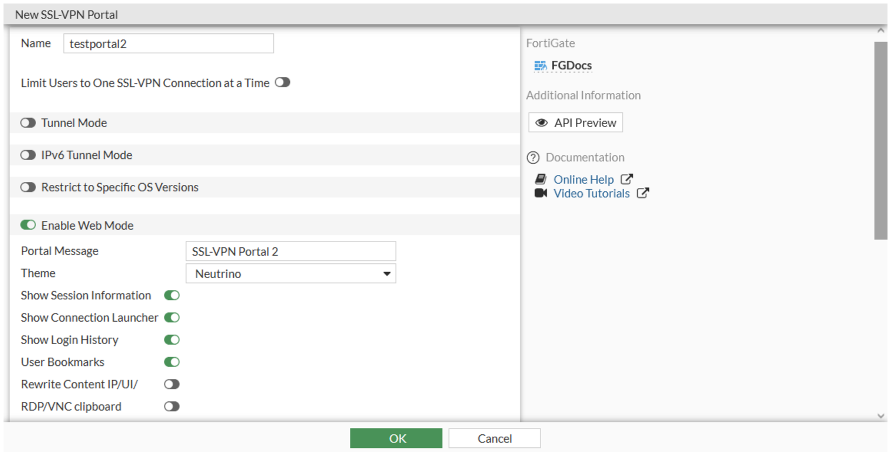
To configure the SSL VPN portals in the GUI:

1. Go to *VPN > SSL-VPN Portals* and click *Create New*.
2. Enter a name for the portal, such as *testportal1*.
3. Enable *Enable Web Mode* and enable *RDP/VNC clipboard* to allow copying and pasting.
4. Configure the remaining settings as needed.



5. Click *OK*.

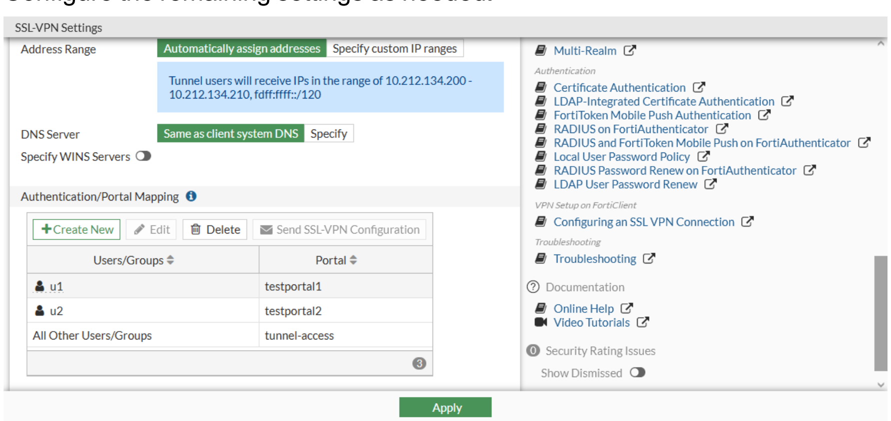
6. Click *Create New* again.
7. Enter a name for the portal, such as *testportal2*.
8. Enable *Enable Web Mode* and disable *RDP/VNC clipboard* to prevent copying and pasting.
9. Configure the remaining settings as needed.



10. Click *OK*.

To configure the SSL VPN settings in the GUI:

1. Go to *VPN > SSL-VPN Settings*.
2. Set *Listen on Interface* to port2.
3. In the *Authentication/Portal Mapping* table, add the users to each of the portals:
   a. Click *Create New*.
   b. Set *Users/Groups* to *u1* and *Portal* to *testportal1*.
   c. Click *OK*, then click *Create New* again.
   d. Set *Users/Groups* to *u2* and *Portal* to *testportal2*.
   e. Click *OK*.
4. Configure the remaining settings as needed.



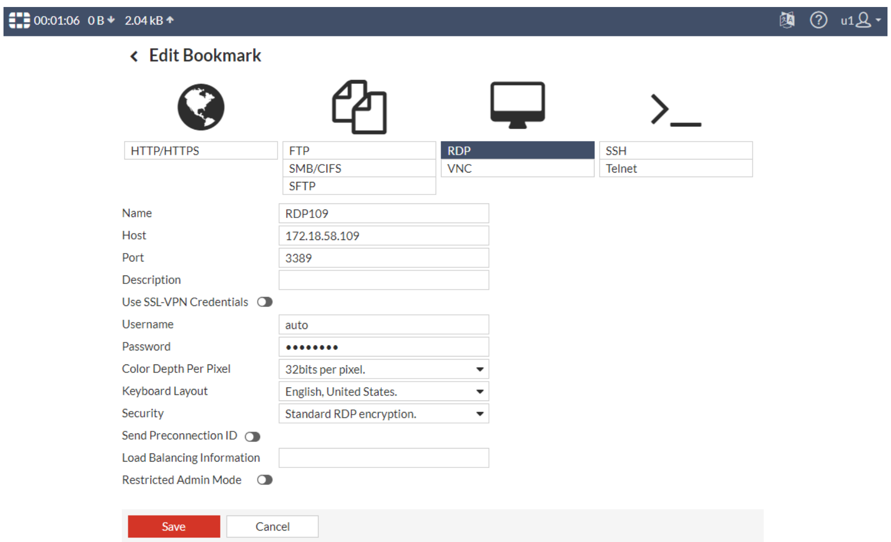
5. Click *Apply*.

To configure a firewall policy for SSL VPN in the GUI:

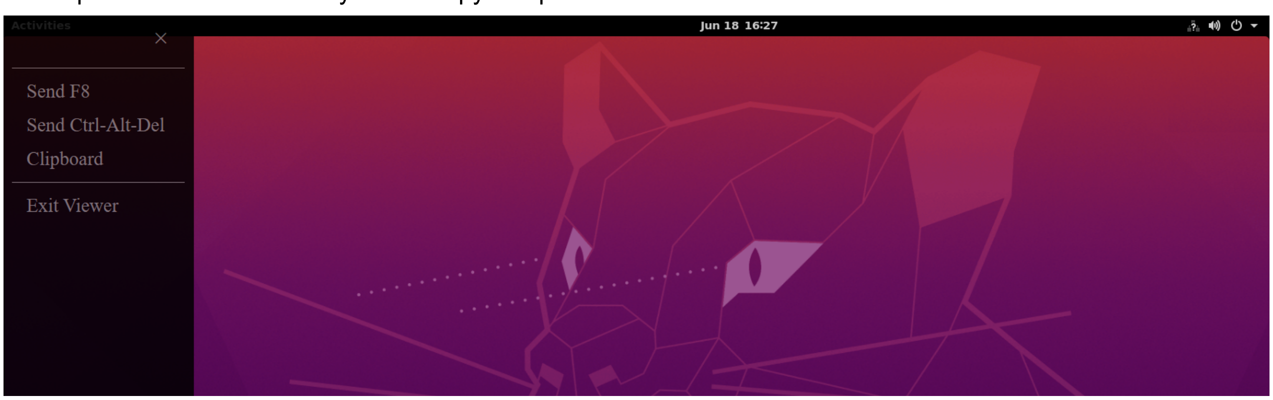
1. Go to *Policy & Objects* > *Firewall Policy* and click *Create New*.
2. Set a name for the policy, such as *policy_to_sslvpn_tunnel*.
3. Set *Incoming Interface* to the SSL VPN tunnel interface and *Outgoing Interface* to port1.
4. Set *Source* to the users, *u1* and *u2*, and all addresses.
5. Set *Destination* to all addresses.
6. Set *Schedule* to *always*, *Service* to *All*, and *Action* to *Accept*.
7. Configure the remaining settings as needed.
8. Click *OK*.

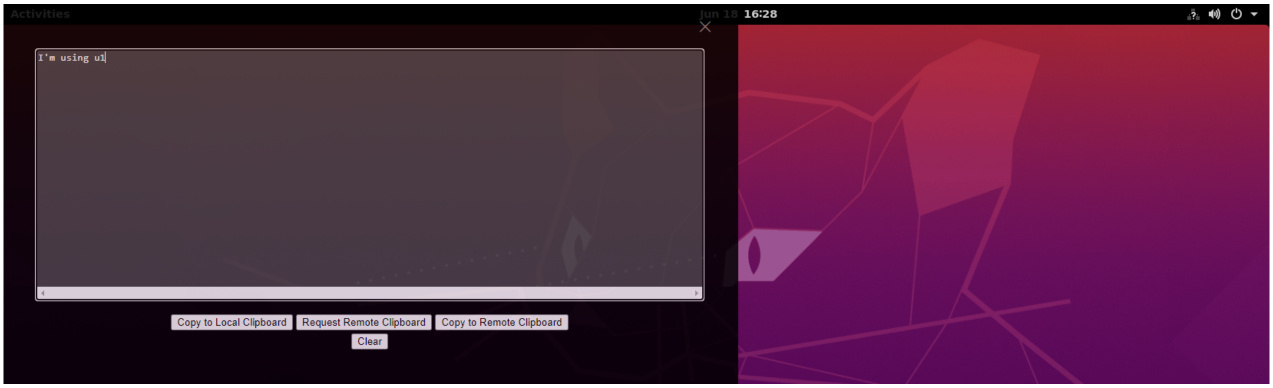
To test the if the users can use the clipboard:

1. On the PC, open a web browser and log in to the web portal as user *u1*.
2. Access the internal server using RDP/VNC.

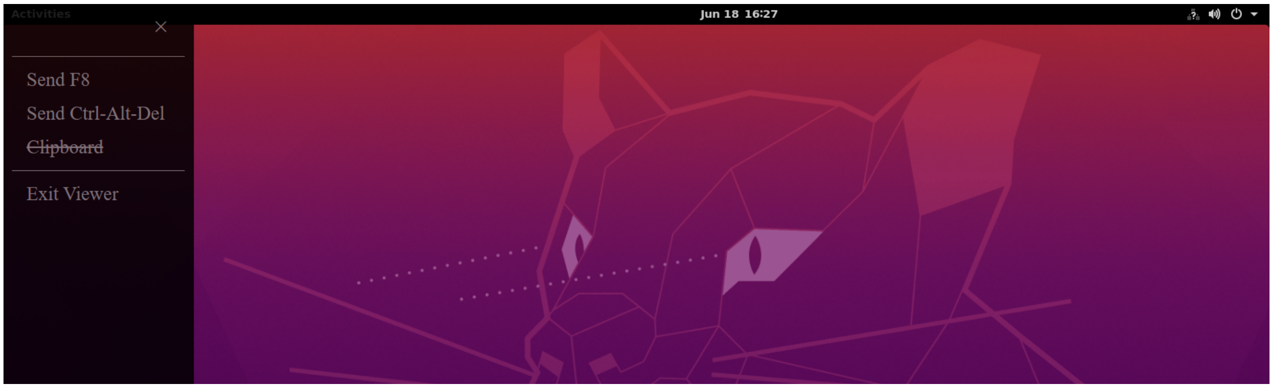


3. The clipboard is available and you can copy and paste content to and from the remote server.





4. Log out of the web portal, then log back in as user *u2* and access the internal server using RDP/VNC.
   The clipboard is disabled.



To configure the SSL-VPN portals and settings in the CLI:

1. Configure the SSL VPN portals:

```
config vpn ssl web portal
    edit "testportal1"
        set web-mode enable
        set clipboard enable
        ...
    next
    edit "testportal2"
        set web-mode enable
        set clipboard disable
        ...
    next
end
```

2. Configure the SSL VPN settings:

```
config vpn ssl settings
    set port 1443
    set source-interface "port2"
    set source-address "all"
    set source-address6 "all"
    set default-portal "tunnel-access"
    config authentication-rule
        edit 1
```

```
            set users "u1"
            set portal "testportal1"
        next
        edit 2
            set users "u2"
            set portal "testportal2"
        next
    end
end
```

3. Configure a firewall policy for SSL VPN:

```
config firewall policy
    edit 1
        set name "policy_to_sslvpn_tunnel"
        set srcintf "ssl.vdom1"
        set dstintf "port1"
        set action accept
        set srcaddr "all"
        set dstaddr "all"
        set srcaddr6 "all"
        set dstaddr6 "all"
        set schedule "always"
        set service "ALL"
        set nat enable
        set users "u1" "u2"
    next
end
```

4. On the PC, open a web browser, log in to the web portal as user *u1*, access the internal server using RDP/VNC, and use the clipboard.
5. Check the SSL VPN session monitor:

```
# get vpn ssl monitor
SSL-VPN Login Users:
 Index   User    Group   Auth Type      Timeout         Auth-Timeout    From     HTTP
in/out    HTTPS in/out    Two-factor Auth
 0       u1             1(1)             N/A     10.1.100.146   0/0     0/364   0

SSL-VPN sessions:
 Index   User    Group   Source IP      Duration        I/O Bytes       Tunnel/Dest IP
 0       u1             10.1.100.146     64      0/700  RDP 172.18.58.109
```

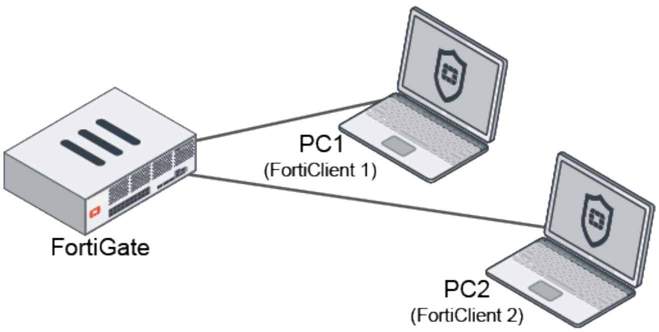
6. On the PC, open a web browser, log in to the web portal as user *u2*, access the internal server using RDP/VNC, and note that the clipboard is not available.
7. Check the SSL VPN session monitor:

```
# get vpn ssl monitor
SSL-VPN Login Users:
 Index   User    Group   Auth Type      Timeout         Auth-Timeout    From     HTTP
in/out    HTTPS in/out    Two-factor Auth
 0       u2             1(1)             N/A     10.1.100.146   0/0     0/2681  0

SSL-VPN sessions:
 Index   User    Group   Source IP      Duration        I/O Bytes       Tunnel/Dest IP
 0       u2             10.1.100.146     7       0/553  RDP 172.18.58.109
```

SSL VPN IP address assignments

When a user disconnects from a VPN tunnel, it is not always desirable for the released IP address to be used immediately. In SSL VPN, IP addresses can be assigned from the pool in a round robin fashion, instead of the default first-available address method.

Example



In this example, two PCs connect to the VPN. SSL VPN is configured to use round robin IP address assignment. Dual stack address assignment (both IPv4 and IPv6) is used.

After a tunnel is disconnected, freeing a low IP address, the next client that connects gets the next address in the round robin instead of the lowest address.

To configure SSL VPN with round robin and dual stack:

1. Create IPv4 and IPv6 address ranges:

```
config firewall address
    edit "sslvpn_ipv4_pool"
        set type iprange
        set start-ip 173.10.1.1
        set end-ip 173.10.1.3
    next
end
```

```
config firewall address6
    edit "sslvpn_ipv6_pool"
        set type iprange
        set start-ip 2000::ad0a:101
        set end-ip 2000::ad0a:103
    next
end
```

2. Set the address ranges as IP pools in the SSL VPN settings:

```
config vpn ssl settings
    set tunnel-ip-pools "sslvpn_ipv4_pool"
    set tunnel-ipv6-pools "sslvpn_ipv6_pool"
end
```

When round-robin is used, any address pools defined in the web portal are ignored and the tunnel IPv4 and IPv6 pool addresses in the SSL VPN settings are used. Only one set of IP pool addresses can be applied.

3. Enable round-robin and dual stack in the SSL VPN settings:

```
config vpn ssl settings
    set dual-stack-mode enable
    set tunnel-addr-assigned-method round-robin
end
```

By default, the IP pool assignment follows the first available rule.

4. Create two users and assign them to an SSL VPN policy:

```
config user local
    edit "u1"
        set type password
        set passwd **********
    next
    edit "u2"
        set type password
        set passwd **********
    next
end
```

```
config firewall policy
    edit 1
        set name "sslvpnd"
        set srcintf "ssl.vdom1"
        set dstintf "link_11" "port1"
        set action accept
        set srcaddr "all"
        set dstaddr "all"
        set srcaddr6 "all"
        set dstaddr6 "all"
        set schedule "always"
        set service "ALL"
        set nat enable
        set users "u1" "u2"
    next
end
```

To test the results:

1. Log in to the SSL VPN on PC1 using user u1 and then check its assigned IP address:

```
# get vpn ssl monitor
SSL-VPN Login Users:
 Index   User    Group   Auth Type      Timeout         Auth-Timeout    From     HTTP
in/out    HTTPS in/out    Two-factor Auth
 0       u1             1(1)             N/A     10.1.100.145   0/0     0/0     0

SSL-VPN sessions:
 Index   User    Group   Source IP      Duration        I/O Bytes       Tunnel/Dest IP
 0       u1             10.1.100.145     13      49935/35251
173.10.1.1,2000::ad0a:101
```

2. Log in to the SSL VPN on PC1 using user u2 and then check its assigned IP address:

```
# get vpn ssl monitor
SSL-VPN Login Users:
 Index    User     Group   Auth Type       Timeout          Auth-Timeout     From      HTTP
in/out    HTTPS in/out    Two-factor Auth
 0        u1              1(1)             N/A      10.1.100.145    0/0      0/0      0
 1        u2              1(1)             N/A      10.1.100.254    0/0      0/0      0

SSL-VPN sessions:
 Index    User     Group   Source IP       Duration         I/O Bytes        Tunnel/Dest IP
 0        u1              10.1.100.145     44       90126/70405
173.10.1.1,2000::ad0a:101
 1        u2              10.1.100.254     10       10563/8158
173.10.1.2,2000::ad0a:102
```

3. Log user u1 off of PC1, then log them back in and check that the assigned IP address is not the same as was previously assigned:

```
# get vpn ssl monitor
SSL-VPN Login Users:
 Index    User     Group   Auth Type       Timeout          Auth-Timeout     From      HTTP
in/out    HTTPS in/out    Two-factor Auth
 0        u1              1(1)             N/A      10.1.100.145    0/0      0/0      0
 1        u2              1(1)             N/A      10.1.100.254    0/0      0/0      0

SSL-VPN sessions:
 Index    User     Group   Source IP       Duration         I/O Bytes        Tunnel/Dest IP
 0        u1              10.1.100.145     10       50992/41159
173.10.1.3,2000::ad0a:103
 1        u2              10.1.100.254     43       30374/21860
173.10.1.2,2000::ad0a:102
```

SSL VPN troubleshooting

The following topics provide information about SSL VPN troubleshooting:

- Debug commands on page 1778
- Troubleshooting common issues on page 1779

Debug commands

SSL VPN debug command

Use the following diagnose commands to identify SSL VPN issues. These commands enable debugging of SSL VPN with a debug level of -1 for detailed results.

```
diagnose debug application sslvpn -1
diagnose debug enable
```

The CLI displays debug output similar to the following:

```
FGT60C3G10002814 # [282:root]SSL state:before/accept initialization (172.20.120.12)
[282:root]SSL state:SSLv3 read client hello A (172.20.120.12)
[282:root]SSL state:SSLv3 write server hello A (172.20.120.12)
[282:root]SSL state:SSLv3 write change cipher spec A (172.20.120.12)
[282:root]SSL state:SSLv3 write finished B (172.20.120.12)
```

```
[282:root]SSL state:SSLv3 flush data (172.20.120.12)
[282:root]SSL state:SSLv3 read finished A:system lib(172.20.120.12)
[282:root]SSL state:SSLv3 read finished A (172.20.120.12)
[282:root]SSL state:SSL negotiation finished successfully (172.20.120.12)
[282:root]SSL established: DHE-RSA-AES256-SHA SSLv3 Kx=DH Au=RSA Enc=AES(256) Mac=SHA1
```

To disable the debug:

```
diagnose debug disable
diagnose debug reset
```

Remote user authentication debug command

Use the following diagnose commands to identify remote user authentication issues.

```
diagnose debug application fnbamd -1
diagnose debug reset
```

Troubleshooting common issues

To troubleshoot getting no response from the SSL VPN URL:

1. Go to *VPN > SSL-VPN Settings*.
   a. Confirm that SSL VPN is enabled.
   b. Check the SSL VPN port assignment.
   c. Check the *Restrict Access* setting to ensure the host you are connecting from is allowed.
2. Go to *Policy > Firewall Policy*.
   a. Check that the policy for SSL VPN traffic is configured correctly.
   b. Check the URL you are attempting to connect to. It should follow this pattern:
      ```
      https://<FortiGate IP>:<Port>
      ```
   c. Check that you are using the correct port number in the URL. Ensure FortiGate is reachable from the computer.
      ```
      ping <FortiGate IP>
      ```
   d. Check the browser has *TLS 1.1*, *TLS 1.2*, and *TLS 1.3* enabled.

To troubleshoot FortiGate connection issues:

1. Check the Release Notes to ensure that the FortiClient version is compatible with your version of FortiOS.
2. FortiClient uses IE security setting, In IE *Internet options > Advanced > Security*, check that *Use TLS 1.1* and *Use TLS 1.2* are enabled.
3. Check that SSL VPN *ip-pools* has free IPs to sign out. The default *ip-poolsSSLVPN_TUNNEL_ADDR1* has 10 IP addresses.
4. Export and check FortiClient debug logs.
   a. Go to *File > Settings*.
   b. In the *Logging* section, enable *Export logs*.
   c. Set the *Log Level* to *Debug* and select *Clear logs*.
   d. Try to connect to the VPN.
   e. When you get a connection error, select *Export logs*.

To troubleshoot SSL VPN hanging or disconnecting at 98%:

1. A new SSL VPN driver was added to FortiClient 5.6.0 and later to resolve SSL VPN connection issues. If your FortiOS version is compatible, upgrade to use one of these versions.
2. Latency or poor network connectivity can cause the login timeout on the FortiGate. In FortiOS 5.6.0 and later, use the following commands to allow a user to increase the SSL VPN login timeout setting.

```
config vpn ssl settings
  set login-timeout 180 (default is 30)
  set dtls-hello-timeout 60 (default is 10)
end
```

To troubleshoot tunnel mode connections shutting down after a few seconds:

This might occur if there are multiple interfaces connected to the Internet, for example, SD-WAN. This can cause the session to become “dirty”. To allow multiple interfaces to connect, use the following CLI commands.

If you are using a FortiOS 6.0.1 or later:

```
config system interface
  edit <name>
    set preserve-session-route enable
  next
end
```

If you are using a FortiOS 6.0.0 or earlier:

```
config vpn ssl settings
   set route-source-interface enable
end
```

To troubleshoot users being assigned to the wrong IP range:

1. Go to *VPN > SSL-VPN Portals* and *VPN > SSL-VPN Settings* and ensure the same *IP Pool* is used in both places. Using the same *IP Pool* prevents conflicts. If there is a conflict, the portal settings are used.

To troubleshoot slow SSL VPN throughput:

Many factors can contribute to slow throughput.

This recommendation tries to improve throughput by using the FortiOS Datagram Transport Layer Security (DTLS) tunnel option, available in FortiOS 5.4 and above.

DTLS allows SSL VPN to encrypt traffic using TLS and uses UDP as the transport layer instead of TCP. This avoids retransmission problems that can occur with TCP-in-TCP.

FortiClient 5.4.0 to 5.4.3 uses DTLS by default. FortiClient 5.4.4 and later uses normal TLS, regardless of the DTLS setting on the FortiGate.

To use DTLS with FortiClient:

1. Go to *File > Settings* and enable *Preferred DTLS Tunnel*.

To enable DTLS tunnel on FortiGate, use the following CLI commands:

```
config vpn ssl settings
  set dtls-tunnel enable
end
```

User & Authentication

In *User & Authentication*, you can control network access for different users and devices in your network. FortiGate authentication controls system access by user group. By assigning individual users to the appropriate user groups you can control each user's access to network resources. You can define local users and peer users on the FortiGate unit. You can also define user accounts on remote authentication servers and connect them to FortiOS.

You can control network access for different device types in your network by doing the following:

- Identifying and monitoring the types of devices connecting to your network
- Using MAC address based access control to allow or deny individual devices
- Using Telemetry data received from FortiClient endpoints to construct a policy to deny access to endpoints with known vulnerabilities or to quarantine compromised endpoints

The following sections provide information about users and devices:

Endpoint control and compliance

The section contains the following topics:

Per-policy disclaimer messages

FortiOS supports a customizable captive portal to direct users to install or enable required software.

Per-policy custom disclaimers in each VDOM are supported. For example, you may want to configure three firewall policies, each of which matches traffic from endpoints with different FortiClient statuses:

Endpoint status	FortiOS behavior
Endpoint does not have FortiClient installed.	Traffic matches a firewall policy that displays an in-browser warning to install FortiClient from the provided link.
Endpoint has FortiClient installed, registered to EMS, and connected to the FortiGate.	Traffic matches a dynamic firewall policy which allows the endpoint to reach its destination via this policy.
Endpoint is deregistered from EMS and disconnected from the FortiGate.	Traffic matches another dynamic firewall policy that displays warning to register FortiClient to EMS.

The replacement message groups and policy disclaimer settings must be enabled.

To enable per-policy disclaimer messages in the GUI:

1. Go to *System* > *Feature Visibility*.
2. Enable *Replacement Message Groups* and *Policy Disclaimer*.
3. Click *Apply*.

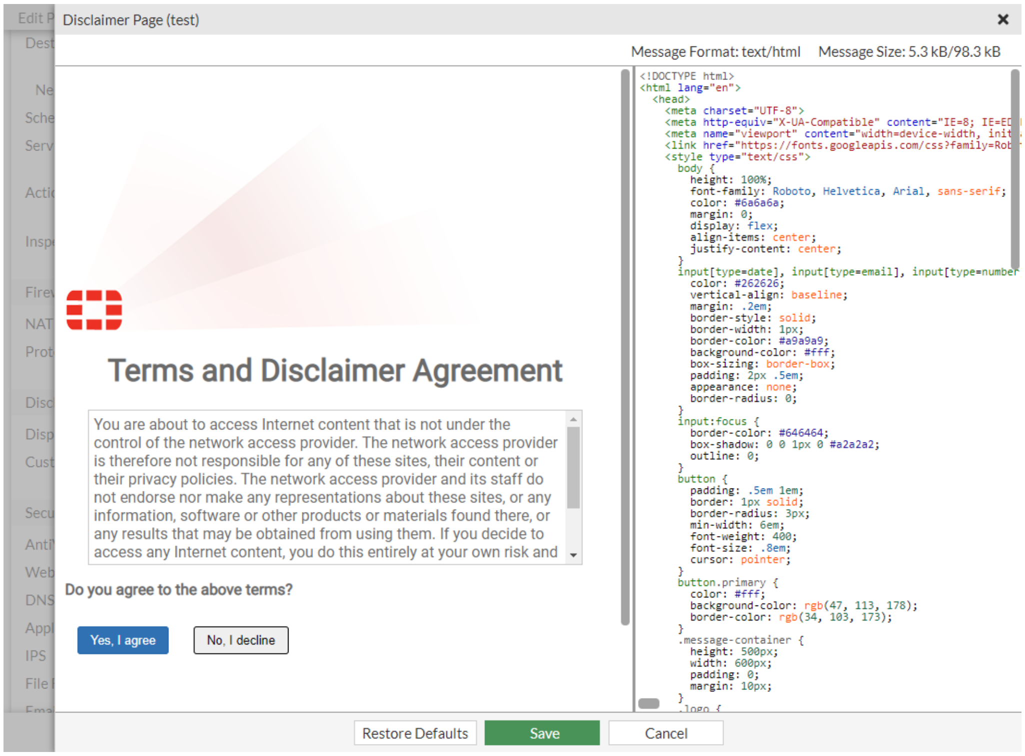
To enable per-policy disclaimer messages in the CLI:

```
config system global
    set gui-replacement-message-groups enable
end

config system settings
    set gui-policy-disclaimer enable
end
```

To configure per-policy disclaimers in the GUI:

1. Ensure the per-policy disclaimer messages option is enabled.
2. Go to *Policy & Objects* > *Firewall Policy*.
3. Edit the policy that applies when an endpoint does not have FortiClient installed.
4. Under *Disclaimer Options*, enable *Display Disclaimer* and *Customize Messages*.
5. Add a replacement message group:
   a. Select an existing replacement message group from the dropdown and click *Edit Disclaimer Message*.
   b. Click *Create*, enter a name, and click *OK*. Select the replacement message group and click *Edit*

Disclaimer Message.



6. Edit the message to warn users to install FortiClient, and provide the FortiClient download link.
7. Click *Save*.
8. Repeat the above steps for each policy that requires a custom disclaimer message.

To configure per-policy disclaimers in the CLI:

```
config firewall policy
    edit 1
        set name "111"
        set srcintf "port12"
        set dstintf "port11"
        set srcaddr "all"
        set dstaddr "pc155_address"
        set action accept
        set schedule "always"
        set service "ALL"
        set wsso disable
        set groups "ems_03_group"
        set disclaimer enable
        set replacemsg-override-group "test"
        set nat enable
    next
    edit 4
        set name "44"
        set srcintf "port12"
        set dstintf "port11"
        set srcaddr "all"
        set dstaddr "pc5-address"
        set action accept
```

```
        set schedule "always"
        set service "ALL"
        set wsso disable
        set groups "ems_03_group"
        set disclaimer enable
        set replacemsg-override-group "test2"
        set nat enable
    next
    edit 6
        set name "66"
        set srcintf "port12"
        set dstintf "port11"
        set srcaddr "all"
        set dstaddr "all"
        set status disable
        set schedule "always"
        set service "ALL"
        set logtraffic all
        set fsso disable
        set block-notification enable
        set replacemsg-override-group "endpoint-override"
    next
end
```

Compliance

The following topics provide information about compliance in FortiOS.

- FortiGate VM unique certificate on page 1784
- Running a file system check automatically on page 1785

FortiGate VM unique certificate

To safeguard against certificate compromise, FortiGate VM and FortiAnalyzer VM use the same deployment model as FortiManager VM where the license file contains a unique certificate tied to the serial number of the virtual device.

A hardware appliance usually comes with a BIOS certificate with a unique serial number that identifies the hardware appliance. This built-in BIOS certificate is different from a firmware certificate. A firmware certificate is distributed in all appliances with the same firmware version.

Using a BIOS certificate with a built-in serial number provides a high trust level for the other side in X.509 authentication.

Since a VM appliance has no BIOS certificate, a signed VM license can provide an equivalent of a BIOS certificate. The VM license assigns a serial number in the BIOS equivalent certificate. This gives the certificate an abstract access ability, which is similar to a BIOS certificate with the same high trust level.

This feature is only supported in new, registered VM licenses.

Sample configurations

Depending on the firmware version and VM license, the common name (CN) on the certificate will be configured differently.

License	Firmware			
	6.0	6.2	6.4	7.0
6.0	CN = FortiGate	CN = FortiGate	CN = FortiGate	CN = FortiGate
6.2	CN = FortiGate	CN = serial number	CN = serial number	CN = serial number
6.4	CN = FortiGate	CN = serial number	CN = serial number	CN = serial number
7.0	CN = FortiGate	CN = serial number	CN = serial number	CN = serial number

To view validated certificates:

1. Go to *System* > *Certificates*.
2. Double-click on a VM certificate. There are two VM certificates:
   - *Fortinet_Factory*
   - *Fortinet_Factory_Backup*

   The *Certificate Detail Information* window displays.

Running a file system check automatically

There is an option in FortiOS to enable automatic file system checks if the FortiGate shuts down ungracefully.

By default, the automatic file system check is disabled. When an administrator logs in after an ungraceful shutdown, a warning message appears advising them to manually run a file system check. A warning also appears in the CLI:

```
WARNING: File System Check Recommended! Unsafe reboot may have caused inconsistency in disk
drive.
It is strongly recommended that you check file system consistency before proceeding.
Please run 'execute disk scan 17'
Note: The device will reboot and scan during startup. This may take up to an hour
```

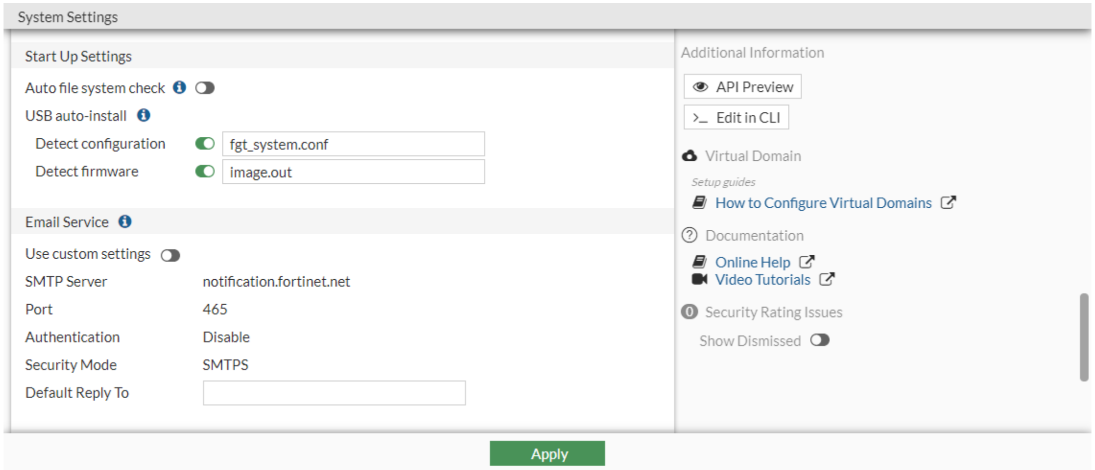
Enabling automatic file system checks

You can enable automatic file system checks in both the GUI and CLI.

To enable automatic file system checks in the GUI:

1. Go to *System* > *Settings*.
2. In the *Start Up Settings* section, enable *Auto file system check*.

3. Click *Apply*.



To enable automatic file system checks using the CLI:

```
config system global
   set autorun-log-fsck enable
end
```

FortiGuard distribution of updated Apple certificates

Push notifications for iPhone (for the purpose of two-factor authentication) require a TLS server certificate to authenticate to Apple. As this certificate is only valid for one year, a service extension allows FortiGuard to distribute updated TLS server certificates to FortiGate when needed.

FortiGuard update service updates local Apple push notification TLS server certificates when the local certificate is expired. FortiGuard update service also reinstalls certificates when the certificates are lost.

You can verify that the feature is working on the FortiGate by using the CLI shell.

To verify certificate updates:

1. Using FortiOS CLI shell, verify that all certificates are installed:

```
/data/etc/apns # ls -al
drwxr-xr-x    2 0        0        Tue Jan 15 08:42:39 2019          1024 .
drwxr-xr-x   12 0        0        Tue Jan 15 08:45:00 2019          2048 ..
-rw-r--r--    1 0        0        Sat Jan 12 00:06:30 2019          2377 apn-dev-cert.pem
-rw-r--r--    1 0        0        Sat Jan 12 00:06:30 2019          1859 apn-dev-key.pem
-rw-r--r--    1 0        0        Sat Jan 12 00:06:30 2019          8964 apn-dis-cert.pem
-rw-r--r--    1 0        0        Sat Jan 12 00:06:30 2019          4482 apn-dis-key.pem
```

2. Rename all current Apple certificates.
   Apple push notification no longer works after you rename the certificates.

```
/data/etc/apns # mv apn-dis-cert.pem apn-dis-cert.pem.save
/data/etc/apns # mv apn-dev-key.pem apn-dev-key.pem.save
/data/etc/apns # mv apn-dev-cert.pem apn-dev-cert.pem.save
/data/etc/apns # mv apn-dis-key.pem apn-dis-key.pem.save
/data/etc/apns # ls -al
```

```
drwxr-xr-x   2 0   0   Tue Jan 15 08:51:15 2019   1024 .
drwxr-xr-x  12 0   0   Tue Jan 15 08:45:00 2019   2048 ..
-rw-r--r--   1 0   0   Sat Jan 12 00:06:30 2019   2377 apn-dev-cert.pem.save
-rw-r--r--   1 0   0   Sat Jan 12 00:06:30 2019   1859 apn-dev-key.pem.save
-rw-r--r--   1 0   0   Sat Jan 12 00:06:30 2019   8964 apn-dis-cert.pem.save
-rw-r--r--   1 0   0   Sat Jan 12 00:06:30 2019   4482 apn-dis-key.pem.save
```

3. Run a FortiGuard update, and verify that all certificates are installed again:

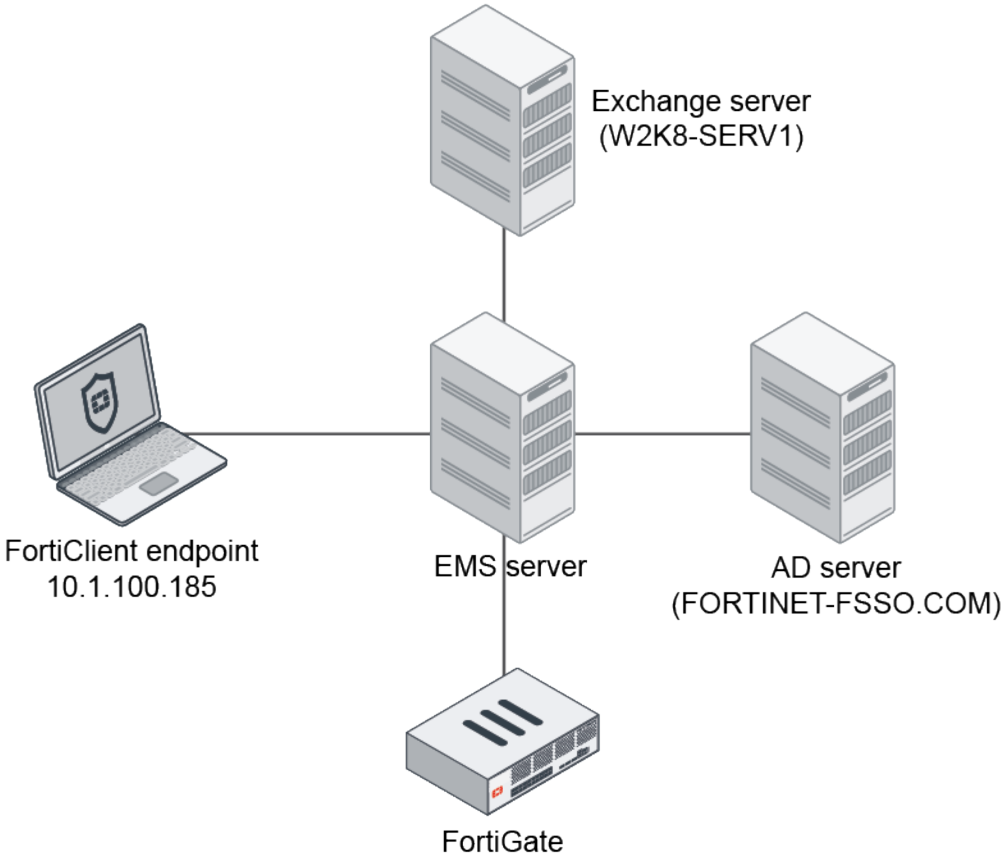
```
/data/etc/apns # ls -al
drwxr-xr-x   2 0   0   Tue Jan 15 08:56:20 2019   1024 .
drwxr-xr-x  12 0   0   Tue Jan 15 08:56:15 2019   2048 ..
-rw-r--r--   1 0   0   Sat Jan 12 00:06:30 2019   2377 apn-dev-cert.pem.save
-rw-r--r--   1 0   0   Sat Jan 12 00:06:30 2019   1859 apn-dev-key.pem.save
-rw-r--r--   1 0   0   Tue Jan 15 08:56:20 2019   2167 apn-dis-cert.pem  <-- downloaded
from FortiGuard
-rw-r--r--   1 0   0   Sat Jan 12 00:06:30 2019   8964 apn-dis-cert.pem.save
-rw-r--r--   1 0   0   Tue Jan 15 08:56:20 2019   1704 apn-dis-key.pem   <-- downloaded
from FortiGuard
-rw-r--r--   1 0   0   Sat Jan 12 00:06:30 2019   4482 apn-dis-key.pem.save
-rw-r--r--   1 0   0   Tue Jan 15 08:56:20 2019     41 apn-version.dat   <-- downloaded
from FortiGuard
/data/etc/apns #
```

Integrate user information from EMS and Exchange connectors in the user store

When a FortiClient endpoint is managed by EMS, logged in user and domain information is shared with FortiOS through the EMS connector. This information can be joined with the Exchange connector to produce more complete user information in the user store.

The `diagnose user-device-store device memory list` command displays detailed device information.

Sample topology



In this example, the FortiClient PC user (test1) logs on to the AD domain (FORTINET-FSSO.COM), which is also the same domain as the Exchange server. The user information is pushed to the EMS server that the user is registered to. The FortiGate synchronizes the information from EMS, and at the same time looks up the user on the Exchange server under the Exchange connector. If the user exists on the Exchange server, additional information is fetched. These details are combined in the user store, which is visible in the *FortiClient* widget in the *Status* dashboard.

To configure the Exchange server:

```
config user exchange
    edit "exchange-140"
        set server-name "W2K8-SERV1"
        set domain-name "FORTINET-FSSO.COM"
        set username "Administrator"
        set password ********
    next
end
```

To configure the EMS server:

```
config endpoint-control fctems
    edit "ems133"
        set server "172.18.62.12"
        set certificate-fingerprint "4F:A6:76:E2:00:4F:A6:76:E2:00:4F:A6:76:E2:00:E0"
```

```
    next
end
```

To view the user information in the GUI:

1. Go to *Dashboard* > *Status*.
2. In the *FortiClient* widget, hover over a device or user name to view the information.

To view the user information in the CLI:

```
# diagnose user-device-store device memory list
...
Record #13:
        device_info
                'ipv4_address' = '10.1.100.185'
                'mac' = '00:0c:29:11:5b:6b'
                'hardware_vendor' = 'VMware'
                'vdom' = 'root'
                'os_name' = 'Microsoft'
                'os_version' = 'Windows 7 Professional Edition, 32-bit Service Pack 1
(build 7601)'
                'hostname' = 'win7-5'
                'unauth_user' = 'Administrator'
                'last_seen' = '1611356490'
                'host_src' = 'forticlient'
                'user_info_src' = 'forticlient'
                'is_forticlient_endpoint' = 'true'
                'unjoined_forticlient_endpoint' = 'false'
                'is_forticlient_unauth_user' = 'true'
                'avatar_source' = 'OS'
                'domain' = 'Fortinet-FSSO.COM'
                'forticlient_id' = '********************************'
                'forticlient_username' = 'Administrator'
                'forticlient_version' = '6.4.2'
                'on_net' = 'true'
                'quarantined_on_forticlient' = 'false'
                'vuln_count' = '0'
                'vuln_count_critical' = '0'
                'vuln_count_high' = '0'
                'vuln_count_info' = '0'
                'vuln_count_low' = '0'
                'vuln_count_medium' = '0'
                'is_online' = 'true'
        interface_info
                'ipv4_address' = '10.1.100.185'
                'mac' = '00:0c:29:11:5b:6b'
                'master_mac' = '00:0c:29:11:5b:6b'
                'detected_interface' = 'port10'
                'last_seen' = '1611356490'
                'is_master_device' = 'true'
                'is_detected_interface_role_wan' = 'false'
                'detected_interface_fortitelemetry' = 'true'
                'forticlient_gateway_interface' = 'port10'
                'on_net' = 'true'
                'is_online' = 'true'
```

User definition and groups

FortiGate authentication controls system access by user groups. By assigning individual users to the appropriate user groups, this controls each user's access to network resources. The user groups members are user accounts, of which there are several types. Local and peer users are defined in FortiOS. User accounts can also be defined on remote authentication servers.

This section contains information about configuring the following:

- Users on page 1790
- User groups on page 1792
- Retail environment guest access on page 1799
- User and user group timeouts on page 1802

For information about configuring authentication servers, see the LDAP servers on page 1803, RADIUS servers on page 1818, TACACS+ servers on page 1850, and SAML on page 1852 sections.

Users

A user is a user account consisting of a username, password, and sometimes other information, that is configured in FortiOS or on an external authentication server. There are several types of user accounts with slightly different methods of authentication.

User type	Authentication method
Local	The username and password must match a user account stored in FortiOS. Authentication is done by a firewall policy.
Remote	Remote users consist of usernames defined in FortiOS that are authenticated by a remote server. For example, RADIUS, TACACS+, LDAP, or FortiNAC. The server must be configured in FortiOS before creating a user.
FSSO	Users on a Microsoft Windows, Citrix, or Novell network can use their network authentication to access resources through the FortiGate. Access is controlled through FSSO user groups, which contain Windows, Citrix, or Novell user groups as members. The FSSO agent must be configured in FortiOS before creating a user (see FSSO on page 1911).
PKI or peer	A PKI or peer user is a digital certificate holder that authenticates using a client certificate. No password is required, unless two-factor authentication is enabled. In the GUI, the *User & Authentication > PKI* menu is only available after a PKI user is configured in the CLI (see Configuring a PKI user on page 1901).

Some user types have an option to enable multi-factor authentication using FortiToken or FortiToken Cloud. In some cases, the user must be defined first, and then can be edited to add multi-factor authentication. See FortiTokens on page 1882 for more information.

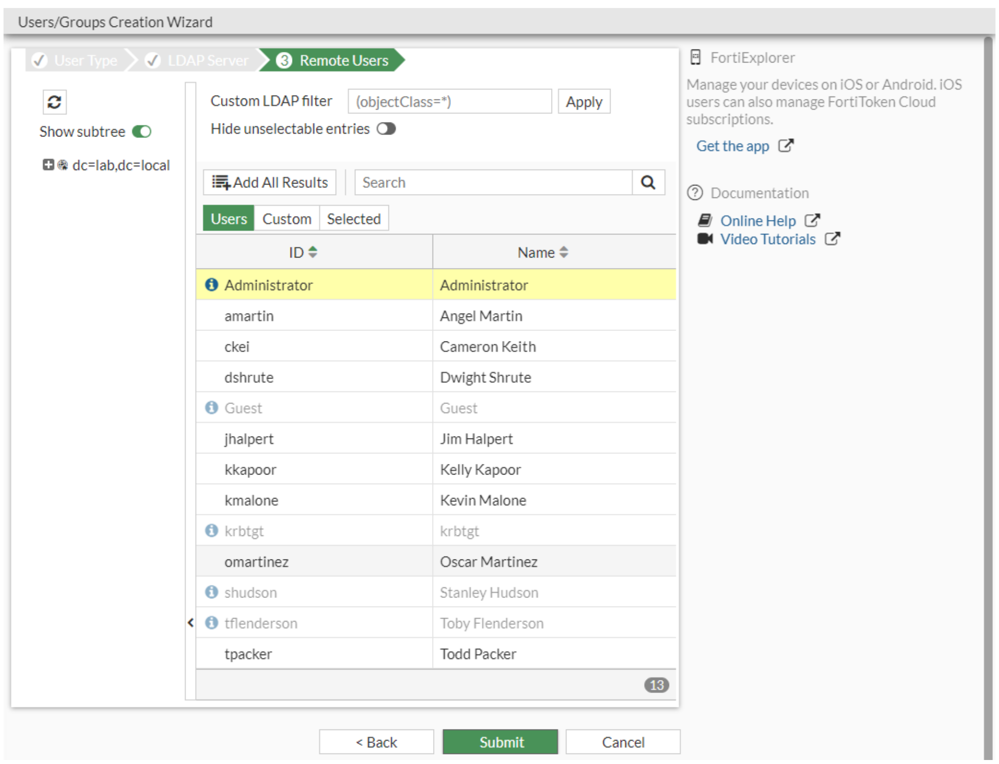
To create a user:

1. Go to User Authentication > User Definition and click Create New. The *Users/Groups Creation Wizard* appears.
2. Select a *User Type* and click *Next*.

3. The remaining wizard steps depend on the user type:
   - *Local User*:
      i. Enter a *Username* and *Password*, then click *Next*.
      ii. Optionally, enable *Two-factor Authentication* and configure the following:

Authentication Type	Select *FortiToken Cloud* or *FortiToken*.
Token	If using FortiToken to authenticate, select a token.
Email Address	Enter an email address.
SMS	Enable to send an SMS message to activate the token.
Country Dial Code	Select the country code.
Phone Number	Enter a phone number.

      iii. Click *Next*, then click *Submit*.
   - *Remote LDAP User*:
      i. Select an *LDAP Server*, then click *Next*.
      ii. Select the users to add from the LDAP server. If the user ID matches an existing configured username, it cannot be added.



      iii. Click *Submit*.
   - *Remote RADIUS User* and *Remote TACACS+ User*:
      i. Enter a *Username* and select the server.
      ii. Click *Next*.
      iii. Optionally, enable *Two-factor Authentication* and configure the settings as needed.
      iv. Click *Next*, then click *Submit*.

- *FSSO*:
    i. Select an *FSSO Agent*, click the + to add *AD Groups*, then click *Next*.
    ii. Select an FSSO group to add the *AD Groups* to. If an FSSO group already exists (see Configuring FSSO user groups on page 1796), click *Choose Existing* and select the group. Otherwise, click *Create New*, enter a name, and click *OK*.
    iii. Click *Submit*.

User groups

A user group is a list of user identities. A user identity can be a:

- Local user account (username/password) stored on the FortiGate
- Remote user account (password stored on a RADIUS, LDAP, or TACACS+ server)
- PKI user account with a digital client authentication certificate stored on the FortiGate
- RADIUS, LDAP, or TACACS+ server, optionally specifying particular user groups on that server
- User group defined on an FSSO server

User groups provide the ability to combine users that require the same permissions so they can be referenced at once, which enables consistency in configurations. User groups allow for remote servers to be referenced by leveraging the pre-existing user accounts, instead of redefining them on the FortiGate.

For example, when a new employee joins a department, they can be added to their respective group, whether in the remote authentication server or local group, and be subject to the same access as their colleagues in the same department. In FortiOS, user groups can be used when configuring firewall policies, traffic shaping policies, proxy policies, SSL VPN portals, IPsec VPN XAUTH, ZTNA, wireless networks (SSIDs), web filtering profiles, identity-based routing, and system administrators with remote authentication.

In most cases, the FortiGate authenticates users by requesting their username and password. The FortiGate checks local user accounts first. If a match is not found, the FortiGate checks the RADIUS, LDAP, or TACACS+ servers that belong to the user group. Authentication succeeds when a matching username and password are found. If the user belongs to multiple groups on a server, those groups will also be matched.

Four types of user groups can be configured:

- Firewall
- FSSO
- RSSO
- Guest

Configuring firewall user groups

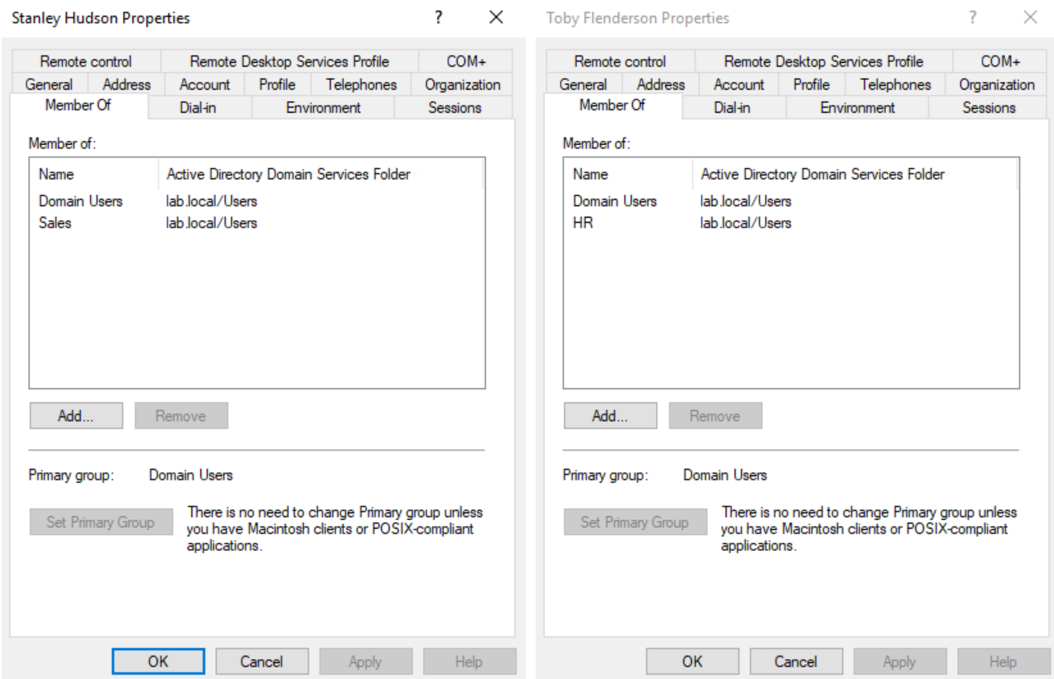
Firewall user groups are used locally as part of authentication. For example, when a firewall policy allows access only to specified user groups, users must authenticate before matching the policy. If the user authenticates successfully and is a member of one of the permitted groups, the policy is applied to the user. A firewall user group may contain local users (defined locally or authenticated remotely), PKI users, or authentication servers.

There are two options to add users in a firewall group configuration: members or remote groups. Members are the individual users who have been defined in FortiOS. Remote groups are remote server that users may authenticate to. One or more user groups can be specified within that server to limit which users can authenticate to the firewall user group. Both options may be used at the same time. The FortiGate attempts to authenticate users in the members list first, and then the remote groups if the initial authentication does not succeed.

When adding remote groups to user groups, FortiTokens cannot be applied to the users. To use remote authentication servers and FortiToken for multi-factor authentication, a remote user type must be created and then added as a user group member.

The following user group configuration examples have local members and a remote authentication server user group. There are two LDAP users, but the principle applies to other remote authentication server types.

Both LDAP users (shudson and tflenderson) belong to the primary group, Domain Users. The user, shudson belongs to the Sales group; tflenderson belongs to the HR group.



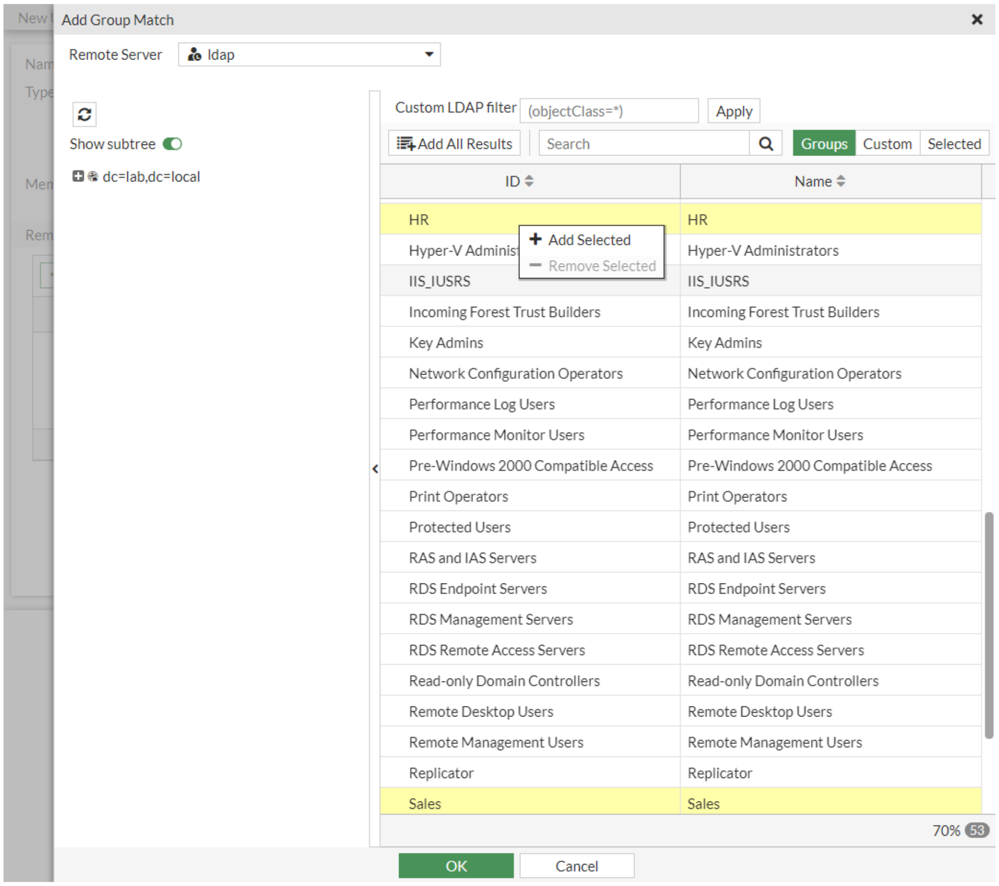
Example 1: Adding multiple remote groups to a user group

In this example, two remote groups (HR and Sales) are added to a firewall group called SSL_VPN_ACCESS.

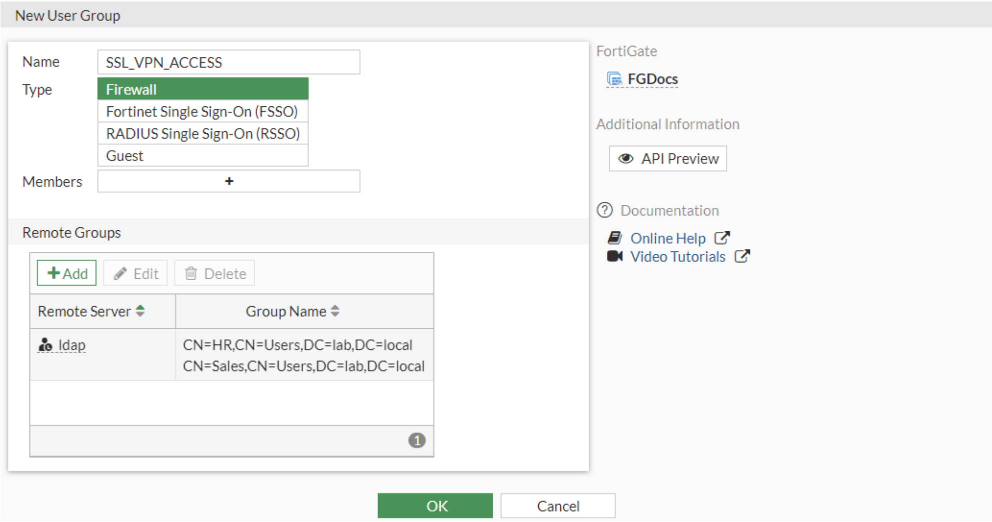
To add multiple remote groups to a user group:

1. Go to *User & Authentication* > *User Groups* and click *Create New*. *Firewall* is selected as the default *Type*.
2. Enter the group name, *SSL_VPN_ACCESS*.
3. In the *Remote Groups Section*, click *Add*.
4. Set *Remote Server* to the LDAP server (*ldap*).
5. In the *Groups* table, select *Sales*, then right-click and select *Add Selected*.

6. Select *HR*, then right-click and select *Add Selected*.



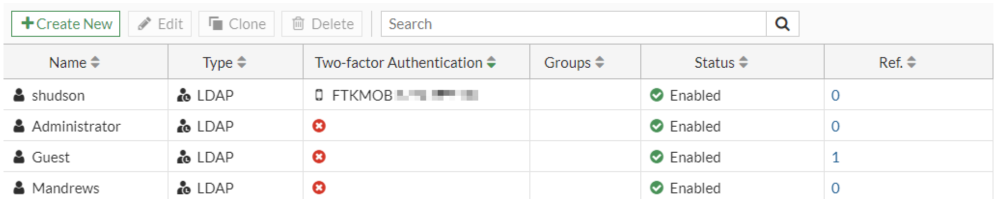
7. Click *OK*.
   Both user group paths are specified under the *Group Name*.



8. Click *OK*.
   In this configuration, shudson and tflenderson would be able to authenticate to this group.

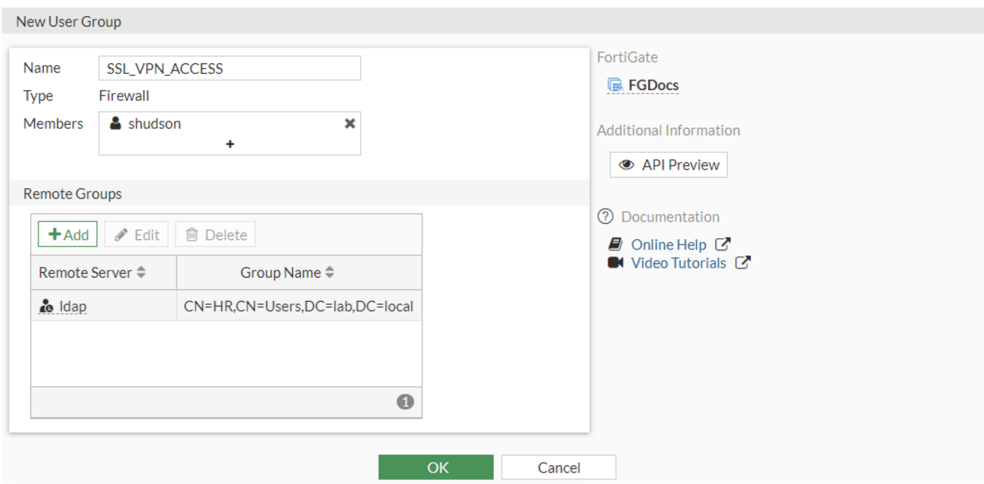
Example 2: combining remote groups and local users in a user group

In this example, the firewall group (SSL_VPN_ACCESS) is configured to contain the HR remote group and a local LDAP user (shudson) with multi-factor authentication.



To combine remote groups and local users in a user group:

1. Go to *User & Authentication > User Groups* and click *Create New*. *Firewall* is selected as the default *Type*.
2. Enter the group name, *SSL_VPN_ACCESS*.
3. In the *Remote Groups Section*, click *Add*.
4. Set *Remote Server* to the LDAP server (*ldap*).
5. In the *Groups* table, select *HR*, then right-click and select *Add Selected*.
6. Click *OK*.
7. In the *Members* field, click the + and add *shudson*.



8. Click *OK*.
   In this configuration, shudson, tflenderson, and any members of the HR LDAP group would be able to authenticate to the user group. Other users in the Sales group are not allowed.

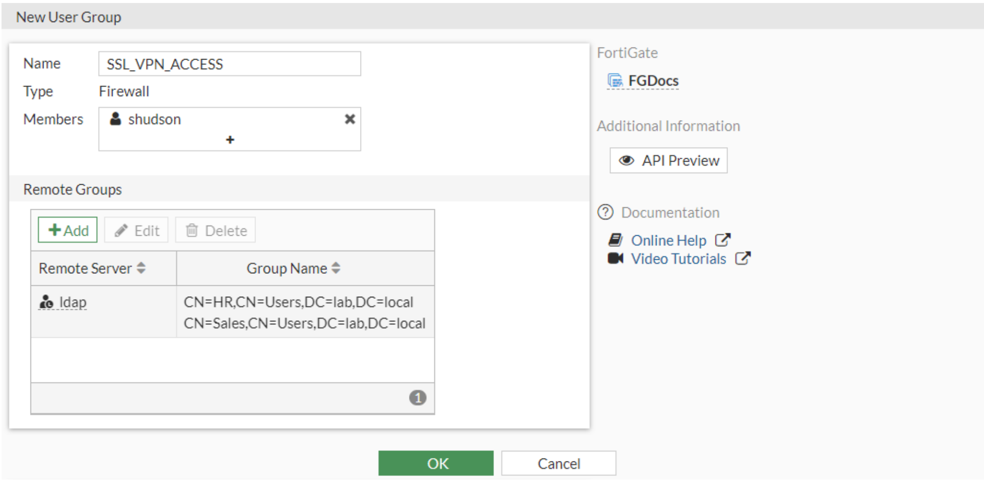
Example 3: adding a user as a member and their group as a remote groups

This example uses a combination of the previous examples. The HR and Sales groups are added as remote groups similar to example 1. The local LDAP user, shudson (using a FortiToken), from example 2 is added as a group member.

This example is for demonstration only. It may cause unwanted results, so this configuration is not advised.

To add a user as a member and their group as a remote groups:

1. Refer to example 1 to configure the two remote groups.
2. In the *Members* field, click the + and add *shudson*.



3. Click *OK*.

One unwanted scenario from this configuration is that a user might be able to bypass multi-factor authentication on LDAP by changing the username case (see the related PSIRT advisory). By default, the username of the remote LDAP user is case sensitive. This means the username has to match what is configured (shudson). If a user types `sHudson`, for example, this will not match the user shudson, so it falls through to remote group authentication. It will match the Sales group in this example. To prevent this, disable username case sensitivity (see SSL VPN for remote users with MFA and user sensitivity on page 1668 for more details).

To disable case sensitivity on the remote user:

```
config user local
    edit <name>
        set type ldap
        set two-factor fortitoken
        set fortitoken "FTKMOBxxxxxxxxxx"
        set email-to <email_address>
        set username-sensitivity disable
        set ldap-server <server_name>
    next
end
```

There is another unwanted scenario from this configuration than can occur to bypass multi-factor authentication. The LDAP server, ldap, has a user named shudson. Another LDAP server, ldap2, also has a user named shudson, but with a different password. If the ldap and ldap2 servers are asdded to the user group in addition to the remote shudson user, if a user tries to log in using `shudson` and the password on the ldap2 server, they would be able to bypass multi-factor authentication.

Configuring FSSO user groups

FSSO user groups contain only Windows, Citrix, and Novell network users. Information about these user groups and their member logon activities are provided by the corresponding FSSO connector. See the FSSO on page 1911 section for more information.

Configuring RSSO user groups

RADIUS single sign-on user groups leverage a RADIUS server to authenticate connecting users. This requires users to log in to their computer using their RADIUS account. The FortiGate does not interact with the remote RADIUS server. It only monitors RADIUS accounting records that the server forwards (originating from the RADIUS client). These records include the user IP address and user group. See RADIUS single sign-on agent on page 2498 for more information.

Configuring guest user groups

In some scenarios, an administrator might need to create temporary user accounts with a defined expiry time to access network resources. For example, if there is a large conference and may attendees require temporary network access for a few days. *Guest Management* can be used to combine many guest users into a group. Many guest accounts can be created at once using randomly-generated user IDs and passwords.

A guest group must be configured first. The guest user account user ID can be an email address, a randomly generated string, or an ID that the assigned by the administrator. The password can be assigned by the administrator or randomly generated. The guest group configuration determines the fields that are provided when creating guest user accounts in *Guest Management*.

To create a guest user group:

1. Go to *User & User & Authentication* > *User Groups* and click *Create New*.
2. Enter a name, and set the *Type* to *Guest*.

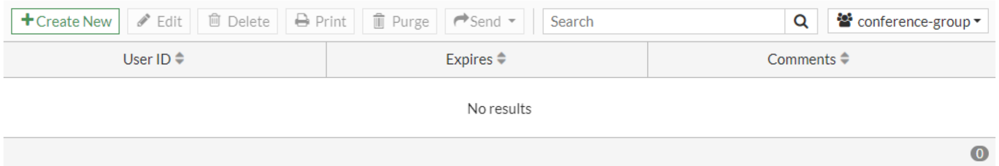
3. Configure the following:

Batch Guest Account Creation	Create multiple accounts automatically. When enabled: • The user ID and password are automatically generated. • The accounts only have user ID, password, and expiration fields. The expiration field is editable in the GUI in the *Start Countdown* and *Time* settings. • An administrator can print the account information. • Users do not receive an email or SMS notification.
User ID	Select one of the following: • *Email*: use the user's email address • *Auto Generate*: FortiOS creates a random user ID • *Specify*: the administrator assigns a user ID
Maximum Accounts	Enable to set a maximum number of guest accounts that can be created for this group (disabled = unlimited).
Guest Details	
Enable Name	If enabled, the user form has a field to enter a name.
Enable Email	If enabled, the user is notified by email.
Enable SMS	If enabled, the user is notified by SMS.
Password	Select one of the following: • *Auto Generate*: FortiOS creates a random password • *Specify*: the administrator assigns a password If the setting is disabled, no password is used.
Sponsor	If enabled, the user form has a field to enter a sponsor (*Optional*). Select *Required* if the sponsor field is mandatory.
Company	If enabled, the user form has a field to enter a company (*Optional*). Select *Required* if the company field is mandatory.
Expiration	
Start Countdown	Select one of the following: • *On Account Creation*: the countdown starts from the time the account is created • *After First Login*: the countdown starts from the time the first time the user logs in
Time	Set the expiry time. There are fields to enter values for *Days*, *Hours*, *Minutes*, and *Seconds*.

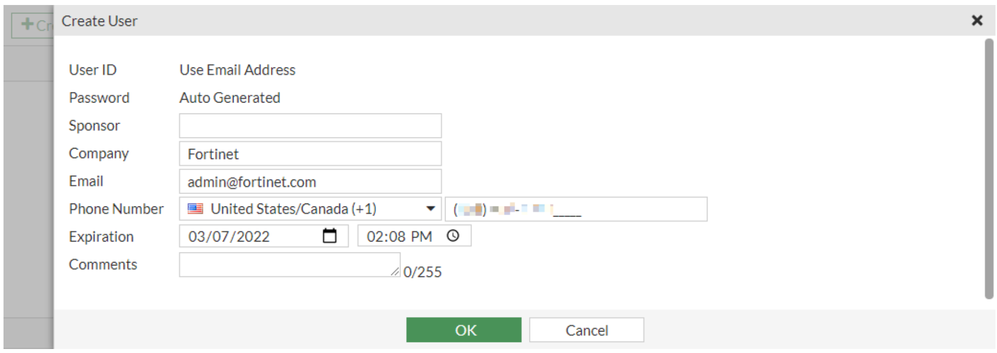
4. Click *OK*.

To manually create a guest user account:

1. Go to *User & User & Authentication* > *Guest Management*.
2. If more than one guest user group is configured, select the group from the dropdown beside the search box.



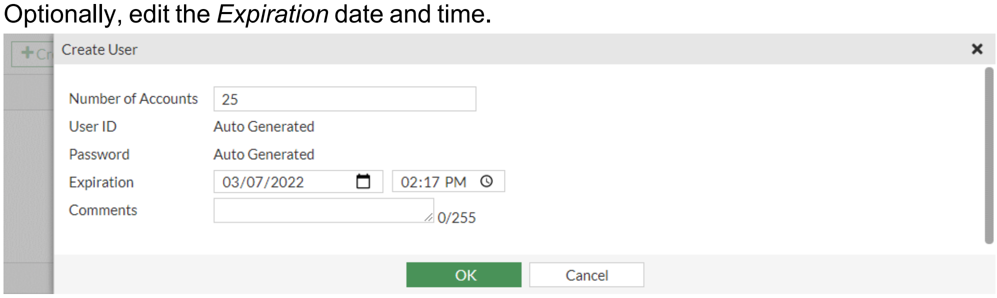
3. Click *Create New* and enter the information in the *Create User* pane. The fields are based on the guest group configuration. Optional fields can be left blank, such as *Sponsor* in this example.



4. Click *OK*.

To automatically create multiple guest user accounts:

1. Go to *User & User & Authentication* > *Guest Management*.
2. If more than one guest user group is configured, select the group from the dropdown beside the search box. The group must have *Batch Guest Account Creation* enabled.
3. Click *Create New* > *Multiple Users* and enter the *Number of Accounts*.
4. Optionally, edit the *Expiration* date and time.



5. Click *OK*.

Retail environment guest access

Businesses such as coffee shops provide free Internet access for customers. In this scenario, you do not need to configure guest management, as customers can access the WiFi access point without logon credentials.

However, consider that the business wants to contact customers with promotional offers to encourage future patronage. You can configure an email collection portal to collect customer email addresses for this purpose. You can configure a firewall policy to grant network access only to users who provide a valid email address. The first time a customer's device attempts WiFi connection, FortiOS requests an email address, which it validates. The customers' subsequent connections go directly to the Internet without interruption.

This configuration consists of the following steps:

1. Creating an email collection portal on page 1800
2. Creating a firewall policy on page 1800
3. Checking for collected emails on page 1801

Creating an email collection portal

The customer's first contact with your network is a captive portal that presents a webpage requesting an email address. When FortiOS has validated the email address, the customer's device MAC address is added to the collected emails device group.

This example modifies the freewifi WiFi interface to present an email collection captive portal.

To configure the freewifi SSID to use an email collection portal in the GUI:

1. Enable email collection:
   a. Go to *System > Feature Visibility*.
   b. In the *Additional Features* section, enable *Email Collection*.
   c. Click *Apply*.
2. Edit the freewifi SSID:
   a. Go to *WiFi & Switch Controller > SSIDs* and edit the *freewifi* SSID.
   b. In the *Security Mode Settings* section, set the *Security mode* to *Captive Portal*.
   c. Set the *Portal type* to *Email Collection*.
   d. Click *OK*.

To configure the freewifi SSID to use an email collection portal in the CLI:

```
config wireless-controller vap
    edit freewifi
        set security captive-portal
        set portal-type email-collect
    next
end
```

Creating a firewall policy

You must configure a firewall policy that allows traffic to flow from the WiFi SSID to the internet interface only for members of the collected emails device group. This policy must be listed first. Unknown devices are not members of the collected emails device group, so they do not match the policy.

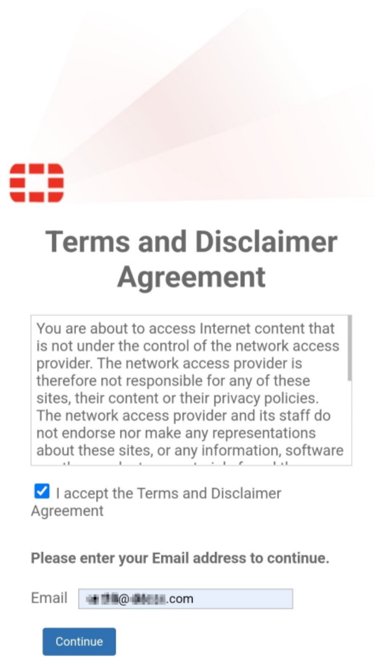
To create a firewall policy:

```
config firewall policy
    edit 3
        set srcintf "freewifi"
        set dstintf "wan1"
        set srcaddr "all"
        set dstaddr "all"
```

```
        set action accept
        set schedule "always"
        set service "ALL"
        set nat enable
        set email-collect enable
    next
end
```

Checking for collected emails

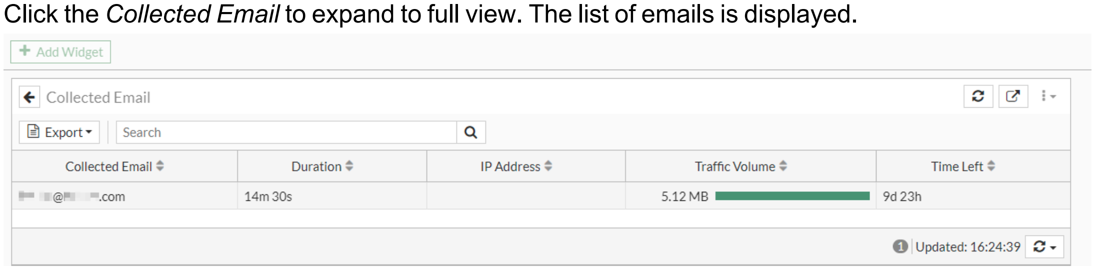
When a WiFi user connects to the freewifi SSID, they are presented with a captive portal to enter their email address.



Once the user enters their email and clicks *Continue*, they will have access to the Internet. The collected emails can be verified in FortiOS.

To check for collected emails in the GUI:

1. Go to *Dashboard* > *Users & Devices* and click *Add Widget*.
2. In the *User & Authentication* section, select *Collected Email* and click *Add Widget*.
3. Click *Close*.
4. Click the *Collected Email* to expand to full view. The list of emails is displayed.



5. Optionally, click *Export* to export the data as a CSV or JSON file.

To check for collected emails in the CLI:

```
# diagnose firewall auth mac list

72:4d:e1:**:**:**, admin@fortinet.com
        type: email, id: 0, duration: 937, idled: 19
        expire: 863980, allow-idle: 864000
        flag(1000): src_idle
        packets: in 4753 out 4592, bytes: in 2662403 out 2458644

----- 1 listed, 0 filtered ------
```

User and user group timeouts

Authenticated user groups can have timeout values per group in addition to FortiGate-wide timeouts. Three types of group timeouts can be configured: idle, hard, and session. These are in addition to any external timeouts, such as those on RADIUS servers.

To configure the timeout type for authenticated users:

```
config user setting
    set auth-timeout-type {idle-timeout | hard-timeout | new-session}
    set auth-timeout <integer>
end
```

Timeouts are measured in minutes (1 - 1440, default = 5). If VDOMs are enabled, the global level `auth-timeout` user setting is the default all VDOMs inherit.

Timeout type	Description
Idle	This is the default setting. The idle timer starts when a user initiates a session. As long as data is transferred in this session, the timer continually resets. If the data flow stops, the timer is allowed to advance until it reaches its limit. When the user has been idle for too long, the user must re-authenticate before traffic is allowed to continue in that session.
Hard	The hard timer starts when a user initiates a session. When the timeout is reached, all the sessions for that user must be re-authenticated. This timeout is not affected by any events.
Session	The session timer starts when a user initiates a session. When the timeout is reached, existing sessions may continue. New sessions are not allowed until the user re-authenticates. This timeout is not affected by any events.

To configure the authentication timeout for a user group:

```
config user group
    edit <name>
        set authtimeout <integer>
    next
end
```

Timeouts are measured in minutes (0 - 43200). A value of zero (the default) means the global timeout is used.

If a user belongs to multiple RADIUS groups, the group `authtimeout` values are ignored. The global `auth-timeout` value is used instead (under `config user setting`).

LDAP servers

The following topics provide information about LDAP servers:

Configuring an LDAP server

FortiOS can be configured to use an LDAP server for authentication.

To configure an LDAP server on the FortiGate:

1. Go to *User & Authentication > LDAP Servers*.
2. Click *Create New*.
3. Configure the following:

Name	This connection name is for reference within the FortiGate only.
Server IP/Name	LDAP server IP address or FQDN resolvable by the FortiGate.
Server Port	By default, LDAP uses port 389 and LDAPS uses 636. Use this field to specify a custom port if necessary.
Common Name Identifier	Attribute field of the object in LDAP that the FortiGate uses to identify the connecting user. The identifier is case sensitive. Common attributes are: • *cn* (Common Name) • *sAMAccountName* (SAMAccountName) • *uid* (User ID)
Distinguished Name	Used to look up user account entries on the LDAP server. It reflects the hierarchy of LDAP database object classes above the CN identifier in which you are doing the lookup. Enter *dc=COMPANY,dc=com* to specify the root of the domain to include all objects.

	Enter *ou=VPN-Users,dc=COMPANY,dc=com* to look up users under a specific organization unit.
Exchange server	Enable to specify the exchange server connector to collect information about authenticated users from a corporate exchange server. See Exchange Server connector on page 2501 for more details.
Bind Type	Select one of the following options: • *Simple*: bind using simple password authentication using the client name. The LDAP server only looks up against the distinguished name (DN), but does not search on the subtree. • *Anonymous*: bind using an anonymous user, and search starting from the DN and recurse over the subtrees. Many LDAP servers do not allow this. • *Regular*: bind using the username and password provided, and search starting from the DN and recurse over the subtrees.
Username	If using regular bind, enter a username with sufficient privileges to access the LDAP server. The following formats are supported: • *username\administrator* • *administrator@domain* • *cn=administrator,cn=users,dc=domain,dc=com*
Password	If using regular bind, enter the password associated with the username.
Secure Connection	Enable to apply security to the LDAP connection through STARTTLS or LDAPS.
Protocol	If *Secure Connection* is enabled, select *STARTTLS* or *LDAPS*. Selecting *STARTTLS* changes the port to 389 and selecting *LDAPS* changes the port to 636.
Certificate	Enable and select the certificate so the FortiGate will only accept a certificate from the LDAP server that is signed by this CA.
Server identity check	Enable to verify the server domain or IP address against the server certificate. This option is enabled by default and it is recommended to leave it enabled for a secure configuration.

When specifying a secure connection, there are some considerations for the certificate used by LDAP to secure the connection. The FortiGate checks the certificate presented by the LDAP server for the IP address or FQDN as specified in the *Server IP/Name* field with the following logic:

- If there is a Subject Alternative Name (SAN), it will ignore any Common Name (CN) value and look for a match in any of the SAN fields.
- If there is no SAN, it will check the CN for a match.

4. Optionally, click *Test User Credentials* to ensure that the account has sufficient access rights.
5. Click *OK*.
   The FortiGate checks the connection and updates the *Connection Status*.

Enabling Active Directory recursive search

By default, nested groups (groups that are members or other groups) are not searched in Windows Active Directory (AD) LDAP servers because this can slow down the group membership search. There is an option in FortiOS to enable the searching of nested groups for user group memberships on AD LDAP servers.

This option is not available for other LDAP servers, such as OpenLDAP-based servers.

The default behavior does not include nested groups:

```
config user ldap
    edit "ldap-ad"
        set server "10.1.100.131"
        set cnid "cn"
        set dn "dc=fortinet-fsso,dc=com"
        set type regular
        set username "cn=Administrator,cn=users,dc=fortinet-fsso,dc=com"
        set password XXXXXXXXXXXXXXXXXXXXXXXX
    next
end
```

The default search results only show groups that have the user as member, and no groups that have groups as members:

```
diagnose test authserver ldap ldap-ad nuser nuser
    authenticate 'nuser' against 'ldap-ad' succeeded!
    Group membership(s) - CN=nested3,OU=Testing,DC=Fortinet-FSSO,DC=COM
                          CN=Domain Users,CN=Users,DC=Fortinet-FSSO,DC=COM
```

To enable recursive search to include nested groups in the results:

```
config user ldap
    edit "ldap-ad"
        set server "10.1.100.131"
        set cnid "cn"
        set dn "dc=fortinet-fsso,dc=com"
        set type regular
        set username "cn=Administrator,cn=users,dc=fortinet-fsso,dc=com"
        set password XXXXXXXXXXXXXXXXXXXXXXXX
        set search-type recursive
    next
end
```

The search results now include groups that have other groups as members:

```
diagnose test authserver ldap ldap-ad nuser nuser
    authenticate 'nuser' against 'ldap-ad' succeeded!
    Group membership(s) - CN=nested3,OU=Testing,DC=Fortinet-FSSO,DC=COM
                          CN=Domain Users,CN=Users,DC=Fortinet-FSSO,DC=COM
                          CN=nested2,OU=Testing,DC=Fortinet-FSSO,DC=COM
                          CN=nested1,OU=Testing,DC=Fortinet-FSSO,DC=COM
```

The group nested3 is a member of the group nested2, which is a member of the group nested1.

Configuring LDAP dial-in using a member attribute

In this configuration, users defined in Microsoft AD can set up a VPN connection based on an attribute that is set to `TRUE`, instead of their user group. You can activate the *Allow Dialin* property in AD user properties, which sets the `msNPAllowDialin` attribute to *TRUE*. You can use this procedure for other member attributes as your system requires.

This configuration consists of the following steps:

1. Ensure that the AD server has the `msNPAllowDialin` attribute set to *TRUE* for the desired users.
2. Configure user LDAP member attribute settings.
3. Configure LDAP group settings.
4. Ensure that you configured the settings correctly.

To configure user LDAP member attribute settings:

```
config user ldap
   edit "ldap_server"
      set server "192.168.201.3"
      set cnid "sAMAccountName"
      set dn "DC=fortilabanz,DC=com,DC=au"
      set type regular
      set username "fortigate@sample.com"
      set password ******
      set member-attr "msNPAllowDialin"
   next
end
```

To configure LDAP group settings:

```
config user group
   edit "ldap_grp"
      set member "ldap_server"
      config match
         edit 1
            set server-name "ldap_server"
            set group-name "TRUE"
         next
      end
   next
end
```

To ensure that you configured the settings correctly:

Users that are members of the `ldap_grp` user group should be able to authenticate. The following shows sample `diagnose debug` output when the `Allow Dial-in` attribute is set to `TRUE`:

```
get_member_of_groups-Get the memberOf groups.
get_member_of_groups- attr='msNPAllowDialin', found 1 values
get_member_of_groups-val[0]='TRUE'
fnbamd_ldap_get_result-Auth accepted
fnbamd_ldap_get_result-Going to DONE state res=0
fnbamd_auth_poll_ldap-Result for ldap svr 192.168.201.3 is SUCCESS
fnbamd_auth_poll_ldap-Passed group matching
```

If the attribute is not set to `TRUE` but is expected, you may see the following output:

```
get_member_of_groups-Get the memberOf groups.
get_member_of_groups- attr='msNPAllowDialin', found 1 values
get_member_of_groups-val[0]='FALSE'
fnbamd_ldap_get_result-Auth accepted
fnbamd_ldap_get_result-Going to DONE state res=0
fnbamd_auth_poll_ldap-Result for ldap svr 192.168.201.3 is SUCCESS
fnbamd_auth_poll_ldap-Failed group matching
```

The difference between the two outputs is the last line, which shows passed or failed depending on whether the member attribute is set to the expected value.

Configuring wildcard admin accounts

To avoid setting up individual admin accounts in FortiOS, you can configure an admin account with the wildcard option enabled, allowing multiple remote admin accounts to match one local admin account. This way, multiple LDAP admin accounts can use one FortiOS admin account.

Benefits include:

- Fast configuration of the FortiOS admin account to work with your LDAP network, saving effort and avoiding potential errors incurred when setting up multiple admin accounts
- Reduced ongoing maintenance. As long as LDAP users belong to the same group and you do not modify the wildcard admin account in FortiOS, you do not need to configure changes on the LDAP accounts. If you add or remove a user from the LDAP group, you do not need to perform changes in FortiOS.

Potential issues include:

- Multiple users may be logged in to the same account simultaneously. This may cause issues if both users make changes simultaneously.
- Security is reduced since multiple users have login access to the same account, as opposed to an account for each user.

Wildcard admin configuration also applies to RADIUS. If configuring for RADIUS, configure the RADIUS server and RADIUS user group instead of LDAP. When using the GUI, wildcard admin is the only remote admin account that does not require you to enter a password on account creation. That password is normally used when the remote authentication server is unavailable during authentication.

This example uses default values where possible. If a specific value is not mentioned, the example sets it to its default value.

You can configure an admin account in Active Directory for LDAP authentication to allow an admin to perform lookups and reset passwords without being a member of the Account Operators or Domain Administrators built-in groups. See Configuring least privileges for LDAP admin account authentication in Active Directory on page 1808.

To configure the LDAP server:

The important parts of this configuration are the username and group lines. The username is the domain administrator account. The group binding allows only the GRP group access.

This example uses an example domain name. Configure as appropriate for your own network.

```
config user ldap
   edit "ldap_server"
      set server "192.168.201.3"
```

```
      set cnid "sAMAccountName"
      set dn "DC=example,DC=com,DC=au"
      set type regular
      set username "CN=Administrator,CN=Users,DC=example,DC=COM”
      set password *
      set group-member-check group-object
      set group-object-filter (&
           (objectcategory=group)member="CN=GRP,OU=training,DC=example,DC=COM"))
   next
end
```

To configure the user group and add the LDAP server:

```
config user group
   edit "ldap_grp"
      set member "ldap_server"
         config match
                edit 1
                   set server-name "ldap_server"
                   set group-name "CN=GRP,OU=training,DC=example,DC=COM"
                next
             end
           next
         end
      end
   end
end
```

To configure the wildcard admin account:

```
config system admin
   edit "test"
      set remote-auth enable
      set accprofile "super_admin"
      set wildcard enable
      set remote-group "ldap_grp"
   next
end
```

Configuring least privileges for LDAP admin account authentication in Active Directory

An administrator should only have sufficient privileges for their role. In the case of LDAP admin bind, you can configure an admin account in Active Directory for LDAP authentication to allow an admin to perform lookups and reset passwords without being a member of the Account Operators or Domain Administrators built-in groups.

For information about Active Directory, see the product documentation.

To configure account privileges for LDAP authentication in Active Directory:

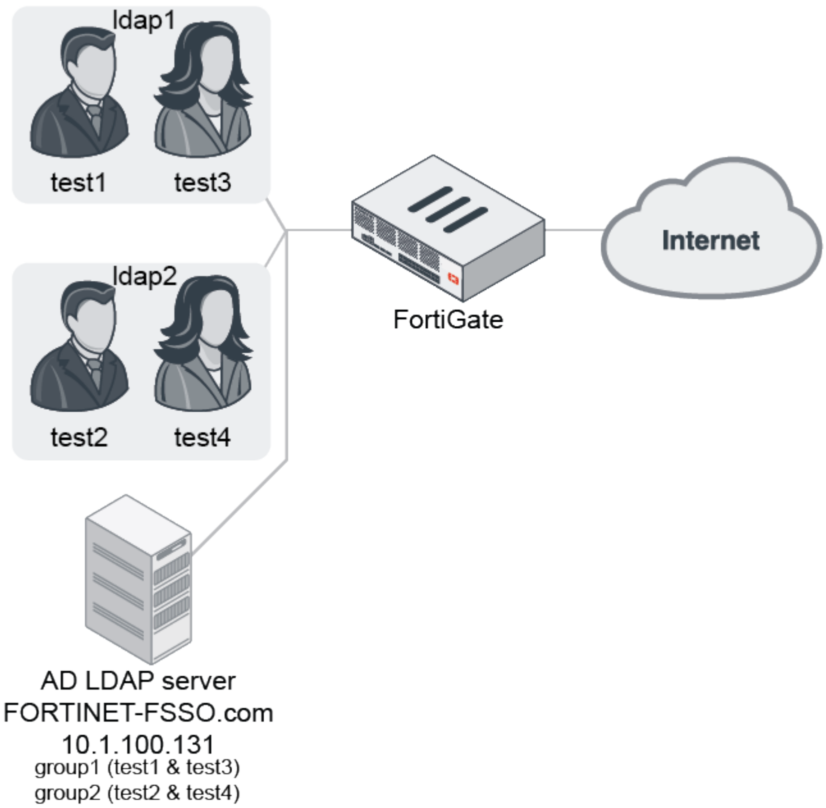
1. In the *Active Directory Users and Computers* administrative console, right-click the Organizational Unit (OU) or the top-level domain you want to configure and select *Delegate Control*.
2. In the *Delegation of Control Wizard* dialog, click *Next*.

3. In the *Users or Groups* dialog, click *Add...* and search Active Directory for the users or groups.
4. Click *OK* and then click *Next*.
5. In the *Tasks to Delegate* dialog, select *Create a custom task to delegate* and click *Next*.
6. Select *Only the following objects in the folder* and scroll to the bottom of the list. Select *User objects* and click *Next*.
7. In the *Permissions* dialog, select *General*.
8. From the *Permissions* list, select the following:
   - *Change password*
   - *Reset password*
9. Clear the *General* checkbox and select *Property-specific*.
10. From the *Permissions* list, select the following:
    - *Write lockoutTime*
    - *Read lockoutTime*
    - *Write pwdLastSet*
    - *Read pwdLastSet*
    - *Write UserAccountControl*
    - *Read UserAccountControl*
11. Click *Next* and click *Finish*.

Tracking users in each Active Directory LDAP group

When LDAP users log on through firewall authentication, the active users per Active Directory LDAP group is counted and displayed in the *Firewall Users* widget and the CLI.

Example



The Active Directory LDAP server, FORTINET-FSSO.com, is configured with two groups that contain two users each: group1 consists of users test1 and test3; group2 consists of users test2 and test4.

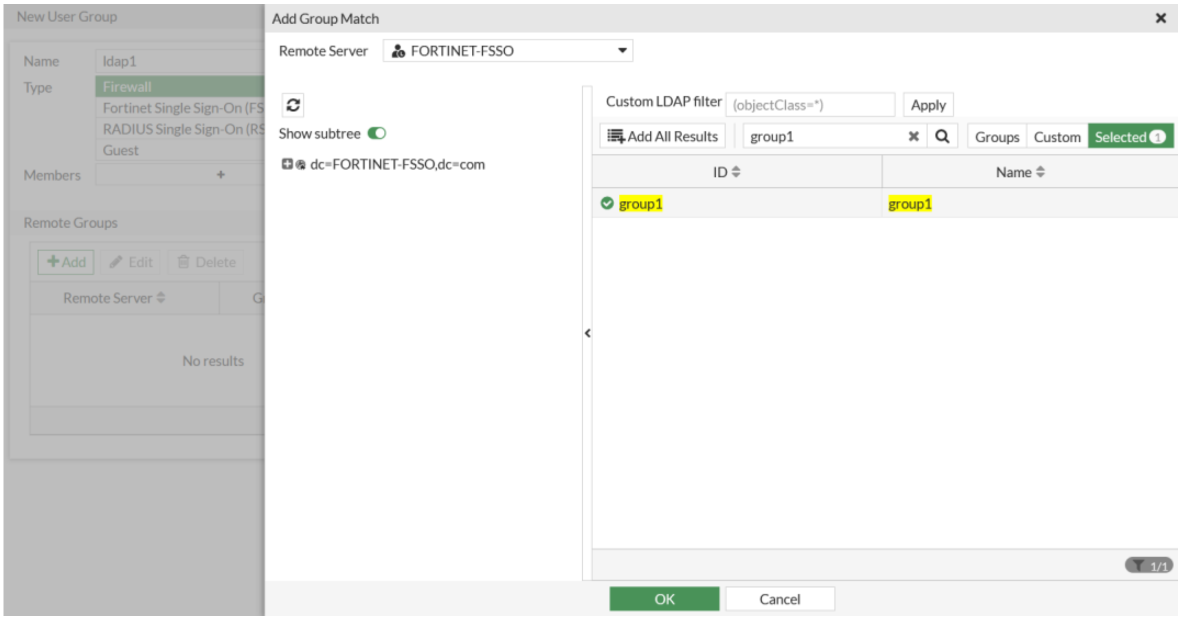
To configure AD LDAP user groups in the GUI:

1. Configure the Active Directory LDAP server, FORTINET-FSSO:
   a. Go to *User & Authentication > LDAP Servers* and click *Create New*.
   b. Enter the following:

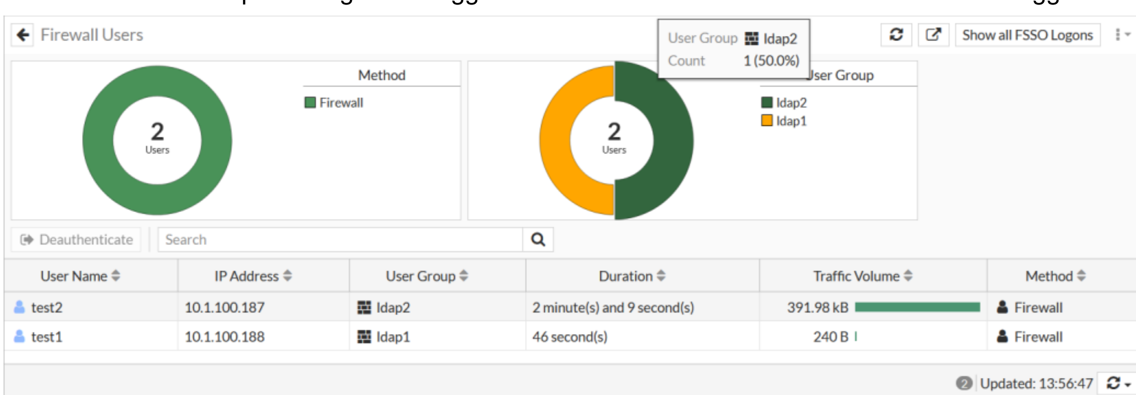
Name	*FORTINET-FSSO*
Server IP/Name	*10.1.100.131*
Distinguished Name	*dc=FORTINET-FSSO,dc=com*
Bind Type	*Regular*
Username	*cn=administrator,cn=users,dc=FORTINET-FSSO,dc=com*
Password	Enter the password.

   c. Click *OK*.
2. Configure the LDAP user groups:
   a. Go to *User & Authentication > User Groups* and click *Create New*.
   b. Enter the name, *ldap1*.

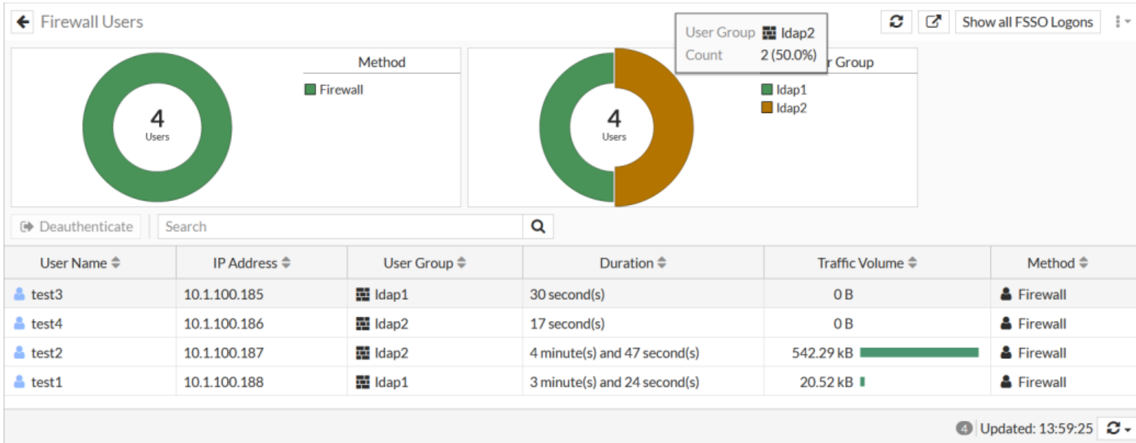
   c. In the *Remote Groups* table, click *Add*. The *Add Group Match* pane opens.
   d. For *Remote Server*, select *FORTINET-FSSO*.
   e. In the search box, enter *group1*, and select the result in the table.
   f. Click *OK*.



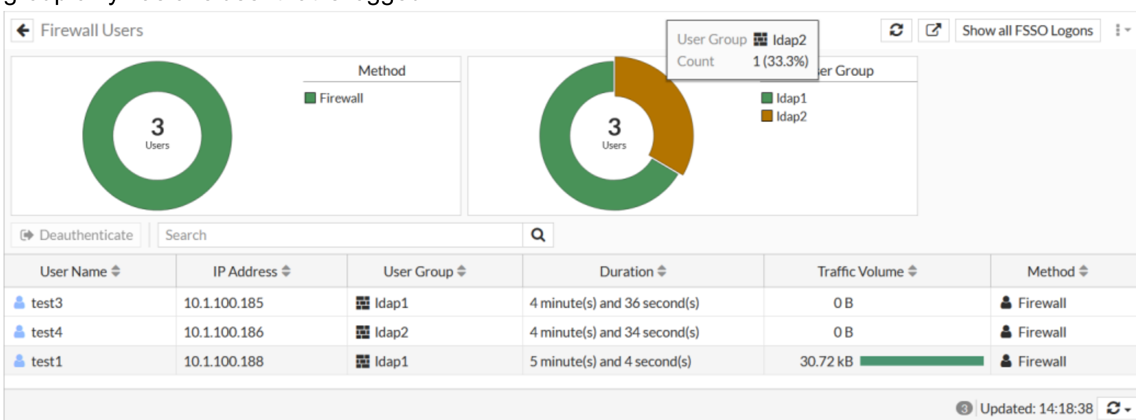
   g. Repeat these steps to configure *ldap2* with the FORTINET-FSSO *group2*.
   h. Click *OK*.
3. Configure a firewall policy with both LDAP groups:
   a. Go to *Policy & Objects > Firewall Policy* and click *Create New*.
   b. For *Source*, select *ldap1* and *ldap2*.
   c. Configure the other settings as needed.
   d. Click *OK*.
4. Get users test1 and test2 to log in.
5. In FortiOS, go to *Dashboard > Users & Devices* and click the *Firewall Users* widget to expand to full screen view. Hover over a group in the *User Group* donut chart to view how many users are logged on from that group, and the number of users as a percentage of all logged on users. The chart shows that two users are logged in.



6. Get users test3 and test4 to log in, and refresh the *Firewall Users* widget. Each LDAP group has two users logged in, with a total of four active users.



7. Get user test2 to log out, and refresh the *Firewall Users* widget. There is a total of three active users, and the ldap2 group only has one user that is logged in.



To verify the user group count in the CLI:

```
# diagnose user-device-store user-count list <integer>
# diagnose user-device-store user-count query <FQDN of AD group>
```

Tracking rolling historical records of LDAP user logins

Authenticated LDAP users can be tracked by logging the users' group memberships, logon timestamps, and logout timestamps into local files on a log disk over a rolling four-week period. The historical records can be queried from the CLI. This feature is only enabled on FortiGate models with a log disk.

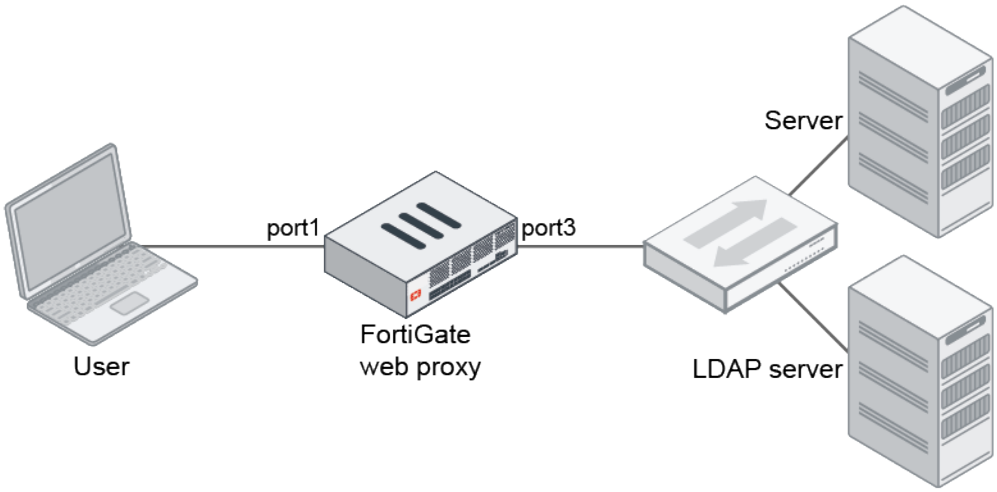
To view active user logged information:

```
# diagnose user-device-store user-stats query <yyyy-mm-dd> <range_in_days>
```

Example

In this example, the FortiGate is configured with an explicit web proxy and an LDAP server. When an LDAP user is authenticated by an IP-based authentication method in WAD, the WAD user is considered to be in an active logon status. This WAD user is listed in the `diagnose wad user list` output. If the user is removed from WAD as an authenticated, such as when the IP-based authentication expires, then the user is considered to become inactive (logout status). The user is no longer listed in the `diagnose wad user list` output.

The WAD user's group membership information and their logon and logout timestamps are written into local files on the FortiGate's disk. There is one log file for each day, and the FortiGate can maintain up to 28 log files over a rolling period of 28 days (four weeks). This means after 28 days with 28 files stored, on the 29th day, the first file will be removed and a new file will be created for the 29th day.



This feature works on other configurations such as firewall authentication, transparent web proxy, ZTNA, and SSL VPN where an LDAP server is used.

To configure the FortiGate:

1. Enable the explicit web proxy on port1:

```
config system interface
    edit "port1"
        set explicit-web-proxy enable
        set explicit-ftp-proxy enable
        set snmp-index 3
    next
end
```

2. Configure the LDAP server:

```
config user ldap
    edit "ldap-test"
        set server "172.16.200.98"
        set cnid "cn"
        set dn "dc=fortinetqa,dc=local"
        set type regular
        set username "CN=root,CN=Users,DC=fortinetqa,DC=local"
        set password **********
    next
end
```

3. Configure the authentication scheme:

```
config authentication scheme
    edit "basic-ldap"
        set method basic
        set user-database "ldap-test"
    next
end
```

4. Configure the authentication rule:

```
config authentication rule
    edit "basic-ldap"
        set srcaddr "all"
        set active-auth-method "basic-ldap"
        set web-portal disable
    next
end
```

5. Configure the user group:

```
config user group
    edit "ldap-group"
        set member "ldap" "ldap-test"
    next
end
```

6. Configure the proxy policy:

```
config firewall proxy-policy
    edit 1
        set proxy explicit-web
        set dstintf "port3"
        set srcaddr "all"
        set dstaddr "all"
        set service "web"
        set action accept
        set schedule "always"
        set groups "ldap-group"
        set utm-status enable
        set ssl-ssh-profile "deep-custom"
        set av-profile "av"
    next
end
```

   When users pass through the explicit proxy and log in and out through LDAP, their login and logout records will be logged to the disk.

In this example, there are two LDAP users, test1 and test3, with the following activity:

1. test3 logs on at 22:30:22 on February 23, 2022, then logs out at 22:31:09 on the same day.
2. test1 logs on at 23:55:02 on February 23, 2022, then logs out at 00:05:02 on February 24, 2022.
3. test3 logs on at 16:29:44 on February 24, 2022, then logs out at 16:39:44 on the same day.

The logon and logout timestamp information, and the group membership information for users test1 and test3 will be logged into two local files on the log disk.

To view the active user logged information for two days back from February 24, 2022:

```
# diagnose user-device-store user-stats query 2022-02-24 2

Record #0:
        'username' = 'test3'
        'groupname' = 'CN=Domain Admins,CN=Users,DC=FORTINETQA,DC=local'
        'groupname' = 'CN=FSSO,OU=QA,DC=FORTINETQA,DC=local'
        'logon' = '2022-02-23 22:30:22'
        'logout' = '2022-02-23 22:31:09'

Record #1:
        'username' = 'test1'
        'groupname' = 'CN=Domain Admins,CN=Users,DC=FORTINETQA,DC=local'
        'groupname' = 'CN=FSSO,OU=QA,DC=FORTINETQA,DC=local'
        'groupname' = 'CN=mytest-grp,OU=QA,DC=FORTINETQA,DC=local'
        'logon' = '2022-02-23 23:55:02'

Record #2:
        'username' = 'test1'
        'groupname' = 'CN=Domain Admins,CN=Users,DC=FORTINETQA,DC=local'
        'groupname' = 'CN=FSSO,OU=QA,DC=FORTINETQA,DC=local'
        'groupname' = 'CN=mytest-grp,OU=QA,DC=FORTINETQA,DC=local'
        'logon' = '2022-02-23 23:55:02'
        'logout' = '2022-02-24 00:05:02'

Record #3:
        'username' = 'test3'
        'groupname' = 'CN=Domain Admins,CN=Users,DC=FORTINETQA,DC=local'
        'groupname' = 'CN=FSSO,OU=QA,DC=FORTINETQA,DC=local'
        'logon' = '2022-02-24 16:29:44'
        'logout' = '2022-02-24 16:39:44'

Returned 4 records.
```

There is one record (`logon`) for test1 on `2022-02-23` because they remained active after midnight (until 00:05:02). There is another record for `2022-02-24` with logon and logout timestamps for test1.

Configuring client certificate authentication on the LDAP server

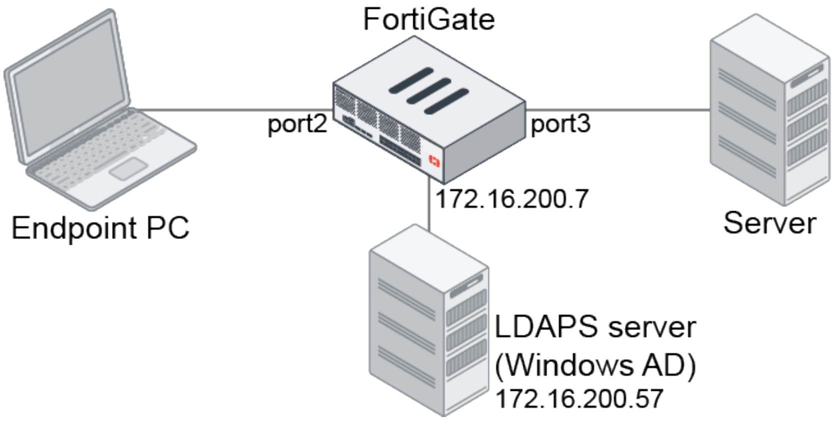
Administrators can configure a FortiGate client certificate in the LDAP server configuration when the FortiGate connects to an LDAPS server that requires client certificate authentication.

```
config user ldap
    edit <ldap_server>
        set client-cert-auth {enable | disable}
        set client-cert <source>
    next
end
```

Example

In this example, the FortiGate is configured as an explicit web proxy. It connects to the Windows AD server through LDAPS, where the Windows server requires a client certificate to connect. The client certificate is configured in the CLI.

The endpoint PC connecting to the web server will first need to authenticate to the explicit web proxy before accessing the server.



While this example demonstrates an LDAP client certificate for an explicit proxy configuration, LDAP client certificates can be used in firewall authentication, transparent proxy, ZTNA, and where ever LDAP configurations are used on the FortiGate.

To configure a client certificate on the LDAP server:

1. Enable the explicit web proxy on port2:

```
config system interface
    edit "port2"
        set explicit-web-proxy enable
    next
end
```

2. Upload the client certificate to the FortiGate:

```
config vpn certificate local
    edit "Zach"
        set password **********
        set private-key <private key>
        set certificate <certificate>
    next
end
```

3. Configure the LDAP server settings:

```
config user ldap
    edit "ldaps"
        set server "172.16.200.57"
        set server-identity-check disable
        set cnid "CN"
        set dn "CN=Users,DC=ftnt,DC=com"
        set secure ldaps
        set port 636
        set client-cert-auth enable
        set client-cert "Zach"
    next
end
```

4. Configure the authentication scheme:

```
config authentication scheme
    edit "1"
        set method basic
        set user-database "ldaps"
    next
end
```

5. Configure the authentication rule:

```
config authentication rule
    edit "1"
        set srcintf "port2"
        set srcaddr "all"
        set dstaddr "all"
        set active-auth-method "1"
    next
end
```

6. Configure the user group:

```
config user group
    edit "test"
        set member "ldaps"
    next
end
```

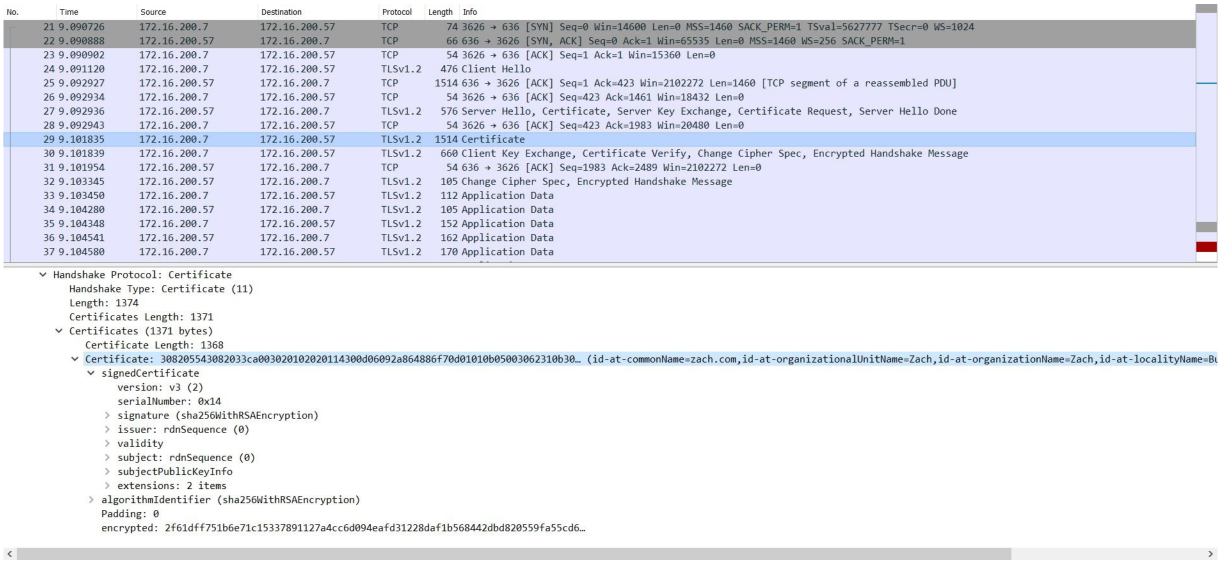
7. Configure the proxy policy with the user group:

```
config firewall proxy-policy
    edit 1
        set proxy explicit-web
        set dstintf "port3"
        set srcaddr "all"
        set dstaddr "all"
        set service "webproxy"
        set action accept
        set schedule "always"
        set srcaddr6 "all"
        set dstaddr6 "all"
        set groups "test"
        set utm-status enable
        set ssl-ssh-profile "deep-inspection-clone"
        set av-profile "av"
    next
end
```

Testing and verification

When traffic from the endpoint PC matches a policy and triggers authentication, the FortiGate starts the LDAPS TLS connection handshake with the Windows AD. The LDAPS server requests a client certificate to identify the FortiGate as a client. The FortiGate provides a configured client certificate, issued to zach.com, to the LDAPS server.

The following communication between the FortiGate and the LDAPS server shows the client certificate is sent by the FortiGate:



RADIUS servers

Remote Authentication and Dial-In User Service (RADIUS) is a broadly supported client-server protocol that provides centralized authentication, authorization, and accounting functions. RADIUS clients are built into gateways that allow access to networks such a VPN server, network access server (NAS), and a network switch or firewall that uses authentication.

RADIUS servers use UDP packets to communicate with the RADIUS clients on the network to authenticate users before allowing them access to the network, authorize access to resources by appropriate users, and account or bill for those resources that are used. RADIUS servers are currently defined by RFC 2865 (RADIUS) and RFC 2866 (RADIUS Accounting), and listen on either UDP ports 1812 (authentication) and 1813 (accounting), or ports 1645 (authentication) and 1646 (accounting) requests. RADIUS servers exist for all major operating systems.

The RADIUS server must be configured to accept the FortiGate as a client so is can use the authentication and accounting functions of the RADIUS server.

RADIUS authentication with a FortiGate requires the following:

- Configuring one or more RADIUS server profiles on the FortiGate.
- Assigning the RADIUS server profile to a user or user group.
- Applying the user or user group to a firewall policy.

RADIUS authentication can be applied to many FortiGate functions, such as firewall authentication, SSL and IPsec VPNs, administrator profiles, ZTNA, explicit proxy, wireless, 802.1X, and more.

The RADIUS server uses a shared secret key with MD5 hashing to encrypt information passed between RADIUS servers and clients. Typically, only user credentials are encrypted. Additional security can be configured through IPsec tunnels by placing the RADIUS server behind another VPN gateway.

The following topics provide more information about RADIUS servers:

- Configuring a RADIUS server on page 1819
- Using multiple RADIUS servers on page 1820
- RADIUS AVPs and VSAs on page 1823

- Restricting RADIUS user groups to match selective users on the RADIUS server on page 1825
- Configuring RADIUS SSO authentication on page 1826
- RSA ACE (SecurID) servers on page 1832
- Support for Okta RADIUS attributes filter-Id and class on page 1836
- Sending multiple RADIUS attribute values in a single RADIUS Access-Request on page 1838
- Traffic shaping based on dynamic RADIUS VSAs on page 1839
- RADIUS Termination-Action AVP in wired and wireless scenarios on page 1846

Configuring a RADIUS server

A RADIUS server can be configured in the GUI by going to *User & Authentication > RADIUS Servers*, or in the CLI under `config user radius`.

Basic configuration

The following table summarizes the common RADIUS settings that can be configured in the GUI and CLI.

GUI field	CLI setting	Description
Name	`edit <name>`	Define the RADIUS server object within FortiOS.
Authentication method	`set auth-type {auto | ms_chap_v2 | ms_chap | chap | pap}`	Specify the authentication method, or select *Default*/`auto` to negotiate PAP, MSCHAP_v2, and CHAP in that order.
NAS IP	`set nas-ip <IPv4_address>`	Optional setting, also known as Calling-Station-Id. Specify the IP address the FortiGate uses to communicate with the RADIUS server. If left unconfigured, the FortiGate will use the IP address of the interface that communicates with the RADIUS server.
Include in every user group	`set all-usergroup {enable | disable}`	Optional setting to add the RADIUS server to each user group. This allows each user group to try and authenticate users against the RADIUS server if local authentication fails.
Primary Server		
IP/Name	`set server <string>`	Enter the IP address or resolvable FQDN of the RADIUS server.
Secret	`set secret <password>`	Enter the password used to connect to the RADIUS server.

There is an option in the GUI to configure a second server, and a third server can be configured in the CLI (see Using multiple RADIUS servers on page 1820).

Advanced settings

Advanced settings for RADIUS servers can be configured in the CLI. The following are some commonly used settings.

To edit the port used to connect with the RADIUS server:

```
config system global
    set radius-port <integer>
end
```

To edit the default setting for password encoding and username case sensitivity:

```
config user radius
    edit <name>
        set password-encoding {auto | ISO-8859-1}
        set username-case-sensitive {enable | disable}
    next
end
```

`password-encoding {auto	ISO-8859-1}`	Set the password encoding to use the original encoding or ISO-8859-1 (default = auto). The `auth-type` must be `auto` or `pap` to change this setting.
`username-case-sensitive {enable	disable}`	Enable/disable case sensitive usernames (default = disable).

Using multiple RADIUS servers

There are several ways to implement multiple RADIUS servers, and each has a different effect on user authentication. The three main options available are:

- Add a second (or third) RADIUS server in the same profile.
- Add a second RADIUS server profile, and add both to the same user group.
- Use two RADIUS server profiles for two user groups (one for each).

Adding a second server in a RADIUS profile

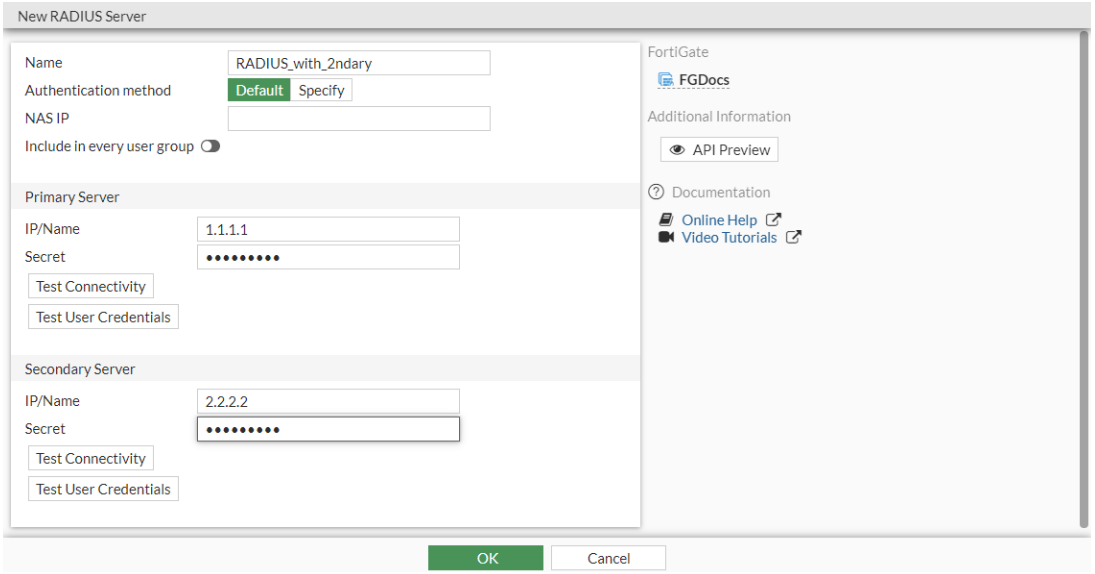
A second RADIUS server can be configured in the same RADIUS profile so in the event the first RADIUS server does not respond, the second server can be checked. If the first RADIUS server responds with an Access-Reject, no further servers are queried.

To add a second server in a RADIUS profile:

1. Go to *User & Authentication > RADIUS Servers* and click *Create New*.
2. Enter the following:

Name	*RADIUS_with_2ndary*
Authentication method	*Default*
Primary Server	
IP/Name	*1.1.1.1*
Secret	Enter the password used to connect to the RADIUS server.
Secondary Server	

IP/Name	*2.2.2.2*
Secret	Enter the password used to connect to the RADIUS server.



3. Click *OK*.

Adding two RADIUS server profiles in the same user group

When two separate RADIUS profiles are added to a user group, the FortiGate sends an Access-Request simultaneously to both RADIUS servers, and authentication succeeds if either server sends back an Access-Accept. This example includes the settings from the previous example where one or more of the RADIUS server profiles has a secondary server configured. In this case, the secondary server profile, *RADIUS_with_2ndary*, is only checked if the primary server of this profile times out and the *fac_radius_server* profile does not return an Access-Accept.

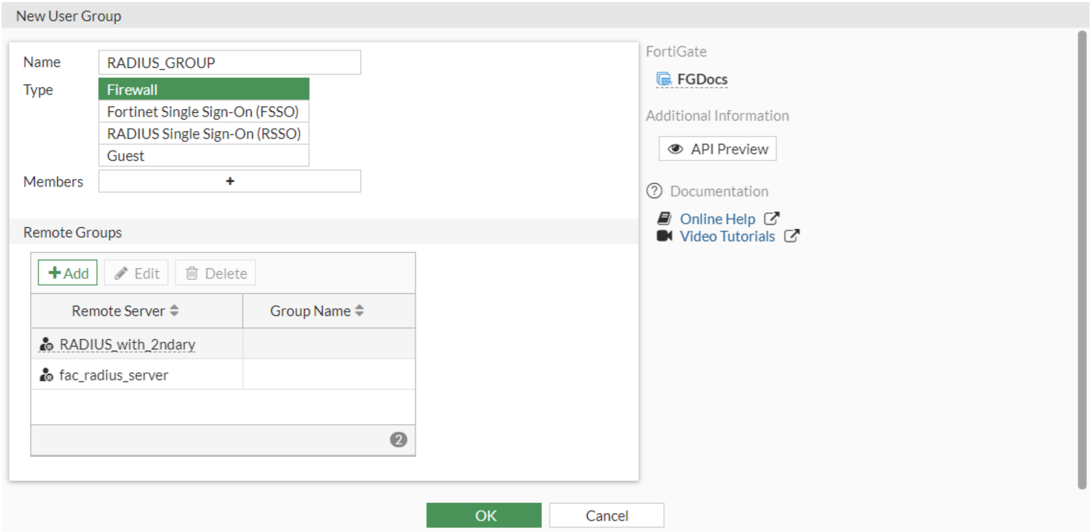
To add two RADIUS server profiles in the same user group:

1. Go to *User & Authentication > RADIUS Servers*, click *Create New*, and configure the RADIUS servers as needed (refer to the previous example).
2. Go to *User & Authentication > User Groups* and click *Create New*.
3. Enter the following:

Name	*RADIUS_GROUP*
Type	*Firewall*

4. In the *Remote Groups* table, click *Add*.
5. Select *RADIUS_with_2ndary* and click *OK*.

6. Click *Add*, select *fac_radius_server*, then click *OK*.



7. Click *OK*.

Using separate RADIUS server profiles for separate user groups

In this example, the FortiGate first evaluates if the user belongs to the first listed group (*radius_group*) in the policy. If the user fails to authenticate to this group, then the FortiGate checks if the user can successfully authenticate to the second user group (*radius_group_2*). Refer to the first and second examples for detailed instructions.

To use separate RADIUS server profiles for separate user groups:

1. Configure the RADIUS server profiles:
   a. Go to *User & Authentication > RADIUS Servers* and click *Create New*.
   b. Configure two RADIUS servers, *fac_radius_server* and *RADIUS_with_2ndary*, as needed (refer to the previous example).
2. Configure the firewall groups:
   a. Go to *User & Authentication > User Groups* and click *Create New*.
   b. Configure two firewall groups, one named *radius_group* with remote server member *fac_radius_server*, and one named *radius_group_2* with remote server member *RADIUS_with_2ndary* (refer to the previous example).

Group Name	Group Type	Members	Ref.
radius_group	Firewall	fac_radius_server	1
radius_group_2	Firewall	RADIUS_with_2ndary	1

3. Configure the firewall policy:
   a. Go to *Policy & Objects > Firewall Policy* and click *Create New*.
   b. For *Source*, click *User* then select *radius_group* and *radius_group_2*. Click *Address* and select *LAN address*.
   c. Configure the other settings as needed.
   d. Click *OK*.

Name	From	To	Source	Destination	Schedule	Service	Action	NAT
LAN to WAN	LAN (LAN)	WAN	radius_group radius_group_2 LAN address	all	always	ALL	ACCEPT	Enabled

RADIUS AVPs and VSAs

This topic describes RADIUS Attribute Value Pairs (AVPs) and Vendor-Specific Attributes (VSAs).

AVPs

RADIUS packets include a set of AVPs to identify information about the user, their location, and other information. The IETF defined a set of 255 standard attributes, which are well known and come in the form of Type, Length, Value (for more details, refer to RFC 2865). Of the standard 255, the FortiGate sends the following RADIUS attributes:

RADIUS attribute number	Name	Description
1	User-Name	Name of the user being authenticated by the RADIUS server.
4	NAS-IP-Address	IP address of the network access server (NAS) that is requesting authentication. The NAS is the FortiGate.
8	Framed-IP-Address	IP address to be configured for the user, by sending the IP address of a user to the RADIUS server in the Access-Request packet.
25	Class	Used in accounting packets and requests for firewall, WiFi, and proxy authentication. The attribute is returned in the Access-Accept message and is added to all accounting packets.
26	Fortinet-VSA	See VSAs.
32	NAS-Identifier	Identifier or IP address of the NAS that is requesting authentication. The NAS is the FortiGate.
42	Acct-Input-Octets	Number of octets received from the port over the course of this service being provided. Used to charge the user for the amount of traffic they used.
43	Acct-Output-Octets	Number of octets sent to the port while delivering this service. Used to charge the user for the amount of traffic they used.
44	Acct-Session-Id	Unique number assigned to each start and stop record to make it easy to match them, and to eliminate duplicate records.
55	Event-Timestamp	Records the time that the event occurred on the NAS. The timestamp is measured in seconds since January 1, 1970 00:00 UTC. Before the Event-Timestamp attribute can be sent in a packet, make sure that the correct time is set on the FortiGate.

VSAs

Some vendors want or need to send attributes that do not match any of the defined IETF attributes. This can be accomplished by using RADIUS attribute type 26, which allows a vendor to encapsulate their own specific attributes in this standard AVP.

In order to support VSAs, the RADIUS server requires a dictionary to define the VSAs. This dictionary is typically supplied by the client or server vendor.

The Fortinet RADIUS vendor ID is 12356 and contains the following attributes:

Attribute name	Attribute number	Attribute value format
Fortinet-Group-Name	1	String
Fortinet-Client-IP-Address	2	IP address
Fortinet-Vdom-Name*	3	String
Fortinet-Client-IPv6-Address	4	Octets
Fortinet-Interface-Name	5	String
Fortinet-Access-Profile	6	String
Fortinet-SSID	7	String
Fortinet-AP-Name	8	String
Fortinet-FAC-Auth-Status	11	String
Fortinet-FAC-Token-ID	12	String
Fortinet-FAC-Challenge-Code	15	String
Fortinet-Webfilter-Category-Allow	16	String
Fortinet-Webfilter-Category-Block	17	Octets
Fortinet-Webfilter-Category-Monitor	18	Octets
Fortinet-AppCtrl-Category-Allow	19	Octets
Fortinet-AppCtrl-Category-Block	20	Octets
Fortinet-AppCtrl-Risk-Allow	21	Octets
Fortinet-AppCtrl-Risk-Block	22	Octets
Fortinet-WirelessController-Device-MAC	23	Ether
Fortinet-WirelessController-WTP-ID	24	String
Fortinet-WirelessController-Assoc-Time	25	Date
Fortinet-FortiWAN-AVPair	26	String
Fortinet-FDD-Access-Profile	30	String
Fortinet-FDD-Trusted-Hosts	31	String
Fortinet-FDD-SPP-Name	32	String
Fortinet-FDD-Is-System-Admin	33	String
Fortinet-FDD-Is-SPP-Admin	34	String
Fortinet-FDD-SPP-Policy-Group	35	String
Fortinet-FDD-Allow-API-Access	36	String
Fortinet-Fpc-User-Role	40	String

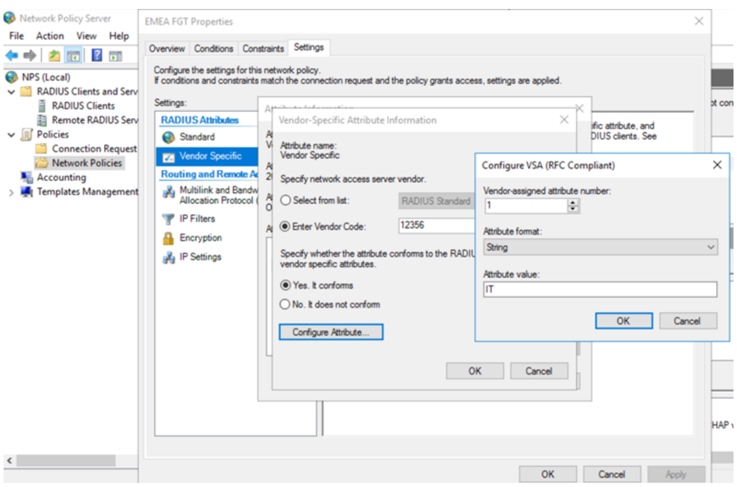
Attribute name	Attribute number	Attribute value format
Fortinet-Tenant-Identification	41	String
Fortinet-Host-Port-AVPair	42	String

[*] For Fortinet-Vdom-Name, users can be tied to a specific VDOM on the FortiGate. Refer to the documentation provided by your RADIUS server for configuration details.

Restricting RADIUS user groups to match selective users on the RADIUS server

When a user group is configured in FortiOS to authenticate against a RADIUS server, it will allow any valid user account on the RADIUS server to match that user group. Sometimes you might want to specify which users on the RADIUS server should match a particular user group on the FortiGate. This can be accomplished using the RADIUS attribute value pair (AVP) 26, known as a Vendor-Specific Attribute (VSA). This attribute allows the Fortinet-Group-Name VSA to be included in the RADIUS response. In FortiOS, the user group must be configured to specifically match this group.

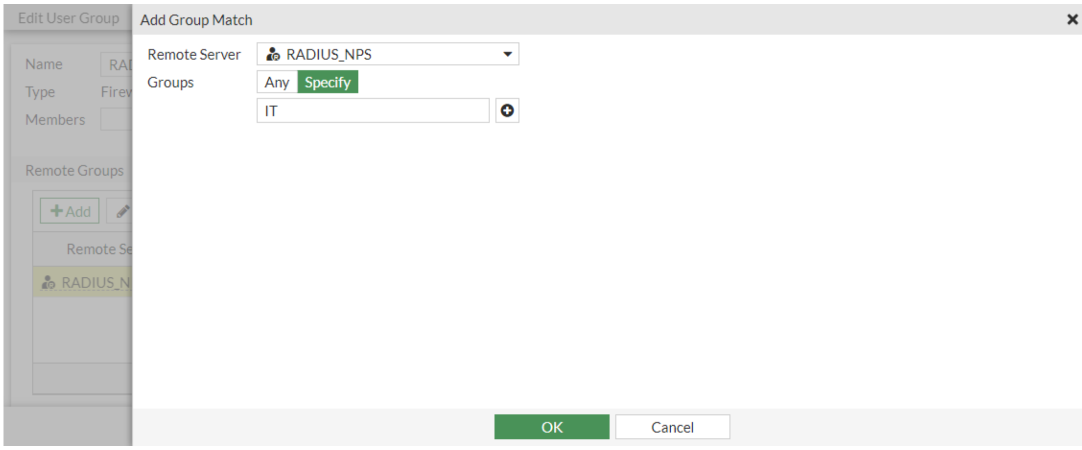
In the following example, a RADIUS Network Policy Server (NPS) has been configured to have the Fortinet-Group-Name be *IT*, and assumes that the user group, *RADIUS_IT* has been created, which authenticates to the *RADIUS_NPS* server.



To configure specific group matching in the GUI:

1. Go to *User & Authentication > User Groups* and edit the *RADIUS_IT* group.
2. In the *Remote Groups* table, select the *RADIUS_NPS* server and click *Edit*. The *Add Group Match* pane opens.
3. For *Groups*, select *Specify* and enter the group name configured on the RADIUS server (*IT*).

4. Click *OK*.



5. Click *OK*.

To configure specific group matching in the CLI:

```
config user group
    edit "RADIUS_IT"
        set member "RADIUS_NPS"
        config match
            edit 1
                set server-name "RADIUS_NPS"
                set group-name "IT"
            next
        end
    next
end
```

To change the matching back to any group, under `config match`, enter `delete 1`. Changing the `group-name` to `"Any"` will cause the FortiGate to match the Fortinet-Group-Name with the literal string, `Any`.

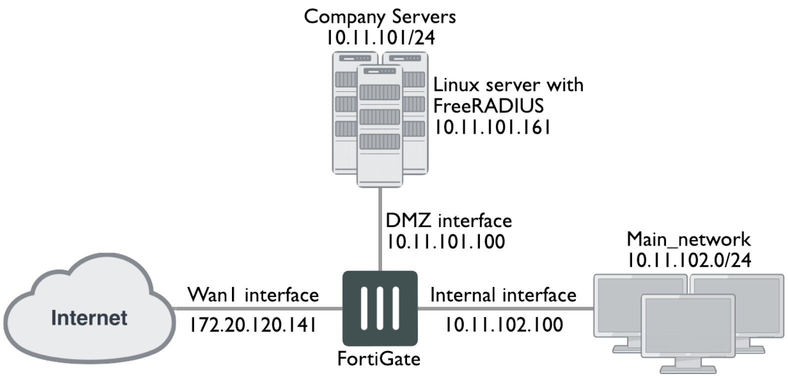
Configuring RADIUS SSO authentication

A common RADIUS SSO (RSSO) topology involves a medium-sized company network of users connecting to the Internet through the FortiGate and authenticating with a RADIUS server. The following describes how to configure FortiOS for this scenario. The example makes the following assumptions:

- VDOMs are not enabled.
- The super_admin account is used for all FortiGate configuration.
- A RADIUS server is installed on a server or FortiAuthenticator and uses default attributes.
- BGP is used for any dynamic routing.
- You have configured authentication event logging under *Log & Report*.

Example.com has an office with 20 users on the internal network who need access to the Internet. The office network is protected by a FortiGate-60C with access to the Internet through the wan1 interface, the user network on the internal interface, and all servers are on the DMZ interface. This includes an Ubuntu sever running FreeRADIUS. This example configures two users:

User	Account
Pat Lee	plee@example.com
Kelly Green	kgreen@example.com



Configuring this example consists of the following steps:

1. Configure RADIUS.
2. Configure FortiGate interfaces.
3. Configure a RSSO agent.
4. Create a RSSO user group.
5. Configure security policies.
6. Test the configuration.

To configure RADIUS:

Configuring RADIUS includes configuring a RADIUS server such as FreeRADIUS on user's computers and configuring users in the system. In this example, Pat and Kelly belong to the exampledotcom_employees group. After completing the configuration, you must start the RADIUS daemon. The users have a RADIUS client installed on their PCs that allow them to authenticate through the RADIUS server.

For any problems installing FreeRADIUS, see the FreeRADIUS documentation.

To configure FortiGate interfaces:

You must define a DHCP server for the internal network, as this network type typically uses DHCP. The wan1 and dmz interfaces are assigned static IP addresses and do not need a DHCP server. The following table shows the FortiGate interfaces used in this example:

Interface	Subnet	Act as DHCP server	Devices
wan1	172.20.120.141	No	Internet service provider
dmz	10.11.101.100	No	Servers including RADIUS server
internal	10.11.102.100	Yes: x.x.x.110-250	Internal user network

1. Go to *Network* > *Interfaces*.
2. Edit wan1:

Alias	Internet
Addressing Mode	Manual
IP/Network Mask	172.20.120.141/255.255.255.0
Administrative Access	HTTPS, SSH
Enable DHCP Server	Not selected
Comments	Internet
Administrative Status	Up

3. Click *OK*.
4. Edit dmz:

Alias	Servers
Addressing Mode	Manual
IP/Network Mask	10.11.101.100/255.255.255.0
Administrative Access	HTTPS, SSH, PING, SNMP
Enable DHCP Server	Not selected
Listen for RADIUS Accounting Messages	Select
Comments	Servers
Administrative Status	Up

5. Click *OK*.
6. Edit internal:

Alias	Internal network
Addressing Mode	Manual
IP/Network Mask	10.11.102.100/255.255.255.0
Administrative Access	HTTPS, SSH, PING
Enable DHCP Server	Select
Address Range	10.11.102.110 - 10.11.102.250
Netmask	255.255.255.0
Default Gateway	Same as Interface IP
Comments	Internal network
Administrative Status	Up

To create a RADIUS SSO agent:

1. Go to *Security Fabric > External Connectors*.
2. Click *Create New*.
3. Under *Endpoint/Identity*, select *RADIUS Single Sign-On Agent*.
4. Enable *Use RADIUS Shared Secret*. Enter the RADIUS server's shared secret.
5. Enable *Send RADIUS Responses*. Click *OK*.

To create a RADIUS SSO user group:

1. Go to *User & Authentication > User Groups*.
2. Click *Create New*.
3. For *Type*, select *RADIUS Single Sign-On (RSSO)*.
4. In *RADIUS Attribute Value*, enter the name of the RADIUS user group that this local user group represents.
5. Click *OK*.

Configuring security policies

The following security policies are required for RADIUS SSO:

Sequence Number	From	To	Type	Schedule	Description
1	internal	wan1	RADIUS SSO	Business hours	Authenticate outgoing user traffic
2	internal	wan1	Regular	Always	Allow essential network services and VoIP
3	dmz	wan1	Regular	Always	Allow servers to access the Internet
4	internal	dmz	Regular	Always	Allow users to access servers
5	any	any	Deny	Always	Implicit policy denying all traffic that has not been matched

You must place the RADIUS SSO policy at the top of the policy list so that it is matched first. The only exception to this is if you have a policy to deny access to a list of banned users. In this case, you must put that policy at the top so that the RADIUS SSO does not mistakenly match a banned user or IP address.

You must configure lists before creating security policies.

Schedule

You must configure a business_hours schedule. You can configure a standard Monday to Friday 8 AM to 5 PM schedule, or whatever days and hours covers standard work hours at the company.

Address groups

You must configure the following address groups:

Name	Interface	Address range included
internal_network	internal	10.11.102.110 to 10.11.102.250
company_servers	dmz	10.11.101.110 to 10.11.101.250

Service groups

You must configure the service groups. The services listed are suggestions and you may include more or less as required:

Name	Interface	Description of services to be included
essential_network_services	internal	Any network protocols required for normal network operation such as DNS, NTP, BGP
essential_server_services	dmz	All the protocols required by the company servers such as BGP, HTTP, HTTPS, FTP, IMAP, POP3, SMTP, IKE, SQL, MYSQL, NTP, TRACEROUTE, SOCKs, and SNMP
user_services	internal	Any protocols required by users such as HTTP, HTTPS, FTP

The following security policy configurations are basic and only include logging and default AV and IPS. These policies allow or deny access to non-RADIUS SSO traffic. These are essential as network services including DNS, NTP, and FortiGuard require access to the Internet.

To configure security policies:

1. Go to *Policy & Objects > Firewall Policy*.
2. Click *Create New*.
3. Configure the policy as follows, then click *OK*:

Incoming Interface	Internal
Source Address	internal_network
Outgoing Interface	wan1
Destination Address	all
Schedule	always
Service	essential_network_services
Action	ACCEPT
NAT	ON
Security Profiles	ON: AntiVirus, IPS
Log Allowed Traffic	ON

Comments	Essential network services

4. Click *Create New*, and configure the new policy as follows, then click *OK*:

Incoming Interface	dmz
Source Address	company_servers
Outgoing Interface	wan1
Destination Address	all
Schedule	always
Service	essential_server_services
Action	ACCEPT
NAT	ON
Security Profiles	ON: AntiVirus, IPS
Log Allowed Traffic	enable
Comments	Company servers accessing the Internet

5. Click *Create New*, and configure the new policy as follows, then click *OK*:

Incoming Interface	Internal
Source Address	internal_network
Outgoing Interface	dmz
Destination Address	company_servers
Schedule	always
Service	all
Action	ACCEPT
NAT	ON
Security Profiles	ON: AntiVirus, IPS
Log Allowed Traffic	enable
Comments	Access company servers

6. Click *Create New*, and configure the RADIUS SSO policy as follows, then click *OK*. This policy allows access for members of specific RADIUS groups.

Incoming Interface	Internal
Source Address	internal_network
Source User(s)	Select the user groups that you created for RSSO.
Outgoing Interface	wan1

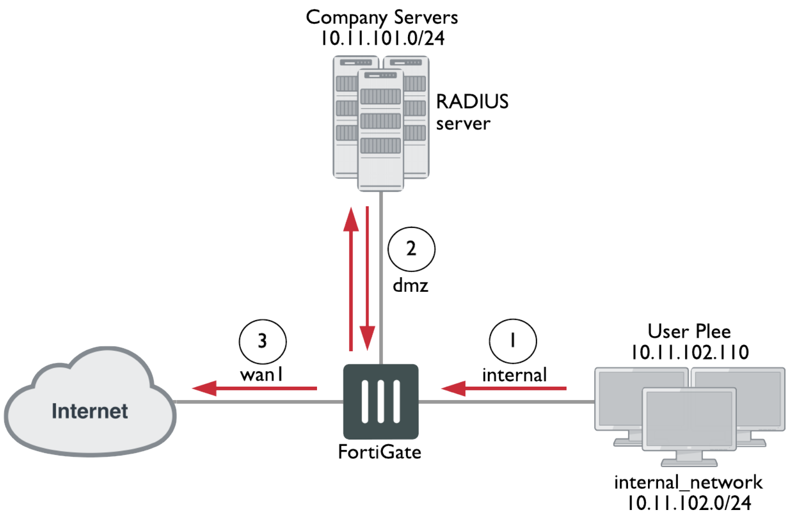
Destination Address	all
Schedule	business_hours
Service	ALL
Action	ACCEPT
NAT	ON
Security Profiles	ON: AntiVirus, Web Filter, IPS, and Email Filter. In each case, select the default profile.

7. Place the RSSO policy higher in the security policy list than more general policies for the same interfaces. Click *OK*.

To test the configuration:

Once configured, a user only needs to log in to their PC using their RADIUS account. After that, when they attempt to access the Internet, the FortiGate uses their session information to get their RADIUS information. Once the user is verified, they can access the website.

1. The user logs on to their PC and tries to access the Internet.
2. The FortiGate contacts the RADIUS server for the user's information. Once confirmed, the user can access the Internet. Each step generates logs that enable you to verify that each step succeeded.
3. If a step does not succeed, confirm that your configuration is correct.



RSA ACE (SecurID) servers

SecurID is a two-factor system produced by the company RSA that uses one-time password (OTP) authentication. This system consists of the following:

- Portable tokens that users carry
- RSA ACE/Server
- Agent host (the FortiGate)

When using SecurID, users carry a small device or "token" that generates and displays a pseudo-random password. According to RSA, each SecurID authenticator token has a unique 64-bit symmetric key that is combined with a powerful algorithm to generate a new code every 60 seconds. The token is time-synchronized with the SecurID RSA ACE/Server.

The RSA ACE/Server is the SecurID system's management component. It stores and validates the information about the SecurID tokens allowed on your network. Alternately, the server can be an RSA SecurID 130 appliance.

The agent host is the server on your network. In this case, this is the FortiGate, which intercepts user logon attempts. The agent host gathers the user ID and password entered from the SecurID token and sends the information to the RSA ACE/Server for validation. If valid, the RSA ACE/Server returns a reply indicating that it is a valid logon and FortiOS allows the user access to the network resources specified in the associated security policy.

Configuring SecurID with FortiOS consists of the following:

1. Configure the RSA and RADIUS servers to work with each other. See RSA server documentation.
2. Do one of the following:
   a. Configure the RSA SecurID 130 appliance.
   b. Configure the FortiGate as an agent host on the RSA ACE/Server.
3. Configure the RADIUS server in FortiOS.
4. Create a SecurID user group.
5. Create a SecurID user.
6. Configure authentication with SecurID.

The following instructions are based on RSA ACE/Server 5.1 and RSA SecurID 130 appliance. They assume that you have successfully completed all external RSA and RADIUS server configuration.

In this example, the RSA server is on the internal network and has an IP address of 192.128.100.000. The FortiOS internal interface address is 192.168.100.3. The RADIUS shared secret is fortinet123, and the RADIUS server is at IP address 192.168.100.202.

To configure the RSA SecurID 130 appliance:

1. Log on to the SecurID IMS console.
2. Go to *RADIUS > RADIUS clients*, then select *Add New*.

RADIUS Client Basics		
	Client Name	FortiGate
	Associated RSA Agent	FortiGate
RADIUS Client Settings		
	IP Address	Enter the FortiOS internal interface. In this example, it is 192.168.100.3.
	Make / Model	Select *Standard Radius*.
	Shared Secret	Enter the RADIUS shared secret. In this example, it is fortinet123.
	Accounting	Leave unselected.
	Client Status	Leave unselected.

3. Configure your FortiGate as a SecurID client:
4. Click *Save*.

To configure the FortiGate as an agent host on the RSA ACE/Server:

1. On the RSA ACE/Server, go to *Start > Programs > RSA ACE/Server*, then *Database Administration - Host Mode*.
2. From the *Agent Host* menu, select *Add Agent Host*.
3. Configure the following:

Name	FortiGate
Network Address	Enter the FortiOS internal interface. In this example, it is 192.168.100.3.
Secondary Nodes	You can optionally enter other IP addresses that resolve to the FortiGate.

For more information, see the RSA ACE/Server documentation.

To configure the RADIUS server in FortiOS:

1. Go to *User & Authentication > RADIUS Servers*, then click *Create New*.
2. Configure the following:

Name	RSA
Authentication method	Select *Default*.
Primary Server	
IP/Name	192.168.100.102. You can click *Test* to ensure the IP address is correct and that FortiOS can contact the RADIUS server.
Secret	fortinet123

3. Click *OK*.

To create a SecurID user group:

1. Go to *User & Authentication > User Groups*. Click *Create New*.
2. Configure the following:

Name	RSA_group
Type	Firewall

3. In *Remote Groups*, click *Add*, then select the RSA server.
4. Click *OK*.

To create a SecurID user:

1. Go to *User & Authentication > User Definition*. Click *Create New*.
2. Configure the following:

User Type	Remote RADIUS User
Type	wloman
RADIUS Server	RSA

Contact Info	(Optional) Enter email or SMS information.
User Group	RSA_group

3. Click *Create*.

You can test the configuration by entering the `diagnose test authserver radius RSA auto wloman 111111111` command. The series of 1s is the OTP that your RSA SecurID token generates that you enter for access.

Configuring authentication with SecurID

You can use the SecurID user group in several FortiOS features that authenticate by user group:

- Security policy on page 1835
- IPsec VPN XAuth on page 1836
- PPTP VPN on page 1836
- SSL VPN

Unless stated otherwise, the following examples use default values.

Security policy

The example creates a security policy that allows HTTP, FTP, and POP3 traffic from the internal interface to WAN1. If these interfaces are not available in FortiOS, substitute other similar interfaces.

To configure a security policy with SecurID authentication:

1. Go to *Policy & Objects > Firewall Policy*.
2. Click *Create New*.
3. Configure the following:

Incoming Interface	internal
Source Address	all
Source User(s)	RSA_group
Outgoing Interface	wan1
Destination Address	all
Schedule	always
Service	HTTP, FTP, POP3
Action	ACCEPT
NAT	On
Shared Shaper	If you want to limit traffic or guarantee minimum bandwidth for traffic that uses the SecurID security policy, enable and use the default shaper, guarantee-100kbps.
Log Allowed Traffic	Enable if you want to generate usage reports on traffic that this policy has authenticated.

4. Click *OK*.

IPsec VPN XAuth

In *VPN > IPsec Wizard*, select the SecurID user group on the *Authentication* page. The SecurID user group members must enter their SecurID code to authenticate.

PPTP VPN

When configuring PPTP in the CLI, set `usrgrp` to the SecurID user group.

SSL VPN

You must map the SecurID user group to the portal that will serve SecurID users and include the SecurID user group in the security policy's *Source User(s)* field.

To map the SecurID group to an SSL VPN portal:

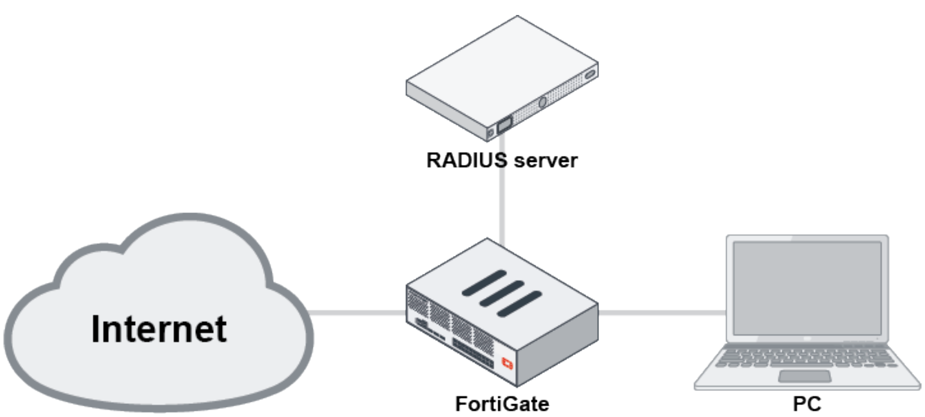
1. Go to *VPN > SSL-VPN Settings*.
2. Under *Authentication/Portal Mapping*, click *Create New*.
3. Configure the following:

Users/Groups	RSA_group
Portal	Select the desired portal.

4. Click *OK*.

Support for Okta RADIUS attributes filter-Id and class

RADIUS user group membership information can be returned in the filter-Id (11) and class (25) attributes in RADIUS Access-Accept messages. The group membership information can be used for group matching in FortiGate user groups in firewall policies and for FortiGate wildcard administrators with remote RADIUS authentication.



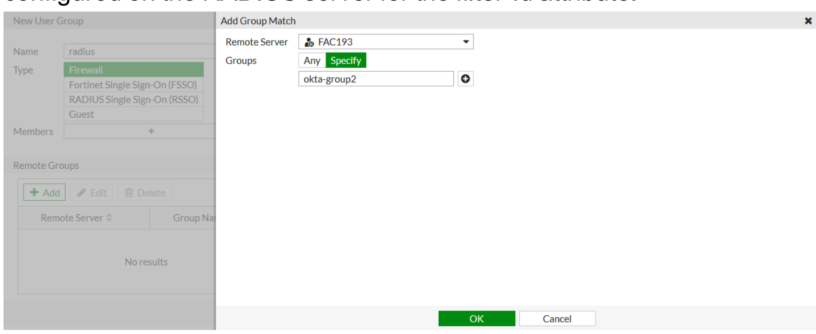
In this example, a FortiAuthenticator is used as the RADIUS server. A local RADIUS user on the FortiAuthenticator is configure with two groups in the filter-Id attribute: *okta-group1* and *okta-group2*.

To create the RADIUS user and set the attribute type to override group information:

```
config user radius
    edit "FAC193"
        set server "10.1.100.189"
        set secret **********
        set group-override-attr-type filter-Id
    next
end
```

FortiOS will only use the configured filter-Id attribute, even if the RADIUS server sends group names in both class and filter-id attributes. To return group membership information from the class attribute instead, set `group-override-attr-type` to `class`.

To configure group match in the user group:

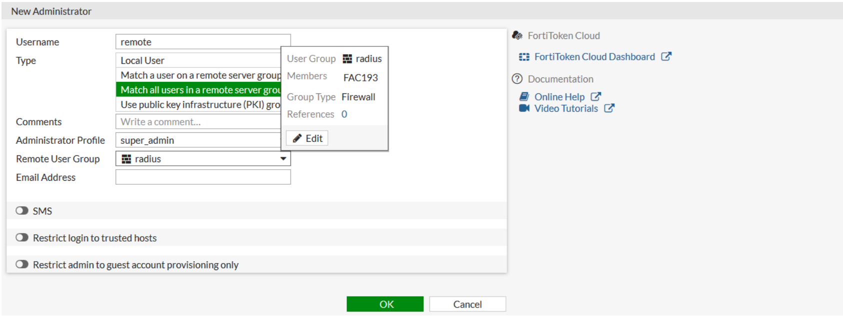
1. Go to *User & Authentication > User Groups*.
2. Click *Create New*.
3. Enter a name for the group, and set *Type* to *Firewall*.
4. In the *Remote Groups* table, click *Add*.
5. Set *Remote Server* to the just created RADIUS server, *FAC193*.
6. Set *Groups* to *Specify*, and enter the group name, `okta-group2`. The string must match the group name configured on the RADIUS server for the filter-Id attribute.


7. Click *OK*.
   The remote server is added to the *Remote Groups* table.
8. Click *OK*.
9. Add the new user group to a firewall policy and generate traffic on the client PC that requires firewall authentication, such as connecting to an external web server.
10. After authentication, on the FortiGate, verify that traffic is authorized in the traffic log:
    a. Go to *Log & Report > Forward Traffic*.
    b. Verify that the traffic was authorized.

To use the remote user group with group match in a system wildcard administrator configuration:

1. Go to *System > Administrators*.
2. Edit an existing administrator, or create a new one.
3. Set *Type* to *Match all users in a remote server group*.

4. Set *Remote User Group* to the remote server.



5. Configure the remaining settings as required.
6. Click *OK*.
7. Log in to the FortiGate using the remote user credentials on the RADIUS server.
   If the correct group name is returned in the filter-Id attribute, administrative access is allowed.

Sending multiple RADIUS attribute values in a single RADIUS Access-Request

A managed FortiSwitch can be configured to send multiple RADIUS attribute values in a single RADIUS Access-Request. This option is configured per RADIUS user, and is set to `none` by default.

The available service type options are:

login	User should be connected to a host.
framed	User use Framed Protocol.
callback-login	User disconnected and called back.
callback-framed	User disconnected and called back, then a Framed Protocol.
outbound	User granted access to outgoing devices.
administrative	User granted access to the administrative unsigned interface.
nas-prompt	User provided a command prompt on the NAS.
authenticate-only	Authentication requested, and no authentication information needs to be returned.
callback-nas-prompt	User disconnected and called back, then provided a command prompt.
call-check	Used by the NAS in an Access-Request packet, Access-Accept to answer the call.
callback-administrative	User disconnected and called back, granted access to the admin unsigned interface.

To configure a managed FortiSwitch to the RADIUS attributes login, framed, and authenticate-only all at the same time:

```
config user radius
    edit "Radius_Server"
        set switch-controller-service-type login framed authenticate-only
```

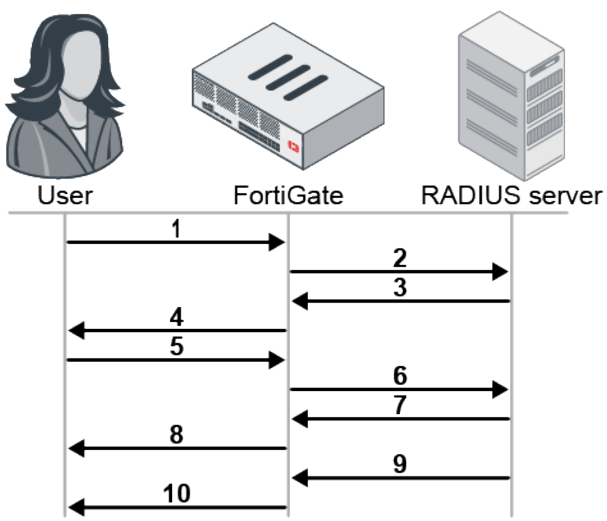
```
        ....
    next
end
```

Traffic shaping based on dynamic RADIUS VSAs

A FortiGate can use the WISPr-Bandwidth-Max-Down and WISPr-Bandwidth-Max-Up dynamic RADIUS VSAs (vendor-specific attributes) to control the traffic rates permitted for a certain device. The FortiGate can apply different traffic shaping to different users who authenticate with RADIUS based on the returned RADIUS VSA values. When the same user logs in from an additional device, the RADIUS server will send a CoA (change of authorization) message to update the bandwidth values to 1/*N* of the total values, where *N* is the number of logged in devices from the same user.

This feature is not supported on NP hardware. NP offloading is automatically disabled on the policy if this feature is enabled.

When a user logs in to two devices through RADIUS authentication. The authentication and authorization flow is as follows:

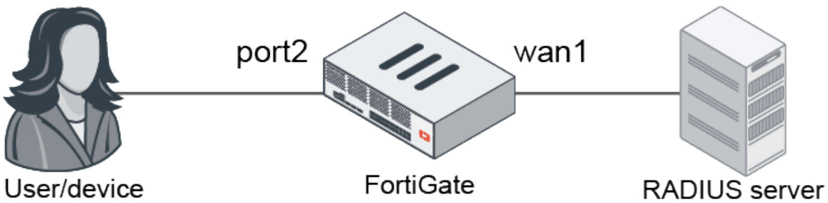


1. The user logs in to a device and the authentication is sent to the FortiGate.
2. The FortiGate sends the Access-Request message to the RADIUS server.
3. The RADIUS server sends the Access-Accept message to the FortiGate. The server also returns the WISPr-Bandwidth-Max-Up and WISPr-Bandwidth-Max-Down VSAs.
4. Based on the VSA values, the FortiGate applies traffic shaping for the upload and download speeds based on its IP.
5. The user logs in to a second device and the authentication is sent to the FortiGate.
6. The FortiGate sends the Access-Request message to the RADIUS server.
7. The RADIUS server sends the Access-Accept message to the FortiGate. The server also returns the WISPr-Bandwidth-Max-Up and WISPr-Bandwidth-Max-Down VSAs at half the value from the first device.
8. Based on the VSA values, the FortiGate applies traffic shaping for the upload and download speeds on the second device based on its IP.
9. The RADIUS server sends a CoA message and returns WISPr-Bandwidth-Max-Up and WISPr-Bandwidth-Max-Down VSAs for the first device at half the value.

10. Based on the VSA values, the FortiGate updates traffic shaping for the upload and download speeds on the first device based on its IP.

Example

In this example, the FortiGate is configured to dynamically shape user traffic based on the WISPr-Bandwidth-Max-Up and WISPr-Bandwidth-Max-Down VSAs returned by the RADIUS server when the user logs in through firewall authentication.



To configure traffic shaping based on dynamic RADIUS VSAs:

1. Configure the RADIUS server users file to identify WISPr-Bandwidth-Max-Up and WISPr-Bandwidth-Max-Down:

The WISPr-Bandwidth is measured in bps, and the FortiOS dynamic shaper is measured in Bps.

```
WISPr-Bandwidth-Max-Up = 1004857,
WISPr-Bandwidth-Max-Down = 504857,
```

2. In FortiOS, configure the RADIUS server:

```
config user radius
    edit "rad1"
        set server "172.16.200.44"
        set secret ************
        set radius-coa enable
        set acct-all-servers enable
        config accounting-server
            edit 1
                set status enable
                set server "172.16.200.44"
                set secret ************
            next
        end
    next
end
```

3. Configure the RADIUS user group:

```
config user group
    edit "group_radius"
        set member "rad1"
    next
end
```

4. Configure the firewall policy with dynamic shaping and the RADIUS group:

```
config firewall policy
    edit 2
        set srcintf "port2"
        set dstintf "wan1"
        set srcaddr "all"
        set dstaddr "all"
        set srcaddr6 "all6"
        set dstaddr6 "all6"
        set action accept
        set schedule "always"
        set service "ALL"
        set dynamic-shaping enable
        set groups "group_radius"
        set nat enable
    next
end
```

Verification

After a client PC is authenticated by the RADIUS server, dynamic shaping is applied to the client based on the IP address.

Use the following commands to monitor the dynamic shaper:

```
# diagnose firewall shaper dynamic-shaper stats
```

```
# diagnose firewall shaper dynamic-shaper list {ip | ipv6 | user} <address or username>
```

Use case 1

User1 is paying for rate plan A that limits their maximum bandwidth to 10 Mbps download and 5 Mbps upload. User2 is paying for rate plan B that limits their maximum bandwidth to 5 Mbps download and 5 Mbps upload. The speeds in both plans are provided by best effort, so there is no guaranteed minimum bandwidth.

User1 logs in to pc1 with RADIUS authentication and IP-based dynamic shaping is applied. User2 logs in to pc2 with RADIUS authentication and IP-based dynamic shaping is applied.

To verify the dynamic shaping:

1. On pc1, verify the bandwidth and transfer speed:

```
root@pc1:~# iperf -c 172.16.200.44 -u -t 25 -b 20M
------------------------------------------------------------
Client connecting to 172.16.200.44, UDP port 5001
Sending 1470 byte datagrams
UDP buffer size:  208 KByte (default)
------------------------------------------------------------
[  3] local 10.1.100.11 port 50510 connected with 172.16.200.44 port 5001
[ ID] Interval       Transfer     Bandwidth
[  3]  0.0-25.0 sec  59.6 MBytes  20.0 Mbits/sec
[  3] Sent 42518 datagrams
[  3] Server Report:
[  3]  0.0-25.3 sec  30.1 MBytes  9.99 Mbits/sec  15.651 ms 21058/42518 (50%)
```

2. On pc2, verify the bandwidth and transfer speed:

```
root@pc2:~# iperf -c 172.16.200.44 -u -t 25 -b 20M
------------------------------------------------------------
Client connecting to 172.16.200.44, UDP port 5001
Sending 1470 byte datagrams
UDP buffer size:  208 KByte (default)
------------------------------------------------------------
[  3] local 10.1.100.22 port 52814 connected with 172.16.200.44 port 5001
[ ID] Interval       Transfer     Bandwidth
[  3]  0.0-25.0 sec  59.6 MBytes  20.0 Mbits/sec
[  3] Sent 42518 datagrams
[  3] Server Report:
[  3]  0.0-25.3 sec  15.1 MBytes  5.03 Mbits/sec  15.652 ms 31710/42514 (75%)
```

3. In FortiOS, check the authentication list:

```
# diagnose firewall auth  list
10.1.100.11, test-shaper1
        src_mac: **:**:**:**:**:**
        type: fw, id: 0, duration: 38, idled: 16
        expire: 562
        flag(814): hard radius no_idle
        server: rad1
        packets: in 8207 out 3999, bytes: in 12306164 out 226963
        group_id: 3
        group_name: group_radius
10.1.100.22, test-shaper2
        src_mac: **:**:**:**:**:**
        type: fw, id: 0, duration: 24, idled: 24
        expire: 156, max-life: 35976
        flag(814): hard radius no_idle
        server: rad1
        packets: in 0 out 5, bytes: in 0 out 300
        group_id: 3
        group_name: group_radius
----- 2 listed, 0 filtered ------
```

4. Check the dynamic shaper list:

```
# diagnose firewall shaper dynamic-shaper list
addr: 10.1.100.11
bandwidth(original/reply): 1250000 Bps/625000 Bps
current bandwidth(original/reply): 1237072 Bps/0 Bps
allow packets(original/reply): 38524/14
allow bytes(original/reply): 55270378/11285
drop packets(original/reply): 10136/0
drop bytes(original/reply): 13516198/0
life: 441
idle: 0/40
idle time limit: 600 s

addr: 10.1.100.22
bandwidth(original/reply): 625000 Bps/625000 Bps
current bandwidth(original/reply): 622909 Bps/0 Bps
allow packets(original/reply): 3232/3
allow bytes(original/reply): 4841536/243
drop packets(original/reply): 2753/0
```

```
drop bytes(original/reply): 4123994/0
life: 10
idle: 0/10
idle time limit: 36000 s
```

5. Check the session list:

```
# diagnose sys session list
session info: proto=6 proto_state=05 duration=3 expire=116 timeout=3600 flags=00000004
socktype=4 sockport=10001 av_idx=0 use=4
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/0
state=redir log local may_dirty auth dst-vis f00 dynamic_shaping
statistic(bytes/packets/allow_err): org=0/0/0 reply=638/4/1 tuples=2
tx speed(Bps/kbps): 0/0 rx speed(Bps/kbps): 185/1
orgin->sink: org pre->post, reply pre->post dev=20->17/17->20 gwy=172.16.200.44/0.0.0.0
hook=pre dir=org act=noop 10.1.100.22:35561->172.16.200.44:80(0.0.0.0:0)
hook=post dir=reply act=noop 172.16.200.44:80->10.1.100.22:35561(0.0.0.0:0)
pos/(before,after) 0/(0,0), 0/(0,0)
src_mac=**:**:**:**:**:**  dst_mac=**:**:**:**:**:**
misc=0 policy_id=2 auth_info=0 chk_client_info=0 vd=1
serial=0005994d tos=ff/ff app_list=0 app=0 url_cat=0
sdwan_mbr_seq=0 sdwan_service_id=0
rpdb_link_id=00000000 rpdb_svc_id=0 ngfwid=n/a
npu_state=0x000001 no_offload
no_ofld_reason:  redir-to-av auth disabled-by-policy

session info: proto=6 proto_state=05 duration=122 expire=38 timeout=3600 flags=00000000
socktype=0 sockport=0 av_idx=0 use=3
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/255
user=test-shaper1 auth_server=rad1 state=log may_dirty authed f00 dynamic_shaping acct-
ext
statistic(bytes/packets/allow_err): org=383611/6604/1 reply=26382470/17592/1 tuples=2
tx speed(Bps/kbps): 0/0 rx speed(Bps/kbps): 0/0
orgin->sink: org pre->post, reply pre->post dev=20->17/17->20 gwy=172.16.200.44/10.2.2.1
hook=post dir=org act=snat 10.1.100.11:54140->172.16.200.44:80(172.16.200.2:54140)
hook=pre dir=reply act=dnat 172.16.200.44:80->172.16.200.2:54140(10.1.100.11:54140)
pos/(before,after) 0/(0,0), 0/(0,0)
src_mac=**:**:**:**:**:**  dst_mac=**:**:**:**:**:**
misc=0 policy_id=2 auth_info=3 chk_client_info=0 vd=1
serial=000598c5 tos=ff/ff app_list=0 app=0 url_cat=0
sdwan_mbr_seq=0 sdwan_service_id=0
rpdb_link_id=00000000 rpdb_svc_id=0 ngfwid=n/a
npu_state=0x000001 no_offload
no_ofld_reason:  disabled-by-policy
total session 2
```

6. Check the policy traffic:

```
# diagnose firewall iprope list 100004
policy index=2 uuid_idx=60 action=accept
flag (8052128): redir auth nat nids_raw master use_src pol_stats
```

```
flag2 (4030): fw wsso resolve_sso
flag3 (200000b0): !sp link-local best-route dynamic-shaping
schedule(always)
cos_fwd=255  cos_rev=255
group=00100004 av=00004e20 au=00000003 split=00000000
host=1 chk_client_info=0x1 app_list=0 ips_view=0
misc=0
zone(1): 20 -> zone(1): 17
source(1): 0.0.0.0-255.255.255.255, uuid_idx=32,
dest(1): 0.0.0.0-255.255.255.255, uuid_idx=32,
user group(1): 3
service(1):
        [0:0x0:0/(0,65535)->(0,65535)] helper:auto
```

Use case 2

A user logs in to a device (pc1, 10.1.100.11) and has a maximum bandwidth of 10 Mbps download and 5 Mbps upload. The same user logs in to a second device (pc2, 10.1.100.22) and the RADIUS server sends a CoA request with the WISPr-Bandwidth-Max to pc1. The maximum bandwidth on pc1 changes to 5 Mbps download and 2.5Mbps upload. On pc2, the maximum bandwidth is also 5 Mbps download and 2.5Mbps upload.

When the user logs out from pc1, the RADIUS server sends CoA request with the new WISPr-Bandwidth-Max for pc2. The FortiGate updates the authentication user list and dynamic shaper for pc2. The maximum bandwidth on pc2 changes to 10 Mbps download and 5 Mbps upload.

To verify the dynamic shaping:

1. Check the dynamic shaper list after the user logs in to pc1:

```
# diagnose firewall shaper dynamic-shaper list
addr: 10.1.100.11
bandwidth(original/reply): 1250000 Bps/625000 Bps
current bandwidth(original/reply): 0 Bps/0 Bps
allow packets(original/reply): 0/3
allow bytes(original/reply): 0/243
drop packets(original/reply): 0/0
drop bytes(original/reply): 0/0
life: 491
idle: 4/4
idle time limit: 86400 s
```

2. Check the dynamic shaper list after the user logs in to pc2:

```
# diagnose firewall shaper dynamic-shaper list
addr: 10.1.100.11
bandwidth(original/reply): 625000 Bps/312500 Bps
current bandwidth(original/reply): 0 Bps/0 Bps
allow packets(original/reply): 0/0
allow bytes(original/reply): 0/0
drop packets(original/reply): 0/0
drop bytes(original/reply): 0/0
life: 652
idle: 5/5
idle time limit: 600 s

addr: 10.1.100.22
bandwidth(original/reply): 625000 Bps/312500 Bps
```

```
current bandwidth(original/reply): 0 Bps/0 Bps
allow packets(original/reply): 0/3
allow bytes(original/reply): 0/243
drop packets(original/reply): 0/0
drop bytes(original/reply): 0/0
life: 3
idle: 3/3
idle time limit: 86400 s
```

3. Check the authentication list:

```
# diagnose firewall auth list
10.1.100.11, test
        src_mac: **:**:**:**:**:**
        type: fw, id: 0, duration: 171, idled: 11
        expire: 589, max-life: 589
        flag(814): hard radius no_idle
        server: rad1
        packets: in 0 out 0, bytes: in 0 out 0
        group_id: 15
        group_name: group_radius
10.1.100.22, test
        src_mac: **:**:**:**:**:**
        type: fw, id: 0, duration: 9, idled: 9
        expire: 86391
        flag(814): hard radius no_idle
        server: rad1
        packets: in 0 out 0, bytes: in 0 out 0
        group_id: 15
        group_name: group_radius
----- 2 listed, 0 filtered ------
```

4. Check the dynamic shaper list after the user logs out from pc1:

```
# diagnose firewall shaper dynamic-shaper list
addr: 10.1.100.22
bandwidth(original/reply): 1250000 Bps/625000 Bps
current bandwidth(original/reply): 0 Bps/0 Bps
allow packets(original/reply): 0/0
allow bytes(original/reply): 0/0
drop packets(original/reply): 0/0
drop bytes(original/reply): 0/0
life: 414
idle: 9/9
idle time limit: 600 s
```

5. Check the authentication list again:

```
# diagnose firewall auth  list
10.1.100.22, test
        src_mac: **:**:**:**:**:**
        type: fw, id: 0, duration: 453, idled: 49
        expire: 551, max-life: 551
        flag(814): hard radius no_idle
        server: rad1
        packets: in 0 out 0, bytes: in 0 out 0
        group_id: 15
```

```
          group_name: group_radius
  ----- 1 listed, 0 filtered ------
```

RADIUS Termination-Action AVP in wired and wireless scenarios

When authenticating with RADIUS in a wired or wireless scenario, the FortiGate can support proper handling of the Termination-Action AVP.

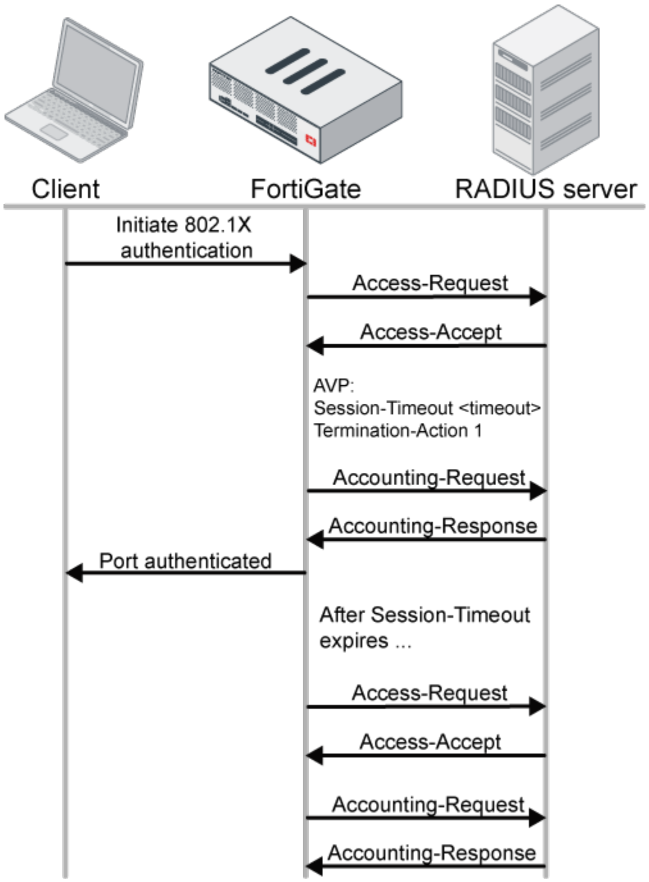
In a wired scenario, a hardware switch configured with 802.1X security authentication can read the Termination-Action attribute value from the RADIUS Access-Accept response. If the Termination-Action is 1, the FortiGate will initiate re-authentication when the session time has expired. During re-authentication, the port stays authorized. If the Termination-Action is 0, the session will be terminated.

In a wireless scenario, when a virtual AP is configured with WPA2-Enterprise security with RADIUS and has CoA enabled, it processes the RADIUS CoA request immediately upon receiving it and re-authenticates when the Termination-Action is 1.

Wired example

This example has a FortiGate configured with a hardware switch with two ports: port3 and port5. The hardware switch is enabled with 802.1X security and assigned to a RADIUS user group. Upon a successful authentication, the RADIUS server responds with an Access-Accept containing the authentication Session-Timeout and Termination-Action attributes. In this example, the Termination-Action value is 1, which informs the client to re-authenticate when the session time expires. During this time, the FortiGate keeps the client/port authorized while it initiates the re-authentication with the RADIUS server.

The message exchange is as follows:



To configure the RADIUS server and the FortiGate to handle the Termination-Action AVP:

1. On the RADIUS server, configure the Termination-Action AVP with the value `RADIUS-Request (1)` to indicate that re-authentication should occur upon expiration of the Session-Time.
2. On the FortiGate, configure the RADIUS server:

```
config user radius
    edit "rad1"
        set server "172.18.60.203"
        set secret ENC **********
        set radius-coa enable
        config accounting-server
            edit 1
                set status enable
                set server "172.18.60.203"
                set secret ENC **********
            next
        end
    next
end
```

3. Configure the RADIUS user group:

```
config user group
    edit "group_radius"
        set member "rad1"
```

```
    next
end
```

4. Configure the hardware switch with 802.1X enabled.
   a. Configure the virtual switch settings:

```
config system virtual-switch
    edit hw2
        set physical-switch "sw0"
        config port
            edit port3
            next
            edit port5
            next
        end
    next
end
```

   b. Configure the interface settings:

```
config system interface
    edit hw2
        set vdom vdom1
        set ip 6.6.6.1 255.255.255.0
        set allowaccess ping https ssh
        set stp enable
        set security-mode 802.1X
        set security-groups "group_radius"
    next
end

WARNING: Changing 802.1X could interrupt network connectivity on affected interfaces.
Do you want to continue? (y/n)y
```

5. On the client device, initiate 802.1X authentication, then verify that the switch port shows as authorized:

```
# diagnose sys 802-1x status
Virtual switch 'hw2' (default mode) 802.1x member status:
  port3: Link up, 802.1X state: unauthorized
  port5: Link up, 802.1X state: authorized
```

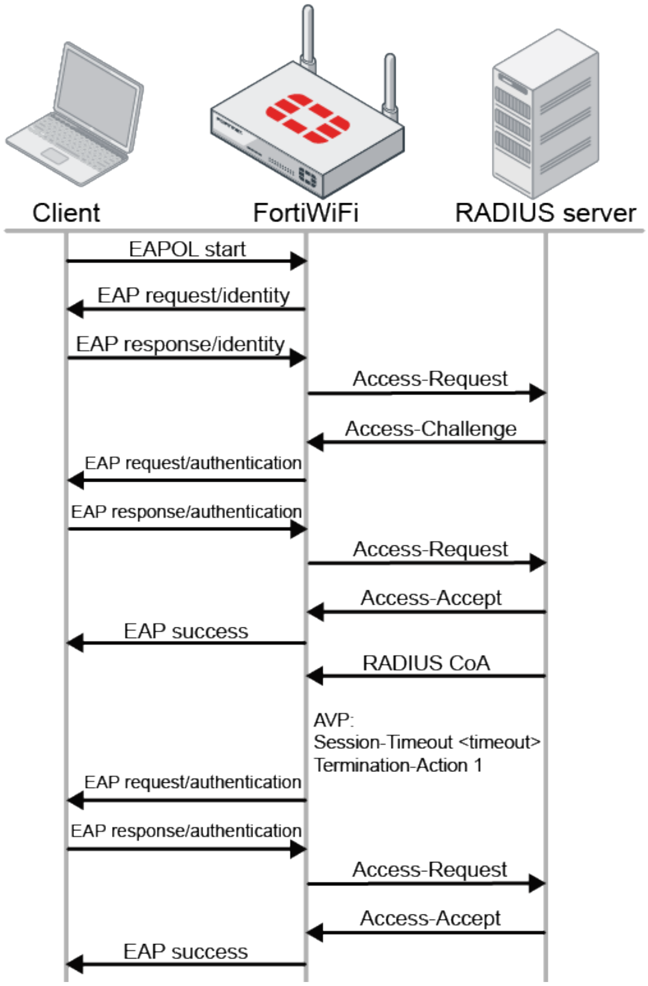
6. After successful authentication, wait for the session to timeout.
7. The FortiGate will keep the 802.1X port authenticated, and initiate re-authentication with the same Acct-Session-Id to the RADIUS server. The 802.1X status of the port remains unchanged:

```
# diagnose sys 802-1x status
Virtual switch 'hw2' (default mode) 802.1x member status:
  port3: Link up, 802.1X state: unauthorized
  port5: Link up, 802.1X state: authorized
```

Wireless example

In this example, a virtual AP is configured with WPA2-Enterprise security with RADIUS and has CoA enabled. After a wireless user authenticates and connects to the wireless SSID, the RADIUS server triggers a CoA event with AVPs Session-timeout and a Termination-Action of 1. This signals the FortiGate to trigger re-authentication of the client, which the client immediately performs to stay connected to the wireless SSID.

The message exchange is as follows:



To configure the FortiGate to handle the Termination-Action AVP:

1. Configure the RADIUS server:

```
config user radius
    edit "peap"
        set server "172.16.200.55"
        set secret **********
        set radius-coa enable
    next
end
```

2. Configure the VAP:

```
config wireless-controller vap
    edit "wifi"
        set ssid "FWF-60E-coa"
        set security wpa2-only-enterprise
        set auth radius
        set radius-server "peap"
        set schedule "always"
    next
end
```

3. Verify that the wireless station connects to the SSID:

```
# diagnose wireless-controller wlac -d sta online
   vf=0 wtp=1 rId=1 wlan=wifi vlan_id=0 ip=10.10.80.2 ip6=:: mac=**:**:**:**:**:** vci=
host=wifi-qa-01 user=test1 group=group1 signal=-28 noise=-95 idle=1 bw=0 use=6 chan=149
radio_type=11AC security=wpa2_only_enterprise mpsk= encrypt=aes cp_authed=no online=yes
mimo=2
```

4. From the RADIUS server, manually trigger a RADIUS CoA event.
   a. RADIUS CoA sent to the FortiGate:

   ```
   Sent CoA-Request Id 7 from 0.0.0.0:54158 to 172.16.200.201:3799 length 39
       User-Name = "test1"
       Session-Timeout = 120
       Termination-Action = RADIUS-Request
   ```

   b. RADIUS CoA-ACK received from the FortiGate:

   ```
   Received CoA-ACK Id 7 from 172.16.200.201:3799 to 0.0.0.0:0 length 44
       Event-Timestamp = "Jan  5 2022 14:43:12 PST"
       Message-Authenticator = 0x3311ba3b763d68da653ab34351b0308
   ```

5. On the wireless station console, verify that the re-authentication happens immediately:

```
root@wifi-qa-01:/home/wpa-test# wlan1: CTRL-EVENT-EAP-STARTED EAP authentication started
wlan1: CTRL-EVENT-EAP-PROPOSED-METHOD vendor=0 method=25
wlan1: CTRL-EVENT-EAP-METHOD EAP vendor 0 method 25 (PEAP) selected
EAP-TLV: TLV Result - Success - EAP-TLV/Phase2 Completed
wlan1: CTRL-EVENT-EAP-SUCCESS EAP authentication completed successfully
wlan1: PMKSA-CACHE-REMOVED **:**:**:**:**:** 0
wlan1: PMKSA-CACHE-ADDED **:**:**:**:**:** 0
wlan1: WPA: Key negotiation completed with **:**:**:**:**:** [PTK=CCMP GTK=CCMP]
```

TACACS+ servers

TACACS+ is a remote authentication protocol that provides access control for routers, network access servers, and other network devices through one or more centralized servers.

FortiOS sends the following proprietary TACACS+ attributes to the TACACS+ server during authorization requests:

Attribute	Description
service=<name>	User must be authorized to access the specified service.
memberof	Group that the user belongs to.
admin_prof	Administrator profile (admin access only).

Only memberof and admin_prof attributes are parsed in authentication replies.

You can configure up to ten remote TACACS+ servers in FortiOS. You must configure at least one server before you can configure remote users.

A TACACS+ server must first be added in the CLI to make the option visible in the GUI.

To configure TACACS+ authentication in the CLI:

1. Configure the TACACS+ server entry:

```
config user tacacs+
    edit "TACACS-SERVER"
        set server <IP address>
        set key <string>
        set authen-type ascii
        set source-ip <IP address>
    next
end
```

2. Configure the remote user group:

```
config user group
    edit "TACACS-GROUP"
        set group-type firewall
        set member "TACACS-SERVER"
    next
end
```

3. Configure the remote user:

```
config system admin
    edit TACACS-USER
        set remote-auth enable
        set accprofile "super_admin"
        set vdom "root"
        set wildcard enable
        set remote-group "TACACS-GROUP"
    next
end
```

To configure a TACACS+ server in the GUI:

1. Go to *User & Authentication > TACACS+ Servers*.
2. Click *Create New*.
3. Configure the following settings:

Name	Enter the TACACS+ server name.
Authentication Type	Select the authentication type used for the TACACS+ server. Selecting *Auto* tries PAP, MSCHAP, and CHAP, in that order.
Server IP/Name	Enter the domain name or IP address for the primary server.
Server Secret	Enter the key to access the primary server.

4. Click *OK*.

SAML

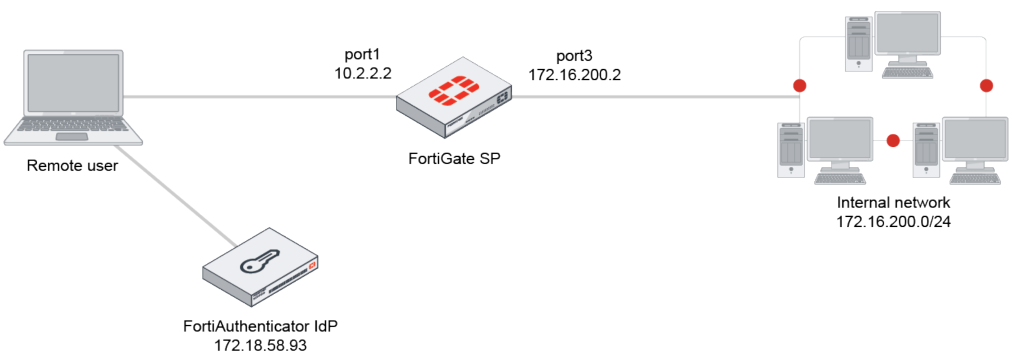
The following topics provide information about SAML:

Outbound firewall authentication for a SAML user

When you configure a FortiGate as a service provider (SP), you can create an authentication profile that uses SAML for firewall authentication.

> You must use the identity provider's (IdP) remote certificate on the SPs.

The following example uses a FortiGate as an SP and FortiAuthenticator as the IdP server:



To configure firewall authentication:

1. Configure the FortiGate SP to be a SAML user:

```
config user saml
    edit "fac-firewall"
        set entity-id "http://10.2.2.2:1000/saml/metadata/"
        set single-sign-on-url "https://10.2.2.2:1003/saml/login/"
        set single-logout-url "https://10.2.2.2:1003/saml/logout/"
        set idp-entity-id "http://172.18.58.93:443/saml-idp/bbbbbb/metadata/"
        set idp-single-sign-on-url "https://172.18.58.93:443/saml-idp/bbbbbb/login/"
        set idp-single-logout-url "https://172.18.58.93:443/saml-idp/bbbbbb/logout/"
        set idp-cert "REMOTE_Cert_3"
        set user-name "username"
        set group-name "group"
```

```
    next
end
```

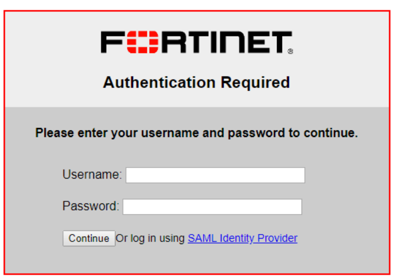
2. Add the SAML user to the user group (optionally, you can configure group matching):

```
config user group
    edit "saml_firewall"
        set member "fac-firewall"
        config match
            edit 1
                set server-name "fac-firewall"
                set group-name "user_group1"
            next
        end
    next
end
```

3. Add the SAML user group to a firewall policy:

```
config firewall policy
    edit 2
        set srcintf "port3"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "pc4"
        set action accept
        set schedule "always"
        set service "ALL"
        set logtraffic all
        set fsso disable
        set groups "saml_firewall" "group_local"
        set users "first"
        set nat enable
    next
end
```

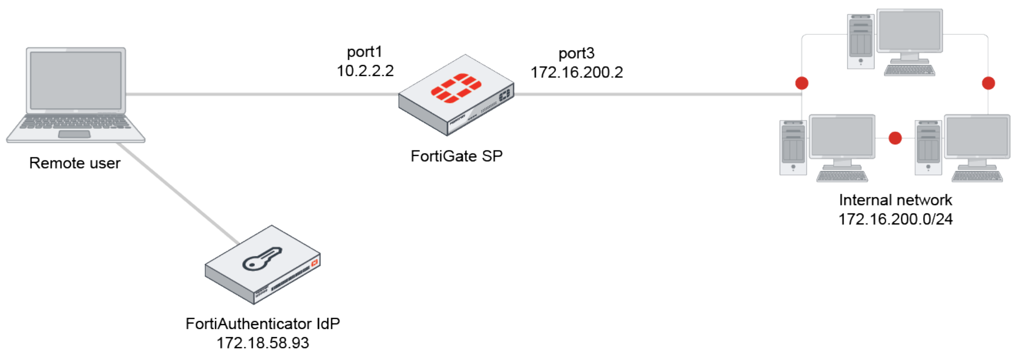
4. Configure the FortiAuthenticator IdP as needed.
5. Run HTTP/HTTPS authentication for a remote user. The SAML login page appears:



SAML SP for VPN authentication

When you configure a FortiGate as a service provider (SP), you can create an authentication profile that uses SAML for SSL VPN web portal authentication.

The following example uses a FortiGate as an SP and FortiAuthenticator as the IdP server:



To configure SSL VPN web portal authentication:

1. Configure the FortiGate SP to be a SAML user:

```
config user saml
    edit "fac-sslvpn"
        set entity-id "https://10.2.2.2:10443/remote/saml/metadata/"
        set single-sign-on-url "https://10.2.2.2:10443/remote/saml/login/"
        set single-logout-url "https://10.2.2.2:10443/remote/saml/logout/"
        set idp-entity-id "http://172.18.58.93:443/saml-idp/ssssss/metadata/"
        set idp-single-sign-on-url "https://172.18.58.93:443/saml-idp/ssssss/login/"
        set idp-single-logout-url "https://172.18.58.93:443/saml-idp/ssssss/logout/"
        set idp-cert "REMOTE_Cert_3"
        set user-name "username"
    next
end
```

2. Add the SAML user to the user group (group matching may also be configured):

```
config user group
    edit "saml_sslvpn"
        set member "fac-sslvpn"
    next
end
```

3. Configure SSL VPN:

```
config vpn ssl settings
    set servercert "Fortinet_Factory"
    set tunnel-ip-pools "SSLVPN_TUNNEL_ADDR1"
    set tunnel-ipv6-pools "SSLVPN_TUNNEL_IPv6_ADDR1"
    set source-interface "port3"
    set source-address "all"
    set source-address6 "all"
    set default-portal "full-access"
    config authentication-rule
        edit 1
            set groups "saml_sslvpn"
            set portal "full-access"
        next
    end
end
```

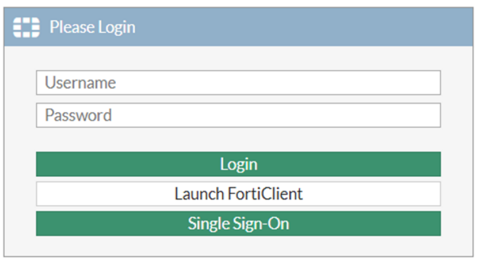
4. Add the SAML user group to a firewall policy:

```
config firewall policy
    edit 8
        set srcintf "ssl.vdom1"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set groups "local" "saml_sslvpn"
        set nat enable
    next
end
```

5. Configure the FortiAuthenticator IdP as needed.

To connect from the SSL VPN web portal:

1. In a web browser, enter the portal address. The SAML login page appears:



2. Enter the user name and password.
3. Click *Login*, or if SSO has been configured, click *Single-Sign-On*.
   Once authenticated, the web portal opens.

Using a browser as an external user-agent for SAML authentication in an SSL VPN connection

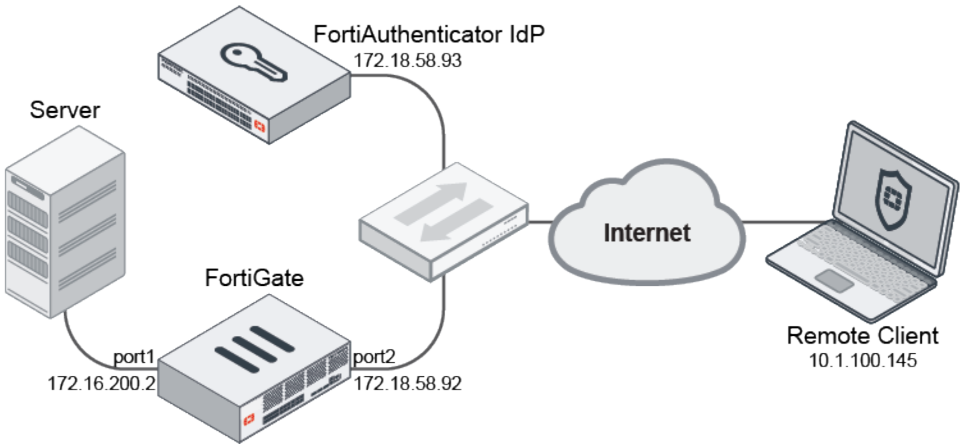
FortiClient can use a browser as an external user-agent to perform SAML authentication for SSL VPN tunnel mode, instead of the FortiClient embedded log in window. If a user has already done SAML authentication in the default browser, they do not need to authenticate again in the FortiClient built-in browser. FortiClient 7.0.1 and later is required.

The following CLI is used to set the SAML local redirect port on the FortiClient endpoint after successful SAML authentication:

```
config vpn ssl settings
    set saml-redirect-port <port>
end
```

Example

In this example, a user wants to use their default browser to connect to IdP for SAML authentication, without needing to separately authenticate in the FortiClient built-in browser. After authenticating in the browser, FortiClient obtains the authentication cookie directly from the browser.



The authentication process proceeds as follows:

1. The remote client uses FortiClient to connect to the FortiGate SSL VPN on 172.16.58.92:1443 with the *Use external browser as user-agent for saml user authentication* option enabled.
2. The SSL VPN redirects FortiClient to complete SAML authentication using the Identity Provider (IdP).
3. FortiClient opens the default browser to authenticate the IdP server.
4. After a successful authentication, the browser redirects to localhost:<port>, where the port is defined by the `saml-redirect-port` variable on the FortiGate.
5. FortiClient reads the authentication ID passed by the successful authentication, then requests that the SAML authentication process continues on the FortiGate with this ID.
6. The FortiGate continues with the remaining SSL-VPN host-check and other steps until it receives the authentication cookie. It then allow the SSL VPN user to connect using tunnel mode.

To configure the VPN:

1. Configure a SAML user:

```
config user saml
    edit "su1"
        set cert "fgt_gui_automation"
        set entity-id "http://172.18.58.92:1443/remote/saml/metadata/"
        set single-sign-on-url "https://172.18.58.92:1443/remote/saml/login/"
        set single-logout-url "https://172.18.58.92:1443/remote/saml/logout/"
        set idp-entity-id "http://172.18.58.93:443/saml-idp/222222/metadata/"
        set idp-single-sign-on-url "https://172.18.58.93:443/saml-idp/222222/login/"
        set idp-single-logout-url "https://172.18.58.93:443/saml-idp/222222/logout/"
        set idp-cert "REMOTE_Cert_1"
        set user-name "Username"
        set group-name "Groupname"
        set digest-method sha1
    next
end
```

2. Add the SAML user to a user group:

```
config user group
    edit "saml_grp"
        set member "su1"
```

```
    next
end
```

3. Create an SSL VPN web portal:

```
config vpn ssl web portal
    edit "testportal1"
        set tunnel-mode enable
        set ipv6-tunnel-mode enable
        set web-mode enable
        ...
    next
end
```

4. Configure the SSL VPN:

```
config vpn ssl settings
    set tunnel-ip-pools "SSLVPN_TUNNEL_ADDR1"
    set tunnel-ipv6-pools "SSLVPN_TUNNEL_IPv6_ADDR1"
    set port 1443
    set source-interface "port2"
    set source-address "all"
    set source-address6 "all"
    set default-portal "testportal1"
    ...
end
```

5. Configure a firewall policy for the SSL VPN and assign the SAML group and a local user to it:

```
config firewall policy
    edit 1
        set name "policy_to_sslvpn_tunnel"
        set srcintf "ssl.root"
        set dstintf "port1"
        set action accept
        set srcaddr "all"
        set dstaddr "all"
        set srcaddr6 "all"
        set dstaddr6 "all"
        set schedule "always"
        set service "ALL"
        set nat enable
        set groups "saml_grp"
        set users "u1"
    next
end
```

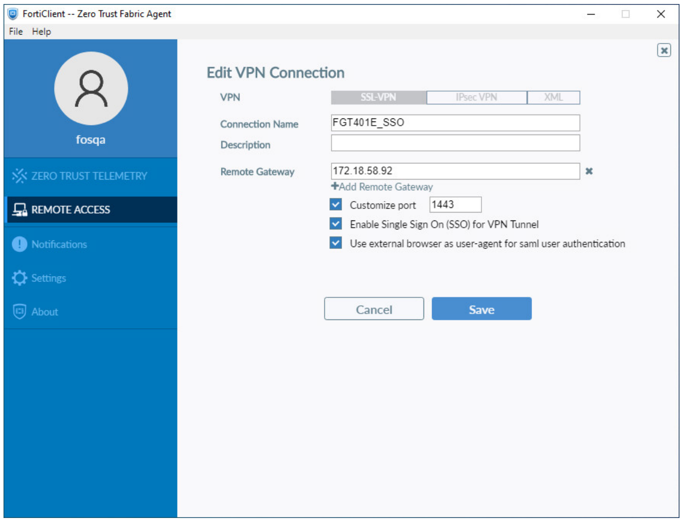
6. Enable the SAML redirect port:

```
config vpn ssl settings
    set saml-redirect-port 8020
end
```

To connect to the VPN using FortiClient:

1. Configure the SSL VPN connection:
   a. Open FortiClient and go to the *Remote Access* tab and click *Configure VPN*.
   b. Enter a name for the connection.

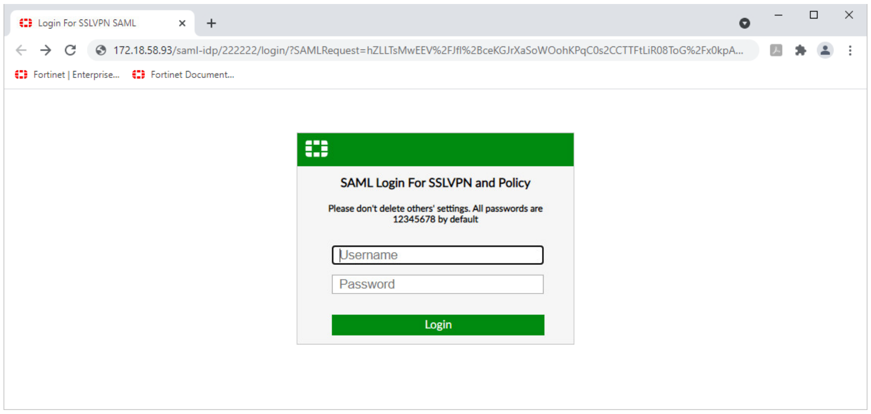
c. Set the *Remote Gateway* to the FortiGate port *172.18.58.92*.
d. Enable *Customize port* and set the port to *1443*.
e. Enable *Enable Single Sign On (SSO) for VPN Tunnel* and *Use external browser as user-agent for saml user authentication*.



f. Click *Save*.

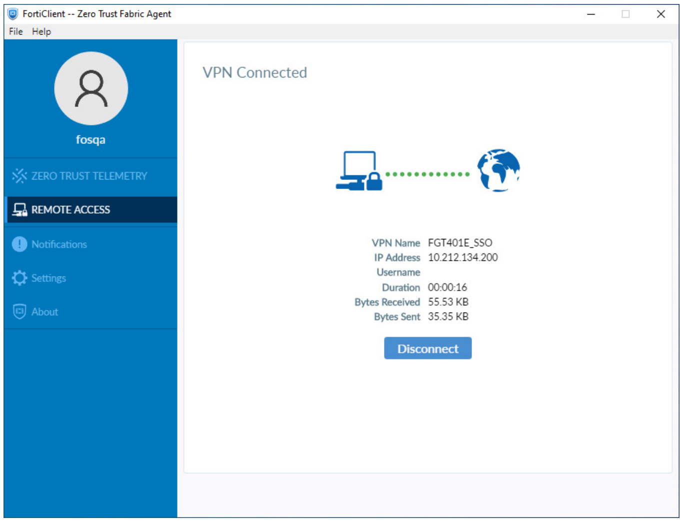
2. On the *Remote Access* tab select the *FGT401E_SSO* VPN connection from the dropdown list.
3. Click *SAML Login*.

   The default browser opens to the IdP authentication page.



4. Enter the username and password, then click *Login*.

   The authenticated result is sent back to FortiClient and the connection is established.



To check the connection on the FortiGate:

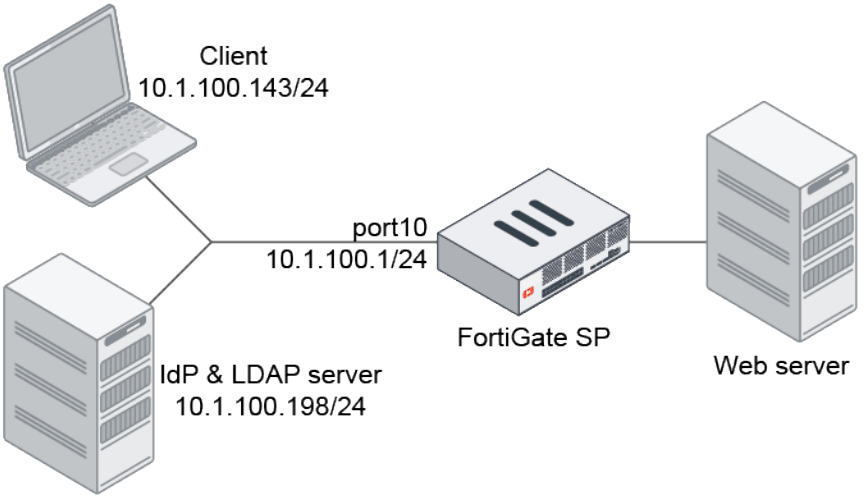
```
# get vpn ssl monitor
SSL-VPN Login Users:
 Index    User     Group    Auth Type      Timeout          Auth-Timeout     From      HTTP in/out
   HTTPS in/out    Two-factor Auth
 1        fac3     saml_grp        256(1)           N/A      10.1.100.254    0/0      0/0      0

SSL-VPN sessions:
 Index    User     Group    Source IP      Duration         I/O Bytes        Tunnel/Dest IP
 0        fac3     saml_grp        10.1.100.254     5       9990/8449
10.212.134.200,fdff:ffff::1
```

```
# diagnose firewall auth list

10.212.134.200, fac3
        type: fw, id: 0, duration: 6, idled: 0
        expire: 259199, allow-idle: 259200
        flag(80): sslvpn
        server: su1
        packets: in 28 out 28, bytes: in 23042 out 8561
        group_id: 5
        group_name: saml_grp
```

SAML authentication in a proxy policy

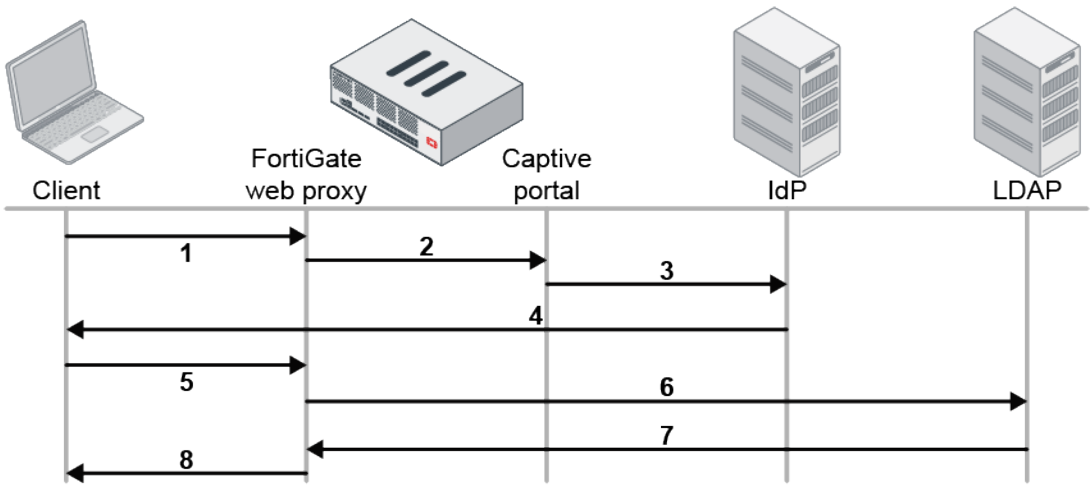
SAML user authentication can be used in explicit web proxies and transparent web proxies with the FortiGate acting as a SAML SP. SAML can be used as an authentication method for an authentication scheme that requires using a captive portal.

Topology



In this configuration, SAML authentication is used with an explicit web proxy. The IdP is a Windows 2016 server configured with ADFS. The LDAP and IdP servers are on the same server. The LDAP server is used as the user backend for the IdP to perform authentication; however, they are not required to be on the same server.

The authentication and authorization flow is as follows:



1. The client opens a browser and visits https://www.google.com.
2. The browser is redirected by the web proxy the captive portal.
3. The request is redirected to the IdP's sign-in page.
4. If the user signs in, the IdP authenticates the user and sends back a SAML assertion message to the user's browser with the user group information.
5. The browser forwards the SAML assertion response as a HTTP POST to the FortiGate SAML assertion consumer service URL (https://fgt9.myqalab.local:7831/XX/YY/ZZ/saml/login/).
6. If the FortiGate authentication scheme has a user database configured, the FortiGate will query the LDAP server for the user group information and ignore the user group information from the SAML message.
7. The user group information is returned. The FortiGate matches the user group information against the LDAP group in the proxy policy group settings. If there is a match, the request is authorized and the proxy policy is matched.
8. If all policy criteria match successfully, then the webpage is returned to the client.

To configure SAML authentication with an explicit web proxy:

1. Enable the web proxy:

```
config web-proxy explicit
    set status enable
    set http-incoming-port 8080
end
```

2. Enable the proxy captive portal:

```
config system interface
    edit "port10"
        set vdom "vdom1"
        set ip 10.1.100.1 255.255.255.0
        set allowaccess ping https ssh snmp http telnet
        set type physical
        set explicit-web-proxy enable
        set explicit-ftp-proxy enable
        set proxy-captive-portal enable
        set snmp-index 12
    next
end
```

3. Configure the LDAP server:

```
config user ldap
    edit "ldap-10.1.100.198"
        set server "10.1.100.198"
        set cnid "cn"
        set dn "dc=myqalab,dc=local"
        set type regular
        set username "cn=fosqa1,cn=users,dc=myqalab,dc=local"
        set password ************
        set group-search-base "dc=myqalab,dc=local"
    next
end
```

4. Configure the user group:

```
config user group
    edit "ldap-group-saml"
        set member "ldap-10.1.100.198"
    next
end
```

5. Configure SAML:

```
config user saml
    edit "saml_user"
        set cert "Fortinet_CA_SSL"
        set entity-id "https://fgt9.myqalab.local:7831/XX/YY/ZZ/saml/metadata/"
        set single-sign-on-url "https://fgt9.myqalab.local:7831/XX/YY/ZZ/saml/login/"
        set single-logout-url "https://fgt9.myqalab.local:7831/XX/YY/ZZ/saml/logout/"
        set idp-entity-id "http://MYQALAB.LOCAL/adfs/services/trust"
        set idp-single-sign-on-url "https://myqalab.local/adfs/ls"
        set idp-single-logout-url "https://myqalab.local/adfs/ls"
        set idp-cert "REMOTE_Cert_4"
        set digest-method sha256
```

```
        set adfs-claim enable
        set user-claim-type name
        set group-claim-type group
    next
end
```

6. Configure the authentication scheme, rule, and setting:

```
config authentication scheme
    edit "saml"
        set method saml
        set saml-server "saml_user"
        set user-database "ldap-10.1.100.198"
    next
end
```

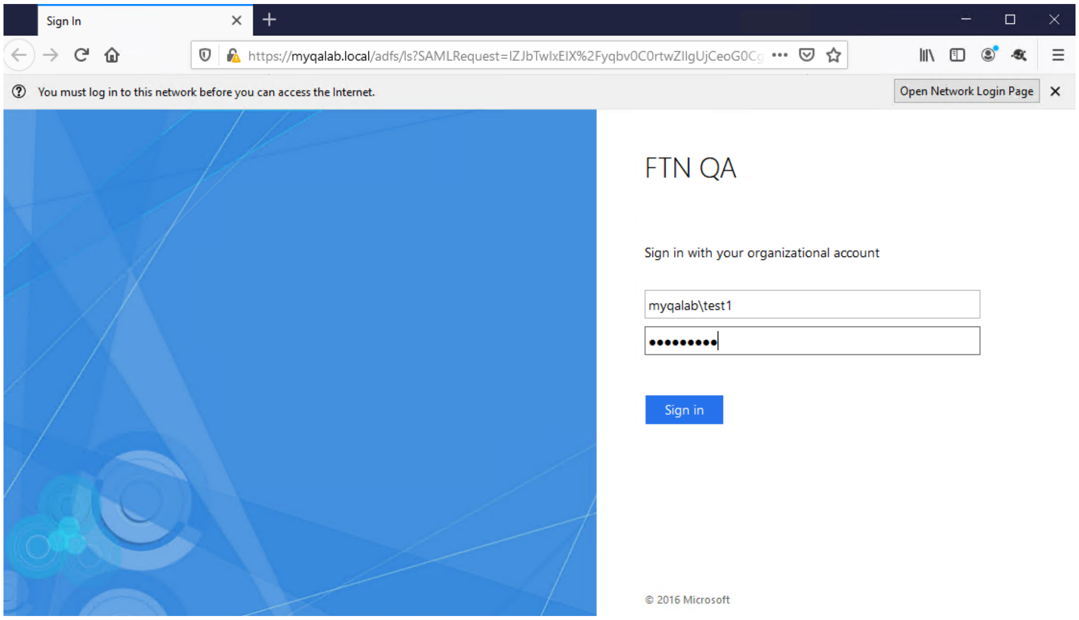
```
config authentication rule
    edit "saml"
        set srcaddr "all"
        set active-auth-method "saml"
    next
end
```

```
config authentication setting
    set captive-portal "fgt9.myqalab.local"
end
```

7. Configure the proxy policy:

```
config firewall proxy-policy
    edit 3
        set proxy explicit-web
        set dstintf "port9"
        set srcaddr "all"
        set dstaddr "all"
        set service "webproxy"
        set action accept
        set schedule "always"
        set logtraffic all
        set groups "ldap-group-saml"
        set utm-status enable
        set profile-protocol-options "protocol"
        set ssl-ssh-profile "deep-custom"
        set av-profile "av"
    next
end
```

When a user goes to www.google.com in a browser that is configured to use the FortiGate as a proxy, the IdP sign-in page appears.

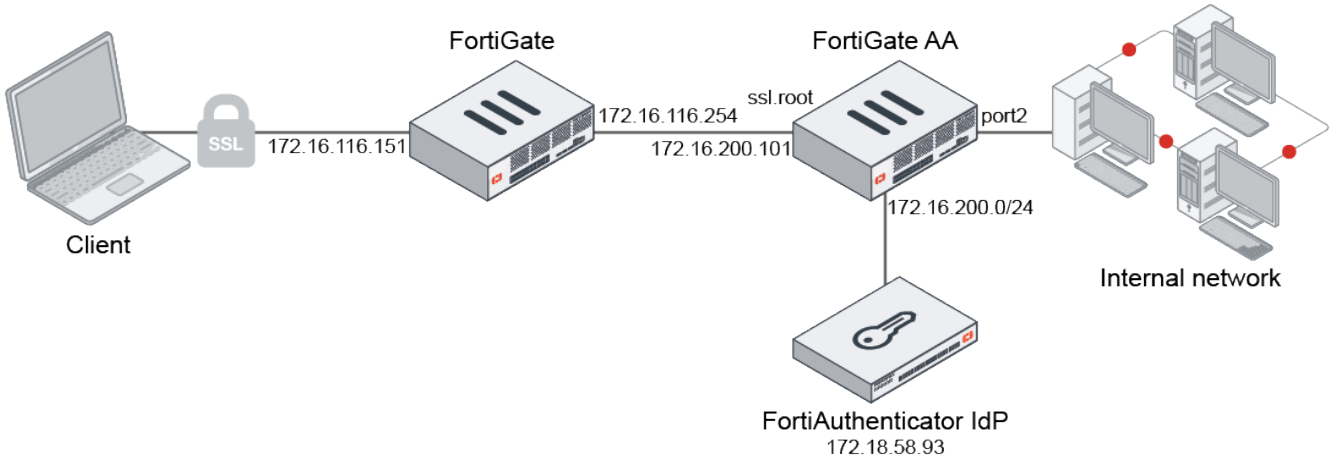


Sample log

```
7: date=2021-03-16 time=21:11:19 eventtime=1615954279072391030 tz="-0700" logid="0000000010"
type="traffic" subtype="forward" level="notice" vd="vdom1" srcip=10.1.100.143 srcport=53544
srcintf="port10" srcintfrole="undefined" dstcountry="United States" srccountry="Reserved"
dstip=173.194.219.99 dstport=443 dstintf="port9" dstintfrole="undefined"
sessionid=1751272387 service="HTTPS" wanoptapptype="web-proxy" proto=6 action="accept"
policyid=3 policytype="proxy-policy" poluuid="052ae158-7d40-51eb-c1d8-19235c4500c2"
trandisp="snat" transip=172.16.200.1 transport=14844 duration=268 user="test1@MYQALAB.local"
group="ldap-group-saml" authserver="ldap-10.1.100.198" wanin=345633 rcvdbyte=345633
wanout=13013 lanin=5098 sentbyte=5098 lanout=340778 appcat="unscanned"
```

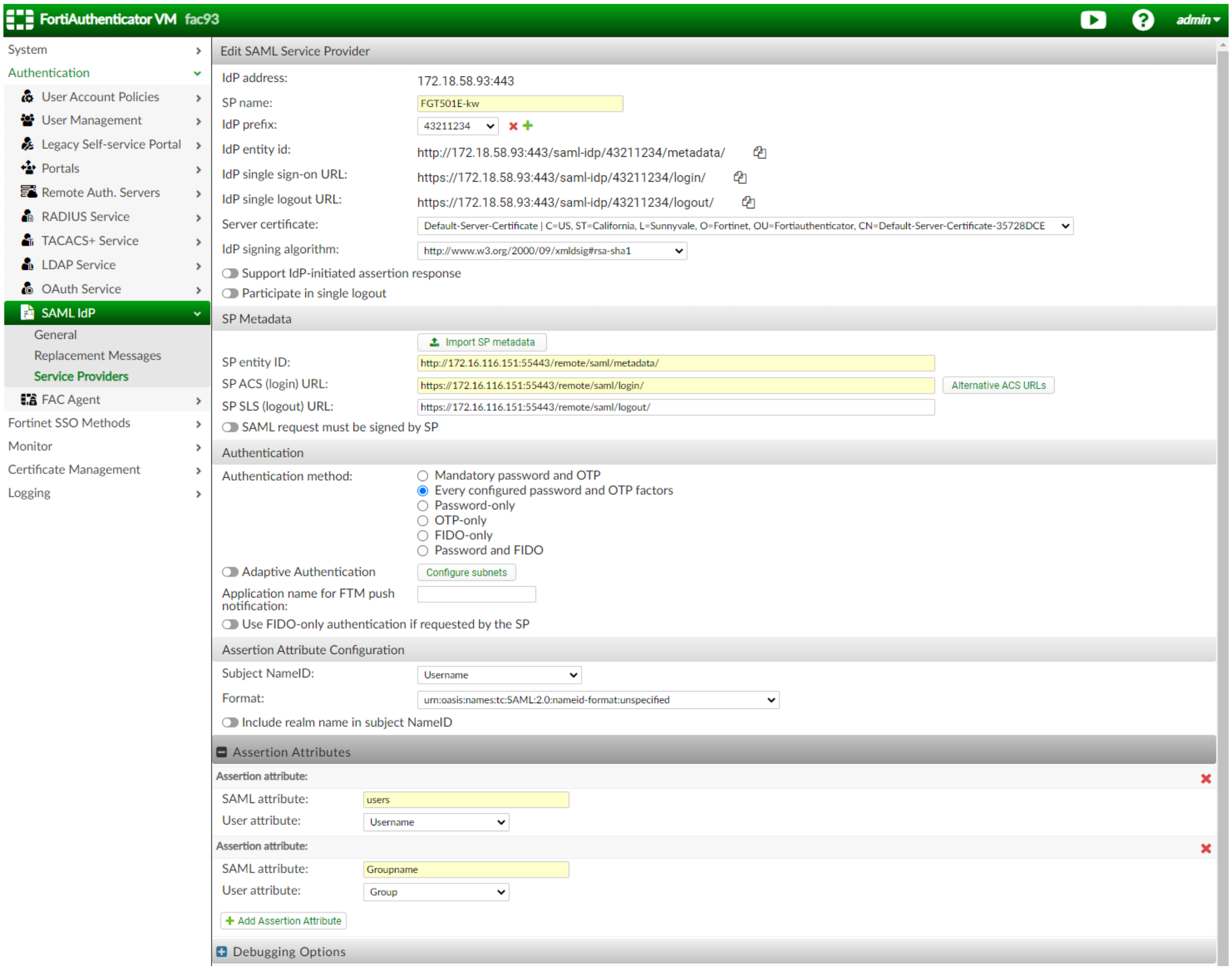
Configuring SAML SSO in the GUI

SAML single sign-on can be configured in the GUI under *User & Authentication* > *User Groups*. The GUI wizard helps generate the service provider (SP) URLs based on the supplied SP address. The SAML object that is created can be selected when defining new user groups.



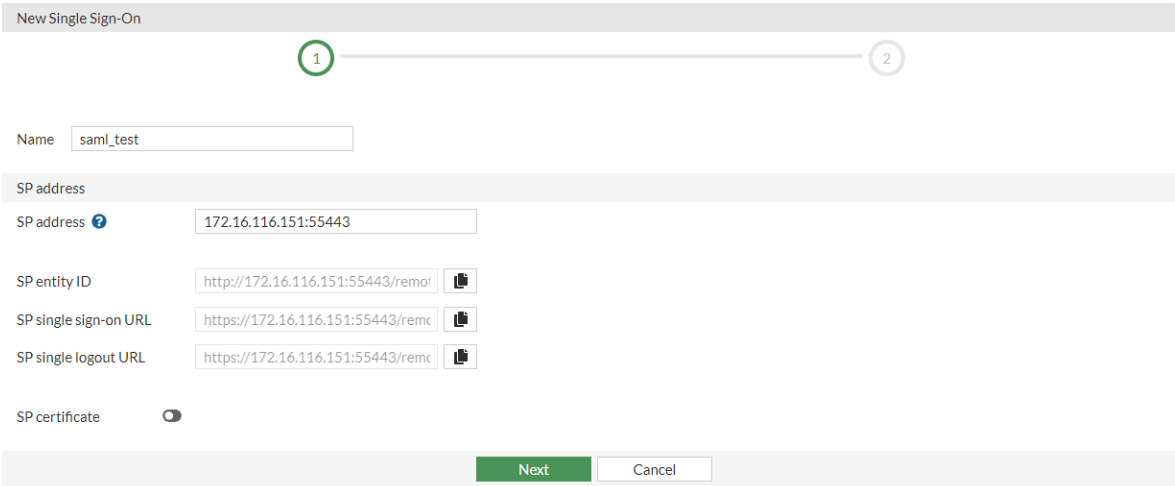
In this example, FortiGate AA is the inside firewall (172.16.200.101). The other FortiGate is the outside firewall that only does port forwarding from 172.16.116.151:55443 to 172.16.200.101:443. FortiGate AA is configured to allow full SSL VPN access to the network in port2. This SSL VPN portal allows users from the user group *saml_grp* and SAML server *saml_test* to log in. In this topology, a FortiAuthenticator acts as the SAML identity provider (IdP), while the FortiGate is the SAML SP. External users are directed to the FortiAuthenticator IdP login URL to authenticate. For more information about configuring a FortiAuthenticator as an IdP, see Service providers.

The FortiAuthenticator in this example has the following configuration:



To configure FortiGate AA as an SP:

1. Create a new SAML server entry:
   a. Go to *User & Authentication > Single Sign-On* and click *Create New*. The single-sign on wizard opens.
   b. Enter a name (*saml_test*). The other fields will automatically populate based on the FortiGate's WAN IP and port.



Click the icon beside the *SP entity ID*, *SP single sign-on URL*, and *SP single logout URL* fields to copy the text.

c. Click *Next*.

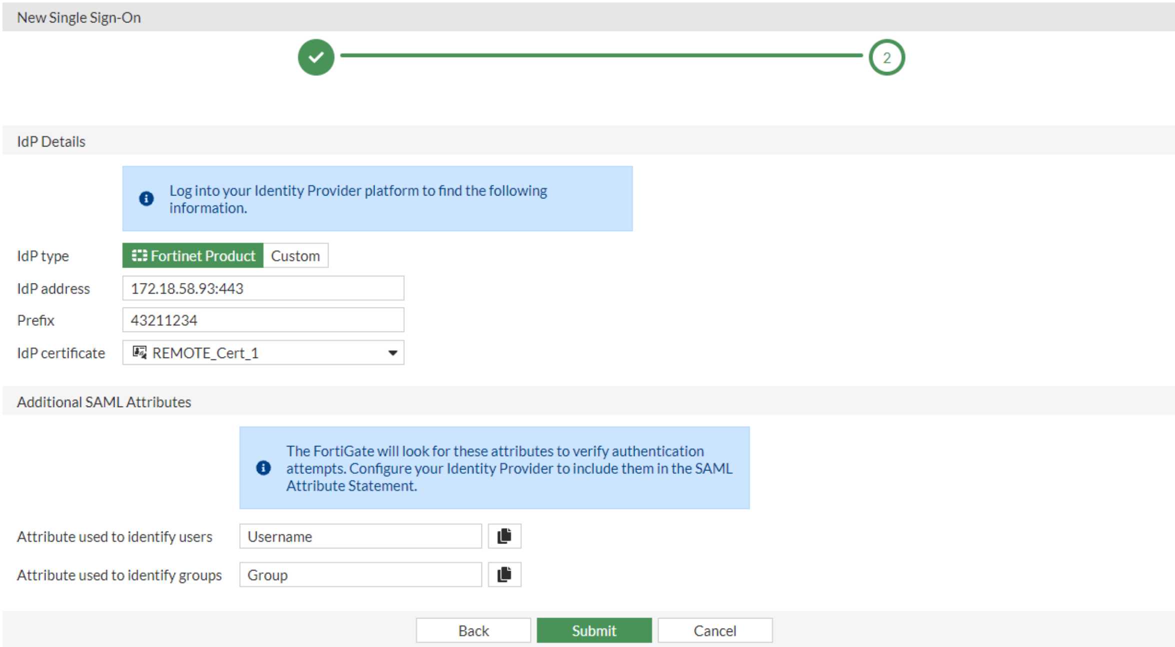
d. Enter the FortiAuthenticator IdP details:

IdP address	172.18.58.93:443
Prefix	43211234
IdP certificate	REMOTE_Cert_1

e. Enter the additional SAML attributes that will be used to verify authentication attempts:

Attribute used to identify users	Username
Attribute used to identify groups	Group

The IdP must be configured to include these attributes in the SAML attribute statement. In FortiAuthenticator, this is configured in the *Assertion Attributes* section.



f. Click *Submit*.
The following is created in the backend:

```
config user saml
    edit "saml_test"
        set cert "fgt_gui_automation"
        set entity-id "http://172.16.116.151:55443/remote/saml/metadata/"
        set single-sign-on-url "https://172.16.116.151:55443/remote/saml/login/"
        set single-logout-url "https://172.16.116.151:55443/remote/saml/logout/"
        set idp-entity-id "http://172.18.58.93:443/saml-idp/43211234/metadata/"
        set idp-single-sign-on-url "https://172.18.58.93:443/saml-
idp/43211234/login/"
        set idp-single-logout-url "https://172.18.58.93:443/saml-
idp/43211234/logout/"
        set idp-cert "REMOTE_Cert_1"
        set user-name "Username"
        set group-name "Group"
        set digest-method sha1
    next
end
```

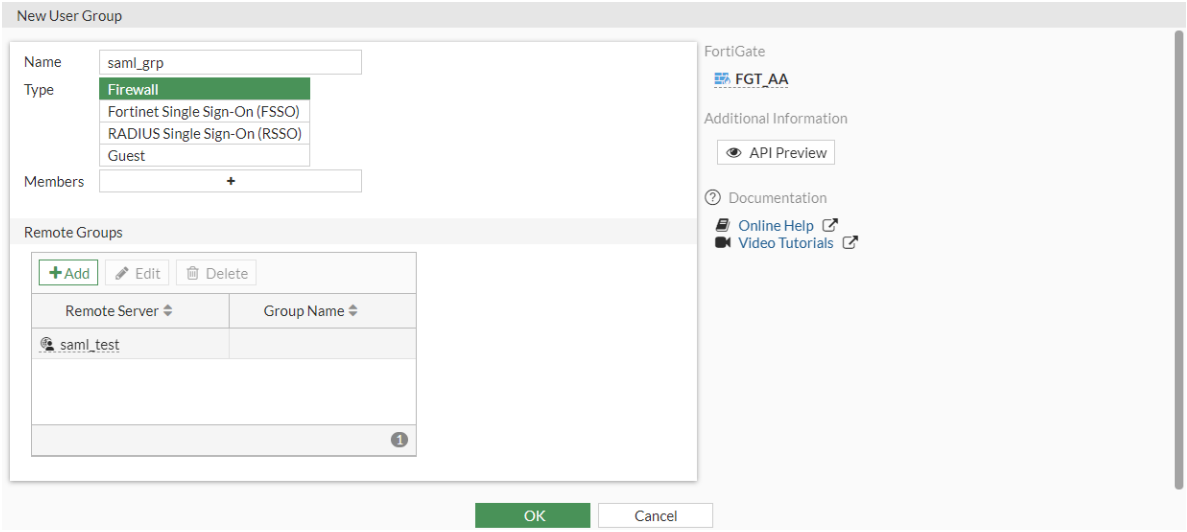
2. Create the SAML group:

a. Go to *User & Authentication > User Groups* and click *Create New*.

b. Enter a name, *saml_grp*.

c. In the *Remote Groups* table, click *Add*.

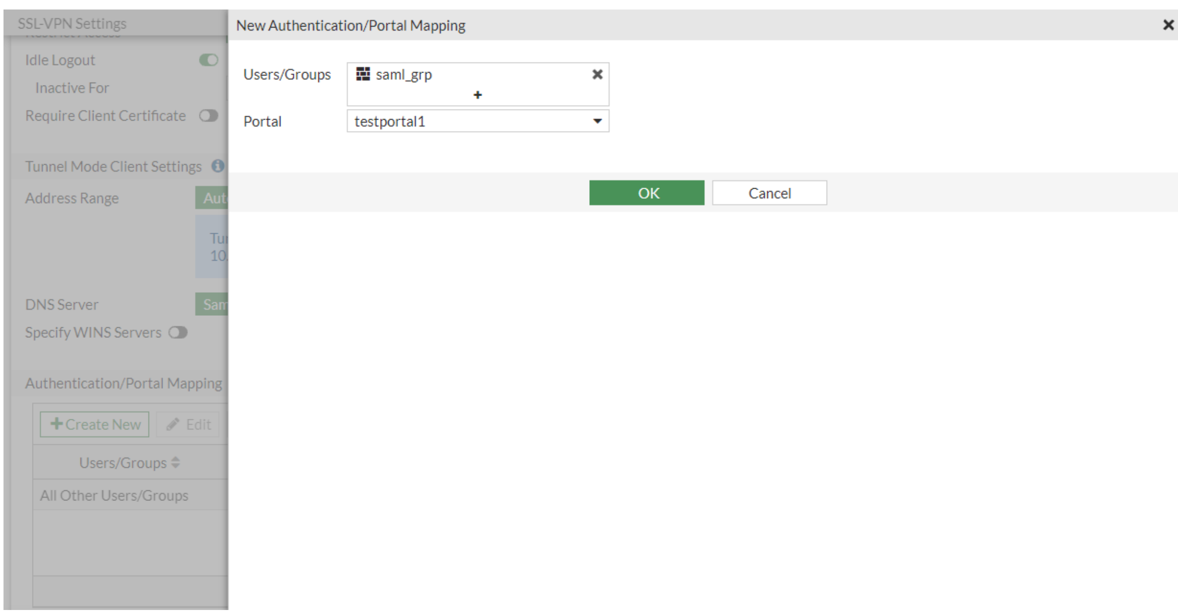
d. In the *Remote Server* dropdown, select *saml_test* and click *OK*.



e. Click *OK*.
The following is created in the backend:

```
config user group
    edit "saml_grp"
        set member "saml_test"
    next
end
```

3. Add the SAML group in the SSL VPN settings:
   a. Go to *VPN > SSL-VPN Settings*.
   b. In the *Authentication/Portal Mapping* table, click *Create New*.
   c. For *Users/Groups*, click the + and select *saml_grp*.
   d. Select the *Portal* (*testportal1*).
   e. Click *OK*.



- **f.** Click *Apply*.

4. Configure the firewall policy:
   - **a.** Go to *Policy & Objects* > *Firewall Policy* and click *Create New*.
   - **b.** Enter the following:

Incoming Interface	ssl.root
Outgoing Interface	port2
Source	all, saml_grp, saml_test

   - **c.** Configure the other settings as needed.
   - **d.** Click *OK*.
5. On the client, log in with SAML using the SSL VPN web portal.

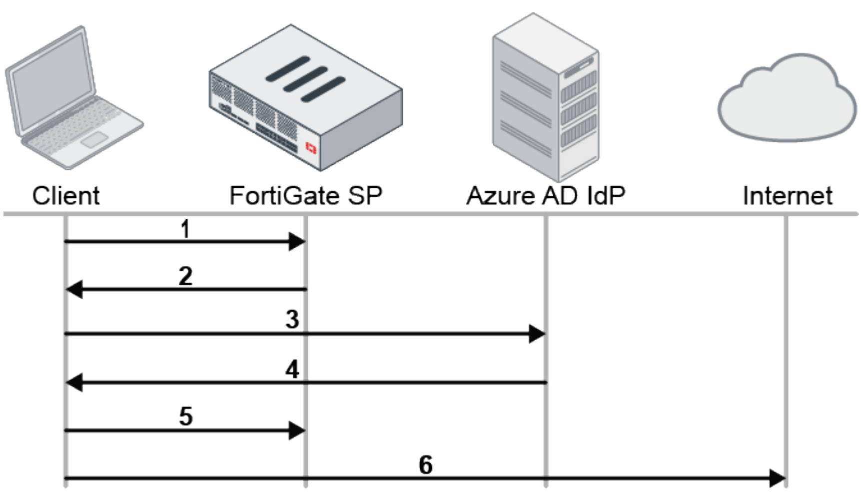
> If you are using FortiClient for tunnel mode access, enable *Enable Single Sign On (SSO) for VPN Tunnel* in the *SSL-VPN* connection settings to use the SAML log in. See Configuring an SSL VPN connection for more information.

6. In FortiOS, go to *Dashboard* > *Network* and click the *SSL-VPN* widget to expand to full view and verify the connection information.

Outbound firewall authentication with Azure AD as a SAML IdP

In this example, users are managed through Microsoft Azure Active Directory (AD). The FortiGate is configured for SSO firewall authentication for outbound traffic, with authentication performed by the Azure AD as a SAML identity provider (IdP).

The SAML interaction occurs as follows:

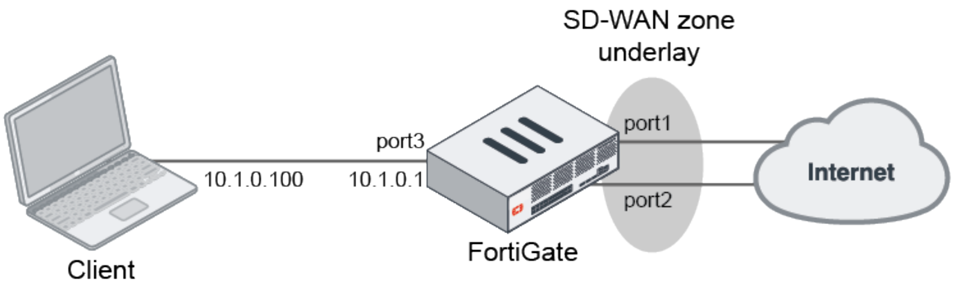


1. The user initiates web traffic to the internet.
2. The FortiGate redirects to the local captive portal port (default is 1003), then redirects the user to the SAML IdP.
3. The user connects to the Microsoft log in page for the SAML authentication request.
4. The SAML IdP sends the SAML assertion containing the user and group.
5. The browser forwards the SAML assertion to the SAML SP.
6. If the user and group are allowed by the FortiGate, the user is allowed to access the internet.

In this example environment, a user is added in the Azure AD belonging to the security group called Firewall.

- Username: John Locus
- User login: jlocus@azure.kldocs.com
- Group: Firewall (ID 62b699ce-4f80-48c0-846e-c1dfde2dc667)

The goal is to allow users in the Firewall group to access the internet after passing firewall authentication.



Configuring the Azure AD

The following Azure AD configuration demonstrates how to add the FortiGate as an enterprise non-gallery application. This application provides SAML SSO connectivity to the Azure AD IdP. Some steps are performed concurrently on the FortiGate.

This example is configured with an Azure AD free-tier directory. There may be limitations to managing users in Azure in this tier that are not limited in other tiers. Consult the Microsoft Azure AD documentation for more information.

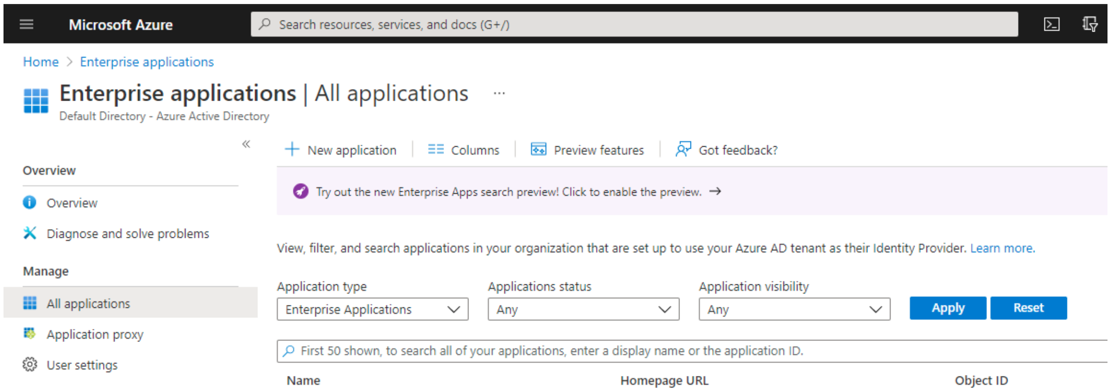
There are three steps to configure the Azure AD:

1. Create a new enterprise application.
2. Configure the SAML SSO settings on the application and FortiGate.
3. Assign Azure AD users and groups to the application.

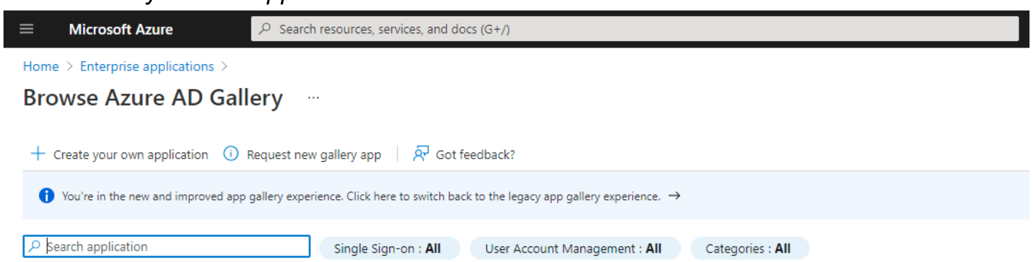
To create a new enterprise application:

1. Log in to the Azure portal.
2. In the Azure portal menu, click *Azure Active Directory*.
3. In the left-side menu go *Manage* > *Enterprise applications*.

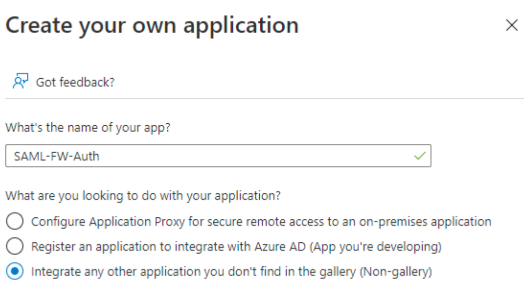
4. Click *New application*.



5. Click *Create your own application*.



6. Enter a name for the application (*SAML-FW-Auth*) and select *Integrate any other application you don't find in the gallery (Non-gallery)*.

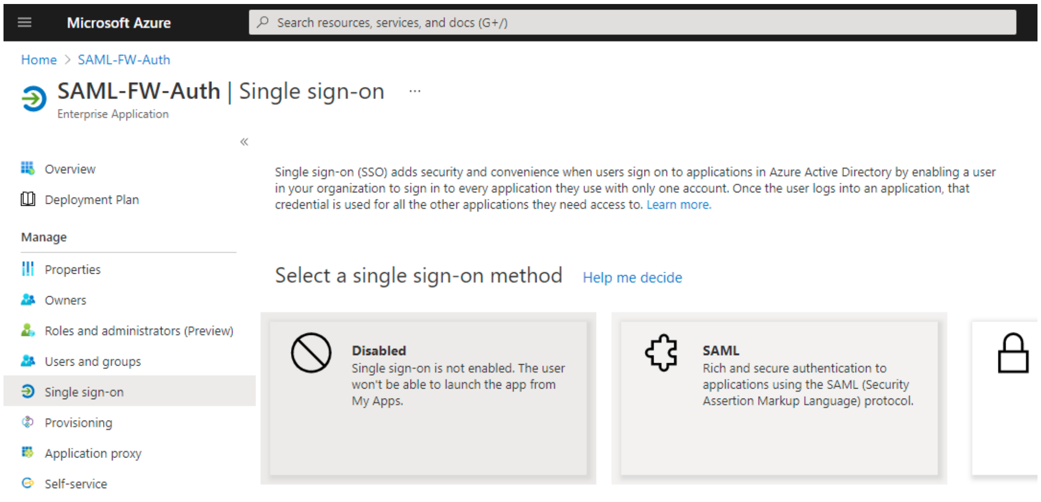


7. Click *Create*.

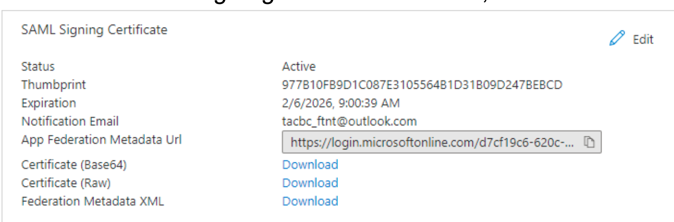
To configure the SAML SSO settings on the application and FortiGate:

This procedure requires going back and forth between Azure and the FortiGate GUI. Leave the FortiGate GUI open for the entire procedure.

1. On the *Enterprise Application Overview* page, go to *Manage* > *Single sign-on* and select *SAML* as the single sign-on method.



2. Under the *SAML Signing Certificate* section, download the Base64 certificate.

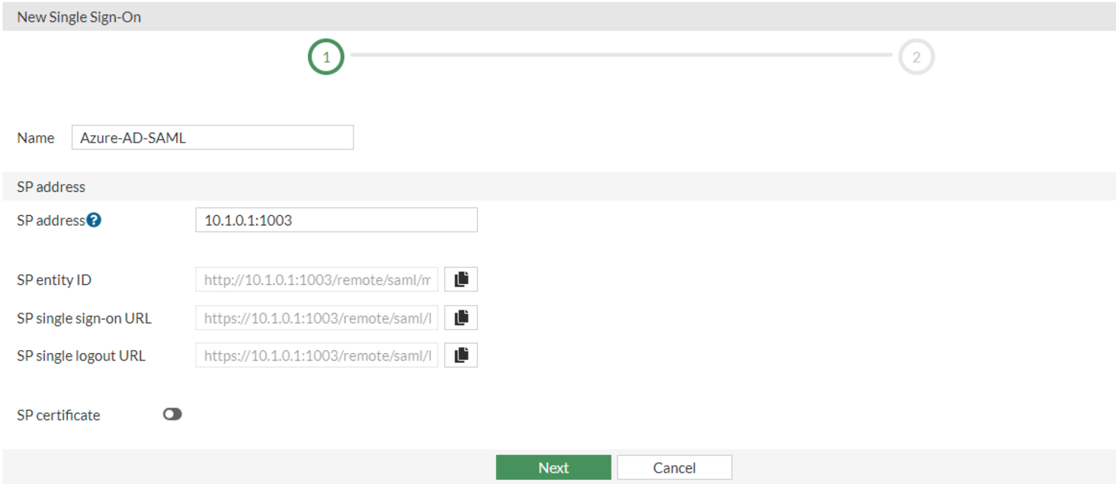


3. Import the certificate from Azure on the FortiGate as the IdP certificate:
   a. Go to *System* > *Certificates* and click *Create/Import* > *Remote Certificate*.
   b. Upload the certificate from Azure and click *OK*. The new certificate appears under the *Remote Certificate* section with the name *REMOTE_Cert_(N)*.
   c. Optionally, rename the certificate in the CLI to give it a more recognizable name:

```
config vpn certificate remote
    rename REMOTE_Cert_3 to AZURE_AD_SAML_FW
end
```

4. The *Basic SAML Configuration* section in Azure describes the SAML SP entity and links that Azure will reference. Configure these settings on the FortiGate by creating a new SAML server object and defining the SP address. The SP (IP or FQDN) address should be accessible by the user who is authenticating against the firewall. The port used should match the port used by the FortiGate firewall authentication captive portal. By default, this is port 1003 for HTTPS. A captive portal does not need to be configured separately.
   a. Go to *User & Authentication* > *Single Sign-On* and click *Create New*.
   b. Enter a *Name* for the SAML object, *Azure-AD-SAML*.

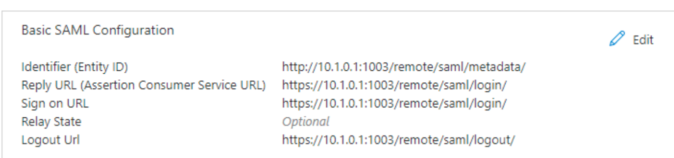
c. Enter the *SP address*, *10.1.0.1:1003*. The three SP URLs are automatically populated.



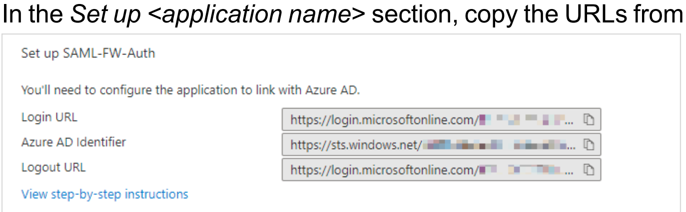
5. In Azure on the *Set up Single Sign-On with SAML* page, copy the following URLs from the FortiGate to the *Basic SAML Configuration* section:

From FortiGate	To Azure field
SP entity ID (http://10.1.0.1:1003/remote/saml/metadata/)	*Identifier (Entity ID)*, set to *Default*
SP single sign-on URL (https://10.1.0.1:1003/remote/saml/login//)	*Reply URL* and *Sign on URL*
SP single logout URL (https://10.1.0.1:1003/remote/saml/logout/)	*Logout URL*

6. Click *Save*.



7. In the *Set up <application name>* section, copy the URLs from Azure to the FortiGate in the *IdP Details* section:



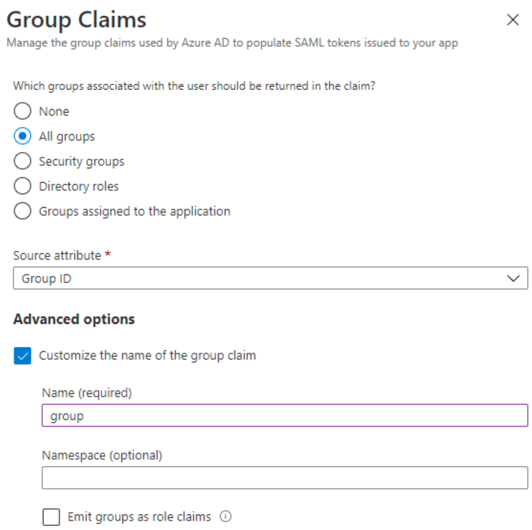
a. On the FortiGate, click *Next*.

b. For *IdP type*, select *Custom* and copy the following from Azure to the corresponding field:

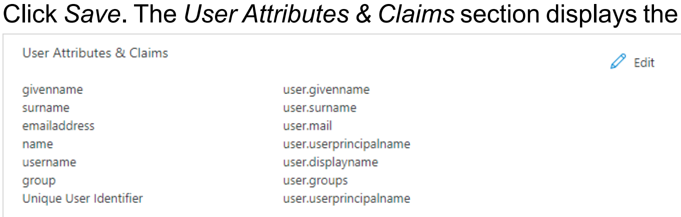
From Azure	To FortiGate field
Azure AD Identifier	*IdP entity ID*
Login URL	*IdP single sign-on URL*
Logout URL	*IdP single logout URL*

c. For *IdP certificate*, select the remote certificate imported earlier.

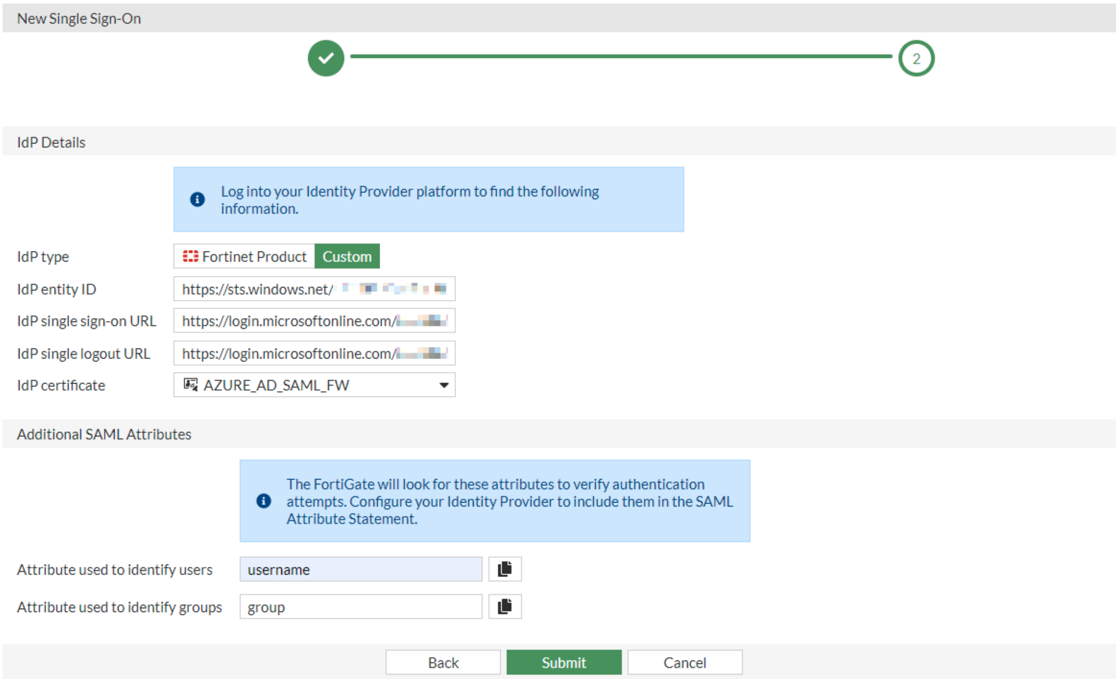
8. In Azure, edit the *User Attributes & Claims* section. The attributes are returned in the SAML assertion, which the FortiGate uses to verify the user and group. Configuring group matching is optional.
    a. Click *Add new claim*, name it *username*, and set the *Source attribute* to *user.displayname*. The source attribute can be any of the related username fields. The value of the username returned to the FortiGate will be used in logs and monitors to identify the user.
    b. Click *Save*.
    c. Click *Add a group claim* and in the *Group Claims* pane, select *All groups*.
    d. In *Advanced Options*, select *Customize the name of the group claim*. Set the name to *group*.



    e. Click *Save*. The *User Attributes & Claims* section displays the update settings.



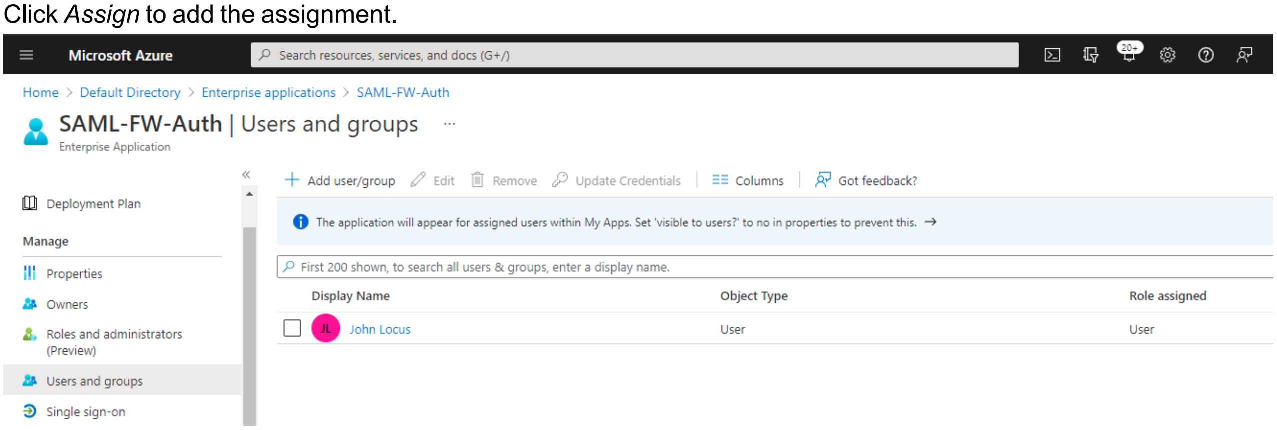
9. On the FortiGate, update the *Additional SAML Attributes* section with the username and group created in Azure:
    a. For *Attribute used to identify users*, enter *username*.
    b. For *Attribute used to identify groups*, enter *group*.



c. Click *Submit*.

To assign Azure AD users and groups to the application:

1. In Azure, go to *Manage* > *Users and groups* and click *Add user/group*.
2. Click *Users* to select the users or groups (*John Locus* is selected in this example).
3. Click *Assign* to add the assignment.



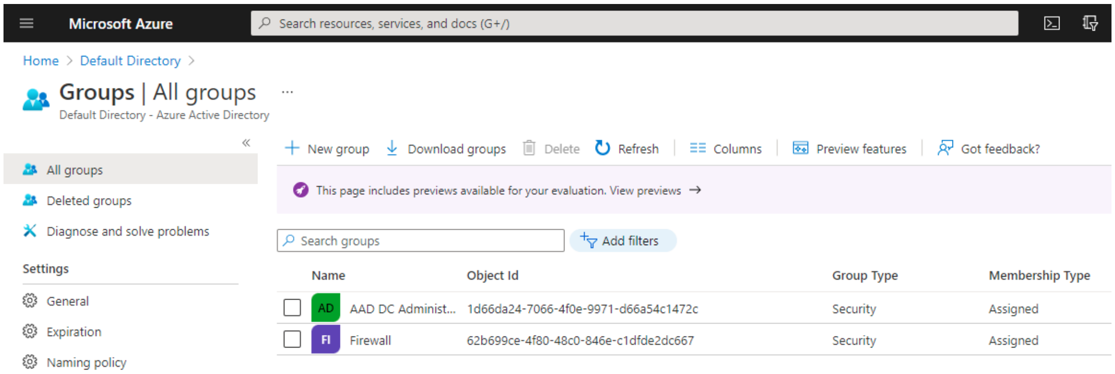
Configuring the FortiGate

The user group, user authentication settings, and firewall policies must be configured on the FortiGate.

Configuring the user group

A user group named *Azure-FW-Auth* is created with the member *Azure-AD-SAML*.

Configuring group matching is optional, and the *Object ID* from Azure is needed for the `config match` settings. In the Azure default directory, go to *Manage* > *Groups* and locate the *Object ID* for the *Firewall* group.



To configure the user group:

```
config user group
    edit "Azure-FW-Auth"
        set member "Azure-AD-SAML"
        config match
            edit 1
                set server-name "Azure-AD-SAML"
                set group-name "62b699ce-4f80-48c0-846e-c1dfde2dc667"
            next
        end
    next
end
```

Configuring the user authentication setting

When a user initiates traffic, the FortiGate will redirect the user to the firewall authentication captive portal before redirecting them to the SAML IdP portal. After the SAML IdP responds with the SAML assertion, the user is again redirected to the firewall authentication captive portal. If the firewall portal's certificate is not trusted by the user, they will receive a certificate warning. Use a custom certificate that the user trusts to avoid the certificate warning.

To configure a custom certificate:

1. Go to *User & Authentication* > *Authentication Settings*.
2. For *Certificate*, select the custom certificate. The custom certificate's SAN field should have the FQDN or IP from the SP URL.

Alternatively, assigning a CA certificate allows the FortiGate to automatically generate and sign a certificate for the portal page. This will override any assigned server certificate. In this example, the built-in Fortinet_CA_SSL is used.

To assign a CA certificate:

1. Edit the user setting:

```
config user setting
    set auth-ca-cert "Fortinet_CA_SSL"
end
```

2. Go to *System* > *Certificates* and download the certificate.
3. Install the certificate into the client's certificate store.

Configuring the firewall policies

Firewall policies must be configured to apply user authentication and still allow users behind the FortiGate to access the Microsoft log in portal without authentication.

To configure the firewall policies:

1. Configure a policy to allow traffic to the Microsoft Azure internet service:
   a. Go to *Policy & Objects* > *Firewall Policy* and click *Create New*.
   b. Enter the following:

Name	*LAN-to-AuthPortal*
Incoming Interface	*port3*
Outgoing Interface	*Underlay*
Source	*all*
Destination	*Microsoft-Azure* (under *Internet Service*)
Schedule	*always*
Service	*ALL*
Action	*ACCEPT*
NAT	Enable and select *NAT*.
Log Allowed Traffic	Enable and select *All Sessions*.

   c. Configure the other settings as needed.
   d. Click *OK*.

2. Configure a policy to apply user authentication:
   a. Click *Create New* and enter the following:

Name	*LAN-auth-policy*
Incoming Interface	*port3*
Outgoing Interface	*Underlay*
Source	*all*, *Azure-FW-Auth*
Destination	*all*
Schedule	*always*
Service	*ALL*
Action	*ACCEPT*
NAT	Enable and select *NAT*.
Log Allowed Traffic	Enable and select *All Sessions*.

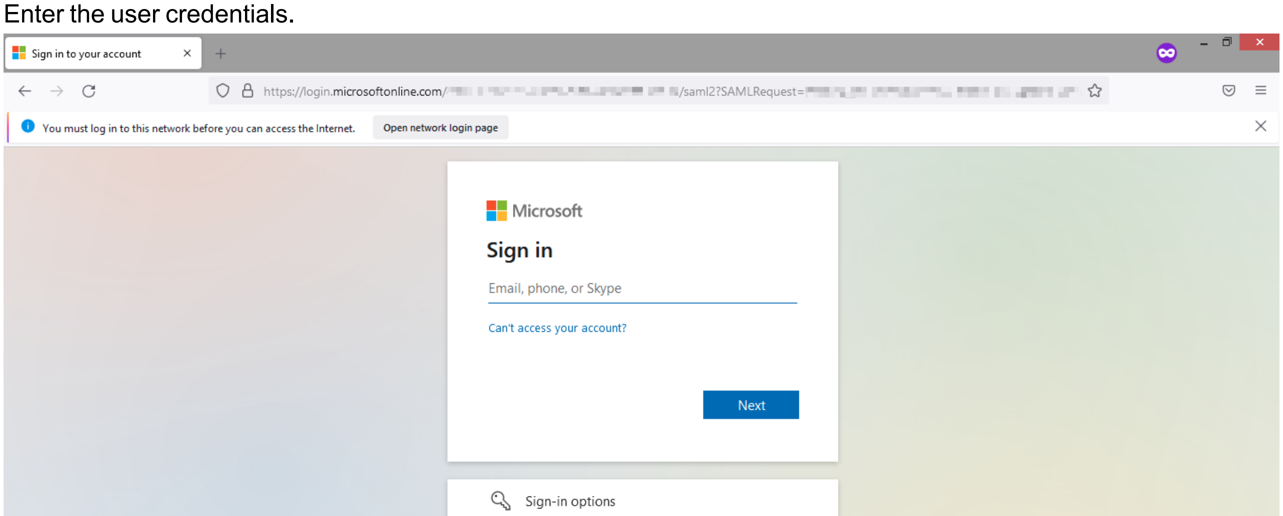
   b. Configure the other settings as needed.
   c. Click *OK*.

Connecting from the client

When the client connects to the internet from a browser, they will be redirected to the Microsoft log in page to authenticate against the Azure AD. The FortiGate's authentication portal certificate should be installed on the client.

To connect from the client:

1. On the client, open a browser (such as Firefox) and go to a website. The user is redirected to the Microsoft log in page.
2. Enter the user credentials.



3. If the log in attempt is successful, the user is allowed to access the internet

Viewing logs and diagnostics

To verify user logins, go to the *Dashboard* > *Users & Devices* > *Firewall Users* widget, or enter the following in the CLI:

```
# diagnose firewall auth list
10.1.0.100, John Locus
        src_mac: 02:09:0f:00:03:03
        type: fw, id: 0, duration: 152, idled: 7
        expire: 292, allow-idle: 300
        server: Azure-AD-SAML
        packets: in 2097 out 932, bytes: in 2208241 out 143741
        group_id: 2
        group_name: Azure-FW-Auth
----- 1 listed, 0 filtered ------
```

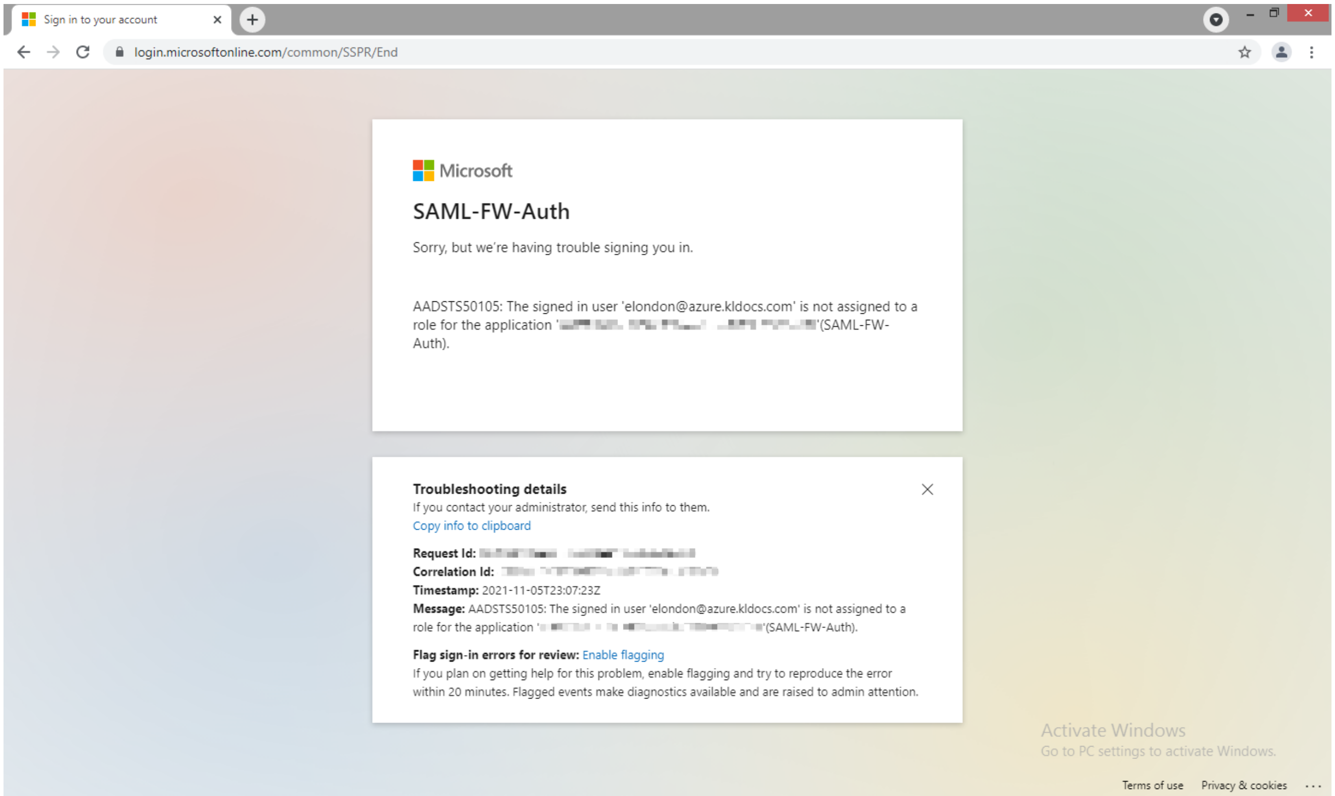
To verify user login logs, go to *Log & Report* > *System Events* and select the *User Events* card, or enter the following in the CLI:

```
# execute log filter category event
# execute log filter field subtype user
# execute log display
17 logs found.
10 logs returned.
7: date=2021-09-30 time=09:49:25 eventtime=1633020565577584390 tz="-0700" logid="0102043039"
type="event" subtype="user" level="notice" vd="root" logdesc="Authentication logon"
srcip=10.1.0.100 user="John Locus" authserver="Azure-AD-SAML" action="auth-logon"
status="logon" msg="User John Locus added to auth logon"

8: date=2021-09-30 time=09:49:25 eventtime=1633020565577075629 tz="-0700" logid="0102043008"
type="event" subtype="user" level="notice" vd="root" logdesc="Authentication success"
srcip=10.1.0.100 dstip=10.1.0.1 policyid=11 interface="port3" user="John Locus"
group="Azure-FW-Auth" authproto="HTTPS(10.1.0.100)" action="authentication" status="success"
reason="N/A" msg="User John Locus succeeded in authentication"
```

If user authentication is successful in Azure AD, but their group does not match the one defined in the FortiGate user group, the user will receive a *Firewall Authentication Failed* message in the browser. A log is also recorded:

```
# execute log filter category event
# execute log filter field subtype user
# execute log display
1: date=2021-09-30 time=10:39:35 eventtime=1633023575381139214 tz="-0700" logid="0102043009"
type="event" subtype="user" level="notice" vd="root" logdesc="Authentication failed"
srcip=10.1.0.100 dstip=10.1.0.1 policyid=11 interface="port3" user="Adam Thompson"
group="N/A" authproto="HTTPS(10.1.0.100)" action="authentication" status="failure"
reason="No matched SAML user or group name in auth resp" msg="User Adam Thompson failed in
authentication"
```

If a user receives the following error message, this means the user is not assigned to the enterprise application *SAML-FW-Auth* in Azure.



To troubleshoot SAML issues:

```
# diagnose debug application samld -1
# diagnose debug enable
```

Authentication settings

You can configure general authentication settings, including timeout, protocol support, and certificates.

You cannot customize FTP and Telnet authentication replacement messages.

To configure authentication settings using the GUI:

1. Go to *User & Authentication > Authentication Settings*.
2. Configure the following settings:

Setting	Description
Authentication Timeout	Enter the desired timeout in minutes. You can enter a number between 1 and 1440 (24 hours). The authentication timeout controls how long an authenticated connection can be idle before the user must reauthenticate. The default value is 5.
Protocol Support	Select the protocols to challenge during firewall user authentication. When you enable user authentication within a security policy, the authentication challenge is normally issued for any of four protocols, depending on the connection protocol: • HTTP (you can set this to redirect to HTTPS) • HTTPS • FTP • Telnet The protocols selected here control which protocols support the authentication challenge. Users must connect with a supported protocol first so they can subsequently connect with other protocols. If HTTPS is selected as a protocol support method, it allows the user to authenticate with a customized local certificate. When you enable user authentication within a security policy, FortiOS challenges the security policy user to authenticate. For user ID and password authentication, the user must provide their username and password. For certificate authentication (HTTPS or HTTP redirected to HTTPS only), you can install customized certificates on the unit and the user can also install customized certificates on their browser. Otherwise, users see a warning message and must accept a default Fortinet certificate. The network user's web browser may deem the default certificate invalid.
Certificate	If using HTTPS protocol support, select the local certificate to use for authentication. This is available only if *HTTPS* and/or *Redirect HTTP Challenge to a Secure Channel (HTTPS)* are selected.

To configure authentication settings using the CLI:

```
config user setting
   set auth-timeout 5
   set auth-type ftp http https telnet
   set auth-cert Fortinet_Factory
end
```

FortiTokens

FortiTokens are security tokens used as part of a multi-factor authentication (MFA) system on FortiGate and FortiAuthenticator. A security token is a 6-digit or 8-digit (configurable) one-time password (OTP) that is used to authenticate one's identity electronically as a prerequisite for accessing network resources. FortiToken is available as either a mobile or a physical (hard) token. Mobile tokens can be purchased as a license, or consumed with points as part of the FortiToken Cloud service.

FortiToken Mobile and physical FortiTokens store their encryption seeds on the cloud. FortiToken Mobile seeds are generated dynamically when the token is provisioned. They are always encrypted whether in motion or at rest.

You can only register FortiTokens to a single FortiGate or FortiAuthenticator for security purposes. This prevents malicious third parties from making fraudulent requests to hijack your FortiTokens by registering them on another FortiGate or FortiAuthenticator. If re-registering a FortiToken Mobile or Hard Token on another FortiGate is required, you must contact Fortinet Customer Support.

Common usage for FortiTokens includes:

- Applying MFA to a VPN dialup user connecting to the corporate network
- Applying MFA to FortiGate administrators
- Applying MFA to firewall authentication and captive portal authentication

---

The MFA process commonly involves:

- **Something you know**: User password
- **Something you have**: The FortiToken OTP

A third factor of authentication is added to the authentication process:

- **Something you are**: Your fingerprint or face

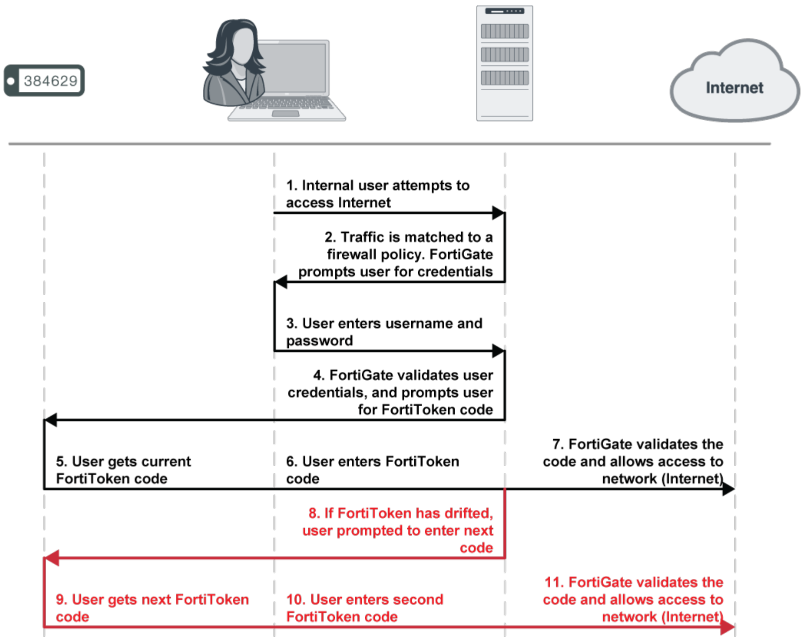
To enable the third factor, refer to the Activating FortiToken Mobile on a mobile phone on page 1886 section.

The following illustrates the FortiToken MFA process:

1. The user attempts to access a network resource.
2. FortiOS matches the traffic to an authentication security policy and prompts the user for their username and password.
3. The user enters their username and password.
4. FortiOS verifies their credentials. If valid, it prompts the user for the FortiToken code.
5. The user views the current code on their FortiToken. They enter the code at the prompt.
6. FortiOS verifies the FortiToken code. If valid, it allows the user access to network resources.

If the FortiToken has drifted, the following must take place for the FortiToken to resynchronize with FortiOS:

1. FortiOS prompts the user to enter a second code to confirm.
2. The user gets the next code from the FortiToken. They enter the code at the prompt.
3. FortiOS uses both codes to update its clock to match the FortiToken.



This section includes the following topics to quickly get started with FortiTokens:

- FortiToken Mobile quick start on page 1883
- FortiToken Cloud on page 1891
- Registering hard tokens on page 1891
- Managing FortiTokens on page 1893
- FortiToken Mobile Push on page 1895
- Troubleshooting and diagnosis on page 1897

FortiToken Mobile quick start

FortiToken Mobile is an OATH compliant, event- and time-based one-time password (OTP) generator for mobile devices. It provides an easy and flexible way to deploy and provision FortiTokens to your end users through mobile devices. FortiToken Mobile produces its OTP codes in an application that you can download onto your Android or iOS mobile device without the need for a physical token.

You can download the free FortiToken Mobile application for Android from the Google Play Store, and for iOS from the Apple App Store.

This section focuses on quickly getting started and setting up FortiToken Mobile for use on a FortiGate:

- Registering FortiToken Mobile on page 1883
- Provisioning FortiToken Mobile on page 1884
- Activating FortiToken Mobile on a mobile phone on page 1886
- Applying multi-factor authentication on page 1890

Registering FortiToken Mobile

To deploy FortiToken Mobile for your end users, you must first register the tokens on your FortiGate. After registering the tokens, you can assign them to your end users.

Each FortiGate comes with two free FortiToken Mobile tokens. These tokens should appear under *User & Authentication > FortiTokens*. If no tokens appear, you may import them. Ensure that your FortiGate is registered and has internet access to connect to the FortiToken servers to import the tokens.

To import FortiTokens from the FortiGate GUI:

1. Go to *User & Authentication > FortiTokens*.
2. Click the *Import Free Trial Tokens* icon at the top. The two free tokens are imported.

To import FortiTokens from the FortiGate CLI:

```
# execute fortitoken-mobile import 0000-0000-0000-0000-0000
# show user fortitoken
```

If only one free token appears, you can first delete that token and then follow the procedure to import the two free tokens from either the GUI or the CLI.

If you have the FortiToken Mobile redemption certificate, you can register FortiToken Mobile on a FortiGate.

To register FortiToken Mobile from the FortiGate GUI:

1. Go to *User & Authentication > FortiTokens* and click *Create New*. The *New FortiToken* dialog appears.
2. For the *Type* field, select *Mobile Token*.
3. Locate the 20-digit code on the redemption certificate and type it in the *Activation Code* field.
4. Click *OK*. The token is successfully registered.

If you attempt to add invalid FortiToken serial numbers, there is no error message. FortiOS does not add invalid serial numbers to the list.

To register FortiToken Mobile from the FortiGate CLI:

```
# execute fortitoken-mobile import <20-digit activation code>
# show user fortitoken
```

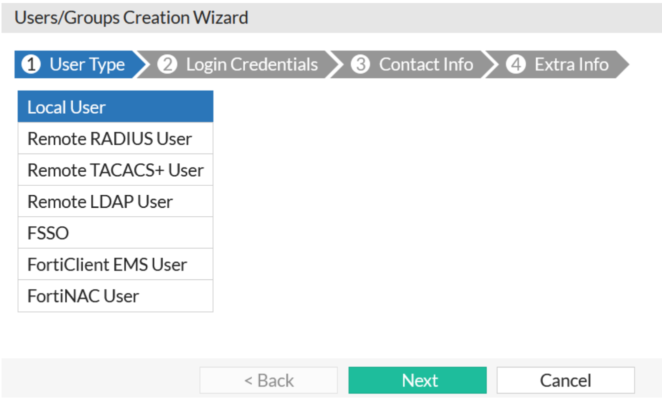
FortiToken Mobile stores its encryption seeds on the cloud. You can only register it to a single FortiGate or FortiAuthenticator.

Provisioning FortiToken Mobile

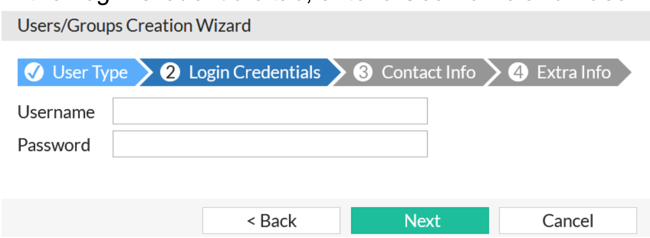
Once registered, FortiTokens need to be provisioned for users before they can be activated. In this example, you will provision a mobile token for a local user. Similar steps can be taken to assign FortiTokens to other types of users.

To create a local user and assign a FortiToken in the FortiGate GUI:

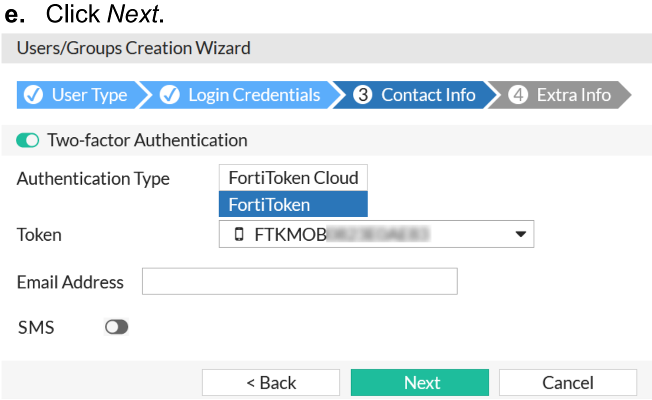
1. Go to *User & Authentication > User Definition*, and click *Create New*. The *Users/Groups Creation Wizard* appears.
2. In the *User Type* tab, select *Local User*, and click *Next*.



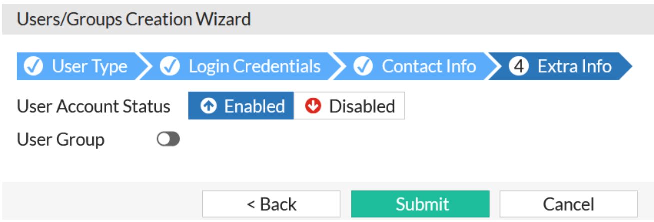
3. In the *Login Credentials* tab, enter a *Username* and *Password* for the user, and click *Next*.



4. In the *Contact Info* tab:
   a. Enable the *Two-factor Authentication* toggle.
   b. Select *FortiToken* for *Authentication Type*.
   c. Select a *Token* to assign to the user from the drop-down list.
   d. Enter the user's email address in the *Email Address* field. This is the email where the user will receive the QR code for activation of the FortiToken.
   e. Click *Next*.



5. In the *Extra Info* tab, make sure the *User Account Status* field is set to Enabled. You can also optionally assign the user to a user group by enabling the *User Group* toggle.



6. Click *Submit*. An activation code should be sent to the created user by email or SMS, depending upon the delivery method configured above.

FortiGate has the *Email Service* setting configured using the server *notifications.fortinet.net* by default. To see configuration, go to *System* > *Settings* > *Email Service*.

The activation code expires if not activated within the 3-day time period by default. However, the expiry time period is configurable.

To configure the time period (in hours) for FortiToken Mobile, using the CLI:

```
config system global
   set two-factor-ftm-expiry <1-168>
end
```

To resend the email or SMS with the activation code, refer to the Managing FortiTokens on page 1893 section.

Activating FortiToken Mobile on a mobile phone

After your system administrator provisions your token, you receive a notification with an activation code and expiry date via SMS or email. If you do not activate your token by the expiry date, you must contact your system administrator so that they can reassign your token for activation.

Platforms that support FortiToken Mobile:

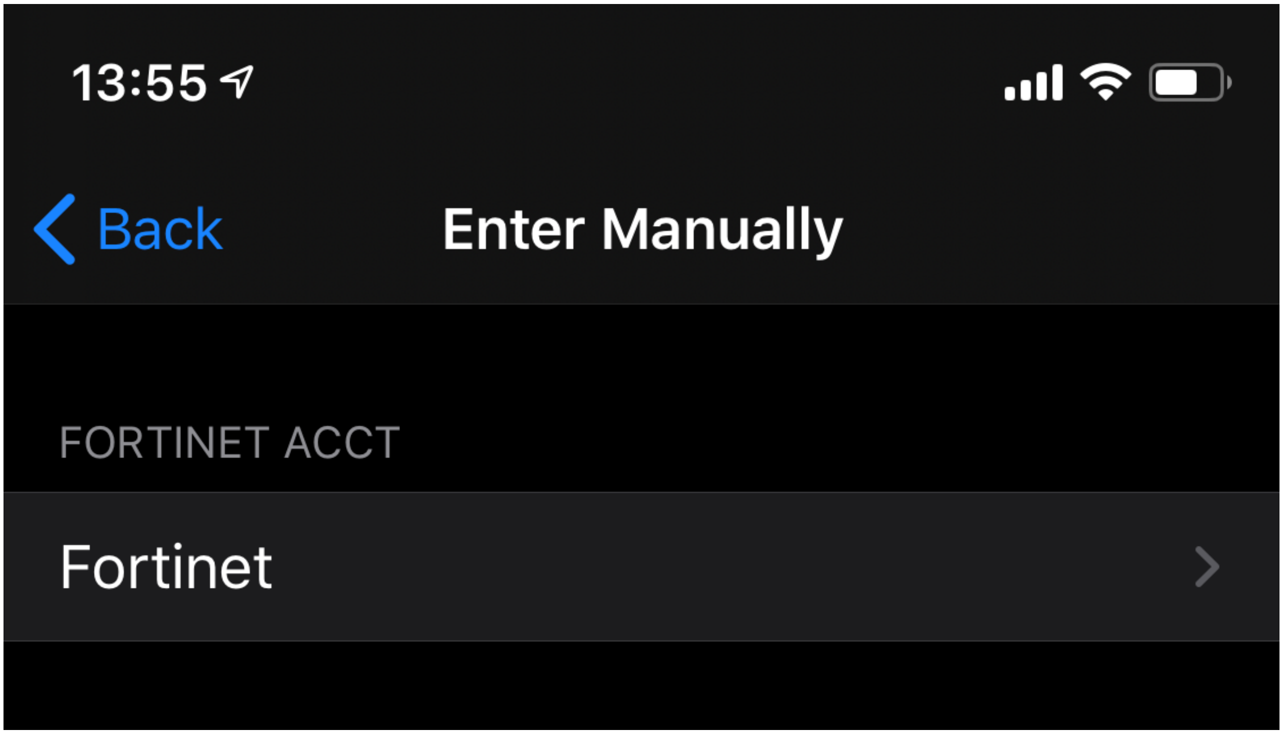
Platform	Device and firmware support
iOS	iPhone, iPad, and iPod Touch with iOS 6.0 and later.
Android	Phones and tablets with Android Jellybean 4.1 and later.
Windows	Windows 10 (desktop and mobile), Windows Phone 8.1, and Windows Phone 8. FortiToken is a Windows Universal Platform (UWP) application. To download FortiToken for Windows 10 desktop and mobile platforms, see FortiToken for Windows on the Microsoft Store.

The following instructions describe procedures when using FortiToken Mobile for iOS on an iPhone. Procedures may vary depending on your device and firmware.

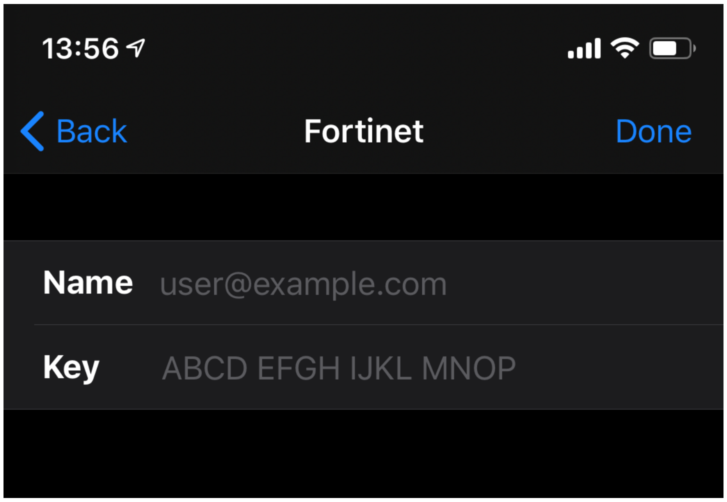
To activate FortiToken Mobile on iOS:

1. On your iOS device, tap on the FortiToken application icon to open the application. If this is your first time opening the application, it may prompt you to create a PIN for secure access to the application and tokens.

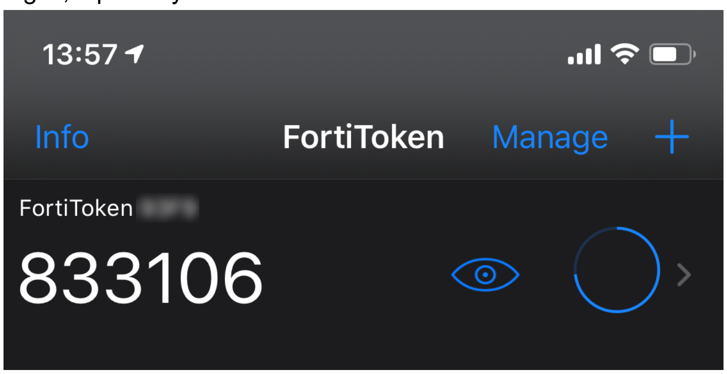

2. Tap on the + icon. The *Scan Barcode* screen appears.


3. If you received the QR code via email, locate and scan the QR code in your email.
OR
If you received the activation key via SMS, tap on *Enter Manually* at the bottom of the screen, and tap on *Fortinet*.


Enter your email address in the *Name* field, the activation key in the *Key* field, and tap *Done*.



4. FortiToken Mobile activates your token, and starts generating OTP digits immediately. To view or hide the OTP digits, tap the eye icon.

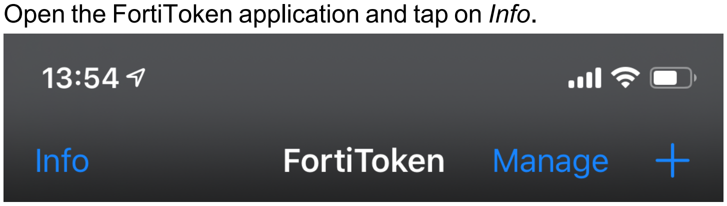


After you open the application, FortiToken Mobile generates a new 6-digit OTP every 30 seconds. All configured tokens display on the application homescreen.

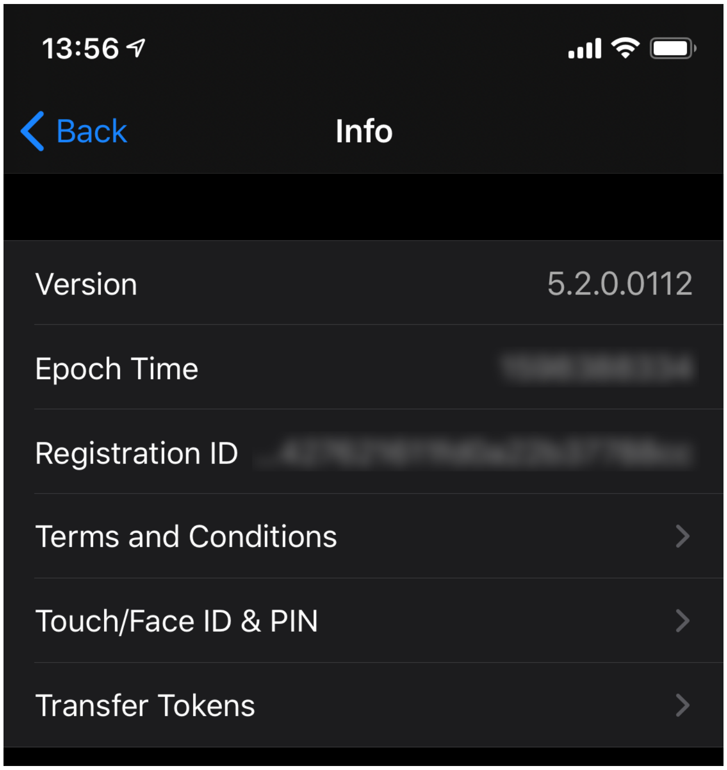
The FortiToken Mobile activation process described above caters to the MFA process that involves two factors (password and OTP) of the authentication process. A third factor (fingerprint or face) can be enabled as well.

To enable *Touch/Face ID* on iOS for FortiToken Mobile:

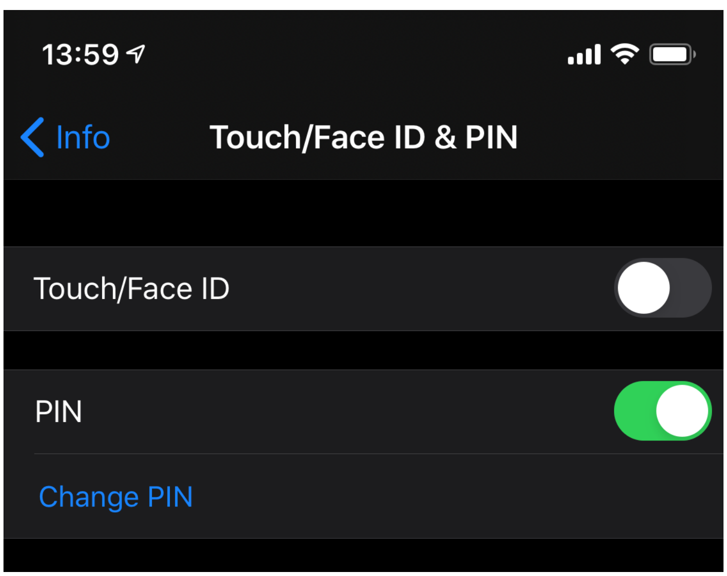
1. Open the FortiToken application and tap on *Info*.



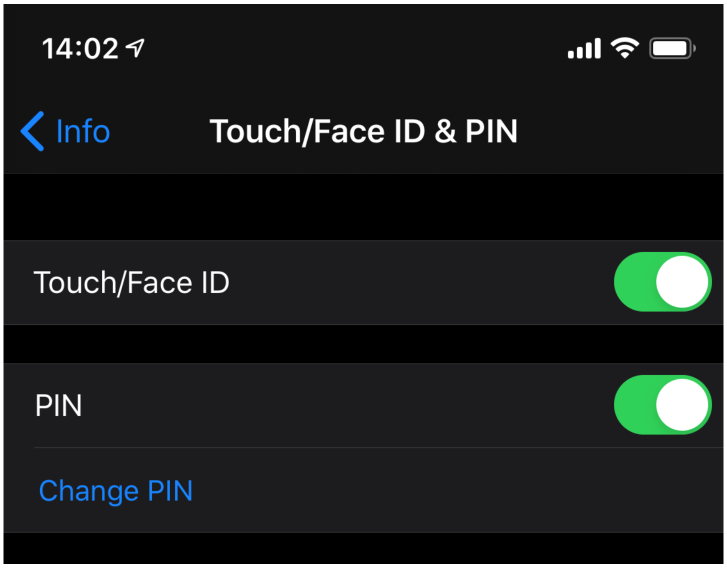
2. Tap on *Touch/Face ID & PIN*.



3. Enable and set up a 4-digit *PIN* for the application. The *PIN* is required to be enabled before you can enable *Touch/Face ID*.

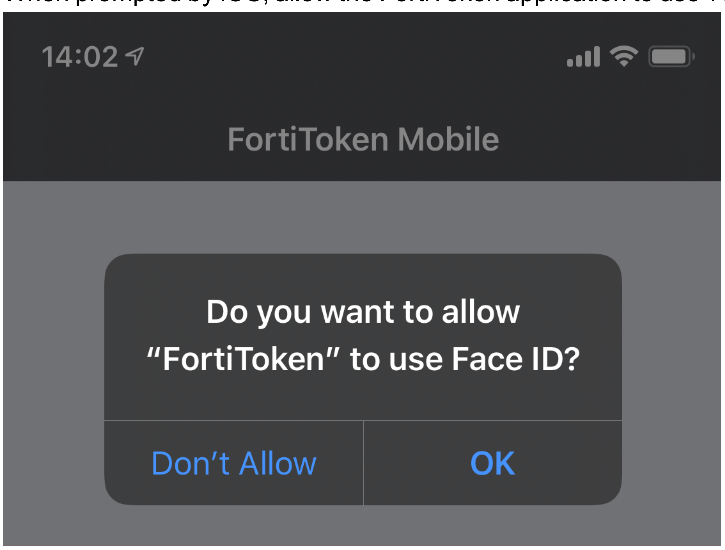


4. Enable *Touch/Face ID*.



You cannot enable *Touch/Face ID* for FortiToken if *Touch/Face ID* is not set up and enabled for device unlock (*iPhone Unlock* in this case) on iOS. You must first set up and enable *Touch/Face ID* from *Settings* on your iOS device.

5. When prompted by iOS, allow the FortiToken application to use *Touch/Face ID* by tapping on *OK* in the prompt.



Applying multi-factor authentication

Multi-factor authentication (MFA) may also be set up for SSL VPN users, administrators, firewall policy, wireless users, and so on. The following topics explain more about how you may use the newly created user in such scenarios:

- MFA for SSL VPN: Set up FortiToken multi-factor authentication on page 1625
- MFA for IPsec VPN: Add FortiToken multi-factor authentication on page 1422
- MFA for Administrators: Administrator account options on page 1941
- MFA with Captive Portal

- MFA for wireless users via Captive Portal
- Configuring firewall authentication on page 1904

FortiToken Cloud

FortiToken Cloud is an Identity and Access Management as a Service (IDaaS) cloud service offering by Fortinet. It enables FortiGate and FortiAuthenticator customers to add MFA for their respective users, through the use of Mobile tokens or Hard tokens. It protects local and remote administrators as well as firewall and VPN users.

For information, see Getting started—FGT-FTC users in the FortiToken Cloud Administration Guide.

Registering hard tokens

Registering FortiTokens consists of the following steps:

1. Adding FortiTokens to FortiOS.
2. Activating FortiTokens.
3. Associating FortiTokens with user accounts.

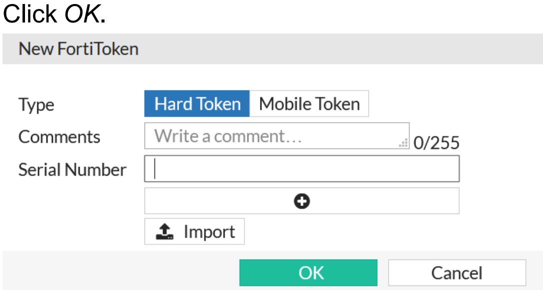
Adding FortiTokens to FortiOS

You can add FortiTokens to FortiOS in the following ways:

- Add FortiToken serial numbers using the GUI
- Add FortiToken serial numbers using the CLI
- Import FortiTokens using a serial number or seed file using the GUI

To manually add single hard token to FortiOS using the GUI:

1. Go to *User & Authentication > FortiTokens*.
2. Click *Create New*.
3. For *Type*, select *Hard Token*.
4. In the *Serial Number* field, enter one or more FortiToken serial numbers.
5. Click *OK*.



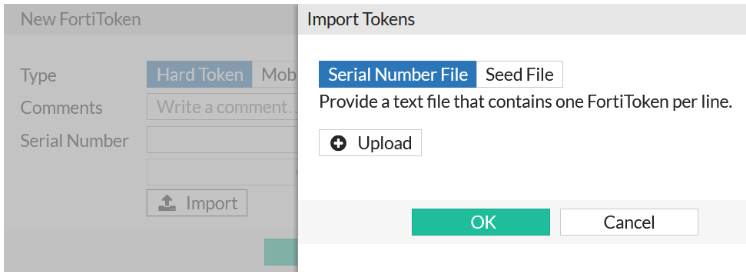
To add multiple FortiTokens to FortiOS using the CLI:

```
config user fortitoken
   edit <serial_number>
   next
   edit <serial_number2>
   next
```

```
end
```

To import multiple FortiTokens to FortiOS using the GUI:

1. Go to *User & Authentication > FortiTokens*.
2. Click *Create New*.
3. For *Type*, select *Hard Token*.
4. Click *Import*. The *Import Tokens* section slides in on the screen.



5. Select *Serial Number File*.

Seed files are only used with FortiToken-200CD. These are special hardware tokens that come with FortiToken seeds on a CD. See the FortiToken Comprehensive Guide for details.

6. Click *Upload*.
7. Browse to the file's location on your local machine, select the file, then click *OK*.
8. Click *OK*.

Activating FortiTokens

You must activate the FortiTokens before starting to use them. FortiOS requires connection to FortiGuard servers for FortiToken activation. During activation, FortiOS queries FortiGuard servers about each token's validity. Each token can only be used on a single FortiGate or FortiAuthenticator. If tokens are already registered, they are deemed invalid for re-activation on another device. FortiOS encrypts the serial number and information before sending for added security.

To activate a FortiToken using the GUI:

1. Go to *User & Authentication > FortiTokens*.
2. Select the desired FortiTokens that have an *Available* status.
3. Click *Activate* from the menu above.
4. Click *Refresh*. The selected FortiTokens are activated.

To activate a FortiToken using the CLI:

```
config user fortitoken
   edit <token_serial_num>
      set status activate
   next
end
```

Associating FortiTokens with user accounts

You can associate FortiTokens with local user or administrator accounts.

To associate a FortiToken to a local user account using the GUI:

1. Ensure that you have successfully added your FortiToken serial number to FortiOS and that its status is *Available*.
2. Go to *User & Authentication > User Definition*. Edit the desired user account.
3. Enable *Two-factor Authentication*.
4. From the *Token* dropdown list, select the desired FortiToken serial number.
5. In the *Email Address* field, enter the user's email address.
6. Click *OK*.

To associate a FortiToken to a local user account using the CLI:

```
config user local
   edit <username>
      set type password
      set passwd "myPassword"
      set two-factor fortitoken
      set fortitoken <serial_number>
      set email-to "username@example.com"
      set status enable
   next
end
```

Before you can use a new FortiToken, you may need to synchronize it due to clock drift.

To associate a FortiToken to an administrator account, refer to the Administrator account options on page 1941 section.

Managing FortiTokens

This section focuses on the following:

Resending an activation email

To resend an activation email/SMS for a mobile token on a FortiGate:

1. Go to *User & Authentication > User Definition* and edit the user.
2. Click *Send Activation Code Email* from the *Two-factor Authentication* section.

Locking/unlocking FortiTokens

To change FortiToken status to active or to lock:

```
config user fortitoken
  edit <token_serial_num>
    set status <active | lock>
  next
end
```

A user attempting to log in using a locked FortiToken cannot successfully authenticate.

Managing FortiTokens drift

If the FortiToken has drifted, the following must take place for the FortiToken to resynchronize with FortiOS:

1. FortiOS prompts the user to enter a second code to confirm.
2. The user gets the next code from the FortiToken. They enter the code at the prompt.
3. FortiOS uses both codes to update its clock to match the FortiToken.

If you still experience clock drift, it may be the result of incorrect time settings on your mobile device. If so, make sure that the mobile device clock is accurate by confirming the network time and the correct timezone.

If the device clock is set correctly, the issue could be the result of the FortiGate and FortiTokens being initialized prior to setting an NTP server. This will result in a time difference that is too large to correct with the synchronize function. To avoid this, selected Tokens can be manually drift adjusted.

To show current drift and status for each FortiToken:

```
diagnose fortitoken info
FORTITOKEN DRIFT STATUS
FTK200XXXXXXXXXC 0 token already activated, and seed won't be returned
FTK200XXXXXXXXXE 0 token already activated, and seed won't be returned
FTKMOBXXXXXXXXXA 0 provisioned
FTKMOBXXXXXXXXX4 0 new
Total activated token: 0
Total global activated token: 0
Token server status: reachable
```

This command lists the serial number and drift for each configured FortiToken. You can check if it is necessary to synchronize the FortiGate and any particular FortiTokens.

To adjust Mobile FortiToken for drift:

```
# execute fortitoken sync <FortiToken_ID> <token_code1> <next_token_code2>
```

Deactivating FortiTokens

To deactivate FortiToken on a FortiGate:

1. Go to *User & Authentication > User Definition*.
2. Select and edit the user for which you want to deactivate the token.
3. Disable the *Two-factor Authentication* toggle.

4. Click *OK*. The token will be removed from the user's *Two-factor Authentication* column. The user will also be removed from the token's *User* column under *User & Authentication > FortiTokens*.

Moving FortiTokens to another device

FortiTokens can only be activated on a single FortiGate or FortiAuthenticator. To move FortiTokens to another device, you would first have to reset the registered FortiTokens on a device and then reactivate them on another device.

To reset Hard tokens registered to a FortiGate appliance (non-VM model), you can reset all hardware FTK200 tokens from the Support Portal, or during RMA transfer. See the Migrating users and FortiTokens to another FortiGate KB article, for more information.

The above process will reset all Hard tokens and you cannot select individual tokens to reset.

To reset FortiToken Mobile, a single Hard token, a Hard token registered to a VM, and so on, an administrator must contact Customer Support and/or open a ticket on the Support Portal.

Once reset, the FortiTokens can be activated on another FortiGate or FortiAuthenticator.

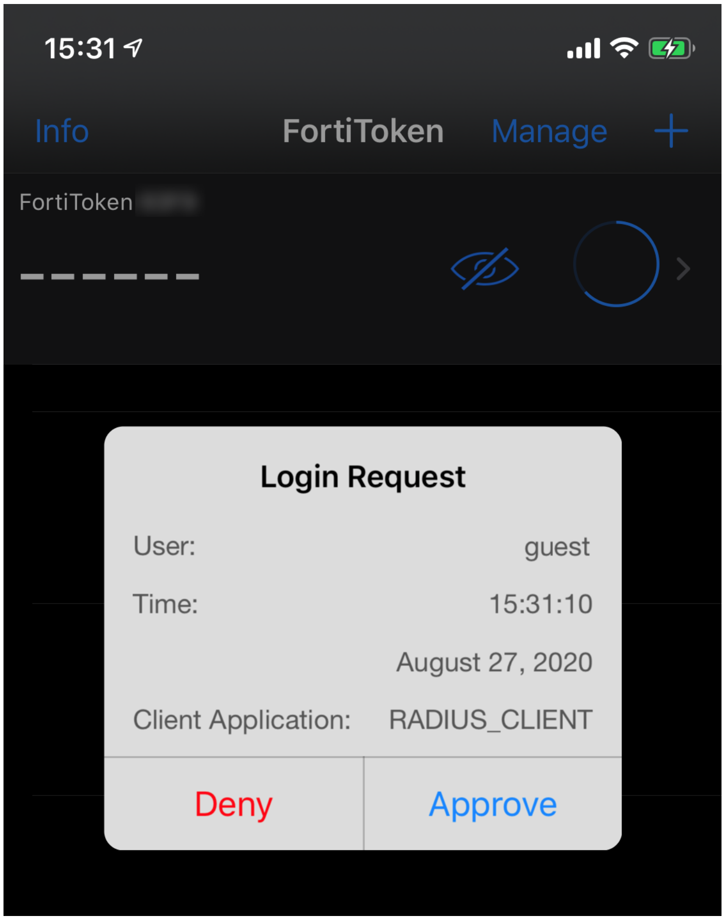
FortiToken Mobile Push

FortiToken Mobile Push allows authentication requests to be sent as push notifications to the end user's FortiToken Mobile application.

The FortiToken Mobile push service operates as follows:

1. FortiGate sends a DNS query to the FortiToken Mobile Push proxy server (*push.fortinet.com*).
2. FortiGate connects to the proxy server via an encrypted connection over TCP/443.
3. The proxy server handles the notification request by making a TLS connection with either Apple (for iOS) or Google (for Android) notification servers. Notification data may include the recipient, session, FortiGate callback IP and port, and so on.
4. The notification service from either Apple or Google notifies the user's mobile device of the push request.
5. The FortiToken Mobile application on the user's mobile displays a prompt for the user to either *Approve* or *Deny* the

request.



To configure FortiToken Mobile push services using the CLI:

```
config system ftm-push
   set status enable
   set server-ip <ip-address>
   set server-port [1-65535]
end
```

The default server port is `4433`.

The server IP address is the public IP address of the FortiOS interface that FortiToken Mobile calls back to. FortiOS uses this IP address for incoming FortiToken Mobile calls.

If an SSL VPN user authenticates with their token, then logs out and attempts to reauthenticate within a minute, a *Please wait* `x` *seconds to login again* message displays. This replaces a previous error/permission denied message. The `x` value depends on the calculation of how much time is left in the current time step.

```
config system interface
   edit "guest"
      set allowaccess ftm
   next
end
```

FortiOS supports FortiAuthenticator-initiated FortiToken Mobile Push notifications for users attempting to authenticate through an SSL VPN and/or RADIUS server (with FortiAuthenticator as the RADIUS server).

Troubleshooting and diagnosis

This section contains some common scenarios for FortiTokens troubleshooting and diagnosis:

- FortiToken Statuses on page 1897
- Recovering trial FortiTokens on page 1898
- Recovering lost Administrator FortiTokens on page 1898
- SSL VPN with multi-factor authentication expiry timers on page 1899

FortiToken Statuses

When troubleshooting FortiToken issues, it is important to understand different FortiToken statuses. FortiToken status may be retrieved either from the CLI or the GUI, with a slightly different naming convention.

Before you begin, verify that the FortiGate has Internet connectivity and is also connected to both the FortiGuard and registration servers:

```
# execute ping fds1.fortinet.com
# execute ping directregistration.fortinet.com
# execute ping globalftm.fortinet.net
```

The `globalftm.fortinet.net` server is the Fortinet Anycast server added in FortiOS 6.4.2.

If there are connectivity issues, retrieving FortiToken statuses or performing FortiToken activation could fail. Therefore, troubleshoot connectivity issues before continuing.

To retrieve FortiToken statuses:

- In the CLI:
  ```
  # diagnose fortitoken info
  ```
- In the GUI:
  Go to *User & Authentication > FortiTokens*.

Various FortiToken statuses in either the CLI or the GUI may be described as follows:

CLI	GUI	Description
`new`	*Available*	Newly added, not pending, not activated, not yet assigned.
`active`	*Assigned*	Assigned to a user, hardware token.
`provisioning`	*Pending*	Assigned to a user and waiting for activation on the FortiToken Mobile app.
`provisioned`	*Assigned*	Assigned to user and activated on the FortiToken Mobile app.

CLI	GUI	Description
`provision timeout`		Token provided to user but not activated on the FortiToken Mobile app. To fix, the token needs to be re-provisioned and activated in time.
`token already activated, and seed won't be returned`	*Error*	Token is locked by FortiGuard FDS. The hardware token was already activated on another device and locked by FDS.
`locked`		Either manually locked by an Administrator (`set status lock`), or locked automatically, for example, when the token is unassigned and the FortiCare FTM provisioning server was unreachable to process that change.

Recovering trial FortiTokens

You can recover trial FortiTokens if deleted from a FortiGate, or if stuck in a state where it is not possible to provision to a user.

When a token is stuck in an unusual state or with errors, delete the FortiTokens from the unit and proceed to recover trial FortiTokens.

To recover trial tokens via the GUI:

1. Go to *User & Authentication > FortiTokens*.
2. Click the *Import Free Trial Tokens* button at the top. The two free trial tokens are recovered.

To recover trial tokens via the CLI:

```
# execute fortitoken-mobile import 0000-0000-0000-0000-0000
```

- Before attempting to recover the trial tokens, both the tokens should be deleted from the unit first.
- If VDOMs are enabled, trial tokens are in the management VDOM (`root` by default).

Following error codes might come up in the CLI:

- If the device is not registered:
  ```
  # execute fortitoken-mobile import 0000-0000-0000-0000-0000
  import fortitoken license error: -7571
  ```
- If the serial number format is incorrect:
  ```
  # execute fortitoken-mobile import 0000-0000-0000-0000-00
  import fortitoken license error: -7566
  ```

Recovering lost Administrator FortiTokens

If an Administrator loses their FortiToken or the FortiToken is not working, they will not be able to log into the admin console through the GUI or the CLI. If there is another Administrator that can log into the device, they may be able to reset the two-factor settings configured for the first Administrator, or create a new Admin user for them. Note that a *super_admin* user will be able to edit other admin user settings, but a *prof_admin* user will not be able to edit *super_admin* settings.

In the case where there are no other administrators configured, the only option is to flash format the device and reload a backup config file. You must have console access to the device in order to format and flash the device. It is recommended to be physically on site to perform this operation.

The process of resetting an Admin user password using the maintainer account cannot be used to reset or disable two-factor authentication.

Before formatting the device, verify that you have a backup config file. You may or may not have the latest config file backed up, though you should consider using a backed up config file, and reconfigure the rest of the recent changes manually. Otherwise, you may need to configure your device starting from the default factory settings.

To recover lost Administrator FortiTokens:

1. If you have a backed up config file:
   a. Open the config file and search for the specific admin user. For representational purposes we will use `Test` in our example.
   ```
   # edit "Test"
      set accprofile "super_admin"
      set vdom "root"
      set two-factor fortitoken
      set fortitoken "FTKXXXXXXXXXX"
      set email-to "admin@email.com"
      set password *********
      next
   end
   ```
   b. Once you find the settings for the `Test` user, delete the `fortitoken`-related settings:
   ```
   # edit "Test"
      set accprofile "super_admin"
      set vdom "root"
      set password *********
      next
   end
   ```
2. Format the boot device during a maintenance window and reload the firmware image using instructions in the Formatting and loading FortiGate firmware image using TFTP KB article.
3. Once the reload is complete, log into the admin console from the GUI using the default admin user credentials, and go to *Configuration* > *Restore* from the top right corner to reload your config file created in Step 1 above.
4. Once the FortiGate reboots and your configuration is restored, you can log in with your admin user credentials.

SSL VPN with multi-factor authentication expiry timers

When SSL VPN is configured with multi-factor authentication (MFA), sometimes you may require a longer token expiry time than the default 60 seconds.

To configure token expiry timers using the CLI:

```
config system global
    set two-factor-ftk-expiry <number of seconds>
    set two-factor-ftm-expiry <number of seconds>
    set two-factor-sms-expiry <number of seconds>
    set two-factor-fac-expiry <number of seconds>
```

```
    set two-factor-email-expiry <number of seconds>
end
```

These timers apply to the tokens themselves and remain valid for as long as configured above. However, SSL VPN does not necessarily accept tokens for the entire duration they are valid. To ensure SSLVPN accepts the token for longer durations, you need to configure the remote authentication timeout setting accordingly.

To configure the remote authentication timeout:

```
config system global
    set remoteauthtimeout <1-300 seconds>
end
```

SSL VPN waits for a maximum of five minutes for a valid token code to be provided before closing down the connection, even if the token code is valid for longer.

The `remoteauthtimout` setting shows how long SSL VPN waits not only for a valid token to be provided before closing down the connection, but also for other remote authentication like LDAP, RADIUS, and so on.

Configuring the maximum log in attempts and lockout period

Failed log in attempts can indicate malicious attempts to gain access to your network. To prevent this security risk, you can limit the number of failed log in attempts. After the configured maximum number of failed log in attempts is reached, access to the account is blocked for the configured lockout period.

To configure number of maximum log in attempts:

This example sets the maximum number of log in attempts to five.

```
config user setting
   set auth-lockout-threshold 5
end
```

To configure the lockout period in seconds:

This example sets the lockout period to five minutes (300 seconds).

```
config user setting
   set auth-lockout-duration 300
end
```

PKI

The following topics include information about public key infrastructure (PKI):

- Configuring a PKI user on page 1901
- SSL VPN with certificate authentication on page 1658

- SSL VPN with LDAP-integrated certificate authentication on page 1663

Configuring a PKI user

PKI users are users who are identified by a digital certificate they hold. Defining a PKI user in FortiOS specifies:

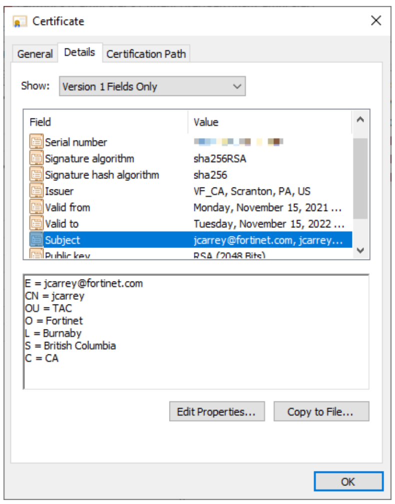
- Which CA certificate to use to validate the user's certificate
- The field and value of the user's certificate that FortiOS will check to verify a user

These peer users can then be used in a FortiGate user group, or as a peer certificate group used for IPsec VPN configurations that accept RSA certificate authentication.

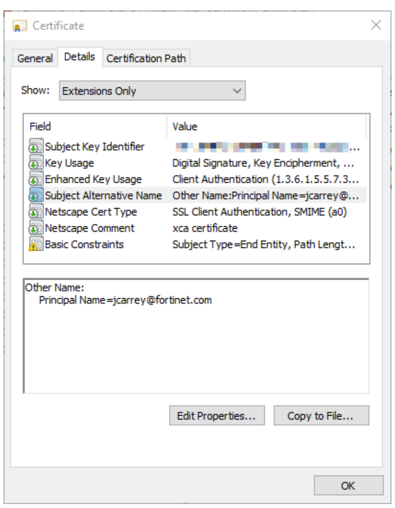
Example X.509 certificate

The following certificate demonstrates which FortiGate settings can be used to match on different fields.

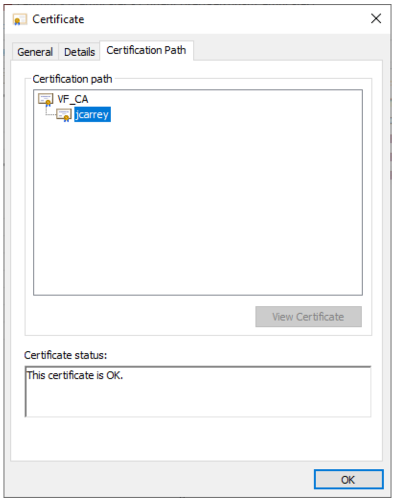
Subject:



Subject Alternative Name:



Certification path:



To configure a PKI user:

```
config user peer
    edit <name>
        set ca <string>
        set mandatory-ca-verify {enable | disable}
        set subject <string>
        set cn <string>
        set cn-type {string | email | FQDN | ipv4 | ipv6}
        set ldap-server <string>
        set ldap-username <string>
        set ldap-password <string>
        set ldap-mode {password | principal-name}
    next
end
```

`ca <string>`	Specify which certificate on the FortiGate is used to validate the client's certificate. This can be any CA in the client's certificate chain. You may need to upload a CA certificate to the FortiGate specifically to identify PKI peer users (see Uploading a certificate using the GUI on page 2164).
`mandatory-ca-verify {enable | disable}`	Control the action if the CA certificate used to sign the client's certificate is not installed on the FortiGate (default = enable). Disabling this setting makes the FortiGate consider any certificate presented by the peer as valid. In the example certificate, the certification path shows that VF_CA signed jcarrey's certificate.
`subject <string>`	Enter the peer certificate name constraints.
`cn <string>`	Enter the peer certificate common name.
`cn-type {string | email | FQDN | ipv4 | ipv6}`	Set the peer certificate common name type: string, email, FQDN, IPv4 address, or IPv6 address. See CN on page 1904 for more details.
`ldap-server <string>`	Enter the name of an LDAP server defined under `config user ldap` for performing client access rights checks. See LDAP servers on page 1803 for more details.

`ldap-mode {password | principal-name}`	Set the mode for LDAP peer authentication, either by password or principal name (default = password). See LDAP on page 1904 for more details.
`ldap-username <string>`	Enter the username for the LDAP server bind when the LDAP mode is password.
`ldap-password <string>`	Enter the password for the LDAP server bind when the LDAP mode is password.

Identifying users based on their client certificate

When the client's certificate is valid, or `mandatory-ca-verify` is disabled, the FortiGate can then inspect the certificate to check specific fields for matching values. There are three ways of specifying which certificate field to verify: by subject, CN, or LDAP. All string comparisons are case sensitive.

Subject

This basic method verifies that the subject string defined in the PKI user setting matches a value or substring in the subject field of the user certificate. Further matching is controlled in the following VPN certificate settings.

```
config vpn certificate setting
    set subject-match {substring | value}
    set subject-set {superset | subset}
    set cn-match {substring | value}
    set cn-allow-multi {enable | disable}
end
```

`subject-match {substring | value}`	Control how to do relative distinguished name (RDN) value matching with the certificate subject name: • substring: find a match if any string in the certificate subject name matches the name being searched for (such as `set subject jcarrey`). • value: find a match if any attribute value string in a certificate subject name is an exact match with the name being searched for (such as `set subject "OU=TAC"` or `set subject "C=CA, CN=jcarrey, OU=TAC"`).
`set subject-set {superset | subset}`	Control how to do RDN value matching with the certificate subject name: • superset: a certificate only passes verification if it contains all the RDNs defined in the subject settings (such as `set subject "E = jcarrey@fortinet.com, CN = jcarrey, OU = TAC, O = Fortinet, L = Burnaby, S = British Columbia, C = CA"`). • subset: a certificate passes verification if the RDN is a subset of the certificate subject (such as `set subject "CN = jcarrey, OU = TAC"`).
`cn-match {substring | value}`	Control how to do CN value matching with the certificate subject name: • substring: find a match if any string in the certificate subject name matches the name being searched for. • value: find a match if any attribute value string in a certificate subject name is an exact match with the name being searched for.
`cn-allow-multi {enable | disable}`	Enable/disable allowing multiple CN entries with the certificate subject name (default = enable).

CN

Common name (CN) certificate verification compares the CN in the subject field with the configured string (such as `set cn "jcarrey"`. The following logic is used when configuring different CN types:

Type	Action
`string`	Based on the `cn-match` setting, perform a substring or exact match in the certificate subject.
`email`	Look for a match in the certificate subject.
`FQDN`	Look for a match in the certificate subject, then compare the mapped IP and client IP. The FQDN is only retrieved from the CN.
`ipv4`	Look for a match in the certificate subject, then compare the IP.
`ipv6`	Look for a match in the certificate subject, then compare the IP.

The CN type also controls the format checking of the CN string. In this example, if the CN type is set to email, the CN must be in email format (`set cn "jcarrey@fortinet.com"`).

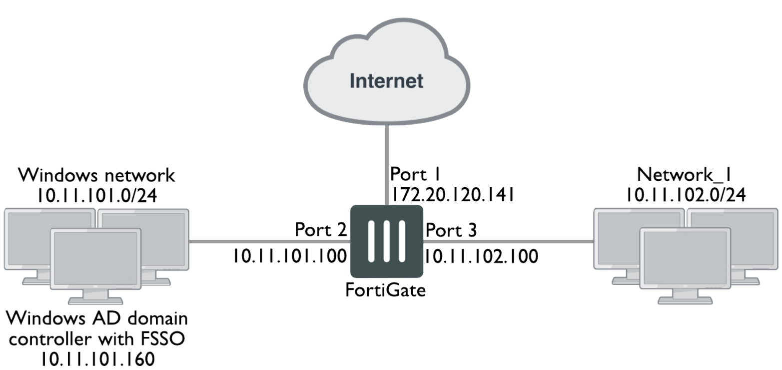
LDAP

LDAP-integrated user authentication allows the FortiGate to check the connecting user against an LDAP server in two ways: through a username and password, or the certificate's principal name. The `password` method requires the username and password of each authenticating user to be entered, so it is not recommended when configuring PKI users. The `principal-name` method is recommended.

The UPN in the user certificate's Subject Alternative Name field is used to look up the user in the LDAP directory. If a match is found, then authentication succeeds. This type of configuration scales well since only one PKI user needs to be created on the FortiGate. Connecting clients use their unique user certificate to match within the configured LDAP server.

Configuring firewall authentication

In this example, a Windows network is connected to the FortiGate on port 2, and another LAN, Network_1, is connected on port 3.



All Windows network users authenticate when they log on to their network. Engineering and Sales groups members can access the Internet without reentering their authentication credentials. The example assumes that you have already installed and configured FSSO on the domain controller.

LAN users who belong to the Internet_users group can access the Internet after entering their username and password. The example shows two users: User1, authenticated by a password stored in FortiOS; and User 2, authenticated on an external authentication server. Both users are local users since you create the user accounts in FortiOS.

1. Create a locally authenticated user account.
2. Create a RADIUS-authenticated user account.
3. Create an FSSO user group.
4. Create a firewall user group.
5. Define policy addresses.
6. Create security policies.

Creating a locally authenticated user account

User1 is authenticated by a password stored in FortiOS.

To create a locally authenticated user account in the GUI:

1. Go to *User & Authentication > User Definition* and click *Create New*.
2. Configure the following settings:

User Type	Local User
User Name	User1
Password	hardtoguess1@@1
User Account Status	Enabled

3. Click *Submit*.

To create a locally authenticated user account in the CLI:

```
config user local
   edit user1
      set type password
      set passwd hardtoguess1@@1
   next
end
```

Creating a RADIUS-authenticated user account

You must first configure FortiOS to access the external authentication server, then create the user account.

To create a RADIUS-authenticated user account in the GUI:

1. Go to *User & Authentication > RADIUS Servers* and click *Create New*.
2. Configure the following settings:

Name	OurRADIUSsrv
Authentication method	Default
Primary Server	
IP/Name	10.11.101.15
Secret	OurSecret

3. Click *OK*.
4. Go to *User & Authentication > User Definition* and click *Create New*.
5. Configure the following settings:

User Type	Remote RADIUS User
User Name	User2
RADIUS Server	OurRADIUSsrv
User Account Status	Enabled

6. Click *Submit*.

To create a RADIUS-authenticated user account in the CLI:

```
  config user radius
  edit OurRADIUSsrv
    set server 10.11.102.15
    set secret OurSecret
    set auth-type auto
  next
end
config user local
  edit User2
    set name User2
    set type radius
    set radius-server OurRADIUSsrv
  next
end
```

Creating an FSSO user group

This example assumes that you have already set up FSSO on the Windows network and that it used advanced mode, meaning that it uses LDAP to access user group information. You must do the following:

- Configure LDAP access to the Windows AD global catalog
- Specify the collector agent that sends user log in information to FortiOS
- Select Windows user groups to monitor
- Select and add the Engineering and Sales groups to an FSSO user group

To create an FSSO user group in the GUI:

1. Configure LDAP for FSSO:
   a. Go to *User & Authentication > LDAP Servers* and click *Create New*.
   b. Configure the following settings:

Name	ADserver
Server Name / IP	10.11.101.160
Distinguished Name	dc=office,dc=example,dc=com
Bind Type	Regular
Username	cn=FSSO_Admin,cn=users,dc=office,dc=example,dc=com
Password	Enter a secure password.

   c. Leave other fields as-is. Click *OK*.
2. Specify the collector agent for FSSO;
   a. Go to *Security Fabric > External Connectors* and click *Create New*.
   b. Under *Endpoint/Identity*, select *FSSO Agent on Windows AD*.
   c. Configure the following settings:

Name	Enter the Windows AD server name. This name appears in the Windows AD server list when you create user groups. In this example, the name is WinGroups.
Server IP/Name	Enter the IP address or name of the server where the agent is installed. The maximum name length is 63 characters. In this example, the IP address is 10.11.101.160.
Password	Enter the password of the server where the agent is installed. You only need to enter a password for the collector agent if you configured the agent to require authenticated access. If the TCP port used for FSSO is not the default, 8000, you can run the `config user fsso` command to change the setting in the CLI.
Collector Agent AD access mode	Advanced
LDAP Server	Select the previously configured LDAP server. In this example, it is ADserver.
User/Groups/Organization Units	Select the users, groups, and OUs to monitor.

   d. Click *OK*.
3. Create the FSSO_Internet_users user group:
   a. Go to *User & Authentication > User Groups* and click *Create New*.
   b. Configure the following settings:

Name	FSSO_Internet_users

Type	Fortinet Single Sign-On (FSSO)
Members	Engineering, Sales

c. Click *OK*.

To create an FSSO user group in the CLI:

```
config user ldap
   edit "ADserver"
      set server "10.11.101.160"
      set dn "cn=users,dc=office,dc=example,dc=com"
      set type regular
      set username "cn=administrator,cn=users,dc=office,dc=example,dc=com"
      set password set_a_secure_password
   next
end
config user fsso
   edit "WinGroups"
      set ldap-server "ADserver"
      set password ********
      set server "10.11.101.160"
   next
end
config user group
   edit FSSO_Internet_users
      set group-type fsso-service
      set member CN=Engineering,cn=users,dc=office,dc=example,dc=com
            CN=Sales,cn=users,dc=office,dc=example,dc=com
   next
end
```

Creating a firewall user group

This example shows a firewall user group with only two users. You can add additional members.

To create a firewall user group in the GUI:

1. Go to *User & Authentication > User Groups* and click *Create New*.
2. Configure the following settings:

Name	Internet_users
Type	Firewall
Members	User1, User2

3. Click *OK*.

To create a firewall user group in the CLI:

```
config user group
   edit Internet_users
         set group-type firewall
         set member User1 User2
```

```
  next
end
```

Defining policy addresses

To define policy addresses:

1. Go to *Policy & Objects > Addresses*.
2. Click *Create New > Address*.
3. Configure the following settings:

Name	Internal_net
Type	Subnet
IP/Netmask	10.11.102.0/24
Interface	Port 3

4. Click *OK*.
5. Create another new address by repeating steps 2-4 using the following settings:

Name	Windows_net
Type	Subnet
IP/Netmask	10.11.101.0/24
Interface	Port 2

Creating security policies

You must create two security policies: one for the firewall group connecting through port 3, and one for the FSSO group connecting through port 2.

To create security policies using the GUI:

1. Go to *Policy & Objects > Firewall Policy*.
2. Click *Create New*.
3. Configure the following settings:

Incoming Interface	Port2
Source Address	Windows_net
Source User(s)	FSSO_Internet_users
Outgoing Interface	Port1
Destination Address	all
Schedule	always
Service	ALL

NAT	Enabled.
Security Profiles	You can enable security profiles as desired.

4. Click *OK*.
5. Create another new policy by repeating steps 2-4 using the following settings:

Incoming Interface	Port3
Source Address	Internal_net
Source User(s)	Internet_users
Outgoing Interface	Port1
Destination Address	all
Schedule	always
Service	ALL
NAT	Enabled.
Security Profiles	You can enable security profiles as desired.

6. Click *OK*.

To create security policies using the CLI:

```
config firewall policy
     edit 0
     set srcintf port2
     set dstintf port1
     set srcaddr Windows_net
     set dstaddr all
     set action accept
     set groups FSSO_Internet_users
     set schedule always
     set service ANY
     set nat enable
   next
end
config firewall policy
   edit 0
     set srcintf port3
     set dstintf port1
     set srcaddr internal_net
     set dstaddr all
     set action accept
     set schedule always
     set groups Internet_users
     set service ANY
     set nat enable
   next
end
```

FSSO

FortiOS can provide single sign-on capabilities to Windows AD, Citrix, VMware Horizon, Novell eDirectory, and Microsoft Exchange users with the help of agent software installed on these networks. The agent software sends information about user logons to the FortiGate unit. With user information such as IP address and user group memberships from the network, FortiGate security policies can allow authenticated network access to users who belong to the appropriate user groups without requesting their credentials again.

Fortinet Single Sign-On (FSSO), through agents installed on the network, monitors user logons and passes that information to the FortiGate unit. When a user logs on at a workstation in a monitored domain, FSSO:

- Detects the logon event and records the workstation name, domain, and user,
- Resolves the workstation name to an IP address,
- Determines which user groups the user belongs to,
- Sends the user logon information, including IP address and groups list, to the FortiGate unit, and
- Creates one or more log entries on the FortiGate unit for this logon event as appropriate.

When the user tries to access network resources, the FortiGate unit selects the appropriate security policy for the destination. If the user belongs to one of the permitted user groups associated with that policy then the connection is allowed, otherwise the connection is denied.

Agent-based FSSO

Several different FSSO agents can be used in an FSSO implementation:

- Domain Controller (DC) agent
- eDirectory agent
- Citrix/Terminal Server (TS) agent
- Collector Agent

Consult the latest FortiOS Release Notes for operating system compatibility information.

Domain Controller agent

The Domain Controller (DC) agent must be installed on every domain controller when you use DC Agent mode. The DC agents monitor user logon events and pass the information to the Collector agent, which stores the information and sends it to the FortiGate unit.

eDirectory agent

The eDirectory agent is installed on a Novell network to monitor user logons and send the required information to the FortiGate unit. It functions much like the Collector agent on a Windows AD domain controller. The agent can obtain information from the Novell eDirectory using either the Novell API or LDAP.

Terminal Server agent

The Terminal Server (TS) agent can be installed on a Citrix, VMware Horizon 7.4, or Windows Terminal Server to monitor user logons in real time. It functions much like the DC Agent on a Windows AD domain controller.

Collector agent

The Collector Agent (CA) is installed as a service on a server in the Windows AD network to monitor user logons and send the required information to the FortiGate unit. The Collector agent can collect information from a DC agent (Windows AD) and TS agent (Citrix or VMware Horizon Terminal Server).

In a Windows AD network, the Collector agent can optionally obtain logon information by polling the AD domain controllers. In this case, DC agents are not needed.

The CA is responsible for DNS lookups, group verification, workstation checks, and updating FortiGates on logon records. The FSSO CA sends Domain Local Security Group and Global Security Group information to FortiGate units. The CA communicates with the FortiGate over TCP port 8000 and it listens on UDP port 8002 for updates from the DC agents.

The FortiGate device can have up to five CAs configured for redundancy. If the first CA on the list is unreachable, the next is attempted, and so on down the list until one is contacted.

All DC agents must point to the correct CA port number and IP address on domains with multiple DCs.

A FortiAuthenticator device can act much like a CA, collecting Windows AD user logon information and sending it to the FortiGate device. It is particularly useful in large installations with several FortiGate units. For more information, see the FortiAuthenticator Administration Guide.

Agentless FSSO

For Windows AD networks, FortiGate devices can also provide SSO capability by directly polling Windows Security Event log entries on Windows DC for user log in information. This configuration does not require a CA or DC agent.

FortiGate configuration

To configure FSSO on a FortiGate, go to *Security Fabric > External Connectors*.

When creating a new connector, several options for connectors are available under Endpoint/Identity:

- Fortinet single sign-on agent on page 2490
  For most FSSO Agent-based deployments, this connector option will be used. Specify either Collector Agent or Local as User Group Source to collect user groups from the Collector Agent, or to match users to user groups from a LDAP server.
- Poll Active Directory server on page 2491
  This connection option directly polls Windows Security Event log entries on Windows DC for user log in information.
- RADIUS single sign-on agent on page 2498
  FortiGate can authenticate users who have authenticated on a remote RADIUS server by monitoring the RADIUS accounting records forwarded by the RADIUS server to the FortiGate.
- Exchange Server connector on page 2501
  FortiGate collects information about authenticated users from corporate Microsoft Exchange Servers.
- Symantec endpoint connector on page 2492
  This connector uses client IP information from Symantec Endpoint Protection Manager (SEPM) to assign dynamic IP addresses on FortiOS.

Since FSSO is commonly associated with Agent-based FSSO and Agentless FSSO, this chapter will primarily focus on the first two Security Fabric External Connector options.

FSSO polling connector agent installation

This topic gives an example of configuring a local FSSO agent on the FortiGate. The agent actively pools Windows Security Event log entries on Windows Domain Controller (DC) for user log in information. The FSSO user groups can then be used in a firewall policy.

This method does not require any additional software components, and all the configuration can be done on the FortiGate.

To configure a local FSSO agent on the FortiGate:

1. Configure an LDAP server on the FortiGate on page 1913
2. Configure a local FSSO polling connector on page 1913
3. Add the FSSO groups to a policy on page 1914

Configure an LDAP server on the FortiGate

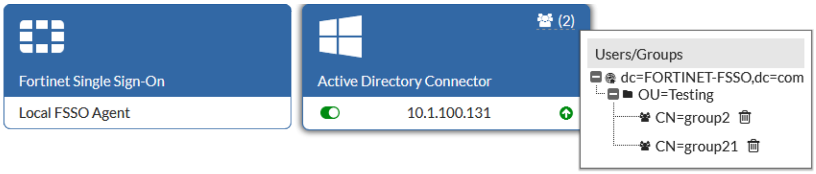
Refer to Configuring an LDAP server on page 1803. The connection must be successful before configuring the FSSO polling connector.

Configure a local FSSO polling connector

To configure a local FSSO polling connector:

1. Go to *Security Fabric > External Connectors* and click *Create New*.
2. In the *Endpoint/Identity* section, select *Poll Active Directory Server*.
3. Fill in the required information.
4. For *LDAP Server*, select the server you just created.
5. Configure the group settings:
   a. For *Users/Groups*, click *Edit*. The structure of the LDAP tree is shown in the *Users/Groups* window.
   b. Click the *Groups* tab.
   c. Select the required groups, right-click on them, and select *Add Selected*. Multiple groups can be selected at one time by holding the CTRL or SHIFT keys. The groups list can be filtered or searched to limit the number of groups that are displayed.
   d. Click the *Selected* tab and verify that the required groups are listed. To remove a group, right-click and select *Remove Selected*.
   e. Click *OK* to save the group settings.
6. Click *OK* to save the connector settings.
7. Go back to *Security Fabric > External Connectors*.

8. There should be two new connectors:



- The *Local FSSO Agent* is the backend process that is automatically created when the first FSSO polling connector is created.
- The *Active Directory Connector* is the front end connector that can be configured by FortiGate administrators.

To verify the configuration, hover the cursor over the top right corner of the connector; a popup window will show the currently selected groups. A successful connection is also shown by a green up arrow in the lower right corner of the connector.

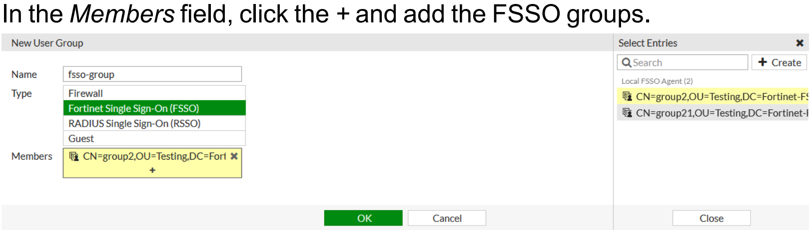
If you need to get log in information from multiple DCs, then you must configure other Active Directory connectors for each additional DC to be monitored.

Add the FSSO groups to a policy

FSSO groups can be used in a policy by either adding them to the policy directly, or by adding them to a local user group and then adding the group to a policy.

To add the FSSO groups to a local user group:

1. Go to *User & Authentication > User Groups* and click *Create New*.
2. Enter a name for the group in the *Name* field.
3. Set the *Type* to *Fortinet Single Sign-On (FSSO)*.
4. In the *Members* field, click the + and add the FSSO groups.



5. Click *OK*.
6. Add the local FSSO group to a policy.

To add the FSSO groups directly to a firewall policy:

1. Go to *Policy & Objects > Firewall Policy* and click *Create New*.
2. In the *Source* field, click the +. In the *Select Entries* pane, select the *User* tab.
3. Select the FSSO groups.
4. Configure the remaining settings as required.
5. Click *OK*.

Troubleshooting

If an authenticated AD user cannot access the internet or pass the firewall policy, verify the local FSSO user list:

```
# diagnose debug authd fsso list
----FSSO logons----
IP: 10.1.100.188 User: test2 Groups: CN=group2,OU=Testing,DC=Fortinet-FSSO,DC=COM
Workstation: MemberOf: CN=group2,OU=Testing,DC=Fortinet-FSSO,DC=COM
Total number of logons listed: 1, filtered: 0
----end of FSSO logons----
```

1. Check that the group in *MemberOf* is allowed by the policy.
2. If the expected AD user is not in list, but other users are, it means that either:
   - The FortiGate missed the log in event, which can happen if many users log in at the same time, or
   - The user's workstation is unable to connect to the DC, and is currently logged in with cached credentials, so there is no entry in the DC security event log.
3. If there are no users in the local FSSO user list:
   a. Ensure that the local FSSO agent is working correctly:

      ```
      # diagnose debug enable
      # diagnose debug authd fsso server-status

      Server Name                    Connection Status    Version           Address
      -----------                    -----------------    -------           -------
      FGT_A (vdom1) # Local FSSO Agent    connected       FSAE server 1.1   127.0.0.1
      ```

      The connection status must be `connected`.

   b. Verify the Active Directory connection status:

      ```
      # diagnose debug fsso-polling detail 1
      AD Server Status (connected):
      ID=1, name(10.1.100.131),ip=10.1.100.131,source(security),users(0)
      port=auto username=Administrator
      read log eof=1, latest logon timestamp: Fri Jul 26 10:36:20 2019

      polling frequency: every 10 second(s) success(274), fail(0)
      LDAP query: success(0), fail(0)
      LDAP max group query period(seconds): 0
      LDAP status: connected

      Group Filter: CN=group2,OU=Testing,DC=Fortinet-
      FSSO,DC=com+CN=group21,OU=Testing,DC=Fortinet-FSSO,DC=COM
      ```

      If the polling frequency shows successes and failures, that indicates sporadic network problems or a very busy DC. If it indicates no successes or failures, then incorrect credentials could be the issue.

      If the LDAP status is connected, then the FortiGate can access the configured LDAP server. This is required for AD group membership lookup of authenticated users because the Windows Security Event log does not include group membership information. The FortiGate sends an LDAP search for group membership of authenticated users to the configure LDAP server.

      FortiGate adds authenticated users to the local FSSO user list only if the group membership is one of the groups in `Group Filter`.
4. If necessary, capture the output of the local FortiGate daemon that polls Windows Security Event logs:

   ```
   # diagnose debug application fssod -1
   ```

This output contains a lot of detailed information which can be captured to a text file.

Limitations

- NTLM based authentication is not supported.
- If there are a large number of user log ins at the same time, the FSSO daemon may miss some. Consider using FSSO agent mode if this will be an issue. See Public and private SDN connectors on page 2426 for information.
- The FSSO daemon does not support all of the security log events that are supported by other FSSO scenarios. For example, only Kerberos log in events 4768 and 4769 are supported.

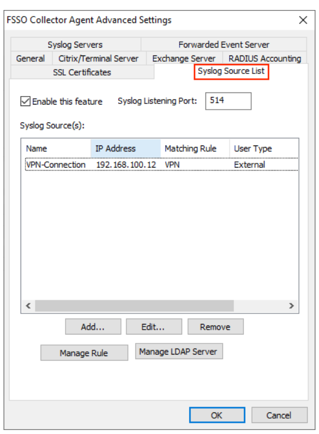
FSSO using Syslog as source

This example describes how to configure Fortinet Single Sign-On (FSSO) agent on Windows using syslog as the source and a custom syslog matching rule.

The FSSO collector agent must be build 0291 or later, and in advanced mode (see How to switch FSSO operation mode from Standard Mode to Advanced Mode).

To configure the FSSO agent on Windows:

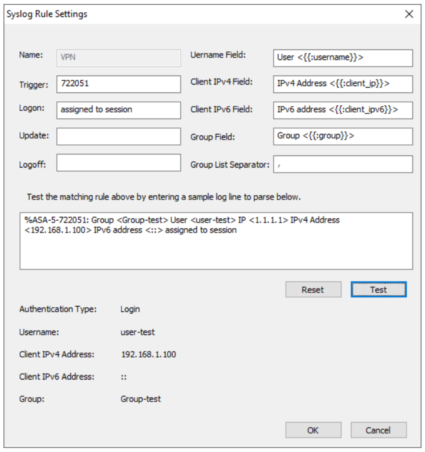
1. Open the FSSO agent on Windows.
2. Click *Advanced Settings*.
3. Go to the *Syslog Source List* tab.
4. Select *Enable this feature*.
5. Set *Syslog Listening Port*, or use the default port.



6. Click *Manage Rule*.
7. Create a new syslog rule:
   a. Click *Add*.
   b. Configure the rule:

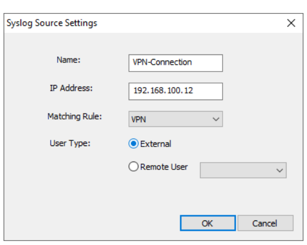
Trigger	*722051*
Logon	*assigned to session*

Username Field	*User <{{:username}}>*
Client IPv4 Field	*IPv4 Address <{{:client_ip}}>*
Client IPv6 Field	*IPv6 Address <{{:client_ipv6}}>*
Group Field	*Group <{{:group}}>*
Groups List Separator	*,*



   c. To test the rule, enter a sample log line, then click *Test*.
   d. Click *OK*.
8. Create a new syslog source:
   a. On the *Advanced Settings* window, click *Add*.
   b. Configure the source:

Name	*VPN-Connection*
IP Address	*192.168.100.12*
Matching Rule	*VPN*
User Type	*External*: Users are not defined on the CA and user groups come from the source. *Remote User*: Users are defined on a remote LDAP server and user groups are retrieved from the specified LDAP server. Any group from the syslog messages are ignored. See Connect to a remote LDAP server on page 1918.



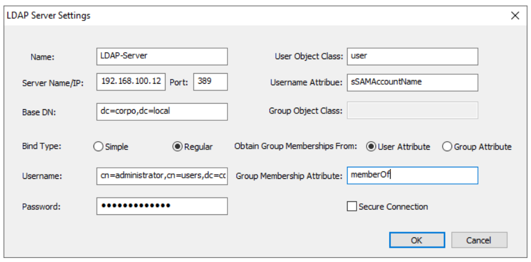
   c. Click *OK*.
9. Click *OK*.

Connect to a remote LDAP server

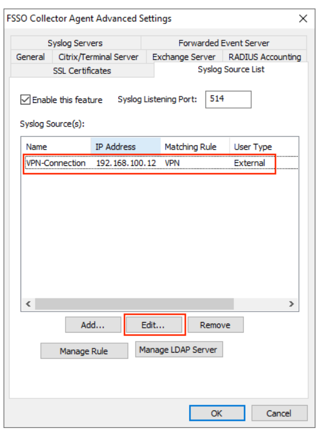
This section describes how to connect to a remote LDAP server to match the user identity from the syslog server with an LDAP server.

To connect to a remote LDAP server:

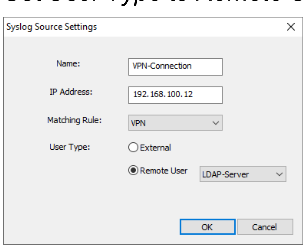
1. Open the FSSO agent on Windows.
2. Click *Advanced Settings*.
3. Go to the *Syslog Source List* tab.
4. Click *Manage LDAP Server*.
5. Click *Add* and configure the LDAP server settings:



6. Click *OK*.
7. Select the syslog source and click *Edit*.



8. Set *User Type* to *Remote User*, and select the LDAP server from the drop-down list.



9. Click *OK*.

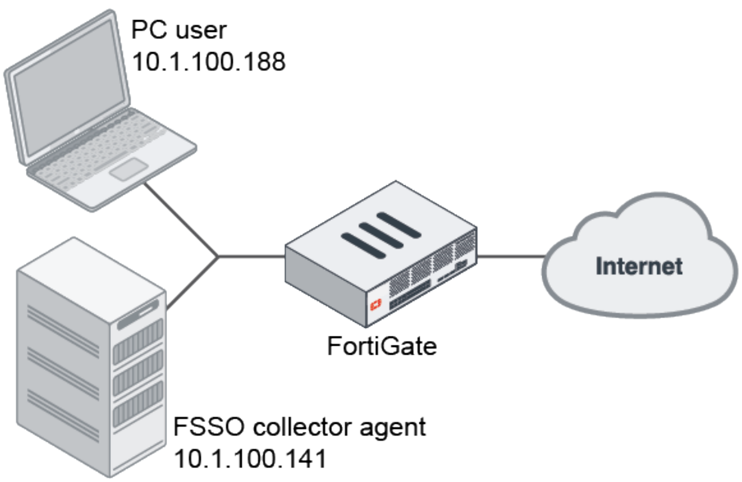
Configuring the FSSO timeout when the collector agent connection fails

The `logon-timeout` option is used to manage how long authenticated FSSO users on the FortiGate will remain on the list of authenticated FSSO users when a network connection to the collector agent is lost.

```
config user fsso
    edit <name>
        set server <string>
        set password <string>
        set logon-timeout <integer>
    next
end
```

`logon-timeout <integer>`	Enter the interval to keep logons after the FSSO server is down, in minutes (1 - 2880, default = 5).

Example



In this example, the logon timeout is set for four minutes.

To configure the FSSO logon timeout:

1. Set the timeout value:
```
config user fsso
    edit "ad"
        set server "10.1.100.141"
        set password ********
        set logon-timeout 4
    next
end
```
2. Log on to a PC with a valid FSSO user account.
3. Enable real-time debugging and check for authd polling collector agent information. During this time, the connection to the collector agent is lost:

```
# diagnose debug enable
# diagnose debug application authd -1
# diagnose debug application fssod -1
021-06-10 16:20:41 authd_timer_run: 2 expired
2021-06-10 16:20:41 authd_epoll_work: timeout 39970
2021-06-10 16:20:46 fsae_io_ctx_process_msg[ad]: received heartbeat 100031
2021-06-10 16:20:46 authd_epoll_work: timeout 1690
2021-06-10 16:20:47 authd_timer_run: 1 expired
2021-06-10 16:20:47 authd_epoll_work: timeout 39990
2021-06-10 16:20:56 fsae_io_ctx_process_msg[ad]: received heartbeat 100032
2021-06-10 16:20:56 authd_epoll_work: timeout 31550
2021-06-10 16:21:00 _event_error[ad]: error occurred in epoll_in: Success
2021-06-10 16:21:00 disconnect_server_only[ad]: disconnecting
2021-06-10 16:21:00 authd_timer_run: 1 expired
2021-06-10 16:21:00 authd_epoll_work: timeout 9620
```

4. After about three minutes, check that the FSSO user is still in the list of authenticated users and can connect to the internet:

```
# diagnose firewall auth l
10.1.100.188, TEST1
        type: fsso, id: 0, duration: 229, idled: 229
        server: ad
        packets: in 0 out 0, bytes: in 0 out 0
        user_id: 16777219
        group_id: 3 33554433
        group_name: ad CN=GROUP1,OU=TESTING,DC=FORTINET-FSSO,DC=COM

----- 1 listed, 0 filtered ------
```

5. After four minutes, check the debugs again. Note that the FSSO users are cleared:

```
...
2021-06-10 16:24:57 authd_timer_run: 3 expired
2021-06-10 16:24:57 authd_epoll_work: timeout 60000
2021-06-10 16:24:59 [fsae_db_logoff:248]: vfid 0, ip 10.1.100.188, id(0), port_range_sz
(0)
2021-06-10 16:24:59 [authd_fp_notify_logoff:444]: vfid 0, ip 10.1.100.188, id 0
2021-06-10 16:24:59 [authd_fp_on_user_logoff:412]: vfid 0, ip 10.1.100.188
2021-06-10 16:24:59 [authd_fp_on_user_logoff:412]: vfid 0, ip 10.1.100.188
2021-06-10 16:24:59 [authd_fp_on_user_logoff:412]: vfid 0, ip 10.1.100.188
2021-06-10 16:24:59 [authd_fpc_on_msg:545]: code 0, type 132, len 28 seq 0
2021-06-10 16:24:59 [authd_fp_on_user_logoff:412]: vfid 0, ip 10.1.100.188
2021-06-10 16:24:59 authd_epoll_work: timeout 21990
```

```
# diagnose firewall auth l

----- 0 listed, 0 filtered ------
```

After the connection to the collector agent is restored, all users remain in the list of authenticated users and are synchronized to the FortiGate. The users do not need to log in again for authentication.

Authentication policy extensions

By default, unauthenticated traffic is permitted to fall to the next policy. This means that unauthenticated users are only forced to authenticate against a policy when there are no other matching policies. To avoid this, you can force authentication to always take place.

To set that authentication requirement:

```
config user setting
    set auth-on-demand {always | implicitly}
end
```

Where:

`always`	Always trigger firewall authentication on demand.
`implicitly` (default)	Implicitly trigger firewall authentication on demand. This is the default setting (and the behavior in FortiOS 6.0 and earlier).

In the following example, authentication is required; traffic that would otherwise be allowed by the second policy is instead blocked by the first policy.

To use forced authentication:

```
config user setting
    set auth-on-demand always
end

config firewall policy
    edit 1
        set name "QA to Database"
        set srcintf "port10"
        set dstintf "port9"
        set srcaddr "QA_subnet"
        set dstaddr "Database"
        set action accept
        set schedule "always"
        set service "ALL"
        set fsso disable
        set groups "qa_group"
        set nat enable
    next
    edit 2
        set name "QA to Internet"
        set srcintf "port10"
        set dstintf "port9"
        set srcaddr "QA_subnet"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set fsso disable
        set nat enable
```

```
    next
end
```

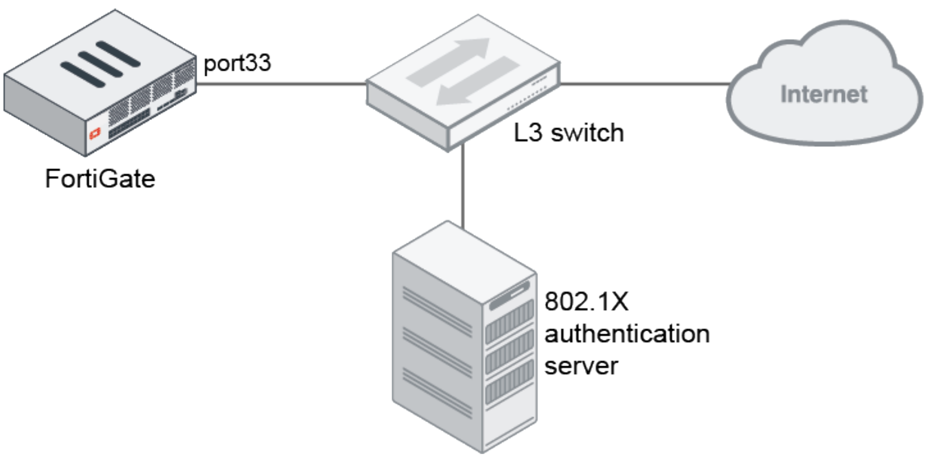
Configuring a FortiGate interface to act as an 802.1X supplicant

A FortiGate interface can be configured to act as a 802.1X supplicant. The settings can be enabled on the network interface in the CLI. The EAP authentication method can be either PEAP or TLS using a user certificate.

```
config system interface
    edit <interface>
        set eap-supplicant {enable | disable}
        set eap-method {peap | tls}
        set eap-identity <identity>
        set eap-password <password>
        set eap-ca-cert <CA_cert>
        set eap-user-cert <user_cert>
    next
end
```

Example

In this example, the FortiGate connects to an L3 switch that is not physically secured. All devices that connect to the internet through the L3 switch must be authenticated with 802.1X on the switch port by either a username and password (PEAP), or a user certificate (TLS). Configuration examples for both EAP authentication methods on port33 are shown.



To configure EAP authentication with PEAP:

1. Configure the interface:

```
config system interface
    edit "port33"
        set vdom "vdom1"
        set ip 7.7.7.2 255.255.255.0
        set allowaccess ping https ssh snmp http telnet fgfm radius-acct probe-response
fabric
```

```
        set stpforward enable
        set type physical
        set snmp-index 42
        set eap-supplicant enable
        set eap-method peap
        set eap-identity "test1"
        set eap-password **********
    next
end
```

2. Verify the interface's PEAP authentication details:

```
# diagnose test app eap_supp 2
Interface: port33
status:Authorized
method: PEAP
identity: test1
ca_cert:
client_cert:
private_key:
last_eapol_src =70:4c:a5:3b:0b:c6
```

Traffic is able to pass because the status is authorized.

To configure EAP authentication with TLS:

1. Configure the interface:

```
config system interface
    edit "port33"
        set vdom "vdom1"
        set ip 7.7.7.2 255.255.255.0
        set allowaccess ping https ssh snmp http telnet fgfm radius-acct probe-response
fabric
        set stpforward enable
        set type physical
        set snmp-index 42
        set eap-supplicant enable
        set eap-method tls
        set eap-identity "test2@fortiqa.net"
        set eap-ca-cert "root_G_CA_Cert_1.cer"
        set eap-user-cert "root_eap_client_global.cer"
    next
end
```

2. Verify the interface's TLS authentication details:

```
# diagnose test application eap_supp 2
Interface: port33
status:Authorized
method: TLS
identity: test2@fortiqa.net
ca_cert: /etc/cert/ca/root_G_CA_Cert_1.cer
client_cert: /etc/cert/local/root_eap_client_global.cer
private_key: /etc/cert/local/root_eap_client_global.key
last_eapol_src =70:4c:a5:3b:0b:c6
```

Traffic is able to pass because the status is authorized.

Include usernames in logs

Usernames can be included in logs, instead of just IP addresses. The benefits of doing this include:

- FortiOS monitors and FortiAnalyzer reports display usernames instead of IP addresses, allowing you to quickly determine who the information pertains to. Without the usernames, it is difficult to correlate the IP addresses with specific users.
- User activity can be correlated across multiple IP addresses.

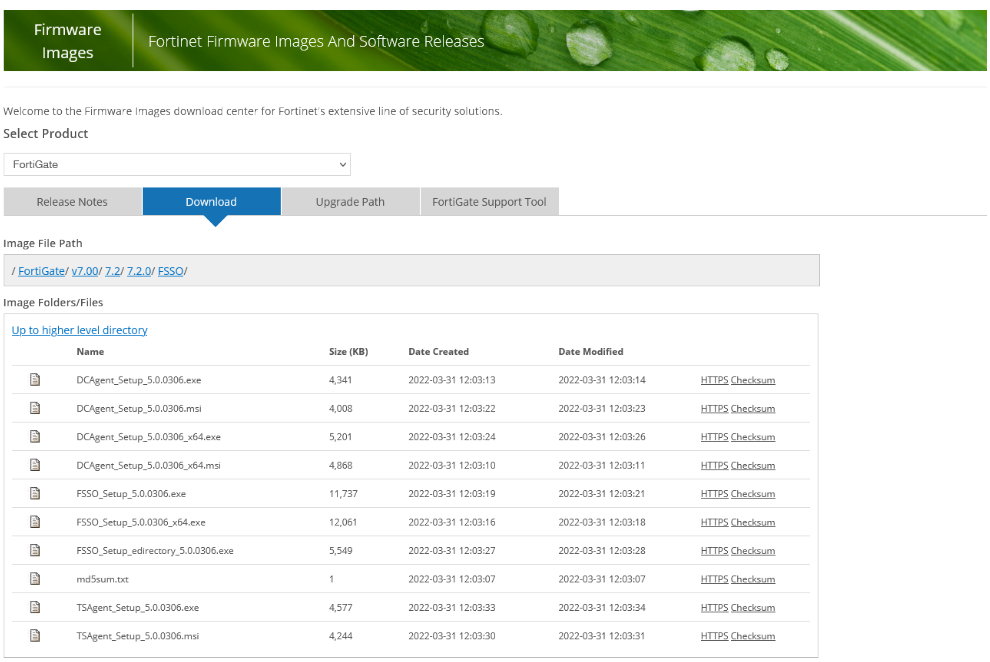
  For example, if DHCP is used a user might receive different IP addresses every day, making it difficult to track a specific user by specifying an IP address as the match criterion.

In this example, a collector agent (CA) is installed on a Windows machine to poll a domain controller (DC) agent (seeFSSO on page 1911 for more information). On the FortiGate, an external connector to the CA is configured to receives user groups from the DC agent. The received group or groups are used in a policy, and some examples of the usernames in logs, monitors, and reports are shown.

Install and configure FSSO Agent

To download the FSSO agent:

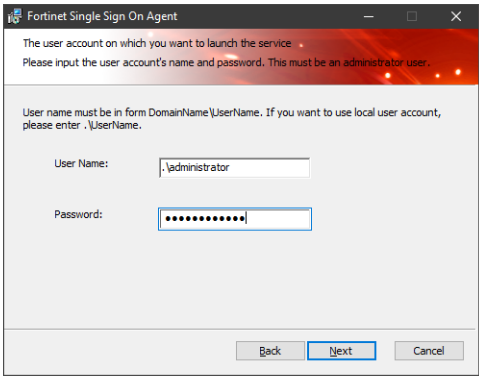
1. Sign in to your FortiCloud account.
2. Go to *Support > Firmware Download* and select the *Download* tab.
3. Browse to the appropriate directory for the version of the FSSO agent that you need to download.



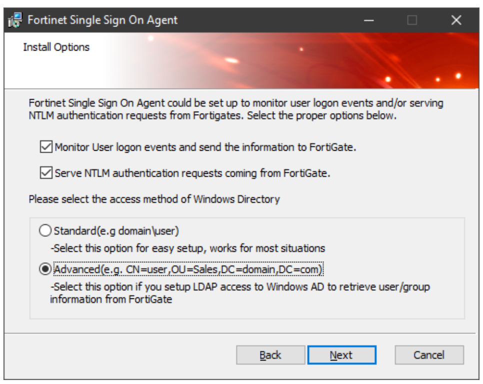
4. Click *HTTPS* to download the appropriate *FSSO_Setup* file.

To install the FSSO agent:

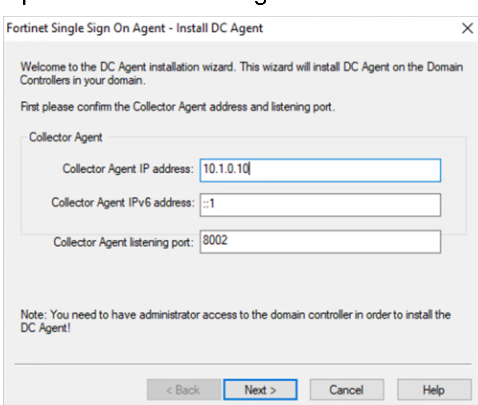
1. Run the *FSSO_Setup* file with administrator privileges.
2. Click *Next*, accept the terms of the license agreement, and click *Next* again.
3. Select the installation directory, or use the default location, then click *Next*.
4. Enter the User Name and Password, then click *Next*.



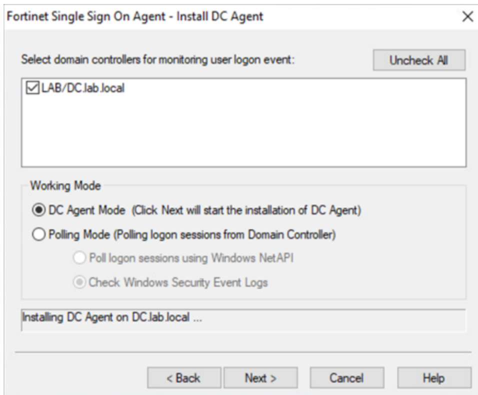
5. On the *Install Options*, select *Advanced*, then click *Next*.



6. Click *Install*.
7. After the FSSO Agent installs, run *Install DC Agent*.
8. Update the Collector Agent IP address and listening port as needed, then click *Next*.



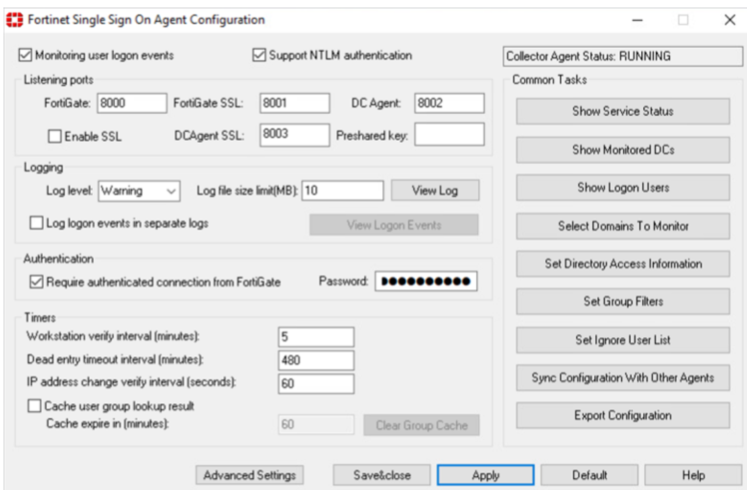
9. Select the domain, in this example *LAB:lab.local*, then click *Next*.
10. Set the *Working Mode* to *DC Agent Mode*, then click *Next* to install the agent.



11. After the DC agent mode installation finishes, Reboot the DC to complete the setup.

To configure the FSSO agent:

1. Open the FSSO agent.
2. Enable *Require authentication from FortiGate* and enter a password for FortiGate authentication.



3. Click *Set Group Filters*, and create a default group filter to limit the groups that are sent to the FortiGate.
4. Click *Save&close*.

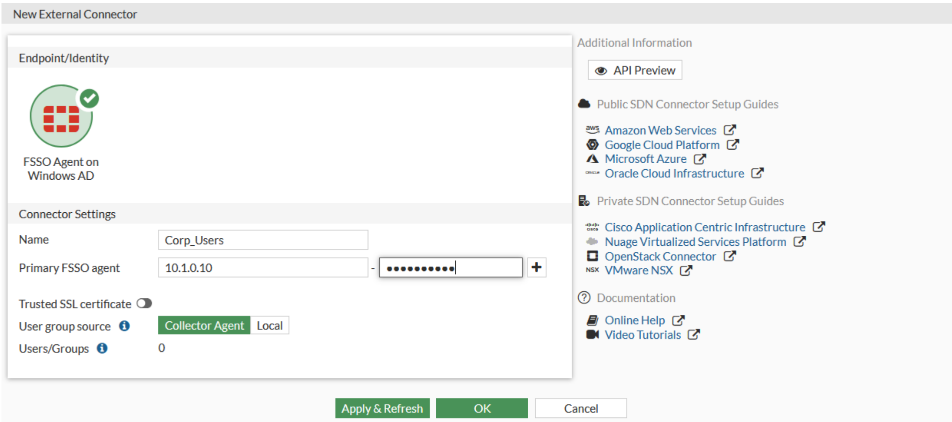
Configure the FortiGate

Create an external connector to the FSSO agent to receive the AD user groups. Add the user group or groups as the source in a firewall policy to include usernames in traffic logs. Enable security profiles, such as web filter or antivirus, in the policy to include the usernames in UTM logs.

Event logs include usernames when the log is created for a user action or interaction, such as logging in or an SSL VPN connection.

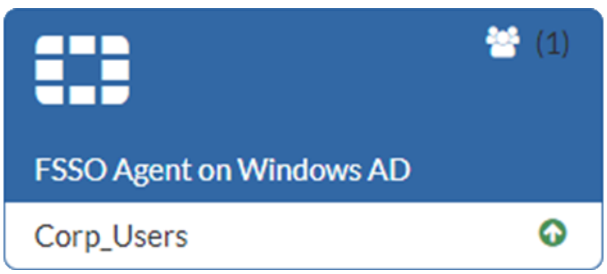
To create an external connector:

1. On the FortiGate, go to *Security Fabric > External Connectors*.
2. Click *Create New* and select *FSSO Agent on Windows AD*.
3. Set the *Primary FSSO agent* to the previously configured *Collector Agent IP address* and authentication password.



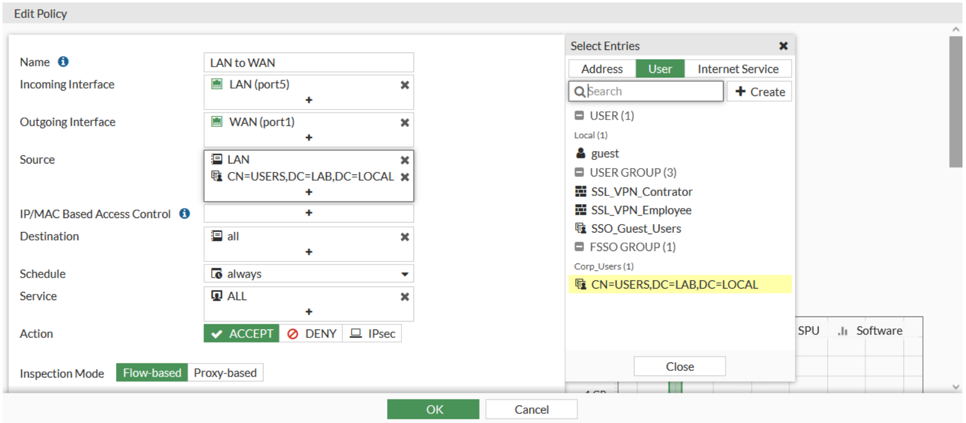
4. Click *OK*

   The connector shows a green arrow when the connection is established, and a number in the top right indicating the number of AD groups received from the DC agent. Edit the connector to view the user groups.



To configure a policy with an imported user group and web filter in the GUI:

1. Go to *Policy & Objects > Firewall Policy*.
2. Edit an existing policy, or create a new one. See Firewall policy on page 759 for information.
3. Add the FSSO groups or groups as sources:
   a. Click in the *Source* field.
   b. Select the *User* tab.
   c. Select the group or groups.



   d. Click *Close*.
4. Under *Security Profiles*, enable *Web Filter* and select a profile that monitors or blocks traffic, such as the *monitor-all* profile. See Web filter on page 1122 for information.
5. Click *OK*.

To configure a policy with an imported user group and web filter in the CLI:

```
config firewall policy
    edit 0
        set name "LAN to WAN"
        set srcintf "port5"
        set dstintf "port1"
        set action accept
        set srcaddr "LAN"
        set dstaddr "all"
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set ssl-ssh-profile "certificate-inspection"
        set webfilter-profile "monitor-all"
        set logtraffic all
        set nat enable
        set fsso-groups "CN=USERS,DC=LAB,DC=LOCAL"
    next
end
```

Log, monitor, and report examples

For more information about logs, see the FortiOS Log Message Reference.

Traffic logs:

Without a web filter profile applied:

```
date=2022-05-24 time=13:50:47 eventtime=1653425447661722283 tz="-0700" logid="0000000015"
type="traffic" subtype="forward" level="notice" vd="root" srcip=10.1.0.11 identifier=0
srcintf="port5" srcintfrole="lan" dstip=192.168.2.200 dstintf="port1" dstintfrole="wan"
srccountry="Reserved" dstcountry="Reserved" sessionid=708558 proto=1 action="start"
policyid=15 policytype="policy" poluuid="5bf426fe-794b-51ec-dedf-4318a843c5b5"
```

```
policyname="LAN to WAN" user="USER2" authserver="Corp_Users" service="PING" trandisp="snat"
transip=192.168.2.99 transport=0 duration=0 sentbyte=0 rcvdbyte=0 sentpkt=0 rcvdpkt=0
appcat="unscanned"
```

With a web filter profile applied:

```
date=2022-05-25 time=12:16:54 id=7101754911016091650 itime=2022-05-25 12:16:07 euid=1039
epid=1037 dsteuid=3 dstepid=101 type=traffic subtype=forward level=notice action=close
utmaction=allow policyid=15 sessionid=683 srcip=10.1.0.11 dstip=104.26.1.188
transip=192.168.2.99 srcport=64494 dstport=443 transport=64494 trandisp=snat duration=7
proto=6 sentbyte=1855 rcvdbyte=18631 sentpkt=16 rcvdpkt=21 logid=0000000013 user=USER2
group=CN=USERS,DC=LAB,DC=LOCAL service=HTTPS app=HTTPS appcat=unscanned srcintfrole=lan
dstintfrole=wan srcserver=0 policytype=policy eventtime=1653506215490475553 countweb=1
poluuid=5bf426fe-794b-51ec-dedf-4318a843c5b5 srcmac=00:0c:29:5e:f5:25
mastersrcmac=00:0c:29:5e:f5:25 srchwvendor=VMware srchwversion=Workstation pro
srcfamily=Virtual Machine srcswversion=10 devtype=Server osname=Windows srccountry=Reserved
dstcountry=United States srcintf=port5 dstintf=port1 authserver=Corp_Users policyname=LAN to
WAN hostname=www.yellow.com catdesc=Reference tz=-0700 devid=FGVM01TM22000459 vd=root
dtime=2022-05-25 12:16:54 itime_t=1653506167
```

UTM log:

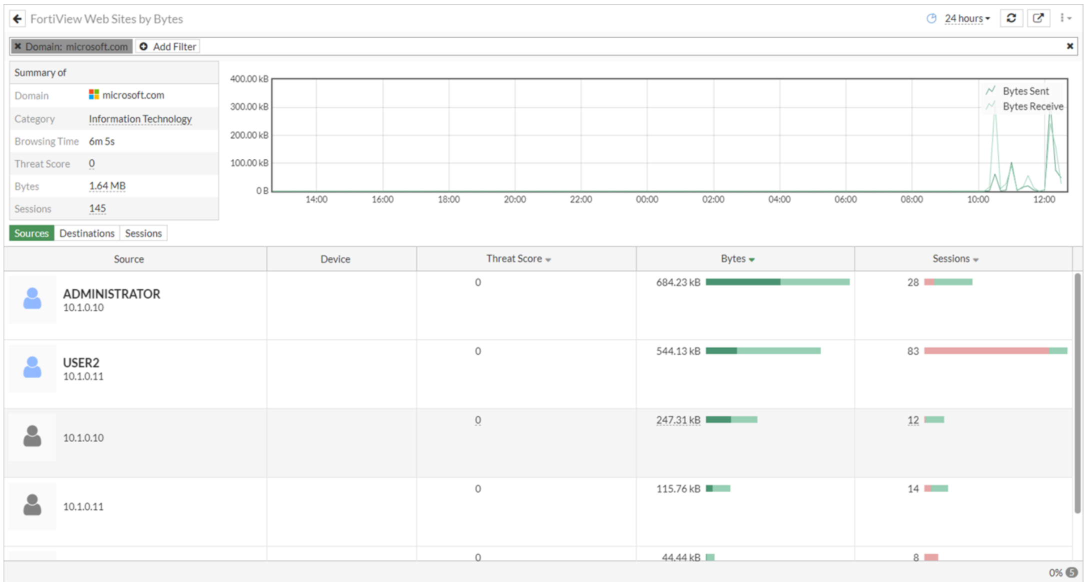
```
date=2022-05-25 time=12:16:46 id=7101754876656353280 itime=2022-05-25 12:15:59 euid=1039
epid=1037 dsteuid=3 dstepid=101 type=utm subtype=webfilter level=notice action=passthrough
sessionid=683 policyid=15 srcip=10.1.0.11 dstip=104.26.1.188 srcport=64494 dstport=443
proto=6 cat=39 logid=0317013312 service=HTTPS user=USER2 group=CN=USERS,DC=LAB,DC=LOCAL
eventtime=1653506207694977460 sentbyte=548 rcvdbyte=0 srcintfrole=lan dstintfrole=wan
direction=outgoing method=domain reqtype=direct url=https://www.yellow.com/
hostname=www.yellow.com profile=default catdesc=Reference eventtype=ftgd_allow srcintf=port5
dstintf=port1 authserver=Corp_Users msg=URL belongs to an allowed category in policy tz=-
0700 srcuuid=41cad638-794b-51ec-a8c9-8128712cb495 dstuuid=e1067f08-8e38-51eb-4b07-
64f219140388 policytype=policy srccountry=Reserved dstcountry=United States
poluuid=5bf426fe-794b-51ec-dedf-4318a843c5b5 devid=FGVM01TM22000459 vd=root dtime=2022-05-25
12:16:46 itime_t=1653506159
```

Event log:

```
date=2019-05-13 time=11:20:54 logid="0100032001" type="event" subtype="system"
level="information" vd="vdom1" eventtime=1557771654587081441 logdesc="Admin login
successful" sn="1557771654" user="admin" ui="ssh(172.16.1.1)" method="ssh"
srcip=172.16.200.254 dstip=172.16.200.2 action="login" status="success" reason="none"
profile="super_admin" msg="Administrator admin logged in successfully from ssh
(172.16.200.254)"
```

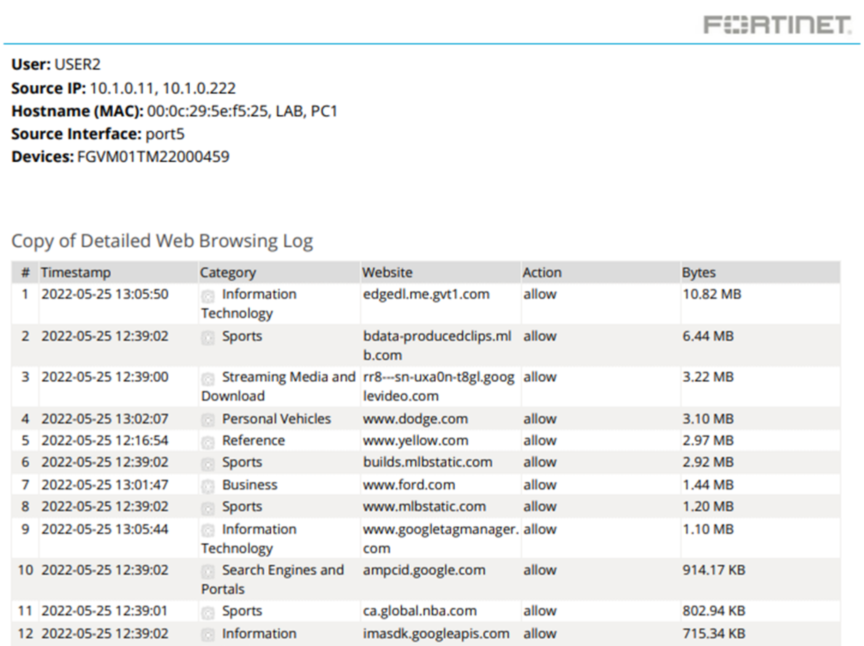
FortiOS monitors:

The *FortiView Web Sites by Bytes* monitor shows a list of visited websites. Double click a specific domain (or manually create a filter), such as *microsoft.com*, to see a breakdown of the usernames and IP addresses that visited that domain. See Monitors on page 106 for more information.



FortiAnalyzer reports:

The *User Detailed Browsing Log* report require a username or IP address to run. If a username is used, the report includes logs related to that user regardless of their IP address. For example, the following report show two source IP addresses:

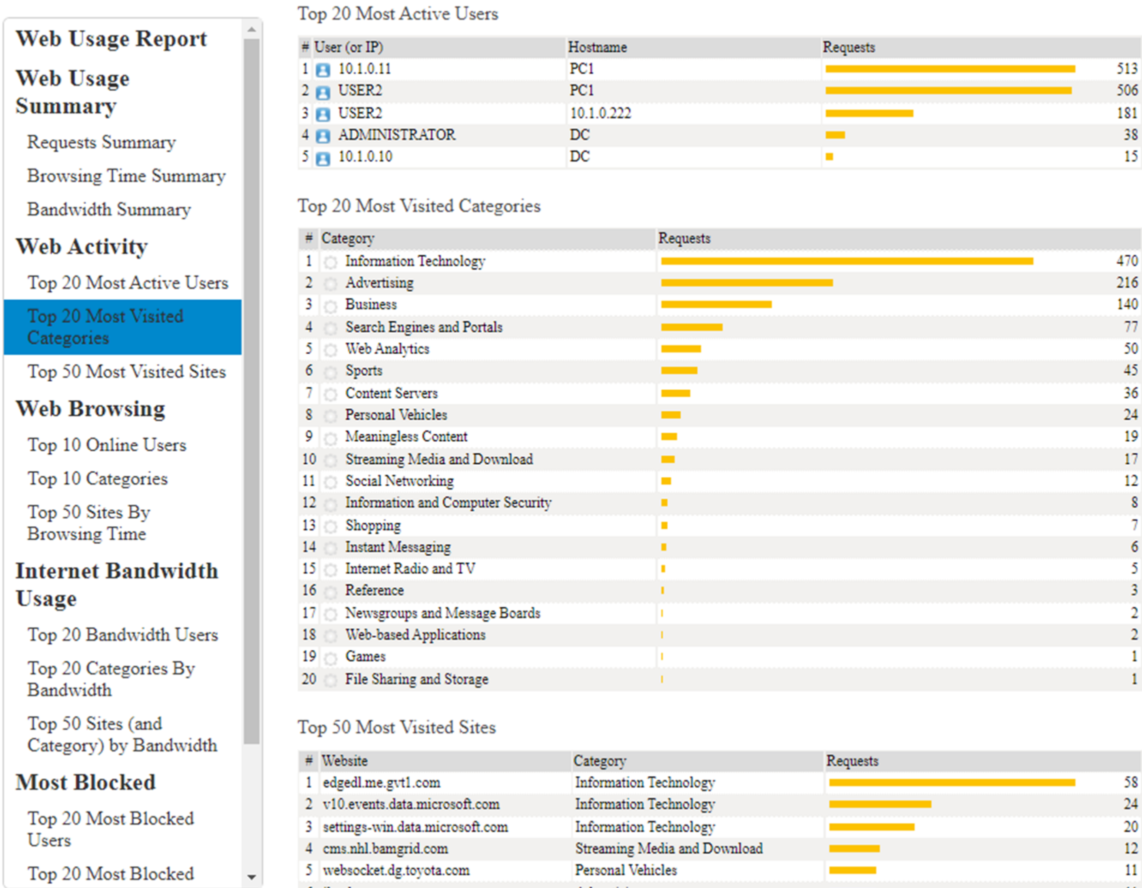


User: USER2
Source IP: 10.1.0.11, 10.1.0.222
Hostname (MAC): 00:0c:29:5e:f5:25, LAB, PC1
Source Interface: port5
Devices: FGVM01TM22000459

Copy of Detailed Web Browsing Log

#	Timestamp	Category	Website	Action	Bytes
1	2022-05-25 13:05:50	Information Technology	edgedl.me.gvt1.com	allow	10.82 MB
2	2022-05-25 12:39:02	Sports	bdata-producedclips.mlb.com	allow	6.44 MB
3	2022-05-25 12:39:00	Streaming Media and Download	rr8---sn-uxa0n-t8gl.googlevideo.com	allow	3.22 MB
4	2022-05-25 13:02:07	Personal Vehicles	www.dodge.com	allow	3.10 MB
5	2022-05-25 12:16:54	Reference	www.yellow.com	allow	2.97 MB
6	2022-05-25 12:39:02	Sports	builds.mlbstatic.com	allow	2.92 MB
7	2022-05-25 13:01:47	Business	www.ford.com	allow	1.44 MB
8	2022-05-25 12:39:02	Sports	www.mlbstatic.com	allow	1.20 MB
9	2022-05-25 13:05:44	Information Technology	www.googletagmanager.com	allow	1.10 MB
10	2022-05-25 12:39:02	Search Engines and Portals	ampcid.google.com	allow	914.17 KB
11	2022-05-25 12:39:01	Sports	ca.global.nba.com	allow	802.94 KB
12	2022-05-25 12:39:02	Information	imasdk.googleapis.com	allow	715.34 KB

The *Web Usage* report includes all usernames and IP addresses that match the specified conditions, like most visited categories.



See Reports in the FortiAnalyzer Administration guide for more information.

Wireless configuration

See the FortiWiFi and FortiAP Configuration Guide.

Switch Controller

Use the Switch Controller function, also known as FortiLink, to remotely manage FortiSwitch units. In the commonly-used layer 2 scenario, the FortiGate that is acting as a switch controller is connected to distribution FortiSwitch units. The distribution FortiSwitch units are in the top tier of stacks of FortiSwitch units and connected downwards with Convergent or Access layer FortiSwitch units. To leverage CAPWAP and the Fortinet proprietary FortiLink protocol, set up data and control planes between the FortiGate and FortiSwitch units.

FortiLink allows administrators to create and manage different VLANs, and apply the full-fledged security functions of FortiOS to them, such as 802.1X authentication and firewall policies. Most of the security control capabilities on the FortiGate are extended to the edge of the entire network, combining FortiGate, FortiSwitch, and FortiAP devices, and providing secure, seamless, and unified access control to users.

See FortiSwitch devices managed by FortiOS.

System

This topic contains information about FortiGate administration and system configuration that you can do after installing the FortiGate in your network.

Basic system settings

Administrators

By default, FortiGate has an administrator account with the username *admin* and no password. See Administrators on page 1937 for more information.

Administrator profiles

An administrator profile defines what the administrator can see and do on the FortiGate. See Administrator profiles on page 1949 for more information.

Password policy

Set up a password policy to enforce password criteria and change frequency. See Password policy on page 1946 for more information.

Interfaces

Physical and virtual interface allow traffic to flow between internal networks, and between the internet and internal networks. See Interfaces on page 135 for more information.

Advanced system settings

SNMP

The simple network management protocol (SNMP) allows you to monitor hardware on your network. See SNMP on page 2124 for more information.

DHCP server

You can configure one or more DHCP servers on any FortiGate interface. See DHCP server on page 287 for more information.

VDOM

You can use virtual domains (VDOMs) to divide a FortiGate into multiple virtual devices that function independently. See Virtual Domains on page 1994 for more information.

High availability

You can configure multiple FortiGate devices, including private and public cloud VMs, in HA mode. See High Availability on page 2030 for more information.

Certificates

You can manage certificates on the FortiGate. See Certificates on page 2164 for more information.

Operating modes

A FortiGate or VDOM (in multi-vdom mode) can operate in either NAT/route mode or transparent mode.

NAT/route mode

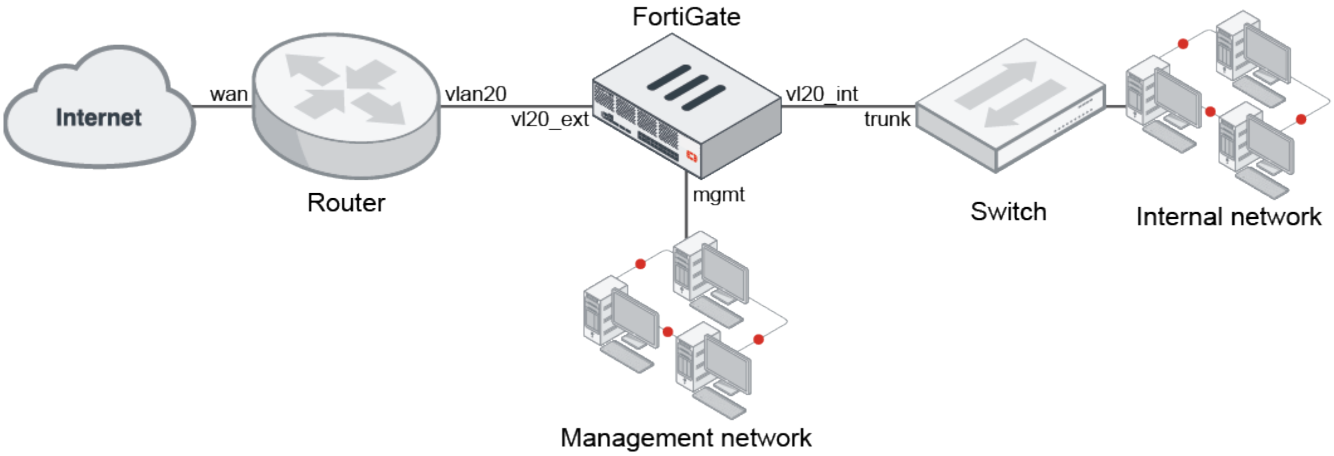
The FortiGate or VDOM is installed as a gateway or router between multiple networks, such as a private network and the internet. One function of NAT/route mode is to allow the FortiGate to hide the IP addresses on the private network using NAT. NAT/route mode can also be used to connect to multiple ISPs in an SD-WAN setup, and to route traffic between different networks. .

By default, new VDOMs are set to NAT/route operation mode.

Transparent mode

The FortiGate or VDOM operates in layer 2 to forward traffic between network devices such as routers, firewalls, and switches. For example. it can be installed inline between a router and a switch to perform security scanning without changing the network topology or modifying the IP addresses. When you add a FortiGate that is in transparent mode to a network, it only needs to be provided with a management IP address in order to access the device. It is recommended that a dedicated interface is used to connect to the management network in transparent mode.

The following topology is an example of a transparent mode FortiGate inserted inline between a router and a switch:



Using transparent mode VDOMs is recommended when multiple VLANs pass through the FortiGate. Otherwise, they must be separated into different forwarding domains within the same VDOM.

Changing modes

The following is a sample configuration for changing from NAT/route operation mode to transparent operation mode in the CLI:

```
config system settings
    set opmode transparent
    set manageip <IP_address>
    set gateway <gateway_address>
end
```

The gateway setting is optional. However, once the operation mode is changed from NAT/route to transparent, the gateway configuration is found under the static router settings:

```
config router static
    edit <seq-num>
        set gateway <IP_address>
    next
end
```

The following is a sample configuration for changing from transparent operation to NAT/route operation mode in the CLI:

```
config system settings
    set opmode nat
    set ip <IP_address>
    set device <interface>
    set gateway <gateway_address>
end
```

The IP and device settings are mandatory. Once the operation mode is changed from transparent to NAT/route, the IP address configuration is found under the corresponding interface settings:

```
config system interface
    edit <interface>
        set ip <IP_address>
    next
end
```

The gateway setting is optional. However, once the operation mode is changed, the gateway configuration is found under the static router settings:

```
config router static
    edit <seq-num>
        set gateway <IP_address>
        device <interface>
    next
end
```

Administrators

By default, FortiGate has an administrator account with the username *admin* and no password. To prevent unauthorized access to the FortiGate, this account must be protected with a password. Additional administrators can be added for various functions, each with a unique username, password, and set of access privileges.

The following topics provide information about administrators:

- Local authentication on page 1937
- Remote authentication for administrators on page 1938
- Administrator account options on page 1941
- REST API administrator on page 1943
- SSO administrators on page 1945
- FortiCloud SSO on page 1945
- Password policy on page 1946
- Public key SSH access on page 1948

Local authentication

By default, FortiGate has one super admin named `admin`. You can create more administrator accounts with different privileges.

To create an administrator account in the GUI:

1. Go to *System > Administrators*.
2. Select *Create New > Administrator*.

3. Specify the *Username*.

> Do not use the characters `< > ( ) # " '` in the administrator username.
>
> Using these characters in an administrator username might have a cross site scripting (XSS) vulnerability.

4. Set *Type* to *Local User*.
5. Set the password and other fields.
6. Click *OK*.

To create an administrator account in the CLI:

```
config system admin
    edit <admin_name>
        set accprofile <profile_name>
        set vdom <vdom_name>
        set password <password for this admin>
    next
end
```

Remote authentication for administrators

Administrators can use remote authentication, such as LDAP, RADIUS, and TACACS+ to connect to the FortiGate.

Configuring remote authentication with an LDAP server is shown. For more information about configuring LDAP, see Configuring an LDAP server on page 1803.

For information about configuring RADIUS or TACACS+ servers, see Configuring a RADIUS server on page 1819 and TACACS+ servers on page 1850. To use a RADIUS or TACACS+ server for remote authentication, configure the server, and then add it to the user group instead of the LDAP server.

Local logins can also be restricted when remote authentication servers are available, see Restricting logins from local administrator accounts when remote servers are available on page 1941.

Configuring remote authentication for administrators using LDAP includes the following steps:

1. Configuring the LDAP server on page 1938
2. Adding the LDAP server to a user group on page 1939
3. Configuring the administrator account on page 1939

Configuring the LDAP server

To configure the LDAP server in the GUI:

1. Go to *User & Authentication > LDAP Servers* and click *Create New*.
2. Enter the server *Name* and *Server IP/Name*.
3. Enter the *Common Name Identifier* and *Distinguished Name*.
4. Set the *Bind Type* to *Regular* and enter the *Username* and *Password*.
5. Click *OK*.

To configure the LDAP server in the CLI:

```
config user ldap
    edit <name>
        set server <server_ip>
        set cnid "cn"
        set dn "dc=XYZ,dc=fortinet,dc=COM"
        set type regular
        set username "cn=Administrator,dc=XYA, dc=COM"
        set password <password>
    next
end
```

Adding the LDAP server to a user group

After configuring the LDAP server, create a user group that includes that LDAP server.

To create a user group in the GUI:

1. Go to *User & Authentication* > *User Groups* and click *Create New*.
2. Enter a *Name* for the group.
3. In the *Remote groups* section, select *Create New*.
4. Select the *Remote Server* from the dropdown list.
5. Click *OK*.

To create a user group in the CLI:

```
config user group
    edit <name>
        set member <ldap_server_name>
    next
end
```

Configuring the administrator account

After configuring the LDAP server and adding it to a user group, create a new administrator. For this administrator, instead of entering a password, use the new user group for authentication.

A remote authentication server can allow authentication of either a single user or any user from a specified group.

Public key infrastructure (PKI) administrator authentication requires a PKI user instead of a remote server. For information about creating a PKI user, see Configuring a PKI user on page 1901.

To create an administrator to match a single user in the GUI:

1. Go to *System* > *Administrators* and click *Create New* > *Administrator*.
2. Specify the *Username*.

   This username is used when the administrator logs in, and is what FortiOS sends to the remote authentication server for authorization.
3. Set *Type* to *Match a user on a remote server group*.

4. In *Remote User Group*, select the user group that you created.
5. Select an *Administrator Profile*.
6. Enter a *Backup Password*, to be used if the remote authentication server is unreachable.
7. Click *OK*.

To create an administrator match a single user in the CLI:

```
config system admin
    edit <name>
        set remote-auth enable
        set accprofile super_admin
        set remote-group <ldap_group_name>
        set password **********
    next
end
```

To create an administrator to match all users in a remote server group in the GUI:

1. Go to *System* > *Administrators* and click *Create New* > *Administrator*.
2. Specify the *Username*.

   This username is only used to identify this administrator group. Administrators can log in with any username in the remote user group.
3. Set *Type* to *Match all users in a remote server group*.
4. In *Remote User Group*, select the user group that you created.
5. Select an *Administrator Profile*.
6. Click *OK*.

To create an administrator to match all users in a remote server group in the CLI:

```
config system admin
    edit <name>
        set remote-auth enable
        set accprofile super_admin
        set wildcard enable
        set remote-group <ldap_group_name>
    next
end
```

To create an administrator that uses a PKI group in the GUI:

1. Go to *System* > *Administrators* and click *Create New* > *Administrator*.
2. Specify the *Username*.
3. Set *Type* to *Use public key infrastructure (PKI) group*.
4. In *Remote User Group*, select the user group that you created.
5. Select an *Administrator Profile*.
6. Click *OK*.

To create an administrator that uses a PKI group in the CLI:

```
config system admin
    edit <name>
        set remote-auth enable
        set accprofile super_admin
        set peer-group <pki_group_name>
    next
end
```

Restricting logins from local administrator accounts when remote servers are available

Logins from local administrator accounts can be restricted when remote servers are available. When enabled, FortiOS will check if all of the remote servers used by administrators are down before allowing a local administrator to log in. This option is applied globally, and is disabled by default.

To restrict local administrator authentication when a remote authentication server available:

```
config system global
    set admin-restrict-local enable
end
```

Administrator account options

Options to further define the access and abilities of an administrator account include:

- Multi-factor authentication on page 1941
- Restricting logins to trusted hosts on page 1943
- Restricting administrators to guest account provisioning on page 1943
- Global and VDOM administrators on page 1943

Multi-factor authentication

Multi-factor authentication (MFA) requires authenticating administrators to supply more than one factor to identify themselves in addition to their password, such as a FortiToken.

Before enabling MFA, it is recommended that you create second administrator account that is configured to guarantee administrator access to the FortiGate if you are unable to authenticate on the main account for any reason.

Multi-factor authentication options include:

- FortiToken
- FortiToken Cloud
- Email
- SMS

FortiToken

To associate a FortiToken to an administrator account using the GUI:

1. Ensure that you have successfully added your FortiToken serial number to FortiOS and that its status is *Available*.
2. Go to *System* > *Administrators*. Edit the admin account. This example assumes that the account is fully configured except forMFA.
3. Enable *Two-factor Authentication* and for *Authentication Type*, select *FortiToken*.
4. From the *Token* dropdown list, select the FortiToken serial number.
5. In the *Email Address* field, enter the administrator's email address.
6. Click *OK*.

For a mobile token, click *Send Activation Code* to send the activation code to the configured email address. The admin uses this code to activate their mobile token. You must have configured an email service in *System* > *Settings* to send the activation code.

To associate a FortiToken to an administrator account using the CLI:

```
config system admin
   edit <username>
      set password "myPassword"
      set two-factor fortitoken
      set fortitoken <serial_number>
      set email-to "username@example.com"
   next
end
```

The `fortitoken` keyword is not visible until you select `fortitoken` for the `two-factor` option.

Before you can use a new FortiToken, you may need to synchronize it due to clock drift.

FortiToken Cloud

FortiToken Cloud is an Identity and Access Management as a Service (IDaaS) cloud service provided by Fortinet. It enables FortiGate and FortiAuthenticator customers to add MFA for their users using Mobile or Hard tokens.

For more information, see Getting started—FGT-FTC users in the FortiToken Cloud Administration Guide.

Email

Enter an email address to send an MFA code to that address.

SMS

Enable *SMS* then select the *Country Dial Code* and enter the *Phone Number* (`sms-phone` in the CLI) to send an MFA code to.

SMS messages can also be sent to the FortiGuard SMS server or a custom server.

```
config system admin
    edit "admin"
        ...
        set sms-server {fortiguard | custom}
        set sms-server-custom <string>
        ...
    next
end
```

Restricting logins to trusted hosts

Administrator accounts can be configured to only be accessible to a user using a trusted host. You can set a specific IP address for the trusted host, or use a subnet. Up to ten trusted hosts can be specified for an administrator.

When trusted hosts are defined for all of the administrators on the FortiGate, the administrative access on each interface will be restricted to the trusted hosts that are defined for the administrator, except for ping. If ping is enabled on an interface, it works regardless of the trusted hosts.

Restricting administrators to guest account provisioning

To simplify guest account creation, an administrator account can be created exclusively for guest user management. This allows new accounts to be created without requiring full administrative access to FortiOS.

When enabling this option, a guest group must be specified for the administrator to provision new accounts to. See Configuring guest user groups on page 1797 for information about creating such a group.

Global and VDOM administrators

When a FortiGate is in multi VDOM mode, it can be managed by either global or per-VDOM administrators. Each type of administrator will have a different view of the GUI that corresponds to their role. For more information, see Administrator roles and views on page 1996.

For information about configuring per-VDOM administrators, see Create per-VDOM administrators on page 2003.

REST API administrator

REST API administrator accounts are used for automated configuration, backup creation, and monitoring of the FortiGate.

For more information about the REST API, see the Fortinet Development Network (FNDN). Note that an account is required to access the FNDN.

To create a REST API admin in the GUI:

1. Go to *System > Administrators*.
2. Select *Create New > REST API Admin*.
3. Configure the administrator:

Username	The username of the administrator.

	Do not use the characters `< > ( ) # " '` in the administrator username. Using these characters in an administrator username might have a cross site scripting (XSS) vulnerability.
Administrator Profile	Where permissions for the REST API administrator are defined. A REST API administrator should have the minimum permissions required to complete the request.
PKI Group	Certificate matching is supported as an extra layer of security. Both the client certificate and token must match to be granted access to the API.
CORS Allow Origin	Cross Origin Resource Sharing (CORS) allows third-party web apps to make API requests to the FortiGate using the token.
Trusted Hosts	The following can be used to restrict access to FortiGate API: • Multiple trusted hosts/subnets can be configured • IPv6 hosts are supported • Allow all (0.0.0.0/0) is not allowed You need your *Source Address* to create the trusted host.

4. Click *OK*.
   An API token is generated. Make note of the token, as it is only shown once.

To create a REST API admin in the CLI:

1. Create the REST API administrator:

```
config system api-user
    edit "api-admin"
        set comments <string>
        set api-key ************
        set accprofile "API profile"
        set vdom "root"
        config trusthost
            edit 1
                set ipv4-trusthost <class_ip&net_netmask>
            next
            ...
        end
    next
end
```

2. Generate the API token:

```
execute api-user generate-key <API username>
```

   Make note of the token, as it is only shown once.

By default, The SSO administrator account can only be assigned the *admin_no_access* or *super_admin_readonly* profile. You can define a new administrator profile with the required permissions for the account. For example, you could use a specific API user to query the FortiGate for just their own status. In that case, the profile would be configured as read-only.

SSO administrators

SSO administrators are automatically created when the FortiGate acts as a SAML service provider (SP) with *SAML Single Sign-On* enabled in the Security Fabric settings.

On the system login page, an administrator can log in with their username and password against the root FortiGate acting as the identity provider (IdP) in the Security Fabric. After the first successful log in, this user is added to the administrators table (*System > Administrators* under *Single Sign-On Administrator*). The default profile selected is based on the SP settings (*Default admin profile*). See Configuring a downstream FortiGate as an SP on page 2319 for more information.

SSO administrators can be manually configured in FortiOS.

To manually configure an SSO administrator in the GUI:

1. Go to *System > Administrators* and click *Create New > SSO Admin*.
2. Enter the username.
3. Select an administrator profile.
4. Click *OK*.

To manually configure an SSO administrator in the CLI:

```
config system sso-admin
    edit <name>
        set accprofile <profile>
        set vdom <vdom>
    next
end
```

FortiCloud SSO

By default, the FortiGate is configured to allow administrators to log in using FortiCloud single sign-on. Both IAM and non-IAM users on the FortiCloud support portal are supported. Non-IAM users must be the FortiCloud account that the FortiGate is registered to.

To configure an IAM user in FortiCloud:

1. Log in to your FortiCloud account at support.fortinet.com.
2. Select *Services > IAM* and *click Add IAM user*.
3. See Adding an IAM user in the *FortiCloud Identity & Access Management (IAM)* guide for more information. The *Portal Permissions* for *SupportSite*, *IAMPortal*, and *FortiOS SSO* must be configured to allow portal access for administrators.

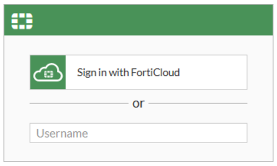
To manually enable FortiCloud single sign-on in the GUI:

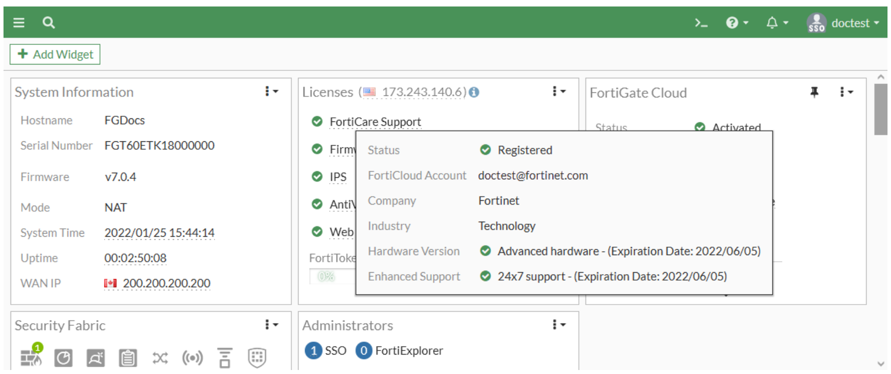
1. Log in to the FortiGate and go to *System > Settings*.
2. In the *Administration Settings* section, enable *Allow administrative login using FortiCloud SSO*.
3. Click *Apply*.

To manually enable FortiCloud single sign-on in the CLI:

```
config system global
    set admin-forticloud-sso-login {enable | disable}
end
```

To log in to the FortiGate with the FortiCloud user:

1. Go to the FortiGate log in screen.


2. Click *Sign in with FortiCloud*. The FortiCloud log in page opens.
3. Enter the FortiCloud account credentials and click *Login*.
You are logged in to the FortiOS GUI. The SSO username is shown in the top right corner of the GUI.


Password policy

Brute force password software can launch more than just dictionary attacks. It can discover common passwords where a letter is replaced by a number. For example, if `p4ssw0rd` is used as a password, it can be cracked.

Using secure passwords is vital for preventing unauthorized access to your FortiGate. When changing the password, consider the following to ensure better security:

- Do not use passwords that are obvious, such as the company name, administrator names, or other obvious words or phrases.
- Use numbers in place of letters, for example: `passw0rd`.
- Administrator passwords can be up to 64 characters.
- Include a mixture of numbers, symbols, and upper and lower case letters.
- Use multiple words together, or possibly even a sentence, for example: `correcthorsebatterystaple`.
- Use a password generator.
- Change the password regularly and always make the new password unique and not a variation of the existing password. For example, do not change from `password` to `password1`.
- Make note of the password and store it in a safe place away from the management computer, in case you forget it; or ensure at least two people know the password in the event one person becomes unavailable. Alternatively, have two different admin logins.

FortiGate allows you to create a password policy for administrators and IPsec pre-shared keys. With this policy, you can enforce regular changes and specific criteria for a password policy, including:

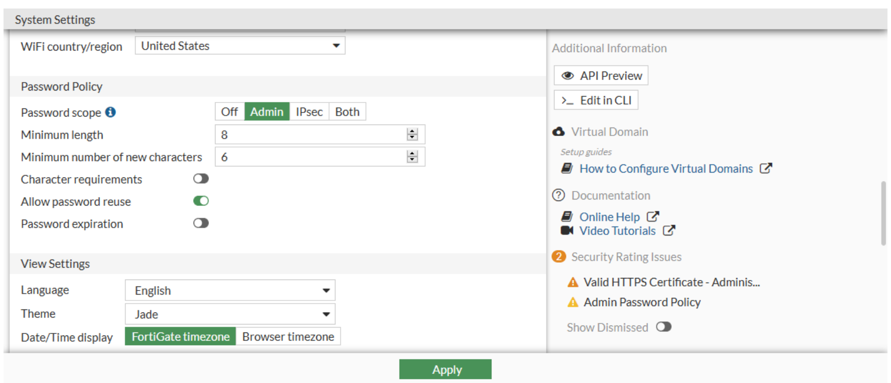
- The minimum length, between 8 and 64 characters.
- If the password must contain uppercase (A, B, C) and/or lowercase (a, b, c) characters.
- If the password must contain numbers (1, 2, 3).
- If the password must contain special or non-alphanumeric characters: !, @, #, $, %, ^, &, *, (, and)
- Where the password applies (admin or IPsec or both).
- The duration of the password before a new one must be specified.
- The minimum number of unique characters that a new password must include.

If you add a password policy or change the requirements on an existing policy, the next time that administrator logs into the FortiGate, the administrator is prompted to update the password to meet the new requirements before proceeding to log in.

For information about setting passwords, see Default administrator password on page 1974.

To create a system password policy the GUI:

1. Go to *System > Settings*.
2. In the *Password Policy* section, change the *Password scope* to *Admin*, *IPsec*, or *Both*.
3. Configure the password policy options.



4. Click *Apply*.

To create a system password policy the CLI:

```
config system password-policy
    set status {enable | disable}
    set apply-to {admin-password | ipsec-preshared-key}
    set minimum-length <8-128>
    set min-lower-case-letter <0-128>
    set min-upper-case-letter <0-128>
    set min-non-alphanumeric <0-128>
    set min-number <0-128>
    set min-change-characters <0-128>
    set expire-status {enable | disable}
    set expire-day <1-999>
    set reuse-password {enable | disable}
end
```

Public key SSH access

Public-private key pairs can be used to authenticate administrators connecting to the CLI using an SSH client. These keys can be RSA, ECDSA, or EdDSA.

Weigh the pros and cons of using key-pair authentication, versus passwords, when considering their use:

	Key-pair	Password
Pros	More secure (higher complexity)	Easy to remember
	Restricts logon to hosts that have the private key	Easy to update
	Never sent to the FortiGate	Can log in from any system
	Can add a password in addition to the key	
Cons	More complex to implement	Might be guessable or brute forced
	The private key is only as secure as the system storing it	Could be reused and compromised on another system
	More complicated to train users and administrators to use keys	Might be stored in plain text on an authenticating device (This does not apply to FortiGates)
		Can be phished or observed if written down

Key-pair authentication is often implemented when connecting to the FortiGate without any human interaction, such as when using a script. The script can leverage existing mechanisms to secure private keys, instead of trying to develop a way to securely store a username and password.

Generating the key pair

Key pairs can be generated and added in multiple different ways. This example shows generating a key pair using *PuTTY Key Generator* and adding the private key to the endpoint using *PuTTY Pageant*.

To create the key pair using PuTTY:

1. Download and install PuTTY.
2. Run *PuTTYgen.exe*.
3. Set *Type of key to generate* to *RSA*, *ECDSA*, or *EdDSA*.
4. Click *Generate*, then move the mouse cursor around in the blank space to generate randomness while the keys are generated.
5. Save both the public and private keys. Optionally, a key passphrase can be entered to protect the private key.

To add the public key to the FortiGate:

1. Delete the *Key comment*, then copy the public key from the *PuTTY Key Generator*.

   Conversely, you can also open the saved public key in Notepad, remove the line breaks from the key, then remove extraneous lines:

```
---- BEGIN SSH2 PUBLIC KEY ----
Comment: "rsa-key-2022XXXX"

---- END SSH2 PUBLIC KEY ----
```

2. Add the key to the FortiGate:

```
config system admin
    edit <admin>
        set ssh-public-key1 "<key_type> <key_value>"
    next
end
```

Where <key_value> is the copied key, and <key_type> depends on the type of key that was generated:

RSA	ssh-rsa
ECDSA	ecdsa-sha2-nistp256 ecdsa-sha2-nistp384 ecdsa-sha2-nistp521
EdDSA	ssh-ed25519

To add the private key to the endpoint:

1. Open *PuTTY Pageant*.
2. Click *Add Key* or *Add Key (encrypted)* and select the previously saved private key.
3. Click *Close*.

You can now log in to the FortiGate on an SSH connection without using a password.

If using PuTTY, the username can be entered under *Connection* > *Data* in the *Auto-login username* field.

The generated keys can also be used in a certificate to authenticate with the FortiGate. See Administrative access using certificates on page 2185 for information about generating and using certificates for administrative authentication.

Administrator profiles

Administrator profiles define what the administrator can do when logged into the FortiGate. When you set up an administrator account, you also assign an administrator profile which dictates what the administrator sees. Depending on the nature of the administrator's work, access level or seniority, you can allow them to view and configure as much or as little as is required. Access to CLI diagnose commands can also be disabled for global and VDOM level administrators.

By default, the FortiGate has an *admin* administrator account that uses the *super_admin* profile.

super_admin profile

This profile has access to all components of FortiOS, including the ability to add and remove other system administrators. For certain administrative functions, such as backing up and restoring the configuration, super_admin access is required. To ensure that there is always a method to administer the FortiGate, the super_admin profile cannot be deleted or modified.

> Lower level administrator profiles cannot backup or restore the FortiOS configuration.

The super_admin profile is used by the default admin account. It is recommended that you add a password and rename this account once you have set up your FortiGate. In order to rename the default account, a second admin account is required.

Creating customized profiles

To create a profile in the GUI:

1. Go to *System* > *Admin Profiles* and click *Create New*.
2. Configure the following settings:
   - Name
   - Access permissions
   - Usage of CLI diagnose commands
   - Override idle timeout
3. Click *OK*.

To create a profile in the CLI:

```
config system accprofile
    edit <name>
        set secfabgrp {none | read | read-write}
        set ftviewgrp {none | read | read-write}
        set authgrp {none | read | read-write}
        set sysgrp {none | read | read-write}
        set netgrp {none | read | read-write}
        set loggrp {none | read | read-write}
        set fwgrp {none | read | read-write}
        set vpngrp {none | read | read-write}
        set utmgrp {none | read | read-write}
        set wanoptgrp {none | read | read-write}
        set wifi {none | read | read-write}
        set admintimeout-override {enable | disable}
        set system-diagnostics {enable | disable}
    next
end
```

Edit profiles

To edit a profile in the GUI:

1. Go to *System > Admin Profiles*.
2. Select the profile to be edited and click *Edit*.
3. Make the required changes.
4. Click *OK* to save any changes.

To edit a profile in the CLI:

```
config system accprofile
    edit "sample"
        set secfabgrp read
    next
end
```

Delete profiles

To delete a profile in the GUI:

1. Go to *System > Admin Profiles*.
2. Select the profile to be deleted and click *Delete*.
3. Click *OK*.

To delete a profile in the CLI:

```
config system accprofile
    delete "sample"
end
```

Fabric Management

The *Fabric Management* page allows administrators to manage the firmware running on each FortiGate, FortiAP, and FortiSwitch in the Security Fabric, and to authorize and register these Fabric devices.

The *Fabric Management* page displays a summary of devices in the Security Fabric that includes:

- Total number of devices in the Fortinet Security Fabric and the types of devices
- Upgrade status
- Device name
- Device status
- Registration status
- Firmware version and maturity level
- Upgrade status

From the *Fabric Management* page, administrators can perform the following actions:

Upgrade	Use to upgrade firmware for the selected device. The *Upgrade* option uses released firmware images from FortiGuard. Alternately you can download a firmware file from the Fortinet Customer Service & Support website, and upload it for the upgrade process. See Upgrading individual device firmware on page 1955 and Upgrading individual device firmware by following the upgrade path (federated update) on page 1957.
Fabric Upgrade	Use to upgrade firmware for the root FortiGate as well as all Fabric devices. You can also use this option to upgrade firmware for a non-Security Fabric FortiGate with managed FortiSwitch and FortiAP devices. The *Fabric Upgrade* option uses released firmware images from FortiGuard. See Upgrading all device firmware on page 1958 and Upgrading all device firmware by following the upgrade path (federated update) on page 1960.
Register	Use the *Register* option to register a selected device to FortiCare. The FortiGate, and then its service contract, must be registered to have full access to Fortinet Customer Service and Support, and FortiGuard services. The FortiGate can be registered in either the FortiGate GUI or the FortiCloud support portal. The service contract can be registered from the FortiCloud support portal. See also Registration on page 55.
Authorize	Use the *Authorize* option to authorize a selected device for use. See Authorizing devices on page 1963.

Before you upgrade FortiGate firmware, it is recommended to learn about firmware updates and firmware maturity levels. See About firmware installations on page 1952 and Firmware maturity levels on page 1953.

This section also includes the following topics:

- Firmware upgrade notifications on page 1965
- Downloading a firmware image on page 1966
- Testing a firmware version on page 1968
- Installing firmware from system reboot on page 1969
- Restoring from a USB drive on page 1971
- Using controlled upgrades on page 1971
- Downgrading individual device firmware on page 1972

About firmware installations

Fortinet periodically updates the FortiGate firmware to include new features and resolve important issues. After you have registered your FortiGate unit, firmware updates are available from FortiGuard and from the Fortinet Customer Service & Support website.

Installing a new firmware image replaces the current antivirus and attack definitions, along with the definitions included with the firmware release that is being installing. After you install new firmware, make sure that the antivirus and attack definitions are up to date.

It is recommended to back up your configuration before making any firmware changes. You will be prompted to back up your configuration as part of the upgrade process. See also Configuration backups on page 65.

Before you install any new firmware, follow the below steps:

1. Understand the maturity level of the current and target firmware releases to help you determine whether to upgrade. See Firmware maturity levels on page 1953.
2. Review the Release Notes for a new firmware release.
3. Review the Supported Upgrade Paths.
4. Download a copy of the currently installed firmware, in case you need to revert to it. See Downloading a firmware image on page 1966 and Downgrading individual device firmware on page 1972 for details.
5. Have a plan in place in case there is a critical failure, such as the FortiGate not coming back online after the update. This could include having console access to the device (Connecting to the CLI on page 29), ensuring that your TFTP server is working (Installing firmware from system reboot on page 1969), and preparing a USB drive (Restoring from a USB drive on page 1971).
6. Back up the current configuration, including local certificates. The upgrade process prompts you to back up the current configuration. See also Configuration backups on page 65 for details.
7. Test the new firmware until you are satisfied that it applies to your configuration. See Testing a firmware version on page 1968 and Using controlled upgrades on page 1971 for details.

Installing new firmware without reviewing release notes or testing the firmware may result in changes to settings and unexpected issues.

Only FortiGate admin users and administrators whose access profiles contain system read and write privileges can change the FortiGate firmware.

Firmware maturity levels

Starting with FortiOS 7.2.0, released FortiOS firmware images use tags to indicate the following maturity levels:

- The *Feature* tag indicates that the firmware release includes new features.
- The *Mature* tag indicates that the firmware release includes no new, major features. Mature firmware will contain bug fixes and vulnerability patches where applicable.

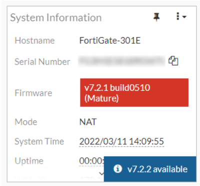
Administrators can use the tags to identify the maturity level of the current firmware in the GUI or CLI.

Administrators can view the maturity level of each firmware image that is available for upgrade on the *Fabric Management* page. When upgrading from mature firmware to feature firmware, a warning message is displayed.

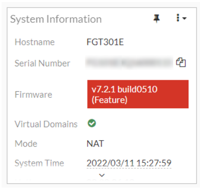
To demonstrate the functionality of this feature, this example uses FortiGates that are running fictitious build numbers.

To view maturity levels for firmware in the GUI:

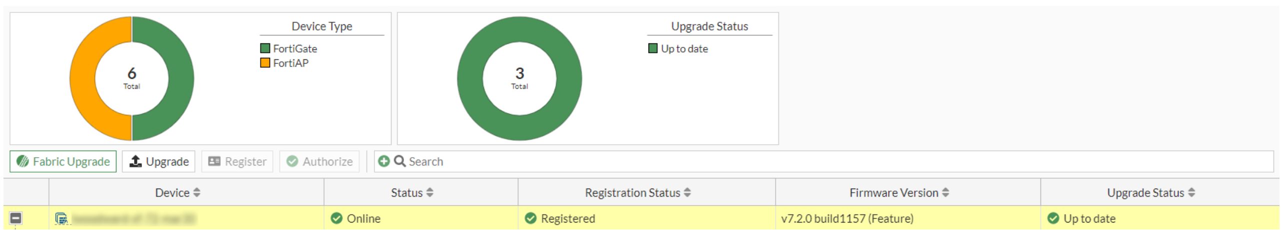
1. Go to *Dashboard > Status*. The *Firmware* field in the *System Information* widget displays the version with build number and either *(Mature)* or *(Feature)*.
   The following is an example of firmware with the *(Mature)* tag:



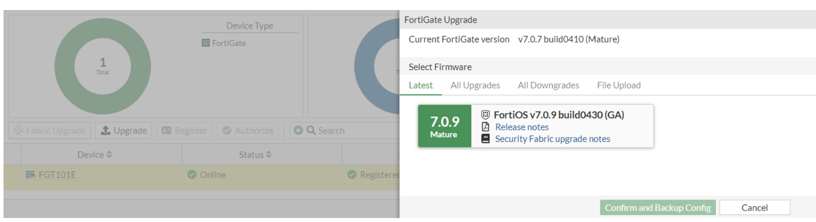
The following is an example of firmware with the *(Feature)* tag:



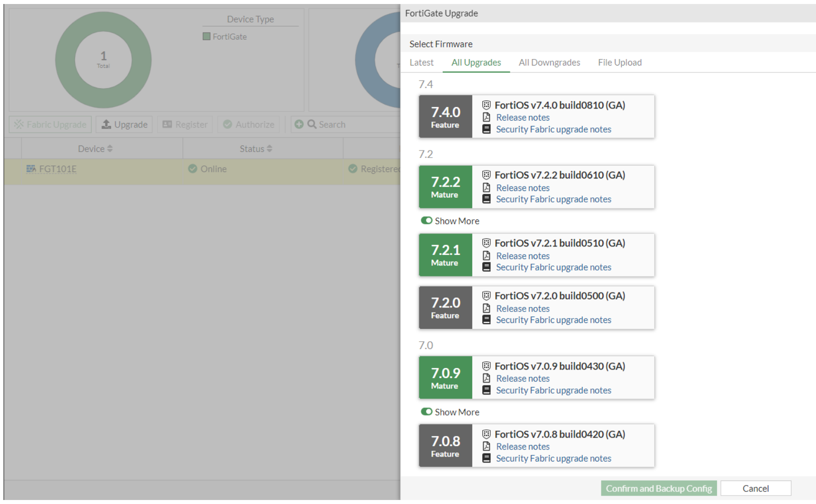
2. Go to *System > Fabric Management*. The *Firmware Version* column displays the version with build number and either *(Mature)* or *(Feature)*.
   The following is an example of firmware with the *(Feature)* tag:



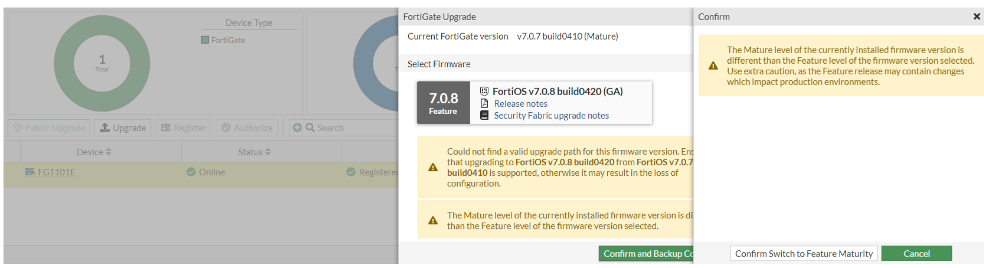
3. Select a device, and click *Upgrade*. The *FortiGate Upgrade* pane is displayed.
4. Click the *Latest* tab to view the latest available firmware version with its maturity level.
   In the following example, the latest firmware version is mature:



5. Click the *All Upgrades* tab to view all available firmware versions and their maturity levels.
   A gray box around the version number and the label *Feature* identifies feature firmware version. A green box around the version with the label *Mature* identifies a mature firmware version.



6. Select a version and click *Confirm and Backup Config*.
   When the firmware version is a feature release, a warning is displayed.



To view maturity levels for firmware in the CLI:

```
# get system status
Version: FortiGate-301E v7.4.0,build0810,220307 (GA.F)
...
```

In this example, the `Version` field includes `.F` to indicate that the maturity level is feature.

```
# get system status
Version: FortiGate-301E v7.2.2,build0610,220304 (GA.M)
...
```

In this example, the `Version` field includes `.M` to indicate that the maturity level is mature.

Upgrading individual device firmware

To upgrade individual device firmware in the GUI:

1. Log into the FortiGate GUI as the admin administrative user.
2. Go to *System > Fabric Management*.
3. Select the FortiGate, and click *Upgrade*. The *FortiGate Upgrade* pane opens.
4. Click the following tabs to view the available firmware:

Latest	Displays the latest, available firmware from FortiGuard.

All Upgrades	Displays all available firmware from FortiGuard.
File Upload	Click the *File Upload* tab to upload a firmware file that you previously downloaded from the Fortinet Customer Service & Support website. See Downloading a firmware image on page 1966.

5. Select a firmware version and click *Confirm and Backup Config*.
   If you are upgrading from a mature to a feature firmware version, the *Confirm* pane opens with a warning message.
6. Review the warning, and click *Confirm* to continue. A warning message is displayed.
7. Click *Continue* to initiate the upgrade.
   The FortiGate unit backs up the current configuration to the management computer, uploads the firmware image file, upgrades to the new firmware version, and restarts. This process takes a few minutes.

To upgrade individual device firmware in the CLI:

1. Make sure that the TFTP server is running.
2. Copy the new firmware image file to the root directory of the TFTP server.
3. Log in to the CLI.
4. Ping the TFTP server to ensure that the FortiGate can connect to it:
   ```
   execute ping <tftp_ipv4>
   ```
5. Enter the following command to copy the firmware image from the TFTP server to the FortiGate unit:
   ```
   execute restore image tftp <filename> <tftp_ipv4>
   ```
   The FortiGate unit responds with the message:
   ```
   This operation will replace the current firmware version!
   Do you want to continue? (y/n)
   ```
6. Type `y`. The FortiGate unit uploads the firmware image file, verifies the signature of the firmware image, and determines the firmware maturity level.
   When you are upgrading to a feature firmware image, you are asked to confirm whether to continue with the upgrade.
   When you proceed with the upgrade, the upgrade image is installed and FortiGate restarts. This process takes a few minutes.
   ```
   Please wait...
   Connect to tftp server 172.16.200.55 ...
   ##############################################################################
   Get image from tftp server OK.
   Verifying the signature of the firmware image.

   Warning: Upgrading to an image with Feature maturity notation.
   Image file uploaded is marked as a Feature image, are you sure you want to upgrade?
   Do you want to continue? (y/n)y
   Please confirm again. Are you sure you want to upgrade using uploaded file?
   Do you want to continue? (y/n)y
   Checking new firmware integrity ... pass
   Please wait for system to restart.
   Firmware upgrade in progress ...
   Done.
   The system is going down NOW !!
   ```
7. Reconnect to the CLI.

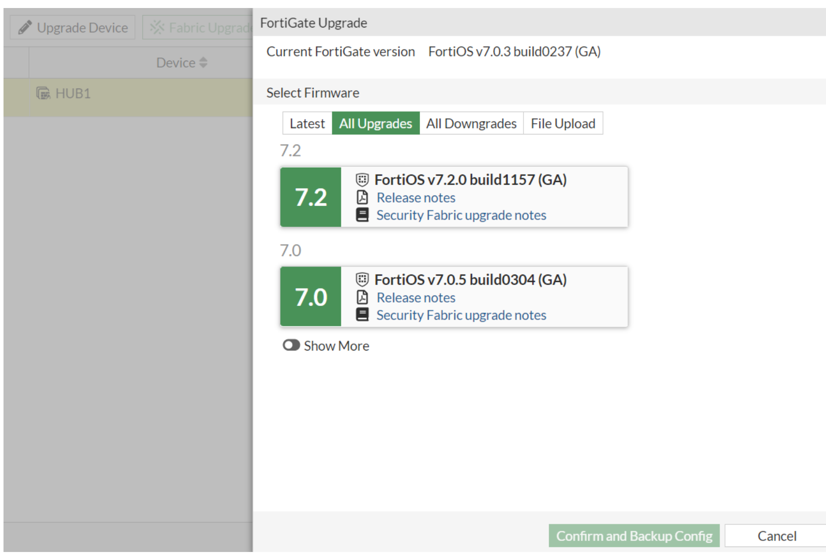
8. Update the antivirus and attack definitions:
```
execute update-now
```

Upgrading individual device firmware by following the upgrade path (federated update)

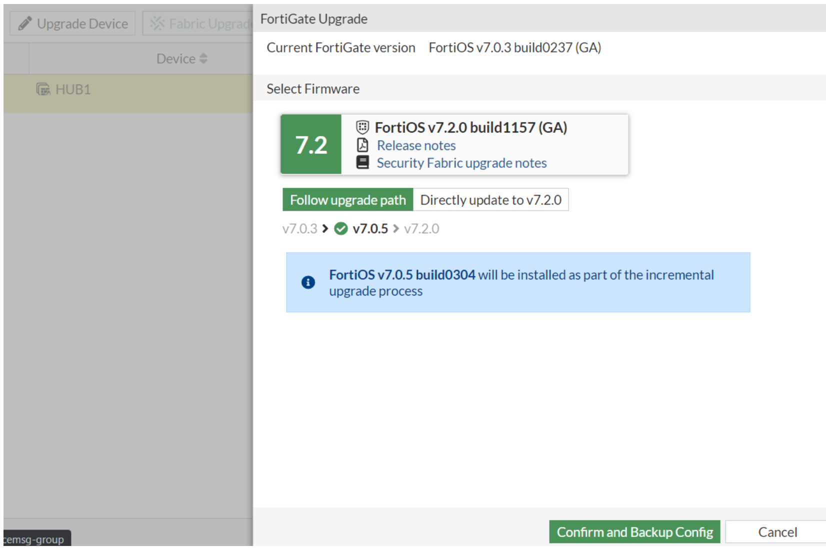
When upgrading a FortiGate to firmware that requires multiple builds in the upgrade path, FortiGate can follow the upgrade path to complete the upgrade automatically. This process is sometimes called a *federated update*. A federated update can be performed immediately or during a scheduled time.

To upgrade individual device firmware by following the upgrade path in the GUI:

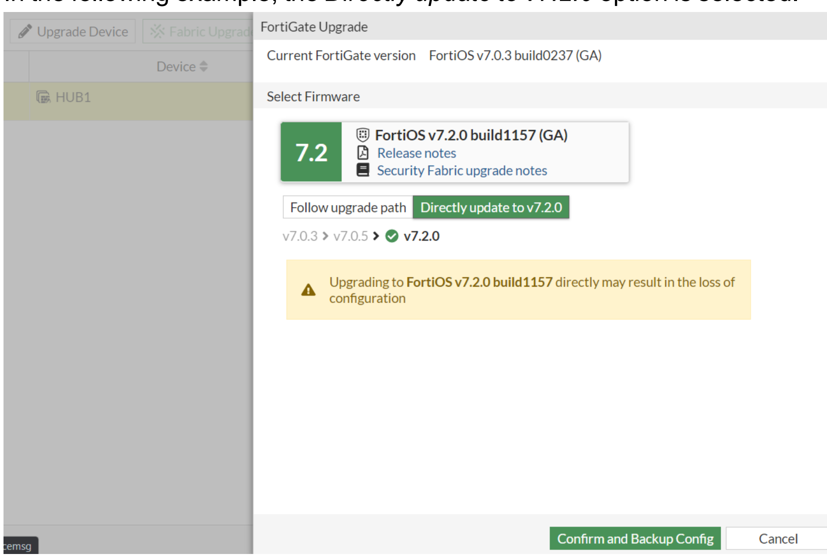
1. Log into the FortiGate GUI as the admin administrative user.
2. Go to *System > Fabric Management*. The *Firmware Version* column displays the version and either *(Feature)* or *(Mature)*.
3. Select the FortiGate, and click *Upgrade*. The *FortiGate Upgrade* pane opens.
4. Click *All Upgrades*. The available firmware versions are displayed.
   In this example, the firmware is available for 7.2 and 7.0.



5. Select the target firmware, and view the upgrade options.
   You can instruct FortiOS to follow the upgrade path (referred to as a federated upgrade) or upgrade directly to the selected firmware version.
   In this example, the target firmware is *7.2.0 build 1157(GA)*, and *Follow upgrade path* is selected. According to the upgrade path, the device can be automatically upgraded to *v7.0.5*, but not all the way to *7.2.0*.



- In the following example, the *Directly update to v7.2.0* option is selected.



6. Select *Follow upgrade path*, and click *Confirm and Backup Config*. A warning message is displayed.
7. Click *Continue* to initiate the upgrade.
   The FortiGate unit backs up the current configuration to the management computer, uploads the firmware image file, upgrades to the new firmware version, and restarts. This process takes a few minutes.

Upgrading all device firmware

Use the *Fabric Upgrade* process to select a firmware version from FortiGuard for the FortiGate. When part of a Security Fabric, the select target firmware is for the root FortiGate.

The target firmware version is used to automatically upgrade firmware for all FortiGate, FortiAP, and FortiSwitch devices. Fabric members or managed devices download the chosen firmware directly from FortiGuard.

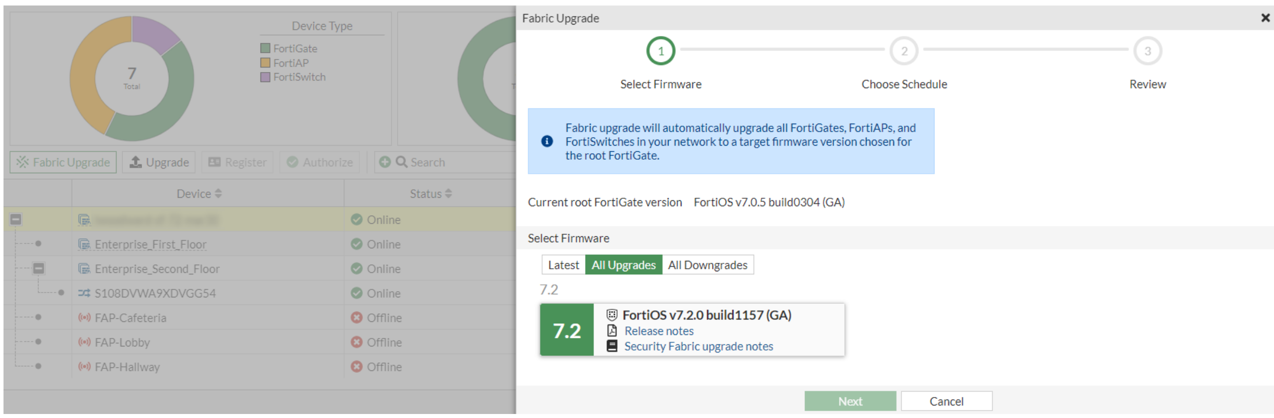
A *Fabric Upgrade* can be performed immediately or during a scheduled time.

To upgrade all device firmware:

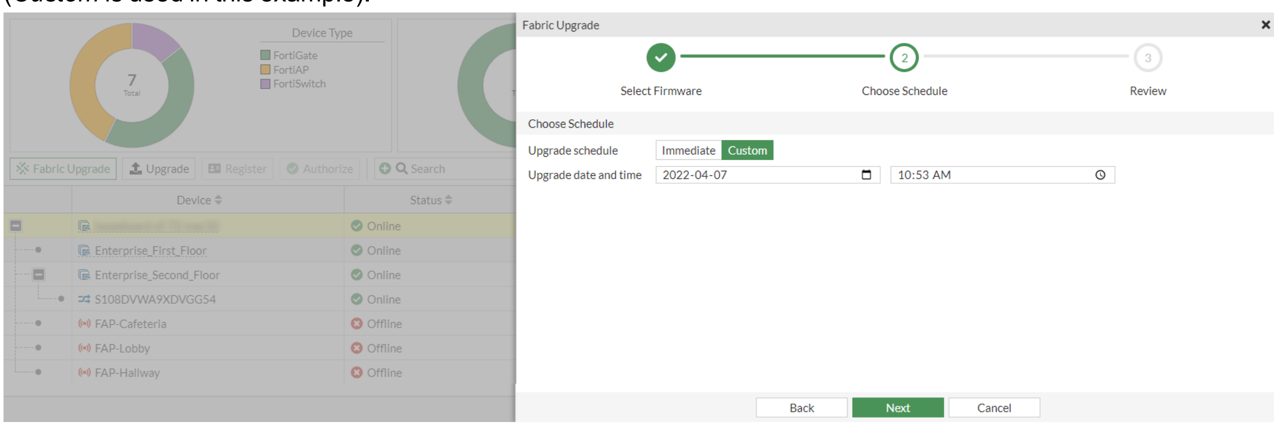
1. Log into the FortiGate GUI as the admin administrative user.
2. Go to *System > Fabric Management* and click *Fabric Upgrade*. The *Fabric Upgrade* pane opens, and the following tabs are available:

Latest	Displays the latest, available firmware from FortiGuard.
All Upgrades	Displays all available firmware from FortiGuard.

3. Select *Latest* or *All Upgrades*, select a target firmware version, and then click *Next*.
   In this example, *All Upgrades* is selected.

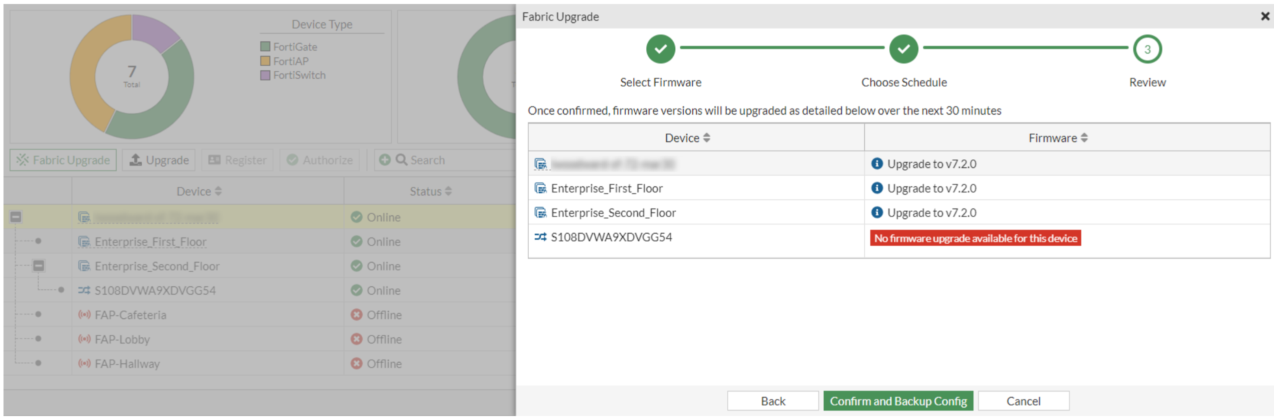


4. Select an upgrade schedule, either *Immediate* or *Custom*. If using *Custom*, enter an upgrade date and time (*Custom* is used in this example).

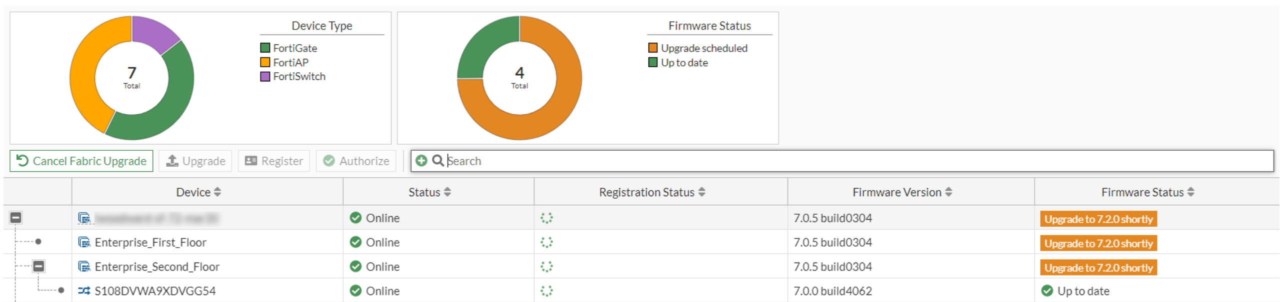


In a custom upgrade, the configuration backups are saved when the administrator schedules the upgrade. If the scheduled upgrade occurs after further configuration changes are made, the latest changes will not be saved in a new backup configuration file.

5. Click *Next* and review the update schedule. For FortiSwitch, a message appears because no firmware upgrade is currently available.



6. Click *Confirm and Backup Config*. The pane goes into a loading state to wait for all FortiGate configurations to save. Once completed, the pane closes and the device list refreshes to reflect the latest changes.



Upgrading all device firmware by following the upgrade path (federated update)

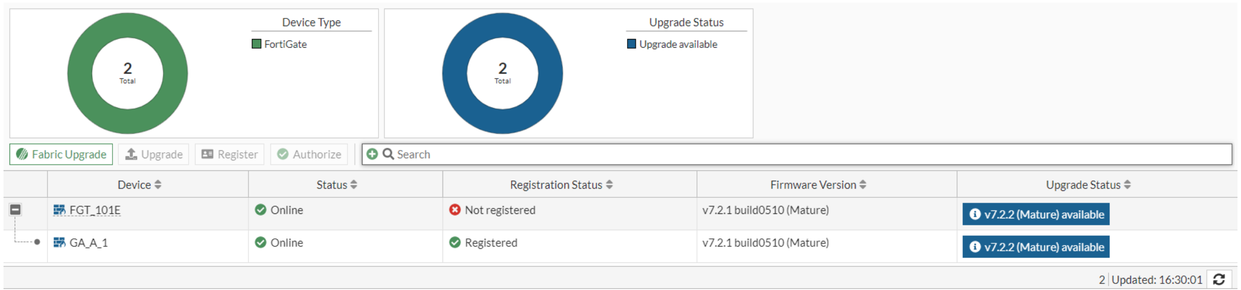
When performing a Fabric upgrade or non-Fabric upgrade under *System* > *Fabric Management* and choosing a firmware that requires multiple builds in the upgrade path, the FortiGate can follow the upgrade path to complete the upgrade automatically. This process is sometimes called a *federated update*. A federated update can be performed immediately or during a scheduled time.

To demonstrate the functionality of this feature, this example uses FortiGates that are running and upgrading to fictitious build numbers.

FortiAPs and FortiSwitches currently cannot follow the upgrade path. They upgrade directly to the target version.

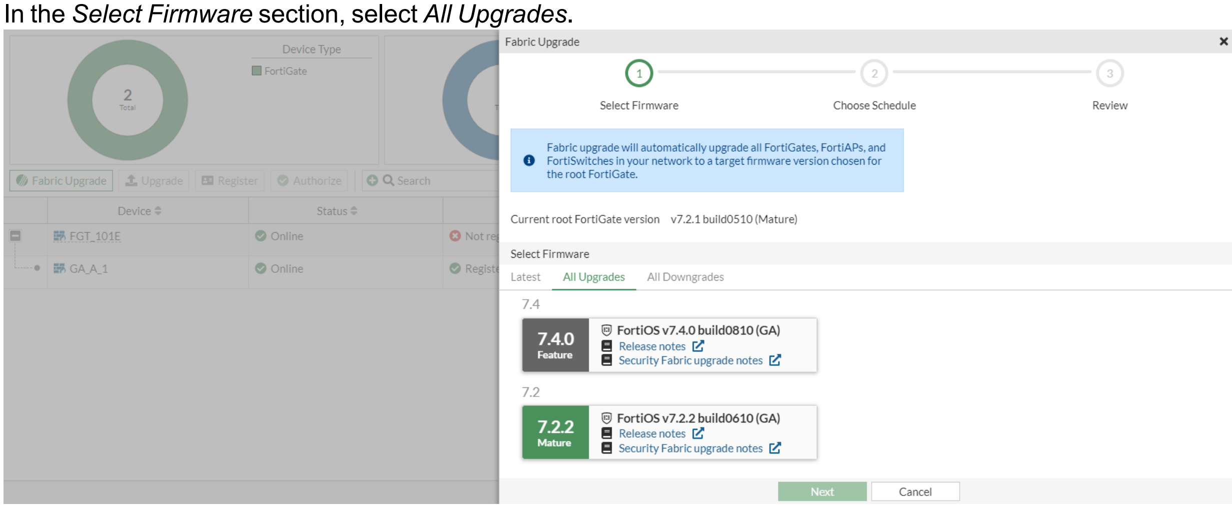
Example

In this example, the Security Fabric consists of a root FortiGate (FGT_101E) and a downstream FortiGate (GA_A_1). The FortiGates are currently running FortiOS 7.2.1 (build 0510). The administrator wants to upgrade the firmware to version 7.4.0 (build 0810). When upgrading the firmware on the *Fabric Management* page, the FortiGate is able to display the upgrade path, 7.2.1 > 7.2.2 > 7.4.0, and perform all of the upgrades in sequence (with multiple reboots).

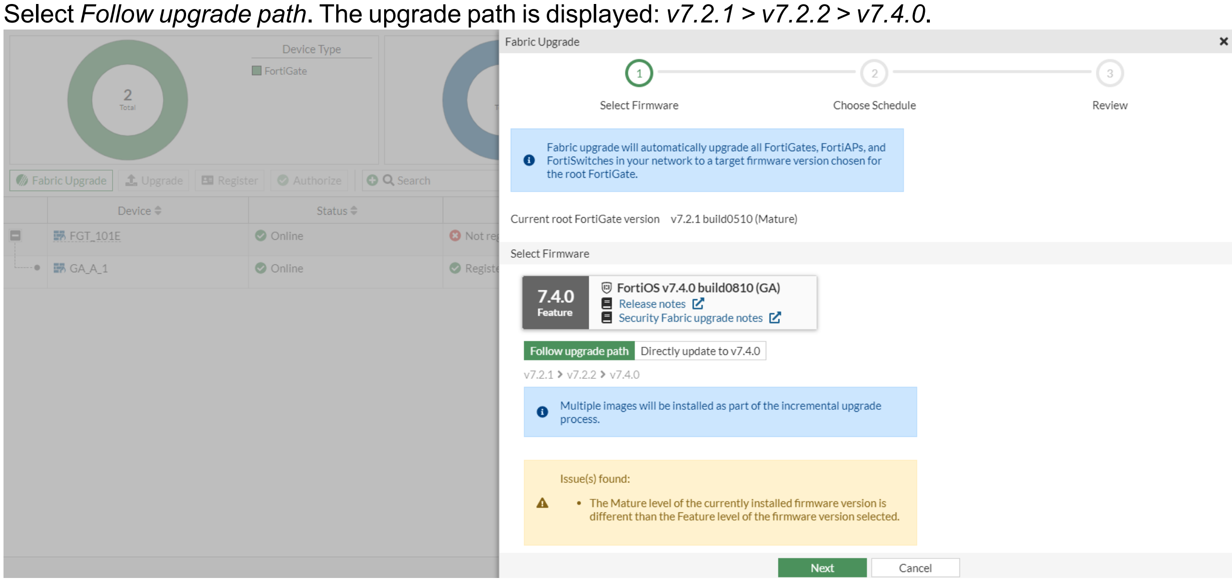


To upgrade the FortiGate firmware:

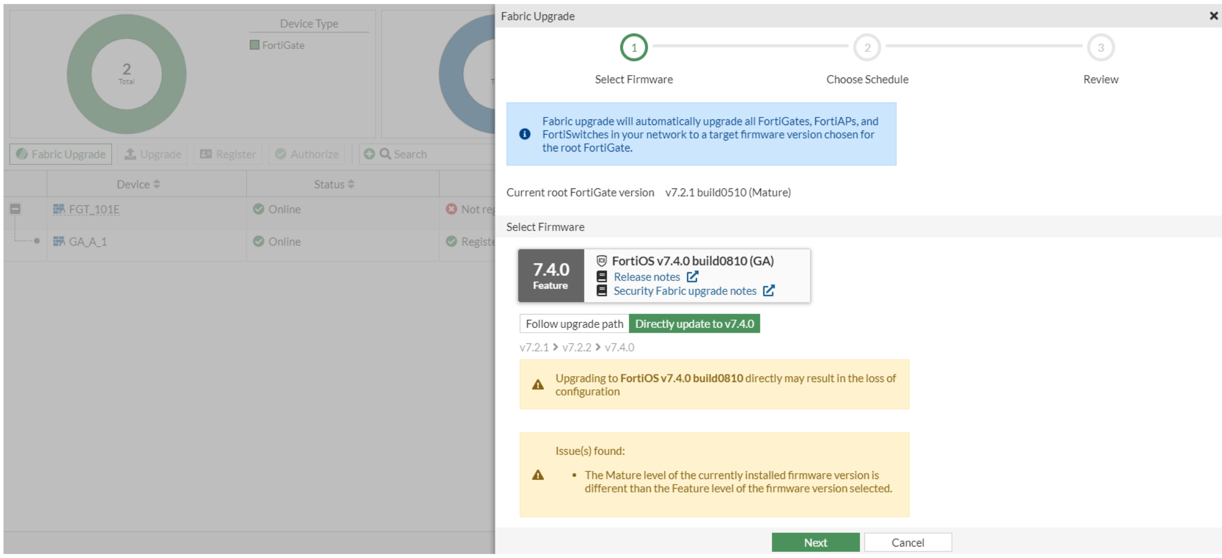
1. Go to *System > Fabric Management* and click *Fabric Upgrade*. The *Fabric Upgrade* pane opens.
2. In the *Select Firmware* section, select *All Upgrades*.



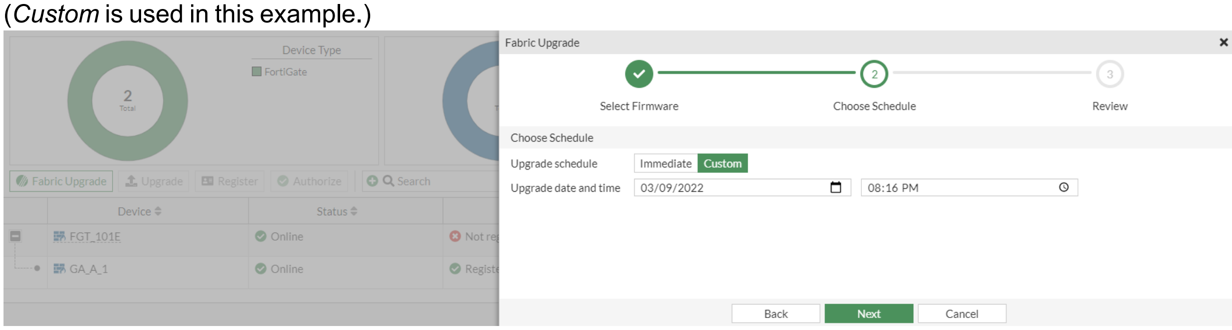
3. Select the 7.4.0 version. Upgrade options appear.
4. Select *Follow upgrade path*. The upgrade path is displayed: *v7.2.1 > v7.2.2 > v7.4.0*.



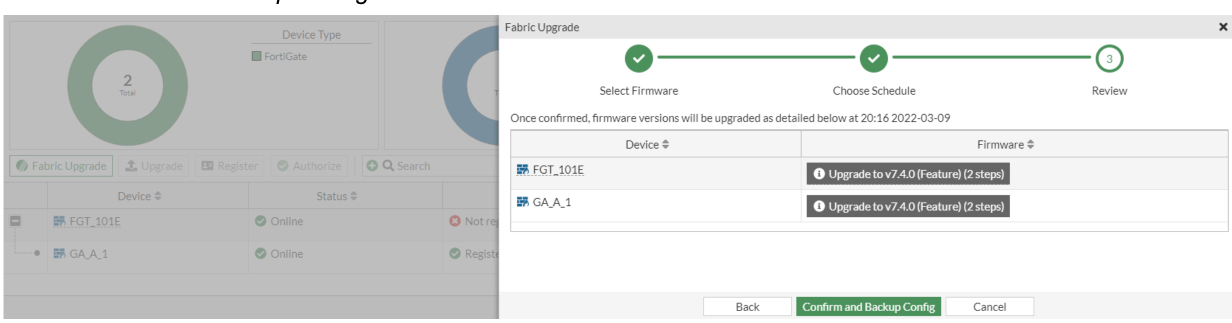
- If *Directly upgrade to v7.4.0* is selected, a warning message appears that this may result in the loss of configuration.



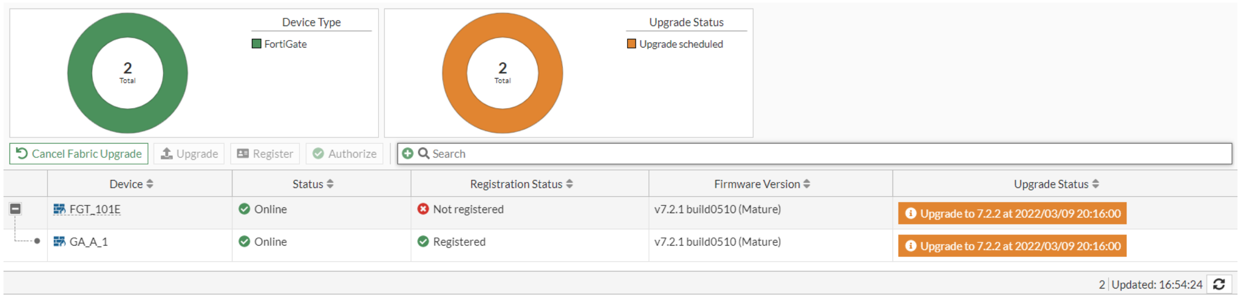
5. Click *Next*.
6. Select an upgrade schedule, either *Immediate* or *Custom*. If using *Custom*, enter an upgrade date and time. (*Custom* is used in this example.)



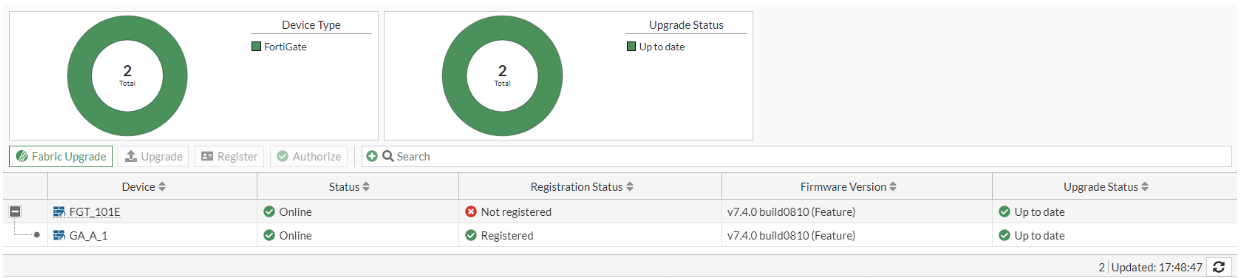
7. Click *Next* and review the update schedule.
8. Click *Confirm and Backup Config*.



The *Upgrade Status* for both FortiGates indicated when the scheduled upgrade will take place. In this example, the first upgrade in the path is to version 7.2.2. The FortiGates will reboot and then upgrade to 7.4.0 as per the upgrade path.



Once the upgrades are complete and both FortiGates are running the desired firmware (7.4.0), the *Upgrade Status* changes to *Up to date*.



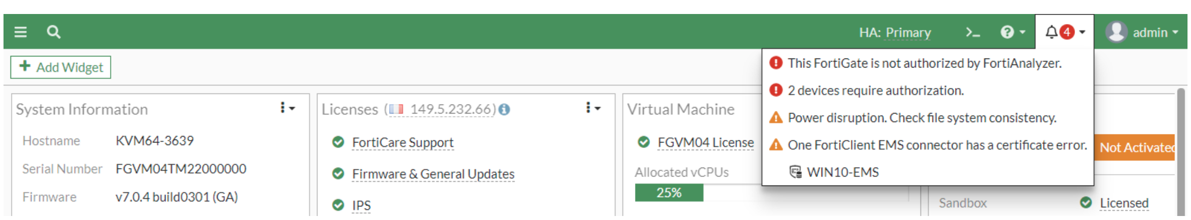
CLI commands

The following options are available in `execute federated-upgrade <option>`:

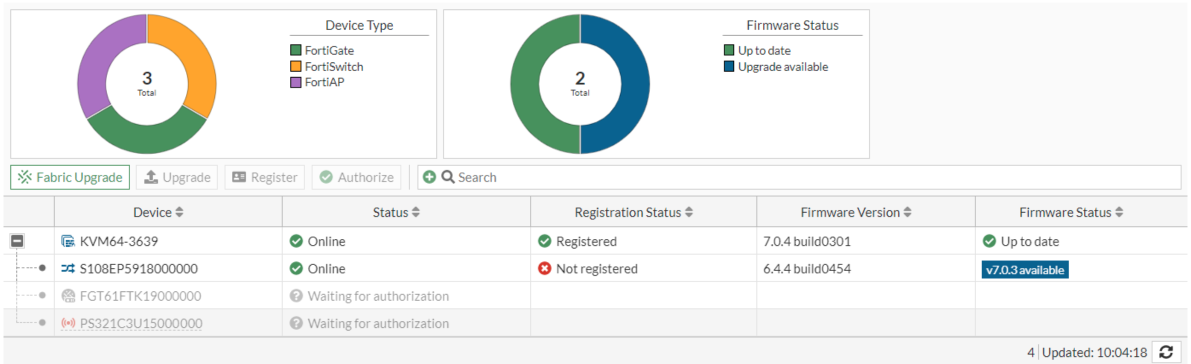
Option	Description
`cancel`	Cancel the currently configured upgrade.
`initialize`	Set up a federated upgrade.
`status`	Show the current status of a federated upgrade.

Authorizing devices

If there are any notifications in the top banner dropdown for unauthorized devices or devices that require authorization, clicking the notification redirects the user to the *System > Fabric Management* page. In this example, two devices require authorization.

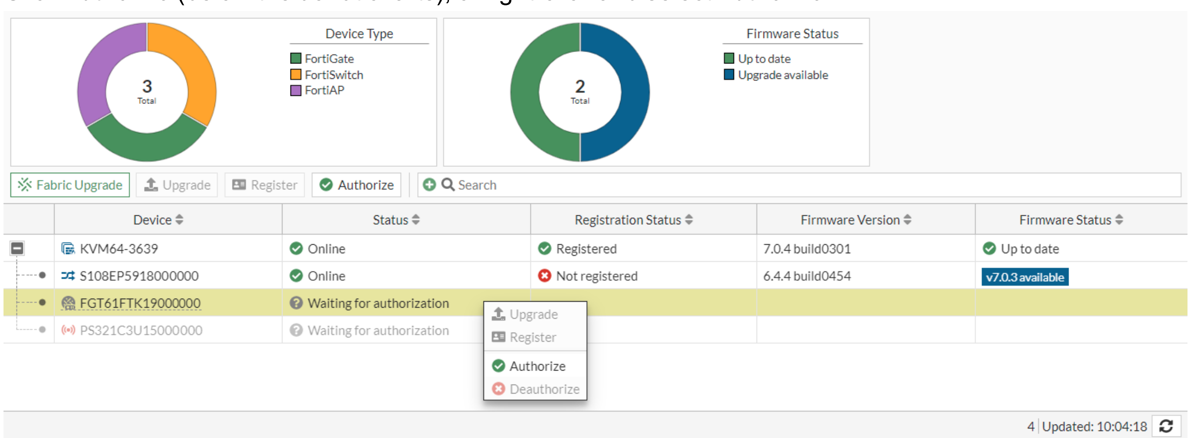


On the *Fabric Management* page, the unauthorized devices (a downstream FortiGate and a FortiAP) are grayed out, and their status is *Waiting for authorization*.

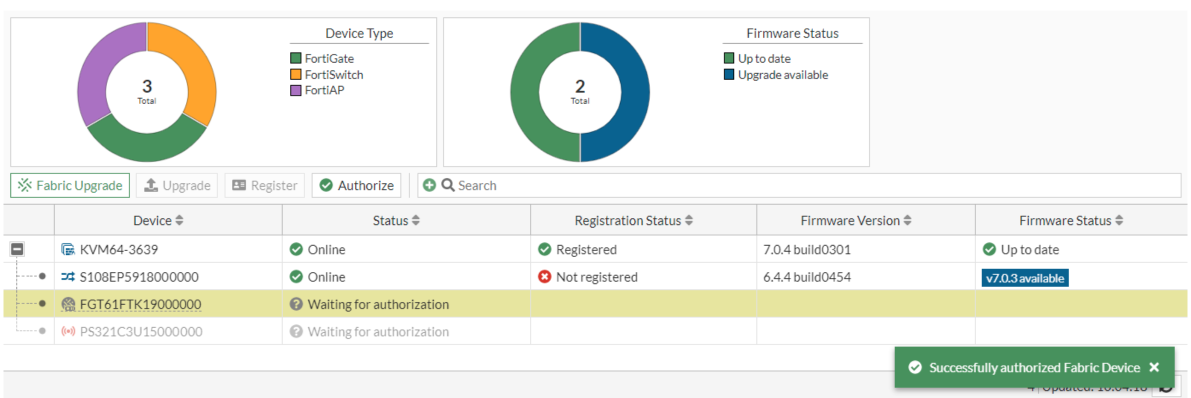


To authorize a Security Fabric device from the Fabric Management page:

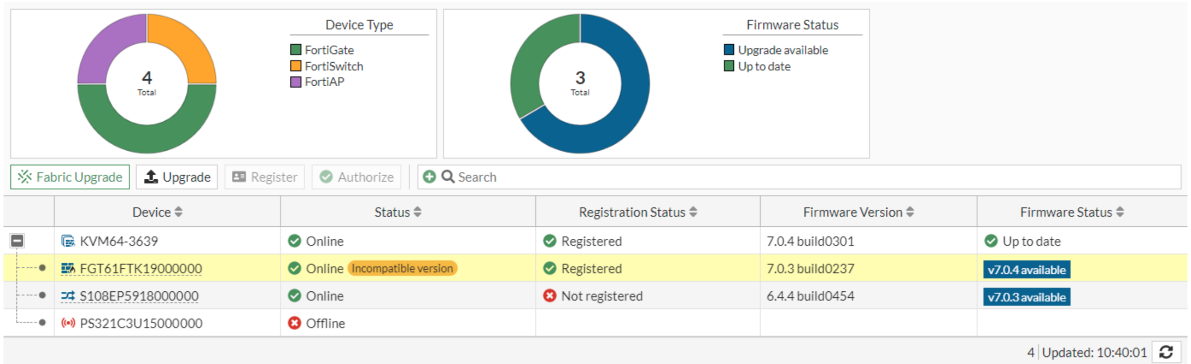
1. Go to *System > Fabric Management*, and select an unauthorized device.
2. Click *Authorize* (below the donut charts), or right-click and select *Authorize*.



A notification appears in the bottom-right corner once the device is authorized.

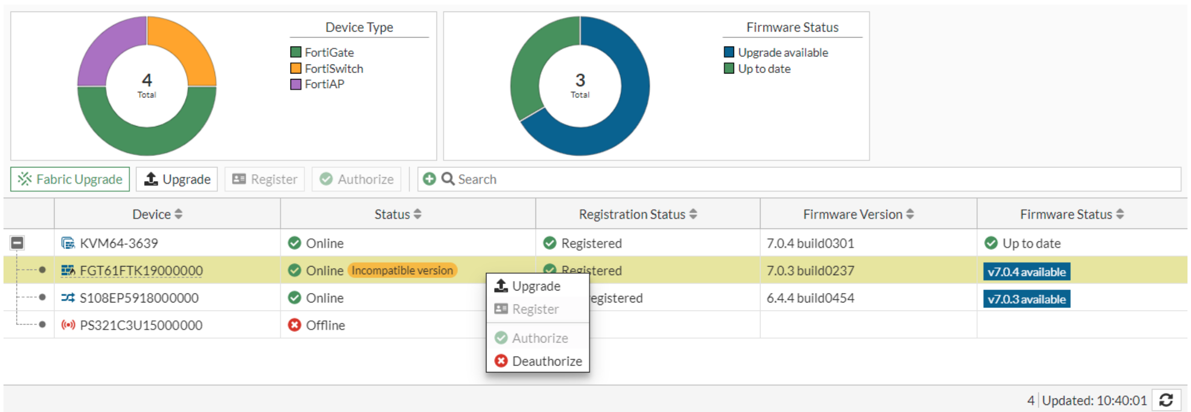


3. Click the subsequent notification to refresh the page. The device's status is now *Online*.



To deauthorize a Security Fabric device from the Fabric Management page:

1. Go to the *System > Fabric Management* page, and select a device.
2. Right-click and select *Deauthorize*.



3. Click the subsequent notification to refresh the page.

Firmware upgrade notifications

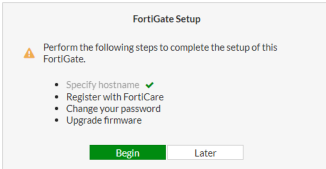
FortiGates with a firmware upgrade license that are connected to FortiGuard display upgrade notifications in the setup window, banner, and FortiGuard menu. The firmware notifications are enabled by default.

To configure firmware notifications in the CLI:

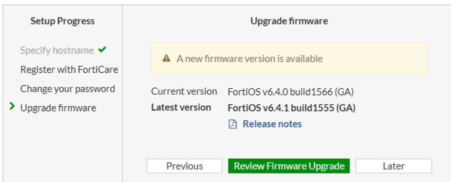
```
config system global
    set gui-firmware-upgrade-warning {enable | disable}
end
```

To use the firmware upgrade notifications in the GUI:

1. When you log in to FortiGate, the *FortiGate Setup* window includes an *Upgrade firmware* step. Click *Begin*.



2. Follow the steps in the *Setup Progress*, then click *Review Firmware Upgrade*.



   The *System > Fabric Management* page opens.
3. Notifications appear below the *Notification* icon in the banner, and beside *Fabric Management* in the tree menu.

Downloading a firmware image

Firmware images for all FortiGate units are available on the Fortinet Customer Service & Support website.

To download firmware:

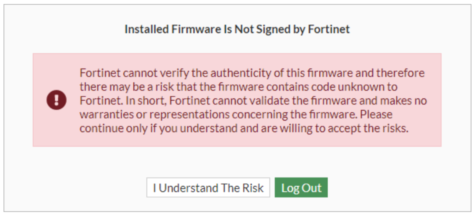
1. Log into the support site with your user name and password.
2. Go to *Support > Firmware Download*.
   A list of Release Notes is shown. If you have not already done so, download and review the Release Notes for the firmware version that you are upgrading your FortiGate unit to.
3. Select the *Download* tab.
4. Navigate to the folder for the firmware version that you are upgrading to.
5. Find your device model on the list. FortiWiFi devices have file names that start with *FWF*.
6. Click *HTTPS* in the far right column to download the firmware image to your computer.

Firmware can also be downloaded using FTP, but as FTP is not an encrypted file transferring protocol, HTTPS downloading is recommended.

FortiOS image signing and verification

Official FortiOS firmware images are signed by the Fortinet CA. The BIOS checks the validity of an image when it is uploaded to the device. If the image is not signed by the Fortinet CA, warning messages appear in the GUI in several locations and in the CLI when the uploaded firmware fails the signature validation.

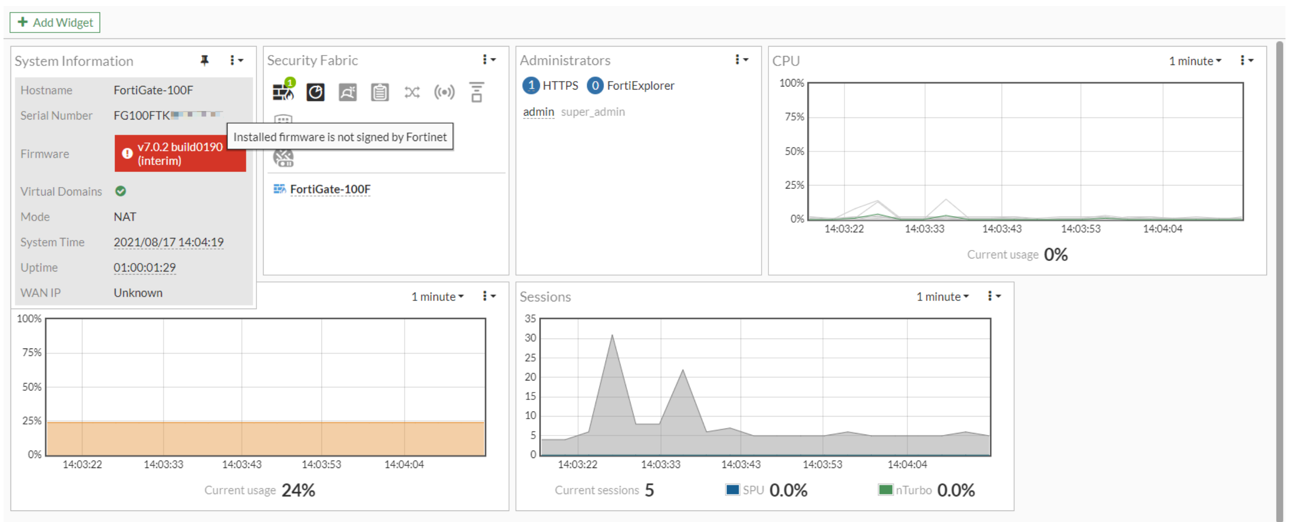
Warning message after an administrator logs in to the GUI:



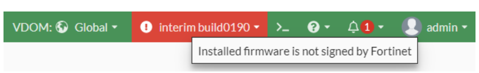
Click *I Understand the Risk* to continue.

Sample GUI warnings

Dashboard > *Status* page:



Banner:



System > *Firmware* page:

Warning message after updating the firmware in the CLI:

```
# execute restore image tftp FGT_100F-v7-build0197-FORTINET.out 172.16.200.55
    This operation will replace the current firmware version!
    Do you want to continue? (y/n)y
```

```
Please wait...
Connect to tftp server 172.16.200.55 ...
############################################################################
Get image from tftp server OK.
Verifying the signature of the firmware image.
******WARNING: This firmware failed signature validation.******
Fortinet cannot verify the authenticity of this firmware and therefore
there may be a risk that the firmware contains code unknown to Fortinet.
In short, Fortinet cannot validate the firmware and makes no warranties
or representations concerning the firmware.
Please continue only if you understand and are willing to accept the risks.
Do you want to continue? (y/n)y

Checking new firmware integrity ... pass
Please wait for system to restart.

Firmware upgrade in progress ...
Done.
```

Testing a firmware version

The integrity of firmware images downloaded from Fortinet's support portal can be verified using a file checksum. A file checksum that does not match the expected value indicates a corrupt file. The corruption could be caused by errors in transfer or by file modification. A list of expected checksum values for each build of released code is available on Fortinet’s support portal.

Image integrity is also verified when the FortiGate is booting up. This integrity check is done through a cyclic redundancy check (CRC). If the CRC fails, the FortiGate unit will encounter an error during the boot process.

Firmware images are signed and the signature is attached to the code as it is built. When upgrading an image, the running OS will generate a signature and compare it with the signature attached to the image. If the signatures do not match, the new OS will not load.

Testing before installation

FortiOS lets you test a new firmware image by installing the firmware image from a system reboot and saving it to system memory. After completing this procedure, the FortiGate unit operates using the new firmware image with the current configuration. The new firmware image is not permanently installed. The next time the FortiGate unit restarts, it operates with the originally installed firmware image using the current configuration. If the new firmware image operates successfully, you can install it permanently using the procedure explained in Upgrading individual device firmware.

For this procedure, you must install a TFTP server that you can connect to from the FortiGate internal interface. The TFTP server should be on the same subnet as the internal interface.

To test the new firmware version:

1. Connect to the CLI using an RJ-45 to USB (or DB-9) or null modem cable.
2. Ensure that the TFTP server is running.
3. Copy the new firmware image file to the root directory on the TFTP server.
4. Ensure that the FortiGate unit can connect to the TFTP server using the `execute ping` command.
5. Restart the FortiGate unit: `execute reboot`. The following message is shown:
   ```
   This operation will reboot the system!
   Do you want to continue? (y/n)
   ```

6. Type `y`. As the FortiGate unit starts, a series of system startup messages appears.
7. When the following messages appears:
   ```
   Press any key to display configuration menu..........
   ```
   Immediately press any key to interrupt the system startup.

   You have only three seconds to press any key. If you do not press a key during this time, the FortiGate will reboot, and you will have to log in and repeat the `execute reboot` command.

   If you successfully interrupt the startup process, the following messages appears:
   ```
   [G]: Get firmware image from TFTP server.
   [F]: Format boot device.
   [B]: Boot with backup firmware and set as default
   [C]: Configuration and information
   [Q]: Quit menu and continue to boot with default firmware.
   [H]: Display this list of options.
   Enter G, F, Q, or H:
   ```
8. Type *G* to get the new firmware image from the TFTP server. The following message appears: `Enter TFTP server address [192.168.1.168]:`
9. Type the address of the TFTP server, then press *Enter*. The following message appears: `Enter Local Address [192.168.1.188]:`
10. Type the IP address of the FortiGate unit to connect to the TFTP server.

    The IP address must be on the same network as the TFTP server.

    Make sure that you do not enter the IP address of another device on this network.

    The following message appears:
    ```
    Enter File Name [image.out]:
    ```
11. Enter the firmware image file name then press *Enter*. The TFTP server uploads the firmware image file to the FortiGate unit and the following message appears:
    ```
    Save as Default firmware/Backup firmware/Run image without saving: [D/B/R]
    ```
12. Type *R*. The FortiGate image is installed to system memory and the FortiGate unit starts running the new firmware image, but with its current configuration.

Test the new firmware image as required. When done testing, reboot the FortiGate unit, and the it will resume using the firmware that was running before you installed the test firmware.

Installing firmware from system reboot

In the event that the firmware upgrade does not load properly and the FortiGate unit will not boot, or continuously reboots, it is best to perform a fresh install of the firmware from a reboot using the CLI. If configured, the firmware can also be automatically installed from a USB drive; see Restoring from a USB drive on page 1971 for details.

This procedure installs a firmware image and resets the FortiGate unit to factory default settings. You can use this procedure to upgrade to a new firmware version, revert to an older firmware version, or re-install the current firmware.

To use this procedure, you must connect to the CLI using the FortiGate console port and a RJ-45 to USB (or DB-9), or null modem cable. You must also install a TFTP server that you can connect to from the FortiGate internal interface. The TFTP server should be on the same subnet as the internal interface.

Before beginning this procedure, ensure that you backup the FortiGate unit configuration. See Configuration backups on page 65 for details. If you are reverting to a previous FortiOS version, you might not be able to restore the previous configuration from the backup configuration file.

Installing firmware replaces your current antivirus and attack definitions, along with the definitions included with the firmware release you are installing. After you install new firmware, make sure that antivirus and attack definitions are up to date.

To install firmware from a system reboot:

1. Connect to the CLI using the RJ-45 to USB (or DB-9) or null modem cable.
2. Ensure that the TFTP server is running.
3. Copy the new firmware image file to the root directory of the TFTP server.
4. Ensure that the FortiGate unit can connect to the TFTP server using the `execute ping` command.
5. Restart the FortiGate unit: `execute reboot`. The following message is shown:
   ```
   This operation will reboot the system!
   Do you want to continue? (y/n)
   ```
6. Type `y`. As the FortiGate unit starts, a series of system startup messages appears.
7. When the following messages appears:
   ```
   Press any key to display configuration menu..........
   ```
   Immediately press any key to interrupt the system startup.

   You have only three seconds to press any key. If you do not press a key during this time, the FortiGate will reboot, and you will have to log in and repeat the `execute reboot` command.

   If you successfully interrupt the startup process, the following messages appears:
   ```
   [C]: Configure TFTP parameters.
   [R]: Review TFTP parameters.
   [T]: Initiate TFTP firmware transfer.
   [F]: Format boot device.
   [I]: System information.
   [B]: Boot with backup firmware and set as default.
   [Q]: Quit menu and continue to boot.
   [H]: Display this list of options.

   Enter C,R,T,F,I,B,Q,or H:
   ```
8. If necessary, type `C` to configure the TFTP parameters, then type `Q` to return to the previous menu:
   ```
   [P]: Set firmware download port.
   [D]: Set DHCP mode.
   [I]: Set local IP address.
   [S]: Set local subnet mask.
   [G]: Set local gateway.
   [V]: Set local VLAN ID.
   [T]: Set remote TFTP server IP address.
   [F]: Set firmware file name.
   [E]: Reset TFTP parameters to factory defaults.
   [R]: Review TFTP parameters.
   [N]: Diagnose networking(ping).
   [Q]: Quit this menu.
   [H]: Display this list of options.

   Enter P,D,I,S,G,V,T,F,E,R,N,Q,or H:
   ```

   The IP address must be on the same network as the TFTP server.

   Make sure that you do not enter the IP address of another device on this network.

9. Type `T` get the new firmware image from the TFTP server.
   The FortiGate unit loads the firmware.
10. Save the firmware as the default (`D`) or backup (`B`) firmware image, or run the image without saving it (`R`).
   The FortiGate unit installs the new firmware image and restarts. The installation might take a few minutes to complete.

Restoring from a USB drive

The FortiGate firmware can be manually restored from a USB drive, or installed automatically from a USB drive after a reboot.

To restore the firmware from a USB drive:

1. Copy the firmware file to the root directory on the USB drive.
2. Connect the USB drive to the USB port of the FortiGate device.
3. Connect to the FortiGate CLI using the RJ-45 to USB (or DB-9) or null modem cable.
4. Enter the following command:
   ```
   execute restore image usb <filename>
   ```
   The FortiGate unit responds with the following message:
   ```
   This operation will replace the current firmware version! Do you want to continue?
        (y/n)
   ```
5. Type `y`. The FortiGate unit restores the firmware and restarts. This process takes a few minutes.
6. Update the antivirus and attack definitions:
   ```
   execute update-now
   ```

To install firmware automatically from a USB drive:

1. Go to *System > Settings*.
2. In the *Start Up Settings* section, enable *Detect firmware* and enter the name of the firmware file.
3. Copy the firmware file to the root directory on the USB drive.
4. Connect the USB drive to the USB port of the FortiGate device.
5. Reboot the FortiGate device.

Using controlled upgrades

Using a controlled upgrade, you can upload a new version of the FortiOS firmware to a separate partition in the FortiGate memory for later upgrade. The FortiGate unit can be configured so that when it is rebooted, it will automatically load the new firmware. Using this option, you can stage multiple FortiGate units to upgrade simultaneously using FortiManager or a script.

To load the firmware for later installation:

```
execute restore secondary-image {ftp | tftp | usb} <filename_str>
```

To set the FortiGate unit so that when it reboots, the new firmware is loaded:

```
execute set-next-reboot {primary | secondary}
```

where `{primary | secondary}` is the partition with the preloaded firmware.

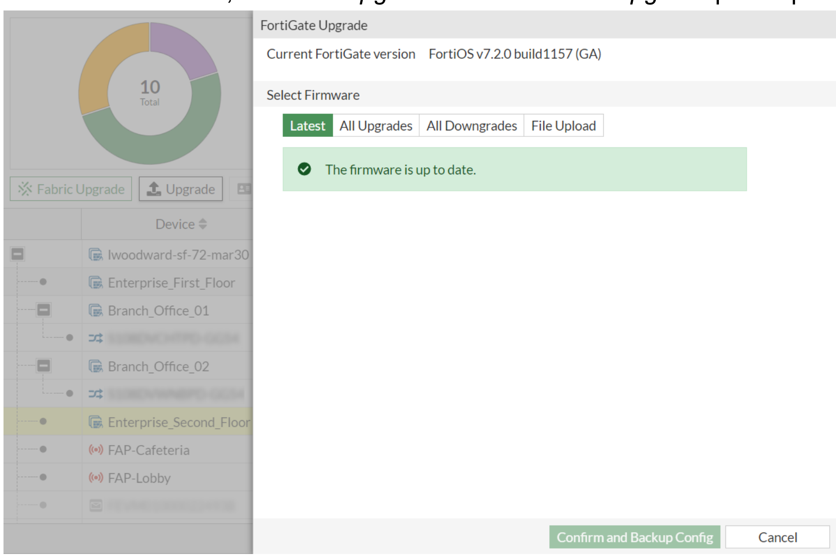
Downgrading individual device firmware

Downgrading the firmware is not recommended.

This procedure downgrades the FortiGate to a previous firmware version. After downgrading, you may be unable to restore the backup configuration.

To downgrade to a previous firmware version in the GUI:

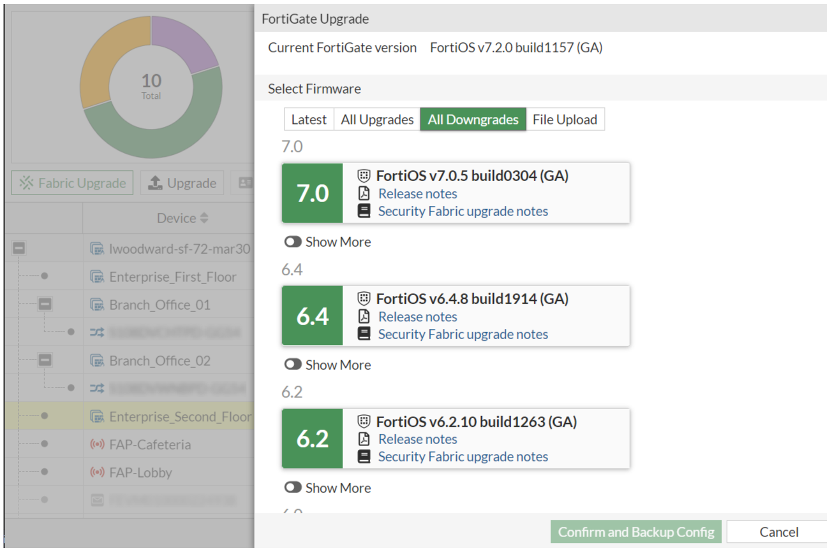
1. Log into the FortiGate GUI as the admin administrative user.
2. Go to *System > Fabric Management*. The *Firmware Version* column displays the version and either *(Feature)* or *(Mature)*.
3. Select the FortiGate, and click *Upgrade*. The *FortiGate Upgrade* pane opens.



4. Click one of the following tabs to select a downgrade method:

All Downgrades	Click the *All Downgrades* tab to view and select all firmware versions that are available from FortiGuard for downgrade.
File Upload	Click the *File Upload* tab to upload a firmware file that you previously downloaded from the Fortinet Customer Service & Support website. See Downloading a firmware image on page 1966.

In this example, the *All Downgrades* tab is selected.



5. Select a firmware version and click *Confirm and Backup Config*. A warning message is displayed.
6. Click *Continue* to continue with the downgrade.
   The FortiGate unit backs up the current configuration to the management computer, uploads the firmware image file, downgrades to the firmware version, and restarts. This process takes a few minutes.

To downgrade to a previous firmware version in the CLI:

1. Make sure that the TFTP server is running.
2. Copy the new firmware image file to the root directory of the TFTP server.
3. Log into the CLI.
4. Ping the TFTP server to ensure that the FortiGate can connect to it:
```
execute ping <tftp_ipv4>
```
5. Enter the following command to copy the firmware image from the TFTP server to the FortiGate unit:
```
execute restore image tftp <filename> <tftp_ipv4>
```
   The FortiGate unit responds with the message:
```
This operation will replace the current firmware version!
Do you want to continue? (y/n)
```
6. Type `y`. The FortiGate unit uploads the firmware image file, then a message similar to the following is shown:
```
Get image from tftp server OK.
Check image OK.
This operation will downgrade the current firmware version!
Do you want to continue? (y/n)
```
7. Type `y`. The FortiGate unit downgrades to the old firmware version and restarts. This process takes a few minutes.
8. Reconnect to the CLI.
9. Update the antivirus and attack definitions:
```
execute update-now
```

Settings

The default administrator password should be configured immediately after the FortiGate is installed, see Default administrator password on page 1974.

After that, there are several system settings that should also be configured in *System > Settings*:

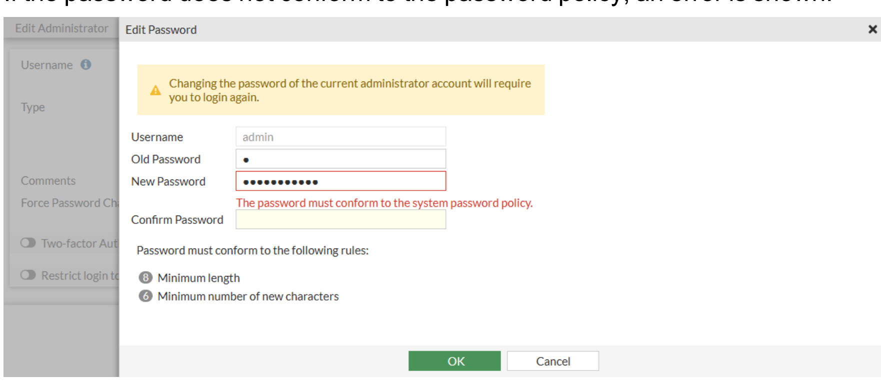
- Changing the host name on page 1975
- Setting the system time on page 1976
- Configuring ports on page 1979
- Setting the idle timeout time on page 1980
- Setting the password policy on page 1981
- Changing the view settings on page 1981
- Setting the administrator password retries and lockout time on page 1982
- TLS configuration on page 1982
- Controlling return path with auxiliary session on page 1983
- Email alerts on page 1987
- Using configuration save mode on page 1991
- Trusted platform module support on page 1992

Default administrator password

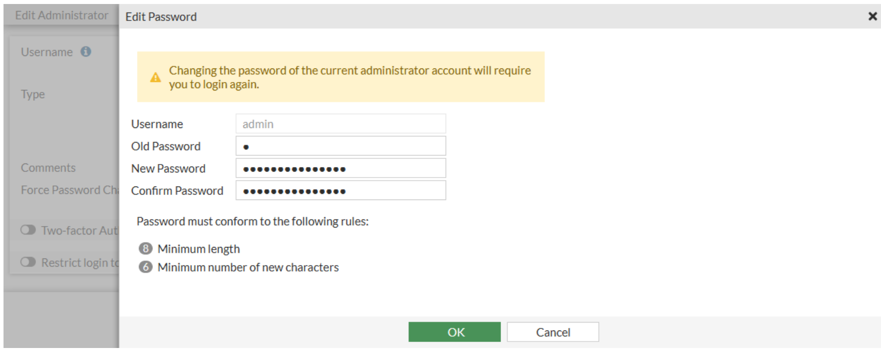
By default, your FortiGate has an administrator account set up with the username `admin` and no password. In order to prevent unauthorized access to the FortiGate, it is highly recommended that you add a password to this account.

Adding a password to the *admin* administrator is mandatory. You will be prompted to configured it the first time you log in to the FortiGate using that account, after a factory reset, and after a new image installation.

To change the default password in the GUI:

1. Go to *System > Administrators*.
2. Edit the *admin* account.
3. Click *Change Password*.
4. If applicable, enter the current password in the *Old Password* field.
5. Enter a password in the *New Password* field, then enter it again in the *Confirm Password* field.
   If the password does not conform to the password policy, an error is shown:
   

   If the password conforms to the password policy, no error message is shown:



6. Click *OK*.

To change the default password in the CLI:

```
config system admin
    edit admin
        set password <old password> <old password>
New password must conform to the password policy enforced on this device:
minimum-length=8; the new password must have at least 1 unique character(s) which don't
exist in the old password.; must not be same as last two passwords

node_check_object fail! for password *

value parse error before '*'
Command fail. Return code -49

        set password <new password> <old password>
    next
end
```

It is also recommended that you change the user name of this account; however, since you cannot change the user name of an account that is currently in use, a second administrator account must be created in order to do this.

Changing the host name

The FortiGate host name is shown in the *Hostname* field in the *System Information* widget on a dashboard, as the command prompt in the CLI, as the SNMP system name, as the device name on FortiGate Cloud, and other places. If the FortiGate is in an HA cluster, use a unique host name to distinguish it from the other devices in the cluster.

An administrator requires *System* > *Configuration* read/write access to edit the host name. See Administrator profiles on page 1949 for details.

To change the host name in the GUI:

1. Go to *System* > *Settings*.
2. In the *Host name* field, enter a new name.
3. Click *Apply*.

To change the host name in the CLI:

```
config system global
    set hostname <hostname>
end
```

Setting the system time

You can either manually set the FortiOS system time, or configure the device to automatically keep its system time correct by synchronizing with a Network Time Protocol (NTP) or Precision Time Protocol (PTP) server.

Daylight savings time is enabled by default, and can only be configured in the CLI.

For many features to work, including scheduling, logging, and SSL-dependent features, the FortiOS system time must be accurate.

To configure the date and time in the GUI:

1. Go to *System > Settings*.
2. In the *System Time* section, configure the following settings to either manually set the time or use an NTP server:

Time Zone	Select a time zone from the list. This should be the time zone that the FortiGate is in.
Set Time	Select either *NTP*, *PTP*, or *Manual settings*.
NTP	To use an NTP server other than FortiGuard, the CLI must be used. In the *Sync interval* field, enter how often, in minutes, that the device synchronizes its time with the NTP server.
PTP	• Set the *Mode* to *Multicast* or *Hybrid*. • Select the *Delay mechanism*: *E2E* or *P2P*. • Set the *Request interval*, in seconds. • Select the *Interface*.
Manual settings	Manually enter the *Date*, and *Time*.
Setup device as local NTP server	Enable to configure the FortiGate as a local NTP server. This option is not available if *Set Time* is *PTP*. In the *Listen on Interfaces* field, set the interface or interfaces that the FortiGate will listen for NTP requests on.

3. Click *Apply*.

To configure the date and time in the CLI:

1. Configure the timezone and daylight savings time:

```
config system global
    set timezone <integer>
```

```
    set dst {enable | disable}
end
```

2. Either manually configure the date and time, or configure an NTP or PTP server:
   - Manual:

     ```
     execute date <yyyy-mm-dd>
     execute time <hh:mm:ss>
     ```

   - NTP server:

     ```
     config system ntp
         set ntpsync enable
         set type {fortiguard | custom}
         set syncinterval <integer>
         set source-ip <ip_address>
         set source-ip6 <ip6_address>
         set server-mode {enable | disable}
         set interface <interface>
         set authentication {enable | disable}
         set key-type {MD5 | SHA1}
         set key <password>
         set key-id <integer>
         config ntpserver
             edit <server_id>
                 set server <ip_address or hostname>
                 set ntpv3 {enable | disable}
                 set authentication {enable | disable}
                 set interface-select-method {auto | sdwan | specify}
                 set key <password>
                 set key-id <integer>
             next
         end
     end
     ```

   - PTP server:

     ```
     config system ptp
         set status enable
         set mode {multicast | hybrid}
         set delay-mechanism {E2E | P2P}
         set request-interval <integer>
         set interface <string>
     end
     ```

SHA-1 authentication support (for NTPv4)

SHA-1 authentication support allows the NTP client to verify that severs are known and trusted and not intruders masquerading (accidentally or intentionally) as legitimate servers. In cryptography, SHA-1 is a cryptographic hash algorithmic function.

SHA-1 authentication support is only available for NTP clients, not NTP servers.

To configure authentication on a FortiGate NTP client:

```
config system ntp
    set ntpsync enable
    set type custom
    set syncinterval 1
    config ntpserver
        edit "883502"
            set server "10.1.100.11"
            set authentication enable
            set key **********
            set key-id 1
        next
    end
end
```

Command	Description
`authentication <enable | disable>`	Enable/disable MD5/SHA1 authentication (default = disable).
`key <passwd>`	Key for MD5/SHA1 authentication. Enter a password value.
`key-id <integer>`	Key ID for authentication. Enter an integer value from 0 to 4294967295.

To confirm that NTP authentication is set up correctly:

```
# diagnose sys ntp status
synchronized: yes, ntpsync: enabled, server-mode: disabled
ipv4 server(10.1.100.11) 10.1.100.11 -- reachable(0xff) S:4 T:6 selected
server-version=4, stratum=3
```

If NTP authentication is set up correctly, the server version is equal to 4.

PTPv2

Precision time protocol (PTP) is used to synchronize network clocks. It is best suited to situations where time accuracy is of the utmost importance, as it supports accuracy in the sub-microsecond range. Conversely, NTP accuracy is in the range of milliseconds or tens of milliseconds.

The following CLI commands are available:

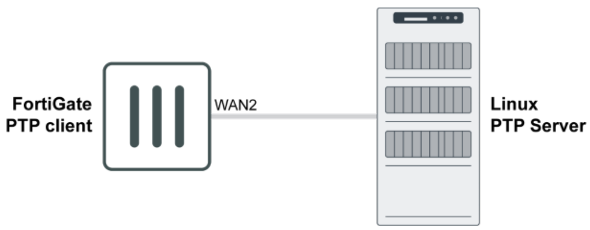
```
config system ptp
    set status {enable | disable}
    set mode {multicast | hybrid}
    set delay-mechanism {E2E | P2P}
    set request-interval <integer>
    set interface <interface>
end
```

Command	Description
status {enable | disable}	Enable or disable the FortiGate system time by synchronizing with a PTP server (default = disable).
mode {multicast | hybrid}	Use multicast or hybrid transmission (default = multicast).

Command	Description
delay-mechanism {E2E | P2P}	Use end-to-end (E2E) or peer-to-peer (P2P) delay detection (default = E2E).
request-interval <integer>	The logarithmic mean interval between the delay request messages sent by the client to the server in seconds (default = 1).
interface <interface>	The interface that the PTP client will reply through.

Sample configuration

This example uses the following topology:



To configure a FortiGate to act as a PTP client that synchronizes itself with a Linux PTP server:

1. Enable debug messages:

   ```
   # diagnose debug application ptpd -1
   ```

   This command will provide details to debug the PTP communication with the server.
2. Check the system date:

   ```
   # execute date
   current date is: 2021-04-01
   ```

3. Configure PTP in global mode:

   ```
   config system ptp
       set status enable
       set interface wan2
   end
   ```

4. Check the system date again after synchronization with the PTP server:

   ```
   # execute date
   current date is: 2021-04-27
   ```

Configuring ports

To improve security, the default ports for administrative connections to the FortiGate can be changed. Port numbers must be unique. If a conflict exists with a particular port, a warning message is shown.

When connecting to the FortiGate after a port has been changed, the port number be included, for example: `https://192.168.1.99:100`.

To configure the ports in the GUI:

1. Go to *System > Settings*.
2. In the *Administration Settings* section, set the HTTP, HTTPS, SSH, and Telnet ports.
3. Enable *Redirect to HTTPS* to prevent HTTP from being used by administrators.
4. Click *Apply*.

To configure the ports in the CLI:

```
config system global
    set admin-port <port>
    set admin-sport <port>
    set admin-https-redirect {enable | disable}
    set admin-ssh-port <port>
    set admin-telnet-port <port>
end
```

Custom default service port range

The default service port range can be customized using the following CLI command:

```
config system global
    set default-service-source-port <port range>
end
```

Where `<port range>` is the new default service port range, that can have a minimum value of 0 and a maximum value up to 65535. The default value is 1 to 65535.

This change effects the TCP/UDP protocol.

Setting the idle timeout time

The idle timeout period is the amount of time that an administrator will stay logged in to the GUI without any activity. This is to prevent someone from accessing the FortiGate if the management PC is left unattended. By default, it is set to five minutes.

A setting of higher than 15 minutes will have a negative effect on a security rating score. See Security rating on page 2335 for more information.

To change the idle timeout in the GUI:

1. Go to *System > Settings*.
2. In the *Administration Settings* section, set the *Idle timeout* to up to 480 minutes.
3. Click *Apply*.

To change the idle timeout in the CLI:

```
config system global
    set admintimeout <1-480>
end
```

Setting the password policy

A password policy can be created for administrators and IPsec pre-shared keys. See Password policy on page 1946 for information.

Changing the view settings

The view settings change the look and language of the FortiOS GUI.

To change the view settings in the GUI:

1. Go to *System* > *Settings*.
2. In the *View Settings* section, configure the following settings:

Language	Set the GUI language: *English*, *French*, *Spanish*, *Portuguese*, *Japanese*, *Traditional Chinese*, *Simplifies Chinese*, *Korean*.
Theme	Set the theme color: *Jade*, *Neutrino*, *Mariner*, *Graphite*, *Melongene*, *Retro*, *Dark Matter*, *Onyx*, or *Eclipse*.
Date/Time Display	Set the date and time to display using the FortiGate's or the browser's timezone.
NGFW Mode	Set the NGFW mode to either *Profile-based* (default) or *Policy-based*.
Central SNAT	Optionally, enable central SNAT. This option is only available in *Profile-based* mode.

3. Click *Apply*.

To change the view settings in the CLI:

```
config system global
    set language {english | french | spanish | portuguese | japanese | trach | simch |
korean}
    set gui-theme {jade | neutrino | mariner | graphite | melongene | retro | dark-matter |
onyx | eclipse}
    set gui-date-time-source {system | browser}
end

config system settings
    set ngfw-mode {profile-based | policy-based}
    set central-nat {enable | disable}
end
```

Setting the administrator password retries and lockout time

By default, the number password retry attempts is set to three, allowing the administrator a maximum of three attempts at logging in to their account before they are locked out for a set amount of time (by default, 60 seconds).

The number of attempts and the default wait time before the administrator can try to enter a password again can be configured using the CLI.

A maximum of ten retry attempts can be configured, and the lockout period can be 1 to 2147483647 seconds (over 68 years). The higher the retry attempts, the higher the risk that someone might be able to guess the password.

To configure the lockout options:

```
config system global
    set admin-lockout-threshold <failed_attempts>
    set admin-lockout-duration <seconds>
end
```

For example, to set the number of retry attempts to 1, and the lockout time to 5 minutes:

```
config system global
    set admin-lockout-threshold 1
    set admin-lockout-duration 300
end
```

If the time span between the first failed log in attempt and the lockout threshold failed attempt is less than lockout time, the lockout will be triggered.

TLS configuration

The minimum TLS version that is used for local out connections from the FortiGate can be configured in the CLI:

```
config system global
    set ssl-min-proto-version {SSLv3 | TLSv1 | TLSv1-1 | TLSv1-2 | TLSv1-3}
end
```

By default, the minimum version is TLSv1.2. The FortiGate will try to negotiate a connection using the configured version or higher. If the server that FortiGate is connecting to does not support the version, then the connection will not be made. Some FortiCloud and FortiGuard services do not support TLSv1.3.

Minimum SSL/TLS versions can also be configured individually for the following settings, not all of which support TLSv1.3:

Setting	CLI
Email server	`config system email-server`
Certificate	`config vpn certificate setting`
FortiSandbox	`config system fortisandbox`
FortiGuard	`config log fortiguard setting`

Setting	CLI
FortiAnalyzer	`config log fortianalyzer setting`
Syslog	`config log syslogd setting`
User Authentication	`config user setting`
LDAP server	`config user ldap`
POP3 server	`config user pop3`
Exchange server	`config user exchange`

A minimum (`ssl-min-proto-ver`) and a maximum (`ssl-max-proto-ver`) version can be configured for SSL VPN. See TLS 1.3 support on page 1744

Controlling return path with auxiliary session

When multiple incoming or outgoing interfaces are used in ECMP or for load balancing, changes to routing, incoming, or return traffic interfaces impacts how an existing sessions handles the traffic. Auxiliary sessions can be used to handle these changes to traffic patterns.

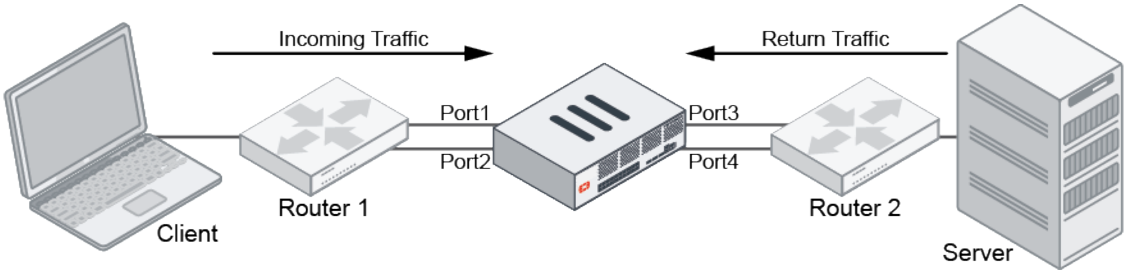
- In FortiOS 6.0 and earlier, the auxiliary session feature is not supported.
- In FortiOS 6.2.0 to 6.2.2, the auxiliary session feature is permanently enabled.
- In FortiOS 6.2.3 and later, the auxiliary session feature is disabled by default, and can be enabled if required.

To enable or disable the auxiliary session feature:

```
config system settings
    set auxiliary-session {enable | disable*}
end
```

When enabling auxiliary sessions, consider the impact of routing in both traffic directions. In topologies such as SD-WAN hub and spoke or ADVPN deployments, the symmetry of the return traffic is important for maintaining the stability of the session. It is expected that the spoke selects the outbound interface and path, and the other nodes obey and reply symmetrically. It is recommended to disable auxiliary in these scenarios, and others where incoming and return traffic symmetry is expected.

Scenarios



Incoming traffic is from the client to the server. Return traffic is from the server to the client.

Scenario 1 - Return traffic returns on the original outgoing interface

In this scenario, a session is established between port1 and port3. When the return traffic hits port3:

Auxiliary sessions disabled:

The reply to the client egresses on the original incoming interface, port1. If policy routes or SD-WAN rules are configured, the next hop gateway is applied if the output device is the same as the original incoming interface.

Auxiliary sessions enabled:

The reply to the client egresses on the best route in the routing table:

- If the best route is port1, then it will egress on port1.
- If the best route is port2, then it will egress on port2.

If policy routes or SD-WAN rules are configured, they must be matched to determine the egress interface. If both are configured, policy routes have higher priority.

Scenario 2 - Return traffic returns on an interfaces other than the original outgoing interfaces

In this scenario, a session is established between port1 and port3. When the return traffic hits port4:

Auxiliary sessions disabled:

- The session is dirtied and then gets refreshed, and interfaces on the session are updated.
- If there is a high traffic volume or flapping between the interfaces, the CPU usage increases.

Auxiliary sessions enabled:

An auxiliary session is created for the existing session, and traffic returns to the client as normal on the auxiliary session.

Scenario 3 - Incoming traffic enters on an interfaces other than the original incoming interfaces

In this scenario, a session is established between port1 and port3. When the incoming traffic hits port2:

Auxiliary sessions disabled:

The session is dirtied and then gets refreshed, and interfaces on the session are updated.

Auxiliary sessions enabled:

An auxiliary session is created for the existing session, and traffic is forwarded to the server as normal on the auxiliary session.

Scenario 4 - the routing table is changed

In this scenario, a session has been established between port1 and port3, when a new route on port4 is updated as the route to the server.

Auxiliary sessions disabled:

As long as there is a route to the destination, the session will not be dirtied or refreshed. Even though there is a better route, traffic continues on the original path between port1 and port3.

Auxiliary sessions enabled:

The session is dirtied and then gets refreshed, and interfaces on the session are updated.

Effect on NPU offloading sessions

When the auxiliary session feature is disabled, there is always one session. If the incoming or return interface changes, the FortiGate marks the session as dirty and updates the session's interfaces. This cannot be done by the NPU, so the session is not offloaded to the NPU, and is processed by the CPU instead. If Equal-Cost Multi-Path (ECMP) causes the interface to keep changing, then it will use significant CPU resources.

When the auxiliary session feature is enabled and the incoming or return interface changes, it creates an auxiliary session, and all traffic can continue to be processed by the NPU.

Verification

When an auxiliary, or reflect, session is created, it will appear as a reflect session below the existing session:

```
# diagnose sys session list
session info: proto=17 proto_state=00 duration=111 expire=175 timeout=0 flags=00000000
socktype=0 sockport=0 av_idx=0 use=4
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/255
state=may_dirty npu
statistic(bytes/packets/allow_err): org=131/4/1 reply=0/0/0 tuples=2
tx speed(Bps/kbps): 0/0 rx speed(Bps/kbps): 0/0
orgin->sink: org pre->post, reply pre->post dev=36->38/38->36 gwy=10.1.2.3/0.0.0.0
hook=pre dir=org act=noop 10.1.100.22:51926->172.16.204.44:5001(0.0.0.0:0)
hook=post dir=reply act=noop 172.16.204.44:5001->10.1.100.22:51926(0.0.0.0:0)
src_mac=90:6c:ac:19:19:58
misc=0 policy_id=1 auth_info=0 chk_client_info=0 vd=2
```

```
serial=00002b11 tos=ff/ff app_list=0 app=0 url_cat=0
sdwan_mbr_seq=0 sdwan_service_id=0
rpdb_link_id=00000000 rpdb_svc_id=0 ngfwid=n/a
npu_state=0x000400
npu info: flag=0x91/0x00, offload=8/0, ips_offload=0/0, epid=129/0, ipid=142/0,
vlan=0x0016/0x0000
vlifid=142/0, vtag_in=0x0016/0x0000 in_npu=1/0, out_npu=1/0, fwd_en=0/0, qid=4/0
no_ofld_reason:
reflect info 0:
dev=37->38/38->37
npu_state=0x000400
npu info: flag=0x91/0x00, offload=8/0, ips_offload=0/0, epid=129/0, ipid=142/0,
vlan=0x0017/0x0000
vlifid=142/0, vtag_in=0x0017/0x0000 in_npu=1/0, out_npu=1/0, fwd_en=0/0, qid=4/0
total reflect session num: 1
total session 1
```

When a session is dirtied, a dirty flag is added to it:

```
# diagnose sys session list
session info: proto=17 proto_state=00 duration=28 expire=152 timeout=0 flags=00000000
socktype=0 sockport=0 av_idx=0 use=3
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/255
state=dirty may_dirty npu
statistic(bytes/packets/allow_err): org=68/2/1 reply=0/0/0 tuples=2
tx speed(Bps/kbps): 2/0 rx speed(Bps/kbps): 0/0
orgin->sink: org pre->post, reply pre->post dev=0->0/0->0 gwy=0.0.0.0/0.0.0.0
hook=pre dir=org act=noop 10.1.100.22:51926->172.16.204.44:5001(0.0.0.0:0)
hook=post dir=reply act=noop 172.16.204.44:5001->10.1.100.22:51926(0.0.0.0:0)
src_mac=90:6c:ac:19:19:58  dst_mac=02:6c:ac:5c:c6:f9
misc=0 policy_id=1 auth_info=0 chk_client_info=0 vd=2
serial=00002b2c tos=ff/ff app_list=0 app=0 url_cat=0
sdwan_mbr_seq=0 sdwan_service_id=0
rpdb_link_id=00000000 rpdb_svc_id=0 ngfwid=n/a
npu_state=0x000400
npu info: flag=0x00/0x00, offload=0/0, ips_offload=0/0, epid=0/0, ipid=0/0,
vlan=0x0000/0x0000
vlifid=0/0, vtag_in=0x0000/0x0000 in_npu=0/0, out_npu=0/0, fwd_en=0/0, qid=0/0
no_ofld_reason:
total session 1
```

When an auxiliary session is created, NPU offloading will continue in the reflect session:

```
# diagnose sys session list
session info: proto=17 proto_state=01 duration=169 expire=129 timeout=0 flags=00000000
socktype=0 sockport=0 av_idx=0 use=4
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/255
state=may_dirty npu
statistic(bytes/packets/allow_err): org=131/4/1 reply=66/2/1 tuples=2
tx speed(Bps/kbps): 0/0 rx speed(Bps/kbps): 0/0
orgin->sink: org pre->post, reply pre->post dev=36->38/38->36 gwy=10.1.2.3/172.17.2.1
hook=pre dir=org act=noop 10.1.100.22:51926->172.16.204.44:5001(0.0.0.0:0)
```

```
hook=post dir=reply act=noop 172.16.204.44:5001->10.1.100.22:51926(0.0.0.0:0)
src_mac=90:6c:ac:19:19:58
misc=0 policy_id=1 auth_info=0 chk_client_info=0 vd=2
serial=00002b11 tos=ff/ff app_list=0 app=0 url_cat=0
sdwan_mbr_seq=0 sdwan_service_id=0
rpdb_link_id=00000000 rpdb_svc_id=0 ngfwid=n/a
npu_state=0x000c00
npu info: flag=0x91/0x81, offload=8/8, ips_offload=0/0, epid=129/142, ipid=142/128,
vlan=0x0016/0x0016
vlifid=142/128, vtag_in=0x0016/0x0016 in_npu=1/1, out_npu=1/1, fwd_en=0/0, qid=4/4
reflect info 0:
dev=37->38/38->37
npu_state=0x000400
npu info: flag=0x91/0x00, offload=8/0, ips_offload=0/0, epid=129/0, ipid=142/0,
vlan=0x0017/0x0000
vlifid=142/0, vtag_in=0x0017/0x0000 in_npu=1/0, out_npu=1/0, fwd_en=0/0, qid=4/0
total reflect session num: 1
total session 1
```

Email alerts

Alert emails are used to notify administrators about events on the FortiGate device, allowing a quick response to any issues.

There are two methods that can be used to configure email alerts:

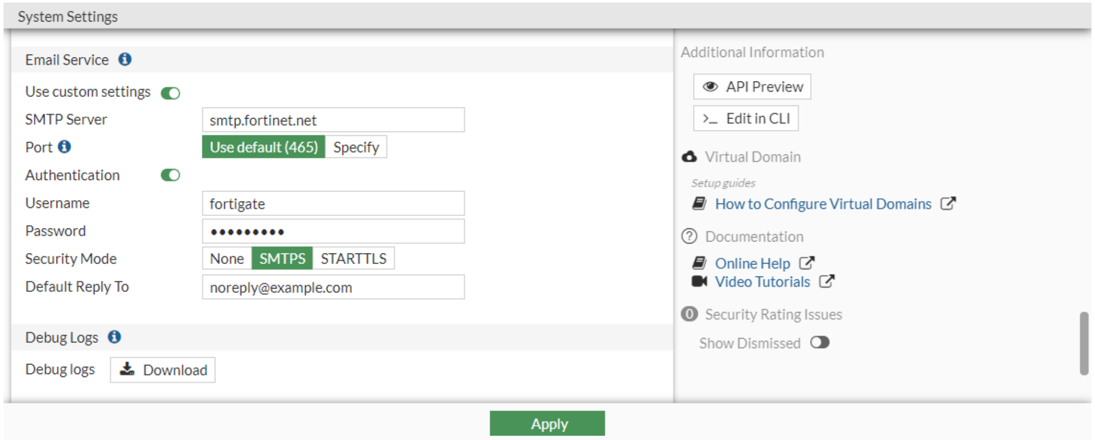
- Automation stitches on page 1988
- Alert emails on page 1991

The FortiGate has a default SMTP server, notification.fortinet.net, that provides secure mail service with SMTPS. It is used for all emails that are sent by the FortiGate, including alert emails, automation stitch emails, and FortiToken Mobile activations. You can also configure a custom email service.

To configure a custom email service in the GUI:

1. Go to *System > Settings*.
2. In the *Email Service* section, enable *Use custom settings*.
3. Configure the following settings:

SMTP Server	Enter the address or name of the SMTP server, such as *smtp.example.com*.
Port	If required, select *Specify* and enter a specific port number. The default is port 465.
Authentication	If required by the email server, enable authentication. If enabled, enter the *Username* and *Password*.
Security Mode	Set the security mode: *None*, *SMTPS*, or *STARTTLS*.
Default Reply To	Optionally, enter the reply to email address, such as noreply@example.com. This address will override the from address that is configured for an alert email.



4. Click *Apply*.

To configure a custom email service in the CLI:

```
config system email-server
    set reply-to "noreply@example.com"
    set server "smtp.fortinet.net"
    set port 465
    set authenticate enable
    set username "fortigate"
    set password **********
    set security smtps
end
```

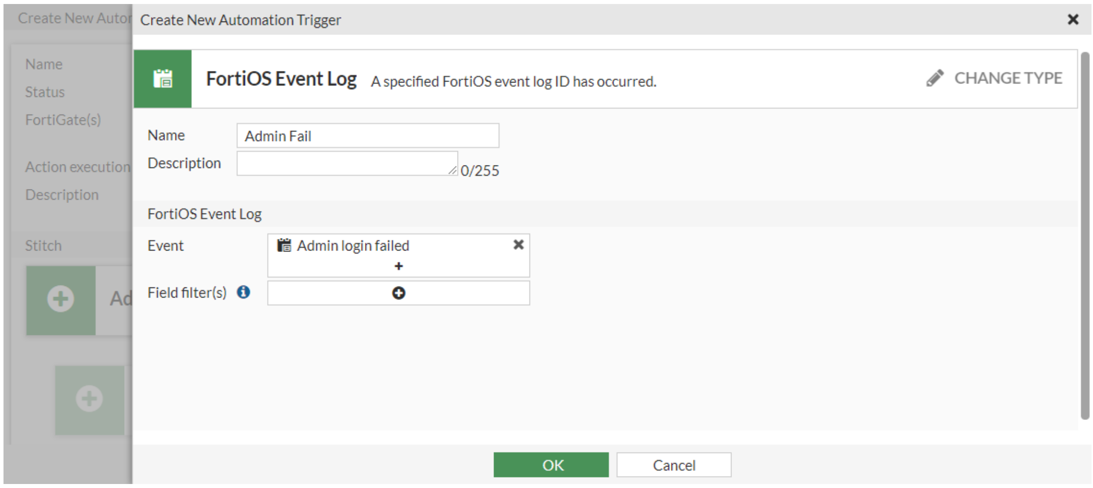
Automation stitches

Automation stitches can be configured to send emails based on a variety of triggers, giving you control over the events that cause an alert, and who gets alerted. For more information, see Automation stitches on page 2342.

In this example, the default mail service sends an email to two recipients when an Admin login failed event occurs or there is a configuration change.

To configure the automation stitch in the GUI:

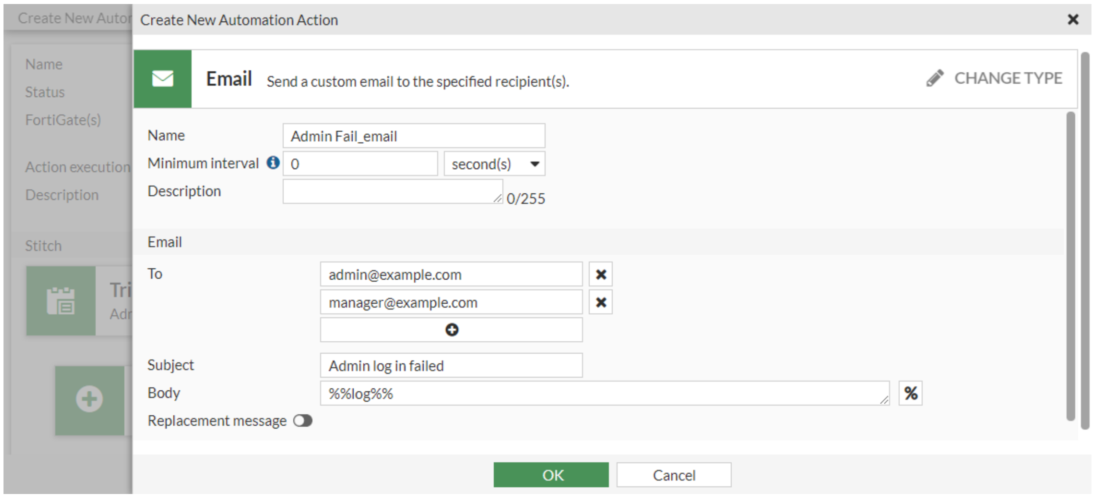
1. On the root FortiGate, go to *Security Fabric > Automation* and click *Create New*.
2. Enter a name for the stitch, such as *Admin Fail*.
3. Configure the trigger:
   a. Click *Add Trigger*.
   b. Click *Create* and select *FortiOS Event Log*.
   c. Enter a name for the trigger, such as *Admin Fail*.
   d. Click in the *Event* field, and in the slide out pane, search for and select *Admin login failed*.



e. Click *OK*.
f. Select the trigger in the list and click *Apply*.

4. Configure the action:
   a. Click *Add Action*.
   b. Click *Create* and select *Email*.
   c. Configure the following settings:

Name	Enter a name for the action, such as *Admin Fail_email*.
To	Enter the two email recipients' addresses, such as *admin@example.com* and *manager@example.com*.
Subject	Enter an subject, such as *Admin log in failed*.
Body	Edit as required. By default, the email body will include all the fields from the log event that triggered the stitch.



   d. Click *OK*.
   e. Select the action in the list and click *Apply*.
5. Click *OK*.

6. Create a second stitch with *Configuration Change* as the trigger, and an email action with a different subject line (such as *Configuration Change Detected*).

To configure the automation stitch in the CLI:

1. Create the automation triggers:

```
config system automation-trigger
    edit "Admin Fail"
        set event-type event-log
        set logid 32002
    next
    edit "Config Change"
        set event-type config-change
    next
end
```

2. Create automation actions to send the email messages:

```
config system automation-action
    edit "Admin Fail_email"
        set action-type email
        set email-to "admin@example.com" "manager@example.com"
        set email-subject "Admin log in failed"
    next
    edit "Config Change_email"
        set action-type email
        set email-to "admin@example.com" "manager@example.com"
        set email-subject "Configuration Change Detected"
    next
end
```

3. Create the automation stitches:

```
config system automation-stitch
    edit "Admin Fail"
        set trigger "Admin Fail"
        config actions
            edit 1
                set action "Admin Fail_email"
                set required enable
            next
        end
    next
    edit "Config Change"
        set trigger "Config Change"
        config actions
            edit 1
                set action "Config Change_email"
                set required enable
            next
        end
    next
end
```

Alert emails

When configuring an alert email, you can define the threshold when an issue becomes critical and requires attention. When the threshold is reached, an email is sent to up to three recipients on the configured schedule to notify them of the issue.

Alert email messages can be configured in the CLI. For more information on the available CLI commands, see Configure alert email settings.

IPS, SSH, violation traffic, antivirus, and web filter logs are supported as triggers in automation stitches. For more information, see Event log category triggers on page 2370.

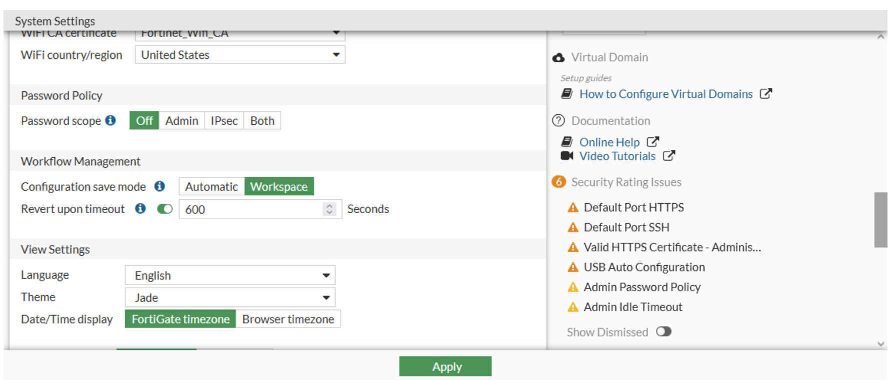
In this example, the FortiGate is configured to send email messages to two addresses, admin@example.com and manager@example.com, every two minutes when multiple intrusions, administrator log in or out events, or configuration changes occur.

To configure an alert email:

```
config alertemail setting
    set username fortigate@example.com
    set mailto1 admin@example.com
    set mailto2 manager@example.com
    set filter-mode category
    set email-interval 2
    set IPS-logs enable
    set configuration-changes-logs enable
    set admin-login-logs enable
end
```

Using configuration save mode

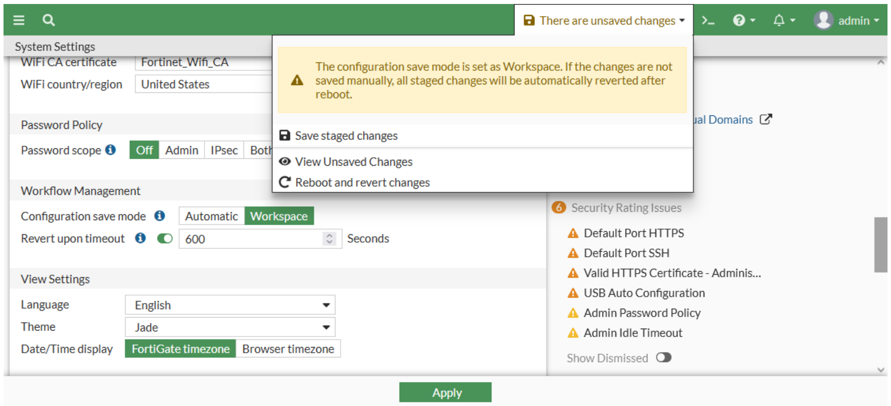
Administrators can use configuration save mode set to *Workplace* to implement strict change control by requiring changes to be manually committed to the flash. To configure the setting in the GUI, go to *System > Settings*.



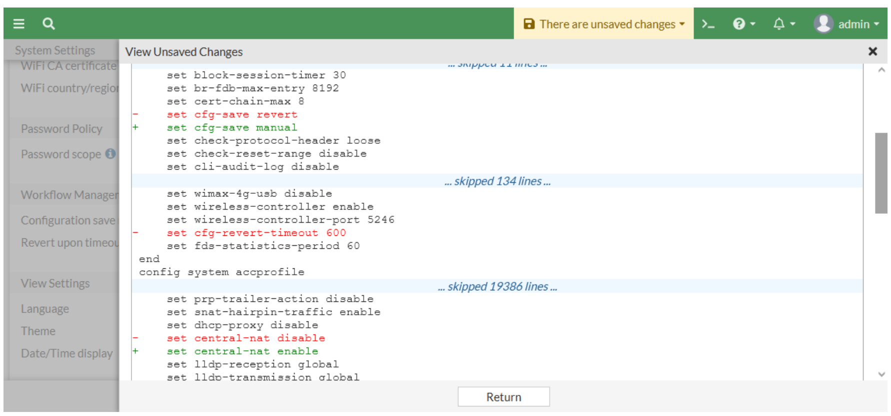
When *Configuration save mode* is set to *Automatic* (default), configuration changes are automatically saved to both memory and flash.

When *Configuration save mode* is set to *Workspace*, configuration changes are saved to memory, but not to flash. The changes take effect immediately, but must be manually saved to flash. Unsaved changes are reverted when the device is rebooted. If *Revert upon timeout* is enabled, the system might be unresponsive for a short time after the configured timeout while it reverts the changes back to the previous save point. Prior to the timeout expiring, a pop-up warning gives you the option to postpone the reboot by one minute, revert the configuration immediately, or save the configuration changes.

In workspace mode, a warning is shown in the banner when there are unsaved changes. Click the warning to save, view, or revert the changes. When you click *Reboot and revert changes*, the system might be unresponsive for a short time while it reverts the changes back to the previous save point.



Clicking *View Unsaved Changes* opens a pane highlighting the changes that have not been committed.



This feature is also available in the CLI:

```
config system global
    set cfg-save {automatic | manual | revert}
    set cfg-revert-timeout <integer>
end
```

```
# execute cfg {reload | save}
```

Trusted platform module support

On supported FortiGate hardware devices, the Trusted Platform Module (TPM) can be used to protect your password and key against malicious software and phishing attacks. The dedicated module hardens the FortiGate by generating, storing, and authenticating cryptographic keys. To help prevent tampering, the chip is soldered on the motherboard to reduce the risk of data transaction interceptions from attackers.

By default, the TPM is disabled. To enable it, you must set the 32 hexadecimal digit master-encryption-password which encrypts sensitive data on the FortiGate using AES128-CBC. With the password, TPM generates a 2048-bit primary key to secure the master-encryption-password through RSA-2048 encryption. The master-encryption-password protects the data. The primary key protects the master-encryption-password.

The TPM module does not encrypt the disk drive of eligible FortiGates.

The primary key binds the encrypted configuration file to a specific FortiGate unit and never leaves the TPM. When backing up the configuration, the TPM uses the primary key to encrypt the master-encryption-password in the configuration file. When restoring a configuration that includes a TPM protected master-encryption-password:

- If TPM is disabled, then the configuration cannot be restored.
- If TPM is enabled but has a different master-encryption-password than the configuration file, then the configuration cannot be restored.
- If TPM is enabled and the master-encryption-password is the same in the configuration file, then the configuration can be restored.

For information on backing up and restoring the configuration, see Configuration backups on page 65.

Passwords and keys that can be encrypted by the master-encryption-key include:

- Alert email user's password
- BGP and other routing related configurations
- External resource
- FortiGuard proxy password
- FortiToken/FortiToken Mobile’s seed
- HA password
- IPsec pre-shared key
- Link Monitor, server side password
- Local certificate's private key
- Local, LDAP. RADIUS, FSSO, and other user category related passwords
- Modem/PPPoE
- NST password
- NTP Password
- SDN connector, server side password
- SNMP
- Wireless Security related password

In HA configurations, each cluster member must use the same master-encryption-key so that the HA cluster can form and its members can synchronize their configurations.

To check if your FortiGate device has a TPM:

Verify all the following commands exist. Otherwise, the platform does not support it.

```
# diagnose hardware test info
List of test cases:
```

```
    bios: sysid
    bios: checksum
    bios: license
    bios: detect

# diagnose hardware deviceinfo tpm
TPM capability information of fixed properties:
=========================================================
TPM_PT_FAMILY_INDICATOR: 2.0
TPM_PT_LEVEL: 0
TPM_PT_REVISION: 138
TPM_PT_DAY_OF_YEAR: 8
TPM_PT_YEAR: 2018
TPM_PT_MANUFACTURER: NTC
# diagnose hardware test tpm
=========== Fortinet Hardware Test Report ===================
TPM
TPM Device Detection.......................................... PASS
================= Fortinet Hardware Test PASSED ==============
# diagnose tpm
get-property Get TPM properties. [Take 0-1 arg(s)]
get-var-property Get TPM var properties.
read-clock Read TPM internal clock.
shutdown-prepare Prepare for TPM power cycle.
selftest Perform self tests.
generate-random-number Generate a 4-byte random number
SHA-1 HASH a sequence of num with SHA-1 algo
SHA-256 HASH a sequence of num with SHA-256 algo
```

To enable TPM and input the master-encryption-password:

```
config system global
    set private-data-encryption enable
end
Please type your private data encryption key (32 hexadecimal numbers):
********************************
Please re-enter your private data encryption key (32 hexadecimal numbers) again:
********************************
Your private data encryption key is accepted.
```

Virtual Domains

Virtual Domains (VDOMs) are used to divide a FortiGate into two or more virtual units that function independently. VDOMs can provide separate security policies and, in NAT mode, completely separate configurations for routing and VPN services for each connected network.

Multiple VDOMs can be created and managed as independent units in multi VDOM mode.

By default, most FortiGate units support 10 VDOMs, and many FortiGate models support purchasing a license key to increase the maximum number. Some exceptions may apply.

The following topics provide an overview of VDOM concepts, topologies, best practices, and the general configurations involved when working with multi VDOM mode:

- VDOM overview on page 1995
- General configurations on page 2000

The following topics provide examples of configuring VDOMs:

- Inter-VDOM routing configuration example: Internet access on page 2007
- Inter-VDOM routing configuration example: Partial-mesh VDOMs on page 2017

VDOM overview

The following sections provide conceptual information on VDOMs:

- Multi VDOM mode on page 1995
- Global settings on page 1995
- Global and per-VDOM resources on page 1995
- Management VDOM on page 1996
- VDOM types on page 1996
- Administrator roles and views on page 1996
- Inter-VDOM routing on page 1997

The following sections provide information on methods of VDOM configuration:

- Topologies on page 1997
- Best practices on page 1999

Multi VDOM mode

In multi VDOM mode, the FortiGate can have multiple VDOMs that function as independent units. When multi VDOM mode is first enabled, all VDOM configurations will move to the root VDOM by default. The root VDOM cannot be deleted, and remains in the configuration even if it is not processing any traffic. New VDOMs can be created, up to the VDOM limit allowable on your device.

Global settings

Global settings are configured outside of a VDOM. They effect the entire FortiGate, and include settings such as interfaces, firmware, DNS, some logging and sandboxing options, and so on. Global settings should only be changed by top level administrators.

Global and per-VDOM resources

Global and per-VDOM resources can be configured when the FortiGate is in multi VDOM mode. Global resources apply to resources that are shared by the whole FortiGate, while per-VDOM resources are specific to each VDOM.

By default, all per-VDOM resource settings are set to have no limits. This means that any single VDOM can use all of the FortiGate device's resources. This could deprive other VDOMs of the resources that they require, to the point that could be unable to function. We recommend setting maximum values on the resources that are vital to you.

Management VDOM

The management VDOM refers to the specific role that must be designated to one of the VDOMs. By default, the root VDOM is the management VDOM, and management-related services such as FortiGuard updates and other local out (self-originating) traffic such as logs to remote servers originate from the management VDOM. The management VDOM cannot be deleted. See Management VDOM on page 2001 for configuration details.

VDOM types

When a FortiGate is in multi VDOM mode, a VDOM can be configured as an *Admin* or *Traffic* type VDOM.

When the VDOM type is set to *Admin*, the VDOM is used to administer and manage the FortiGate. Usually, the *Admin* VDOM resides in a management network which is only accessible by administrators. Global and VDOM administrators can log in to the FortiGate using SSH, HTTPS, and so on but traffic cannot pass through this *Admin* VDOM. A FortiGate does not need to have an *Admin* VDOM and, at most, there can only be one *Admin* VDOM per FortiGate.

When VDOM type is set to *Traffic*, the VDOM can pass traffic like a regular firewall. Most VDOMs will be *Traffic* type VDOMs. Network interfaces on a *Traffic* VDOM can also enable SSH, HTTPS, and so on for administrative and management purposes.

In general, an *Admin* VDOM has a subset of a *Traffic* VDOM's capabilities.

FortiGate-VM supports having at least two VDOMs; one that supports an administrative VDOM and another that supports a traffic VDOM.

See Configure an administrative VDOM type on page 2003 for configuration details.

Administrator roles and views

When a FortiGate has been configured in multi VDOM mode, the device can be managed by global administrators and per-VDOM administrators. Each type of administrator will have a different view of the GUI in multi VDOM mode which corresponds to their role.

Global administrators

Global administrators have complete visibility and access because the scope of their role is to manage the entire physical FortiGate device. An example of a global administrator is an administrator working for a managed security services provider (MSSP) providing the FortiGate as a multi-tenant environment to its clients.

When global administrators log into the GUI, from the *VDOM: Global* view they will see all pages for global settings shared between VDOMs, and VDOM-specific settings.

To create a global administrator that has access to all VDOMs and access to global settings, it must be created at the global level and must use the *super_admin* administrator profile.

See Administrator profiles on page 1949 and Local authentication on page 1937 for configuration details.

VDOM administrators

VDOM administrators will be unable to view global settings or VDOMs not assigned to them because the scope of their role is restricted to managing specific VDOMs only. An example of a VDOM administrator is the administrator working for a company which is a client, or tenant, of an MSSP’s multi-tenant FortiGate.

When VDOM administrators log into the GUI, from the *VDOM:<VDOM>* view they will see pages for settings specific to the VDOM they have been configured to administer such as interfaces, routes, firewall policies, and security profiles.

See Create per-VDOM administrators on page 2003 for configuration details.

Inter-VDOM routing

VDOM links are virtual interfaces that allow VDOMs to communicate internally without using additional physical interfaces. A VDOM link contains a pair of interfaces, each one connected to a VDOM to form each end of the inter-VDOM connection. Inter-VDOM routing can be configured in order to communicate between one VDOM to another.

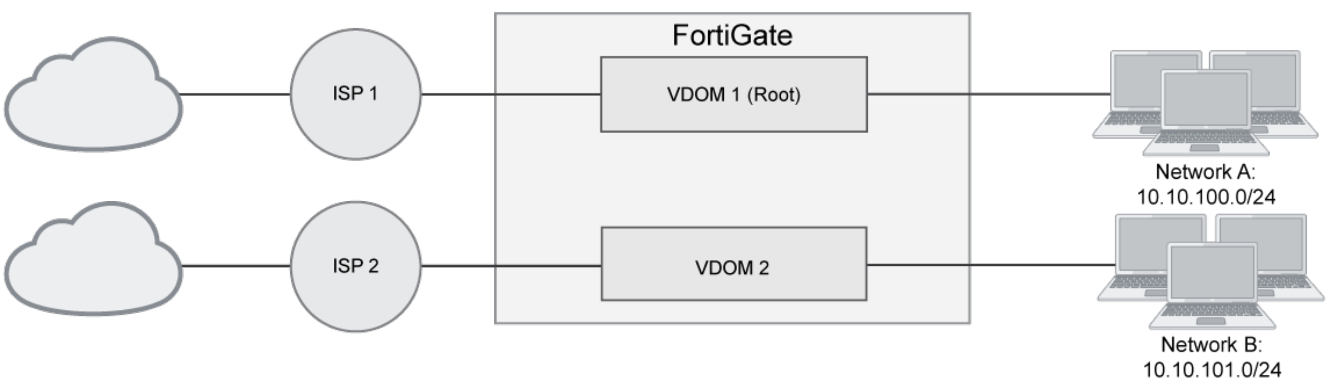
When VDOMs are configured on your FortiGate unit, configuring inter-VDOM routing and VDOM links is similar to creating a VLAN interface. VDOM links can be managed in either the CLI or in the network interface list in the GUI.

See Inter-VDOM routing configuration example: Internet access on page 2007 for more information.

Topologies

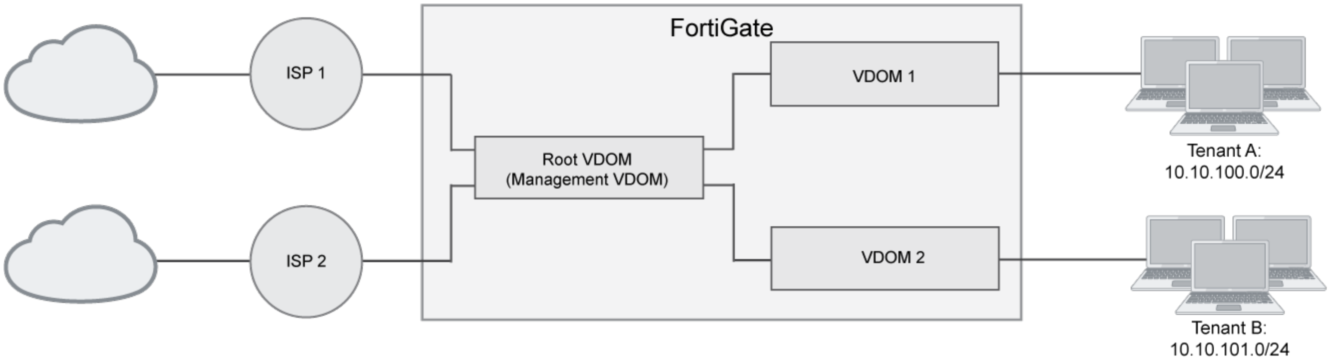
These are the main configuration types in multi VDOM mode:

Independent VDOMs



Multiple, completely separate VDOMs are created. Any VDOM can be the management VDOM, as long as it has Internet access to connect to FortiGuard services and other management resources. There are no inter-VDOM links, and each VDOM is independently managed.

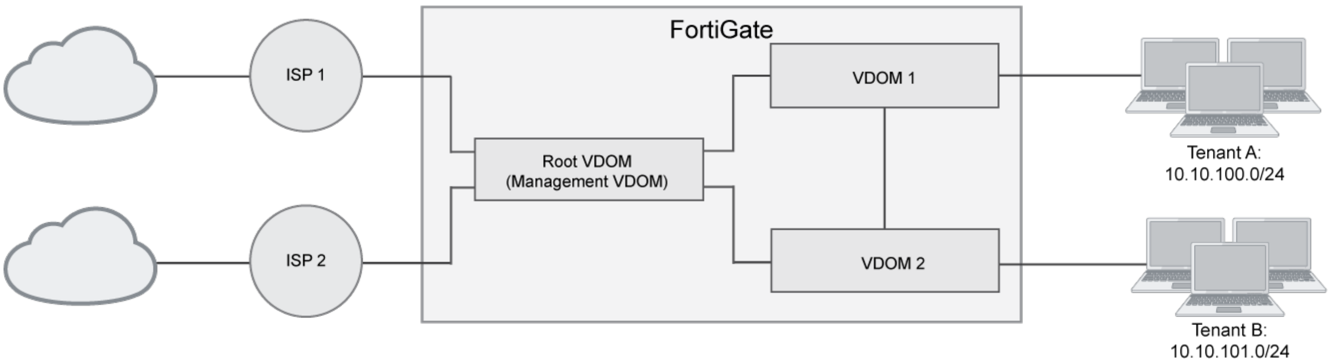
Internet access VDOM



In the Internet access VDOM configuration, Internet access is provided primarily by a single VDOM; for example, the management VDOM (depicted as root VDOM in the preceding diagram). Each tenant connects to the management VDOM via an inter-VDOM link. The management VDOM has complete control over Internet access, including the types of traffic that are allowed in both directions. This can improve security, as there is only one point of ingress and egress.

There is no communication between the other VDOMs.

Meshed VDOMs

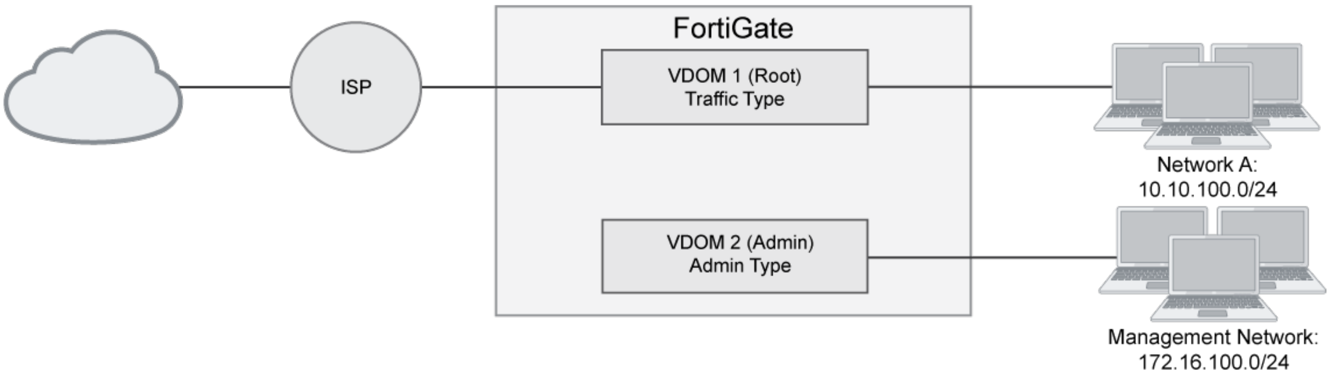


VDOMs can communicate with inter-VDOM links. In full-mesh configurations, all the VDOMs are interconnected. In partial-mesh configurations, only some of the VDOMs are interconnected.

In this configuration, inter-vdom links between tenants are created by the global administrator, but each tenant controls the firewall policies to allow access to other tenants.

See Inter-VDOM routing on page 2005 and Inter-VDOM routing configuration example: Internet access on page 2007 for configuration details.

Administrative VDOM on a management network



The administrative VDOM type can be used to limit administrative access to the FortiGate using SSH, HTTPS and so on to administrators working from a management network. Administrators may be limited to management settings or may have global privileges to access other VDOMs. The user or tenant network (depicted as Network A in the diagram) uses a traffic type VDOM, which allows traffic to pass through it like a regular firewall and allows configuration of firewall-related settings. This configuration can improve security if the management network is a closed network and administrative access is not enabled on any interfaces on the traffic VDOM.

Best practices

VDOMs can provide separate firewall policies and, in NAT mode, completely separate configurations for routing and VPN services for each connected network or organization. This section provides a list of best practices for configuring VDOMs.

Per-VDOM resource setting

All per-VDOM resource settings are set to no limit by default. To ensure proper functionality of all VDOMs, it is recommended that you set some maximum values for the most vital resources. See Global and per-VDOM resources on page 2001 for configuration details.

Virtual domains in NAT mode

Once the virtual domains have been enabled and one or more VDOMs have been created, they must be configured. The following steps provide a general overview of the configuration process.

To configure VDOMs:

1. Change the management virtual domain.
2. Configure FortiGate interfaces for your VDOMs in NAT mode.
3. Configure VDOM routing.
4. Configure security policies for VDOMs in NAT mode.
5. Configure UTM profiles for VDOMs in NAT mode.
6. Test the configuration.

While you may not require all of the steps for your network topology, it is recommended that you perform them in the order given.

See General configurations on page 2000 for configuration details.

Virtual clustering

Virtual clustering is an extension of FGCP HA that provides failover protection between two instances of one or more VDOMs operating on two FortiGates that are in a virtual cluster. A standard virtual cluster consists of FortiGates that are operating in active-passive HA mode with multiple VDOMs enabled. See HA virtual cluster setup on page 2039 for more details.

Typically, virtual clustering is configured with override enabled and uses device priorities to distribute traffic between the primary and secondary FortiGates.

If you decide to disable override for clustering, as a result of persistent renegotiating, you should disable it for both cluster units.

General configurations

VDOMs can be configured in the GUI and the CLI. To ensure that no VDOMs are accidentally configured in the CLI, prompts can be enabled. These prompts will display to ask for confirmation that the VDOM is meant to be configured in the CLI.

To configure confirmation prompts:

```
config system global
    set edit-vdom-prompt enable
end
```

The following topics provide information on general VDOM configurations:

Enable multi VDOM mode

Enable multi VDOM mode and create the VDOMs in the GUI and CLI.

On FortiGate 90 series models and lower, VDOMs can only be enabled using the CLI.

To enable VDOMs in the GUI:

1. Go to *System > Settings*.
2. In the *System Operation Settings* sections, enable *Virtual Domains*.
3. Click *OK*.

To enable VDOMs in the CLI:

```
config system global
    set vdom-mode multi-vdom
end
```

You will be logged out of the device when the VDOM mode is enabled.

Management VDOM

By default, the management VDOM is *root*. The management VDOM can be manually assigned from the GUI or the CLI.

To assign the management VDOM in the GUI:

1. In the *Global* VDOM, go to *System > VDOM*.
2. Select the VDOM you want to assign as the management VDOM.
3. Click *Switch Management*.
4. Click *OK*.

To assign the management VDOM in the CLI:

```
config global
    config system global
        set management-vdom <vdom>
    end
end
```

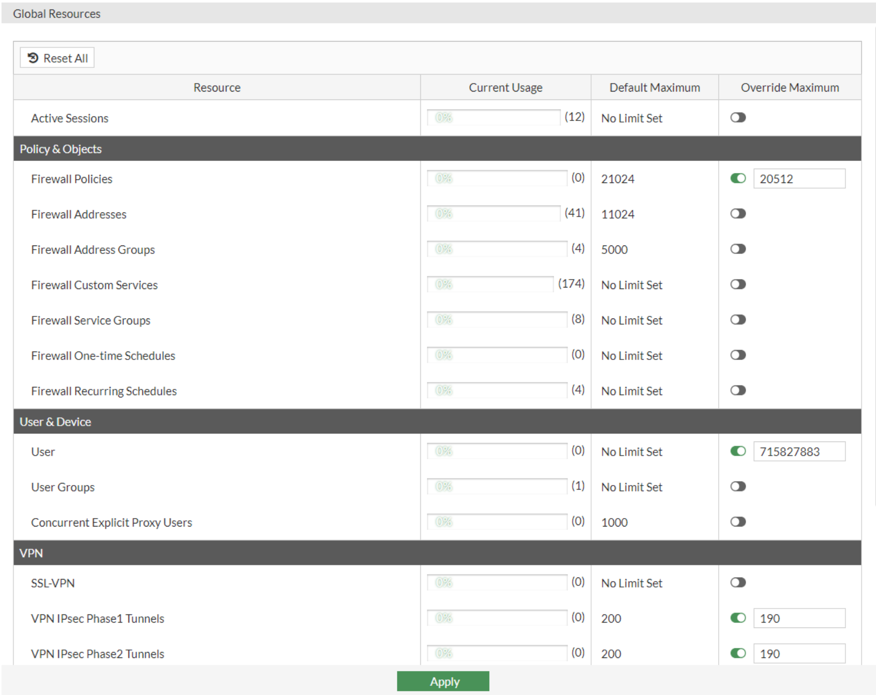
Only one management VDOM can exist at a time. It is strongly recommended that the management VDOM have Internet access otherwise management-related services, such as FortiGuard updates and queries, will not work.

Global and per-VDOM resources

Global resources apply to resources that are shared by the whole FortiGate, while per-VDOM resources are specific to each VDOM.

To configure global resources:

1. In the *Global* VDOM, go to *System > Global Resources*.
2. Enable the resource's override in the *Override Maximum* column, then enter the override value.

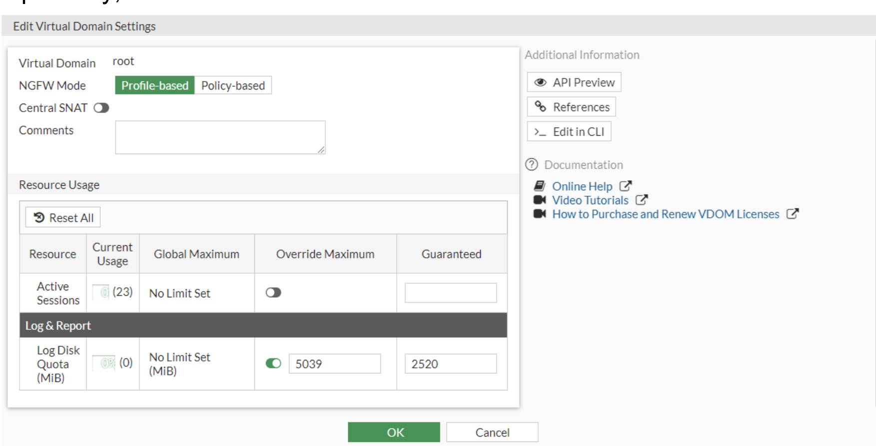


3. Click *Apply*.

   To reset all of the override values, click *Reset All*.

To configure per-VDOM resources:

1. In the *Global* VDOM, go to *System > VDOM*.
2. Select the VDOM whose resources need to be configured and click *Edit*.
3. Enable the resource's override in the *Override Maximum* column, then enter the override value.
4. Optionally, enter a value in the *Guaranteed* column.



5. Click *OK*.

   To reset all of the override values, click *Reset All*.

Create per-VDOM administrators

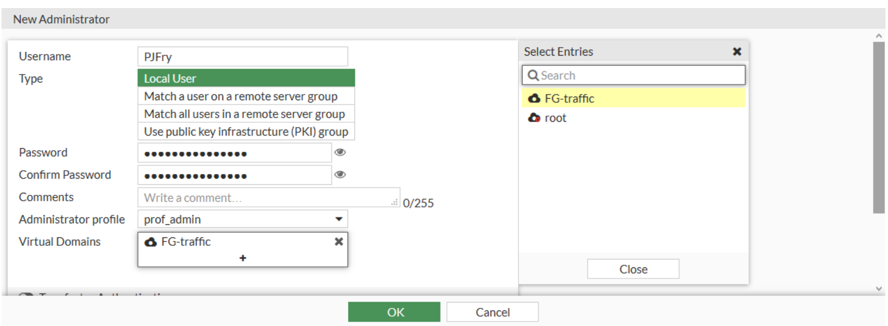
Per-VDOM administrators can be created that can access only the administrative or traffic VDOM. These administrators must use either the *prof_admin* administrator profile, or a custom profile.

A per-VDOM administrator can only access the FortiGate through a network interface that is assigned to the VDOM that they are assigned to. The interface must also be configured to allow management access. They can also connect to the FortiGate using the console port.

To assign an administrator to multiple VDOMs, they must be created at the global level. When creating an administrator at the VDOM level, the *super_admin* administrator profile cannot be used.

To create a per-VDOM administrator in the GUI:

1. On the FortiGate, connect to the *Global* VDOM.
2. Go to *System > Administrators* and click *Create New > Administrator*.
3. Fill in the required information, setting the *Type* as *Local User*.
4. In the *Virtual Domains* field, add the VDOM that the administrator will be assigned to, and if necessary, remove the other VDOM from the list.



5. Click *OK*.

To create a per-VDOM administrator using the CLI:

```
config global
    config system admin
        edit <name>
            set vdom <VDOM_name>
            set password <password>
            set accprofile <admin_profile>
            ...
        next
    end
end
```

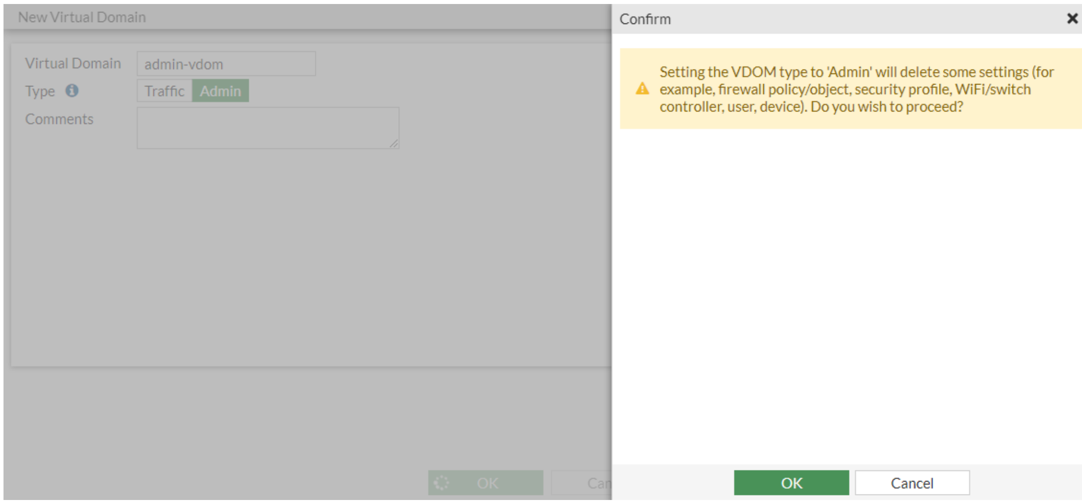
Configure an administrative VDOM type

Individual VDOMs can be configured as an administrative type in multi VDOM mode.

Only one administrative VDOM can exist at a time and cannot be set on a FortiWifi. A VDOM cannot be an administrative type and in transparent mode at the same time.

To configure an administrative VDOM in the GUI:

1. Go to *System > VDOM*.
2. Click *Create New*.
3. Enter a *Virtual Domain* name and set the *Type* to *Admin*.
4. Click *OK*.



5. Click *OK* in the confirmation pane. The administrative VDOM is created.

To configure the VDOM type in the CLI:

```
config system settings
    set vdom-type {traffic | admin}
end
```

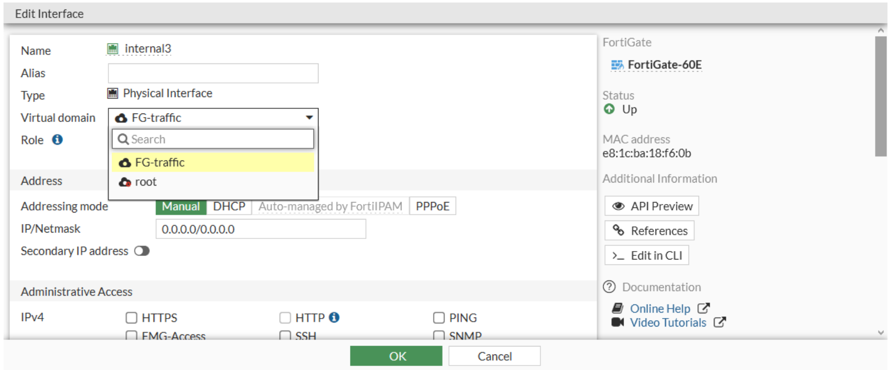
Assign interfaces to a VDOM

An interface can only be assigned to one of the VDOMs. An interface cannot be moved if it is referenced in an existing configuration.

In the GUI, the interface list *Ref.* column shows if the interface is referenced in an existing configuration, and allows you to quickly access and edit those references.

To assign an interface to a VDOM in the GUI:

1. In the *Global* VDOM, go to *Network > Interfaces*.
2. Select the interface that will be assigned to a VDOM and click *Edit*.
3. Select the VDOM that the interface will be assigned to from the *Virtual Domain* list.



4. Click *OK*.

To assign an interface to a VDOM using the CLI:

```
config global
    config system interface
        edit <interface>
            set vdom <VDOM_name>
        next
    end
end
```

Inter-VDOM routing

VDOM links allow VDOMs to communicate internally without using additional physical interfaces.

VDOM link does not support traffic offload. If you want to use traffic offload, use NPU-VDOM-LINK. See Configuring inter-VDOM link acceleration with NP6 processors in the Hardware Acceleration guide for details.

To configure a VDOM link in the GUI:

1. In the *Global* VDOM, go to *Network > Interfaces*.
2. Click *Create New > VDOM Link*.
3. Configure the fields, including the *Name*, *Virtual Domain*, IP information, *Administrative Access*, and so on, then click *OK*.

By default, VDOM links are created as point-to-point (ppp) links. If required, the link type can be changed in the CLI.

For example, when running OSPF in IPv6, a link-local address is required in order to communicate with OSPF neighbors. For a VDOM link to obtain a link-local address, its type must be set to `ethernet`.

To configure a VDOM link in the CLI:

```
config global
    config system vdom-link
```

```
        edit "<vdom-link-name>"
            set type {ppp | ethernet}
        next
    end
    config system interface
        edit "<vdom-link-name0>"
            set vdom "<VDOM Name>"
            set type vdom-link
        next
        edit "<vdom-link-name1>"
            set vdom "<VDOM Name>"
            set type vdom-link
        next
    end
end
```

To delete a VDOM link in the GUI:

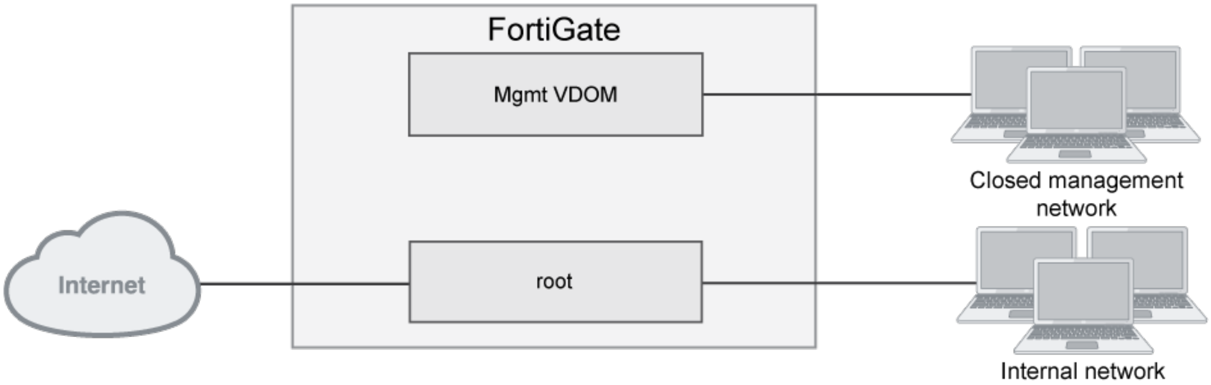
1. In the *Global* VDOM, go to *Network > Interfaces*.
2. Select a *VDOM Link* and click *Delete*.

To delete a VDOM link in the CLI:

```
config global
    config system vdom-link
        delete <VDOM-LINK-Name>
    end
end
```

Allow FortiGuard services and updates to initiate from a traffic VDOM

In multi VDOM mode, users can choose from which VDOM FortiGuard services and updates are initiated from, instead of being locked to the management VDOM. This allows deployment scenarios where the management VDOM resides in a closed management network.



When the management VDOM resides in a closed network, it does not have internet access. FortiGuard services (FortiGuard updates, web filters, DNS proxy, DDNS, and so on) must be configured in a VDOM with Internet access in order to work. Therefore, in the example above, change the FortiGuard settings to initiate from the root VDOM.

To configure FortiGuard services on a traffic VDOM:

1. Set up a traffic VDOM for FortiGuard services:

```
config global
    config system fortiguard
        set vdom "root"
    end
end
```

2. Ensure the traffic VDOM has the correct gateway to reach the internet:

```
config vdom
    edit root
        config router static
            edit 1
                set gateway 172.16.200.254
                set device "wan1"
            next
        end
    next
end
```

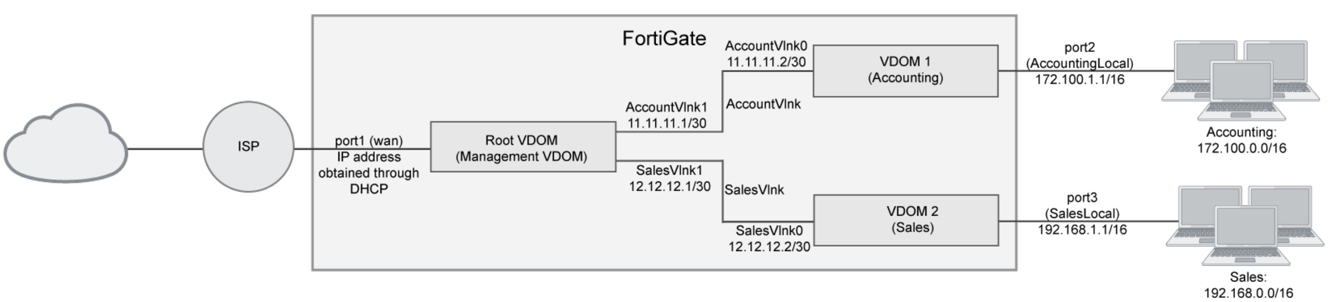
3. Configure the DNS servers to ensure the FortiGuard services can resolve the server name through the traffic VDOM:

```
config vdom
    edit root
        config system vdom-dns
            set vdom-dns enable
            set primary 208.91.112.53
            set secondary 208.91.112.52
        end
    next
end
```

Inter-VDOM routing configuration example: Internet access

This example shows how to configure a FortiGate unit to use inter-VDOM routing to route outgoing traffic from individual VDOMs to a root VDOM with Internet access. See Inter-VDOM routing on page 2005 for more information.

Two departments of a company, Accounting and Sales, are connected to one FortiGate. The company uses a single ISP to connect to the Internet. This is an example of the Internet access configuration. See Topologies on page 1997 for details.



This example assumes that the interfaces of the FortiGate have already been configured with the IP addresses depicted in the preceding diagram.

General steps for this example

This example includes the following general steps. We recommend following the steps in the order below:

1. Enable multi VDOM mode and create the VDOMs on page 2008
2. Assign interfaces to VDOMs on page 2009
3. Configure the VDOM links on page 2009
4. Configure inter-VDOM routing on page 2011
5. Configure the firewall policies on page 2012
6. Test the configuration on page 2013

This example demonstrates how to configure these steps first using the GUI and then, at the end of the section, using the CLI. See Configuration with the CLI on page 2014 for details.

Enable multi VDOM mode and create the VDOMs

Create the Accounting and Sales VDOMs.

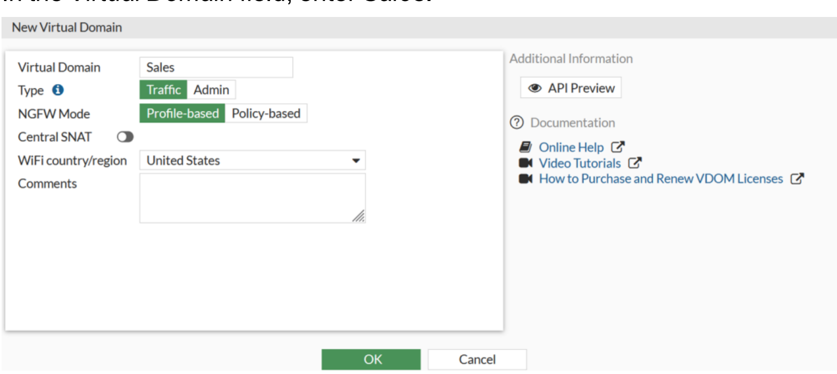
To enable VDOMs in the GUI:

1. Go to *System > Settings*.
2. In the *System Operation Settings* section, enable *Virtual Domains*.
3. Click *OK*.

On FortiGate 90 series models and lower, VDOMs can only be enabled using the CLI.

To create the Sales and Accounting VDOMs in the GUI:

1. In the *Global* VDOM, go to *System > VDOM*.
2. Click *Create New*.
3. In the *Virtual Domain* field, enter *Sales*.



4. If required, set the *NGFW Mode*. If the *NGFW Mode* is *Profile-based*, *Central SNAT* can be enabled.

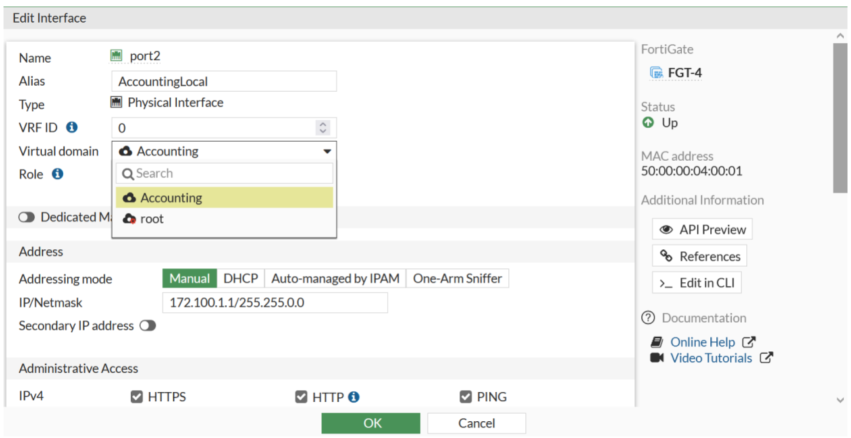
5. Click *OK* to create the VDOM.
6. Repeat the above steps for *Accounting*.

Assign interfaces to VDOMs

This example uses three interfaces on the FortiGate unit: port2 (AccountingLocal), port3 (SalesLocal), and port1 (WAN). Port2 and port3 interfaces each have a department's network connected. Port1 is for all traffic to and from the Internet and uses DHCP to configure its IP address, which is common with many ISPs.

To assign interfaces to VDOMs in the GUI:

1. In the *Global* VDOM, go to *Network > Interfaces*.
2. Select *port2* and click *Edit*.
3. From the *Virtual domain* list, select *Accounting*.



4. Click *OK*.
5. Repeat the preceding steps to assign *port3* to the *Sales* VDOM.
6. Repeat the preceding steps to assign *port1* to the *root* VDOM.

Configure the VDOM links

To complete the connection between each VDOM and the management VDOM, add the two VDOM links. One pair is the Accounting – management link and the other is the Sales – management link. Each side of these links will be assigned IP addresses since they will be handy in configuring inter-VDOM routing in the next step.

To configure the Accounting and management VDOM link in the GUI:

1. In the *Global* VDOM, go to *Network > Interfaces*.
2. Select *Create New > VDOM Link*.

3. Enter the following information:

Name		AccountVlnk
Interface 0		
	Virtual Domain	Accounting
	IP/Netmask	11.11.11.2/255.255.255.252
	Administrative Access	HTTPS, PING, SSH
	Comment	Accounting side of the VDOM link
Interface 1		
	Virtual Domain	root
	IP/Netmask	11.11.11.1/255.255.255.252
	Administrative Access	HTTPS, PING, SSH
	Comment	Management side of the VDOM link

4. Click *OK*.

To configure the Sales and management VDOM link in the GUI:

1. In the *Global* VDOM, go to *Network* > *Interfaces*.
2. Select *Create New* > *VDOM link*.
3. Enter the following information:

Name		SalesVlnk
Interface 0		
	Virtual Domain	Sales
	IP/Netmask	12.12.12.2/255.255.255.252
	Administrative Access	HTTPS, PING, SSH
	Comment	Accounting side of the VDOM link
Interface 1		
	Virtual Domain	root
	IP/Netmask	12.12.12.1/255.255.255.252
	Administrative Access	HTTPS, PING, SSH
	Comment	Management side of the VDOM link

4. Click *OK*.

Configure inter-VDOM routing

A default static route can be configured on each VDOM to provide Internet access. In other words, this static route would provide inter-VDOM routing between each department VDOM and the root VDOM.

For this static route, these settings are used:

- Default Gateway: IP address of the management side of the VDOM link
  - Accounting VDOM: 11.11.11.1
  - Sales VDOM: 12.12.12.1
- Interface: Interface on the department VDOM side of the VDOM link
  - Accounting VDOM: AccountVlnk0
  - Sales VDOM: SalesVlnk0
- IP address: 0.0.0.0/0.0.0.0 (default)

To configure the default static route to the Internet in the Accounting VDOM:

1. In the *Accounting* VDOM, go to *Network > Static Routes*.
2. Click on *Create New* and select the version you need.
3. Enter the following information:

Destination	Subnet
IP address	0.0.0.0/0.0.0.0
Gateway	11.11.11.1
Interface	AccountVlink0
Administrative Distance	10

4. Click *OK*.

To configure the default static route to the Internet in the Sales VDOM:

1. In the *Sales* VDOM, go to *Network > Static Routes*.
2. Click on *Create New* and select the version you need.
3. Enter the following information:

Destination	Subnet
IP address	0.0.0.0/0.0.0.0
Gateway	12.12.12.1
Interface	SalesVlink0
Administrative Distance	10

4. Click *OK*.

Configure the firewall policies

With the VDOMs, physical interfaces, VDOM links, and static routes configured, the firewall must now be configured to allow the proper traffic. Firewalls are configured per-VDOM, and firewall objects and routes must be created for each VDOM separately.

To configure the firewall policies from AccountingLocal to Internet in the GUI:

1. In the *Accounting* VDOM, go to *Policy & Objects > Firewall Policy*.
2. Click *Create New*.
3. Enter the following information:

Name	Account-Local-to-Management
Incoming Interface	port2
Outgoing Interface	AccountVlnk0
Source	All
Destination	All
Schedule	always
Service	ALL
Action	ACCEPT
NAT	enabled

4. Click *OK*.
5. In the *root* VDOM, go to *Policy & Objects > Firewall Policy*.
6. Click *Create New*.
7. Enter the following information:

Name	Account-VDOM-to-Internet
Incoming Interface	AccountVlnk1
Outgoing Interface	port1
Source	All
Destination	All
Schedule	always
Service	ALL
Action	ACCEPT
NAT	enabled

8. Click *OK*.

To configure the firewall policies from SalesLocal to Internet in the GUI:

1. In the *Sales* VDOM, go to *Policy & Objects > Firewall Policy*.
2. Click *Create New*.
3. Enter the following information:

Name	Sales-Local-to-Management
Incoming Interface	port3
Outgoing Interface	SalesVlnk0
Source	All
Destination	All
Schedule	always
Service	ALL
Action	ACCEPT
NAT	enabled

4. Click *OK*.
5. In the *root* VDOM, go to *Policy & Objects > Firewall Policy*.
6. Click *Create New*.
7. Enter the following information:

Name	Sales-VDOM-to-Internet
Incoming Interface	SalesVlnk1
Outgoing Interface	port1
Source	All
Destination	All
Schedule	always
Service	ALL
Action	ACCEPT
NAT	enabled

8. Click *OK*.

Test the configuration

When the inter-VDOM routing has been configured, test the configuration to confirm proper operation. Testing connectivity ensures that physical networking connections, FortiGate unit interface configurations, and firewall policies are properly configured.

The easiest way to test connectivity is to use the `ping` and `traceroute` commands on hosts in the Accounting and Sales networks, respectively, to confirm the connectivity of different routes on the network. Test connectivity with hosts connected to port2 (AccountingLocal) in the Accounting VDOM to the internet and hosts connected to port3 (SalesLocal) in the Sales VDOM to the internet.

Configuration with the CLI

The example can also be configured in the CLI.

To configure inter-VDOM routing in the CLI:

1. Enable multi VDOM mode:

```
config system global
    set vdom-mode multi-vdom
end
```

   You will be logged out of the device when VDOM mode is enabled.

2. Create the Sales and Accounting VDOMs:

```
config vdom
    edit Accounting
    next
    edit Sales
    next
end
```

3. Assign interfaces to the VDOMs:

```
config global
    config system interface
        edit port2
            set vdom Accounting
        next
        edit port3
            set vdom Sales
        next
        edit port1
            set vdom root
        next
    end
end
```

4. Configure the Accounting and management VDOM link:

```
config global
    config system vdom-link
        edit AccountVlnk
        next
    end
    config system interface
        edit AccountVlnk0
            set vdom Accounting
            set ip 11.11.11.2 255.255.255.252
            set allowaccess https ping ssh
            set description "Accounting side of the VDOM link"
        next
        edit AccountVlnk1
```

```
            set vdom root
            set ip 11.11.11.1 255.255.255.252
            set allowaccess https ping ssh
            set description "Management side of the VDOM link"
        next
    end
end
```

5. Configure the Sales and management VDOM link:

```
config global
    config system vdom-link
        edit SalesVlnk
        next
    end
    config system interface
        edit SalesVlnk0
            set vdom Sales
            set ip 12.12.12.2 255.255.255.252
            set allowaccess https ping ssh
            set description "Sales side of the VDOM link"
        next
        edit SalesVlnk1
            set vdom root
            set ip 12.12.12.1 255.255.255.252
            set allowaccess https ping ssh
            set description "Management side of the VDOM link"
        next
    end
end
```

6. Configure the default static route to the Internet in the Accounting VDOM:

```
config vdom
    edit Accounting
    config router static
        edit 1
            set gateway 11.11.11.1
            set device "AccountVlnk0"
        next
    end
end
```

7. Configure the default statis route to the Internet in the Sales VDOM:

```
config vdom
    edit Sales
    config router static
        edit 1
            set gateway 12.12.12.1
            set device "SalesVlnk0"
        next
    end
end
```

8. Configure the firewall policies from AccountingLocal to the Internet:

```
config vdom
    edit Accounting
```

```
            config firewall policy
                edit 1
                    set name "Accounting-Local-to-Management"
                    set srcintf port2
                    set dstintf AccountVlnk0
                    set srcaddr all
                    set dstaddr all
                    set action accept
                    set schedule always
                    set service ALL
                    set nat enable
                next
            end
        next
        edit root
            config firewall policy
                edit 2
                    set name "Accounting-VDOM-to-Internet"
                    set srcintf AccountVlnk1
                    set dstintf port1
                    set srcaddr all
                    set dstaddr all
                    set action accept
                    set schedule always
                    set service ALL
                    set nat enable
                next
            end
        next
    end
```

9. Configure the firewall policies from SalesLocal to the Internet:

```
config vdom
    edit Sales
        config firewall policy
            edit 3
                set name "Sales-local-to-Management"
                set srcintf port3
                set dstintf SalesVlnk0
                set srcaddr all
                set dstaddr all
                set action accept
                set schedule always
                set service ALL
                set nat enable
            next
        end
    next
    edit root
        config firewall policy
            edit 4
                set name "Sales-VDOM-to-Internet"
                set srcintf SalesVlnk1
                set dstintf port1
                set srcaddr all
                set dstaddr all
```

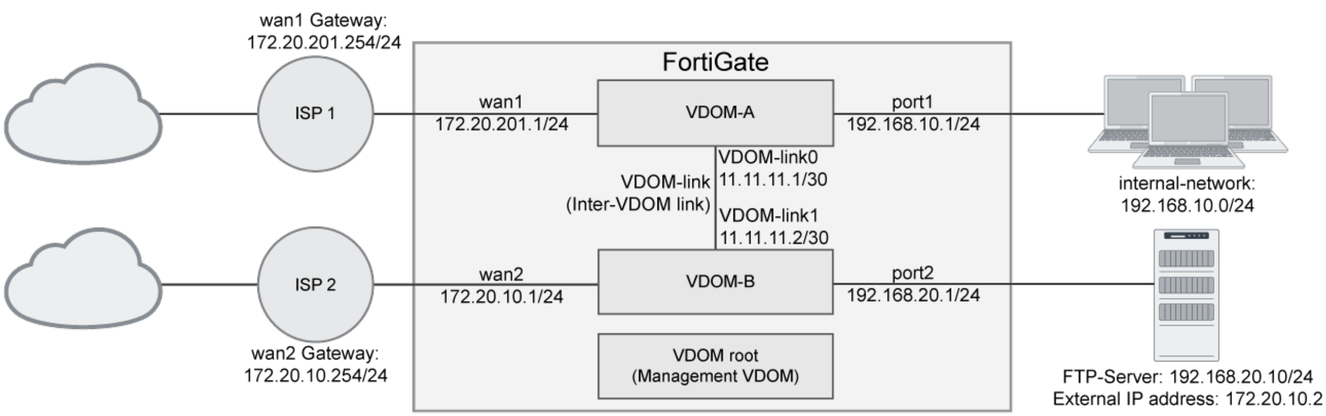
```
                    set action accept
                    set schedule always
                    set service ALL
                    set nat enable
                next
            end
        next
    end
```

Inter-VDOM routing configuration example: Partial-mesh VDOMs

This example shows how to configure a FortiGate unit to use inter-VDOM routing to route traffic between an internal network and FTP server that are each behind separate VDOMs. See Inter-VDOM routing on page 2005 for more information.

The following example shows how to configure per-VDOM settings, such as operation mode, routing, and firewall policies, in a network that includes the following VDOMs:

- VDOM-A: allows the internal network to access the Internet.
- VDOM-B: allows external connections to an FTP server.
- root: the management VDOM.



You can use VDOMs in either NAT or transparent mode on the same FortiGate. By default, VDOMs operate in NAT mode. In this example, both VDOM-A and VDOM-B use NAT mode. An inter-VDOM link is created and inter-VDOM routes configured to allow users on the internal network to access the FTP server.

This is an example of the partial-mesh VDOMs configuration since only VDOM-A is connected to VDOM-B but neither of those VDOMs are connected to the root VDOM. See Topologies on page 1997 for details.

This example assumes that the interfaces of the FortiGate have already been configured with the IP addresses depicted in the preceding diagram.

General steps for this example

This configuration requires the following general steps:

1. Enable Multi VDOM mode and create the VDOMs on page 2018
2. Assign interfaces to VDOMs on page 2018
3. Configure VDOM-A on page 2019

4. Configure VDOM-B on page 2020
5. Configure the VDOM link on page 2022
6. Configure inter-VDOM routing on page 2022
7. Configure firewall policies using the VDOM link on page 2024

This example demonstrates how to configure these steps first using the GUI and then, at the end of the section, using the CLI. See Configuration with the CLI on page 2025 for details.

Enable Multi VDOM mode and create the VDOMs

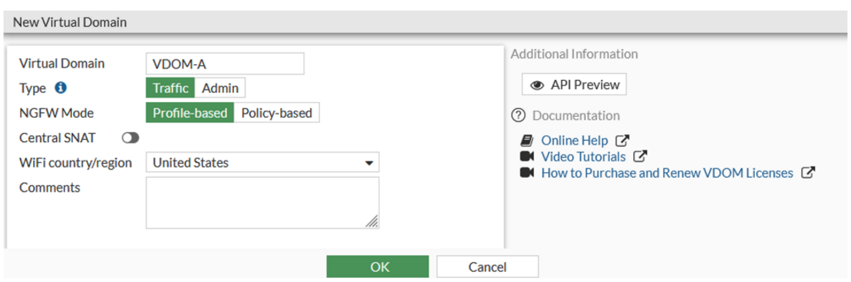
Multi VDOM mode can be enabled in the GUI or CLI. Enabling it does not require a reboot, but does log you out of the device. The current configuration is assigned to the root VDOM.

On FortiGate 90 series models and lower, VDOMs can only be enabled using the CLI.

To enable multi VDOM mode in the GUI:

1. On the FortiGate, go to *System* > *Settings*.
2. In the *System Operation Settings* section, enable *Virtual Domains*.
3. Click *OK*.

To create the VDOMs in the GUI:

1. In the *Global* VDOM, go to *System* > *VDOM*.
   Click *Create New*.
2. In the *Virtual Domain* field, enter *VDOM-A*.
   

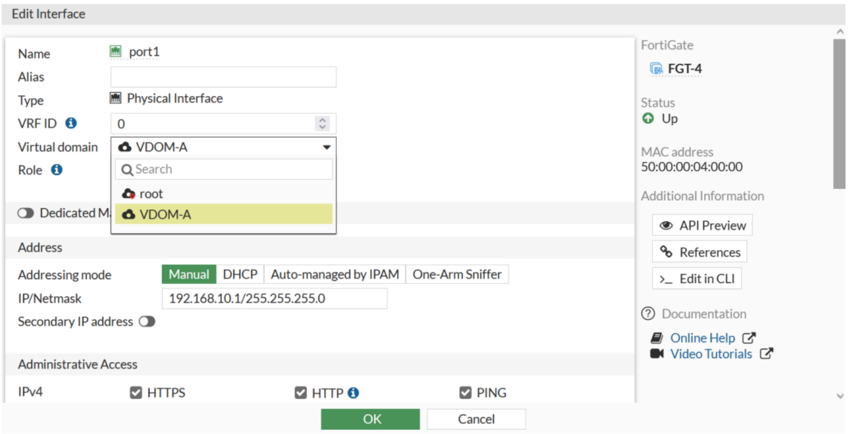
3. If required, set the *NGFW Mode*. If the *NGFW Mode* is *Profile-based*, *Central SNAT* can be enabled.
4. Click *OK* to create the VDOM.
5. Repeat the above steps for *VDOM-B*.

Assign interfaces to VDOMs

This example uses three interfaces on the FortiGate unit: port1 (internal network), port2 (FTP server), wan1 (WAN link for VDOM-A), and wan2 (WAN link for VDOM-B). The port1 and port2 interfaces are connected to the internal network and FTP server, respectively. The wan1 and wan2 interfaces are static assigned with IP addresses and default gateways provided by the ISPs for those WAN links.

To assign interfaces to VDOMs in the GUI:

1. In the *Global* VDOM, go to *Network > Interfaces*.
2. Select *port1* and click *Edit*.
3. From the *Virtual domain* list, select *VDOM-A*.



4. Click *OK*.
5. Repeat the preceding steps to assign *port2* to *VDOM-B*.
6. Repeat the preceding steps to assign *wan1* to *VDOM-A*.
7. Repeat the preceding steps to assign *wan2* to *VDOM-B*.

Configure VDOM-A

VDOM-A allows connections from devices on the internal network to the Internet. WAN1 and port1 are assigned to this VDOM.

The per-VDOM configuration for VDOM-A includes the following:

- A firewall address for the internal network
- A static route to the ISP gateway
- A firewall policy allowing the internal network to access the Internet

All procedures in this section require you to connect to VDOM-A, either using a global or per-VDOM administrator account.

To add the firewall addresses in the GUI:

1. Go to *Policy & Objects > Addresses*.
2. Click *Create New > Address*.
3. Enter the following information:

Name	internal-network
Type	Subnet
IP/Netmask	192.168.10.0/255.255.255.0
Interface	port1

4. Click *OK*.

To add a default route in the GUI:

1. Go to *Network > Static Routes* and create a new route.
2. Enter the following information:

Destination	Subnet
IP address	0.0.0.0/0.0.0.0
Gateway	172.20.201.254
Interface	wan1
Administrative Distance	10

3. Click *OK*.

To add the firewall policy in the GUI:

1. Go to *Policy & Objects > Firewall Policy*.
2. Click *Create New*.
3. Enter the following information:

Name	VDOM-A-Internet
Incoming Interface	port1
Outgoing Interface	wan1
Source	internal-network
Destination	all
Schedule	always
Service	ALL
Action	ACCEPT
NAT	enabled

4. Click *OK*.

Configure VDOM-B

VDOM-B allows external connections to reach an internal FTP server. WAN2 and port2 are assigned to this VDOM.

The per-VDOM configuration for VDOM-B includes the following:

- A firewall address for the FTP server
- A virtual IP address for the FTP server
- A static route to the ISP gateway
- A firewall policy allowing external traffic to reach the FTP server

The procedures described above require you to connect to VDOM-B, either using a global or per-VDOM administrator account.

To add the firewall addresses in the GUI:

1. Go to *Policy & Objects* > *Addresses*.
2. Click *Create New* > *Address*.
3. Enter the following information:

Name	FTP-server
Type	Subnet
IP/Netmask	192.168.10.0/255.255.255.255
Interface	port2

4. Click *OK*.

To add the virtual IP address in the GUI:

1. Go to *Policy & Objects* > *Virtual IPs*.
2. Click *Create New* > *Virtual IP*.
3. Enter the following information:

Name	FTP-server-VIP
Interface	wan2
External IP address/range	172.20.10.2
Map To	192.168.20.10

4. Click *OK*.

To add a default route in the GUI:

1. Go to *Network* > *Static Routes* and create a new route.
2. Enter the following information:

Destination	Subnet
IP address	0.0.0.0/0.0.0.0
Gateway	172.20.201.254
Interface	wan2
Administrative Distance	10

3. Click *OK*.

To add the firewall policy in the GUI:

1. Go to *Policy & Objects* > *Firewall Policy*.
2. Click *Create New*.

3. Enter the following information:

Name	Access-server
Incoming Interface	wan2
Outgoing Interface	port2
Source	all
Destination	FTP-server-VIP
Schedule	always
Service	FTP
Action	ACCEPT
NAT	enabled

4. Click *OK*.

Configure the VDOM link

The VDOM link allows connections from VDOM-A to VDOM-B. The VDOM link interface configured in this step will be used for inter-VDOM routing.

This step requires you to connect to the global VDOM using a global administrator account.

To add the VDOM link in the GUI:

1. In the *Global* VDOM, go to *Network > Interfaces*.
2. Create *New > VDOM link*.
3. Enter the following information:

Name		VDOM-link
Interface 0		
	Virtual Domain	VDOM-A
	IP/Netmask	11.11.11.1/255.255.255.252
Interface 1		
	Virtual Domain	VDOM-B
	IP/Netmask	11.11.11.2/255.255.255.252

4. Click *OK*.

Configure inter-VDOM routing

Inter-VDOM routing allows users on the internal network to route traffic to the FTP server through the FortiGate.

The configuration of inter-VDOM routing includes the following:

- Firewall addresses for the FTP server on VDOM-A and for the internal network on VDOM-B
- Inter-VDOM routing using static routes for the FTP server on VDOM-A and for the internal network on VDOM-B

- Policies allowing traffic using the VDOM link

The procedures described above require you to connect to both VDOM-A and VDOM-B, either using a global or per-VDOM administrator account.

To add the firewall address on VDOM-A in the GUI:

1. In the *VDOM-A* VDOM, go to *Policy & Objects > Addresses*.
2. Click *Create New > Address*.
3. Enter the following information:

Name	FTP-server
Type	Subnet
IP/Netmask	192.168.20.10/32
Interface	VDOM-link2
Static route configuration	enabled

4. Click *OK*.

To add the static route on VDOM-A in the GUI:

1. Connect to *VDOM-A*.
2. Go to *Network > Static Routes* and create a new route.
3. Enter the following information:

Destination	Named Address
Named Address	FTP-server
Gateway	11.11.11.2
Interface	VDOM-link0

4. Click *OK*.

To add the firewall address on VDOM-B in the GUI:

1. In the *VDOM-B* VDOM, go to *Policy & Objects > Addresses*.
2. Click *Create New > Address*.
3. Enter the following information:

Name	internal-network
Type	Subnet
IP/Netmask	192.168.10.0/24
Interface	VDOM-link1
Static route configuration	enabled

4. Click *OK*.

To add the static route on VDOM-B in the GUI:

1. In the *VDOM-B* VDOM, go to *Network > Static Routes* and create a new route.
2. Enter the following information:

Destination	Named Address
Named Address	internal-network
Gateway	11.11.11.1
Interface	VDOM-link1

3. Click *OK*.

Configure firewall policies using the VDOM link

Firewall policies using the VDOM link allows users on the internal network to access the FTP server through the FortiGate.

Configuring policies allowing traffic using the VDOM link require you to connect to both VDOM-A and VDOM-B, respectively, either using a global or per-VDOM administrator account.

To add the firewall policy on VDOM-A in the GUI:

1. In the *VDOM-A* VDOM, go to *Policy & Objects > Firewall Policy*.
2. Click *Create New*.
3. Enter the following information:

Name	Access-FTP-server
Incoming Interface	port1
Outgoing Interface	VDOM-link0
Source	internal-network
Destination	FTP-server
Schedule	always
Service	FTP
Action	ACCEPT
NAT	disabled

4. Click *OK*.

To add the firewall policy on VDOM-B in the GUI:

1. In the *VDOM-B* VDOM, go to *Policy & Objects > Firewall Policy*.
2. Click *Create New*.

3. Enter the following information:

Name	Internal-server-access
Incoming Interface	VDOM-link1
Outgoing Interface	port2
Source	internal-network
Destination	FTP-server
Schedule	always
Service	FTP
Action	ACCEPT
NAT	disabled

4. Click *OK*.

Configuration with the CLI

The example can also be configured in the CLI.

To configure the two VDOMs:

1. Enable multi VDOM mode:

```
config system global
    set vdom-mode multi-vdom
end
```

You will be logged out of the device when VDOM mode is enabled.

2. Create the VDOMs:

```
config vdom
    edit VDOM-A
    next
    edit VDOM-B
    next
end
```

3. Assign interfaces to the VDOMs:

```
config global
    config system interface
        edit port1
            set vdom VDOM-A
        next
        edit port2
            set vdom VDOM-B
        next
        edit wan1
            set vdom VDOM-A
        next
        edit wan2
```

```
            set vdom VDOM-B
        next
    end
end
```

4. Add the firewall addresses to VDOM-A:

```
config vdom
    edit VDOM-A
        config firewall address
            edit internal-network
                set associated-interface port1
                set subnet 192.168.10.0 255.255.255.0
            next
        end
    next
end
```

5. Add a default route to VDOM-A:

```
config vdom
    edit VDOM-A
        config router static
            edit 0
                set gateway 172.20.201.254
                set device wan1
            next
        end
    next
end
```

6. Add the firewall policy to VDOM-A:

```
config vdom
    edit VDOM-A
        config firewall policy
            edit 1
                set name "VDOM-A-Internet"
                set srcintf "port1"
                set dstintf "wan1"
                set srcaddr "internal-network"
                set dstaddr "all"
                set action accept
                set schedule "always"
                set service "ALL"
                set nat enable
            next
        end
    next
end
```

7. Add the firewall addresses to VDOM-B:

```
config vdom
    edit VDOM-B
        config firewall address
            edit FTP-server
                set associated-interface port2
                set subnet 192.168.20.10 255.255.255.255
```

```
            next
        end
    next
end
```

8. Add the virtual IP address to VDOM-B:

```
config vdom
    edit VDOM-B
        config firewall vip
            edit FTP-server-VIP
                set extip 172.20.10.2
                set extintf wan2
                set mappedip 192.168.20.10
            next
        end
    next
end
```

9. Add a default route to VDOM-B:

```
config vdom
    edit VDOM-B
        config router static
            edit 0
                set gateway 172.20.10.254
                set device wan2
            next
        end
    next
end
```

10. Add the firewall policy to VDOM-B:

```
config vdom
    edit VDOM-B
        config firewall policy
            edit 1
                set name "Access-server"
                set srcintf "wan2"
                set dstintf "port2"
                set srcaddr "all"
                set dstaddr "FTP-server-VIP"
                set action accept
                set schedule "always"
                set service "FTP"
                set nat enable
            next
        end
    next
end
```

To configure the VDOM link:

1. Configure the VDOM link:

```
config global
    config system vdom-link
```

```
            edit "VDOM-link"
            next
        end
        config system interface
            edit VDOM-link0
                set vdom VDOM-A
                set ip 11.11.11.1 255.255.255.252
                set allowaccess https ping ssh
                set description "VDOM-A side of the VDOM link"
            next
            edit VDOM-link1
                set vdom VDOM-B
                set ip 11.11.11.2 255.255.255.252
                set allowaccess https ping ssh
                set description "VDOM-A side of the VDOM link"
            next
        end
    end
```

2. Configure the firewall addresses on VDOM-A:

```
config vdom
    edit VDOM-A
        config firewall address
            edit "FTP-server"
                set associated-interface "VDOM-link0"
                set allow-routing enable
                set subnet 192.168.20.10 255.255.255.255
            next
        end
    next
end
```

3. Add the firewall policy to VDOM-B:

```
config vdom
    edit VDOM-B
        config firewall policy
            edit 1
                set name "Access-server"
                set srcintf "wan2"
                set dstintf "port2"
                set srcaddr "all"
                set dstaddr "FTP-server-VIP"
                set action accept
                set schedule "always"
                set service "FTP"
                set nat enable
            next
        end
    next
end
```

4. Add the static route on VDOM-A:

```
config vdom
    edit VDOM-A
        config router static
```

```
            edit 0
                set device VDOM-link0
                set dstaddr FTP-server
                set gateway 11.11.11.2
            next
        end
    next
end
```

5. Configure the firewall addresses on VDOM-B:

```
config vdom
    edit VDOM-B
        config firewall address
            edit internal-network
                set associated-interface VDOM-link1
                set allow-routing enable
                set subnet 192.168.10.0 255.255.255.0
            next
        end
    next
end
```

6. Add the static route on VDOM-B:

```
config vdom
    edit VDOM-B
        config router static
            edit 0
                set device VDOM-link1
                set dstaddr internal-network
                set gateway 11.11.11.1
            next
        end
    next
end
```

7. Add the security policy on VDOM-A:

```
config vdom
    edit VDOM-A
        config firewall policy
            edit 0
                set name Access-FTP-server
                set srcintf port1
                set dstintf VDOM-link0
                set srcaddr internal-network
                set dstaddr FTP-server
                set action accept
                set schedule always
                set service FTP
            next
        end
    next
end
```

8. Add the firewall policy on VDOM-B:

```
config vdom
    edit VDOM-B
        config firewall policy
            edit 0
                set name Internal-server-access
                set srcintf VDOM-link1
                set dstintf port2
                set srcaddr internal-network
                set dstaddr FTP-server
                set action accept
                set schedule always
                set service FTP
            next
        end
    next
end
```

High Availability

Whether your FortiGate is used as a security gateway, an internal segmentation firewall, in the cloud, or in an MSSP environment, as long as there is critical traffic passing through it, there is risk of it being a single point of failure. Physical outages can occur due to power failures, physical link failures, transceiver failures, or power supply failures. Non-physical outages can be caused by routing, resource issues, or kernel panic.

Network outages cause disruptions to business operations, downtime, and frustration for users and in some situations may have financial setbacks. In designing your network and architecture, it is important to weigh the risks and consequences associated with unexpected outages.

There are many ways to build redundancy and resiliency. In a switching network, you can accomplish this by adding redundant links and switches in partial or full mesh topologies. Using redundant and aggregate links, you can avoid a single link failure causing a network to go down. Using SD-WAN, you can build redundant and intelligent WAN load balancing and failover architectures.

FortiGate HA offers several solutions for adding redundancy in the case where a failure occurs on the FortiGate, or is detected by the FortiGate through monitored links, routes, and other health checks. These solutions support fast failover to avoid lengthy network outages and disruptions to your traffic.

FortiGate Clustering Protocol (FGCP)

FGCP provides a solution for two key requirements of critical enterprise networking components: enhanced reliability and increased performance. Enhanced reliability is achieved through device failover protection, link failover protection, and remote link failover protection. Session failover protection for most IPv4 and IPv6 sessions also contributes to enhanced reliability. Increased performance is achieved though active-active HA load balancing.

FortiGate Session Life Support Protocol (FGSP)

In a network that already includes load balancing (either with load balancers or routers) for traffic redundancy, two entities (either standalone FortiGates or FGCP clusters) can be integrated into the load balancing configuration using the FortiGate Session Life Support Protocol (FGSP). The external load balancers or routers can distribute sessions among the FortiGates and the FGSP performs session synchronization of IPv4 and IPv6 TCP, SCTP, UDP, ICMP, expectation, and NAT sessions to keep the session tables of both entities synchronized. In the event of a failure, the load balancer can detect the failed unit and failover the sessions to other active members to continue processing the traffic.

VRRP

FortiGates can function as primary or backup Virtual Router Redundancy Protocol (VRRP) routers. The FortiGates can quickly and easily integrate into a network that has already deployed VRRP. A FortiGate can be integrated into a VRRP group with any third-party VRRP devices, and VRRP can provide redundancy between multiple FortiGates. FortiOS supports VRRP version 2 and 3.

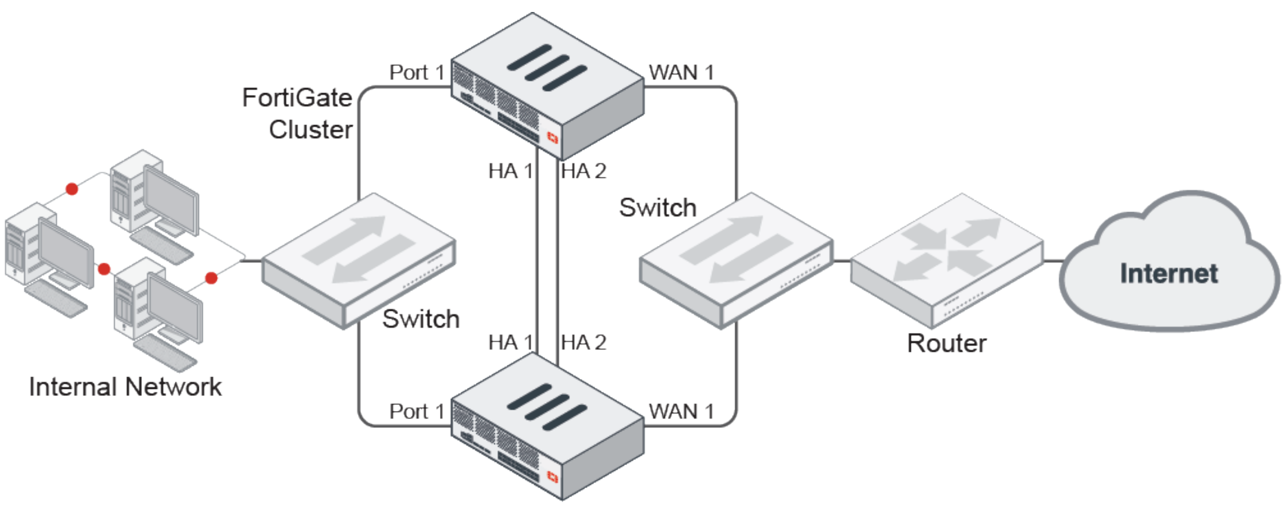
The following topics provide more information about each HA solution and other HA related topics:

- FGCP on page 2031
- FGSP on page 2086
- Standalone configuration synchronization on page 2107
- VRRP on page 2112

FGCP

High availability (HA) is usually required in a system where there is high demand for little downtime. There are usually hot-swaps, backup routes, or standby backup units and as soon as the active entity fails, backup entities will start functioning. This results in minimal interruption for the users.

The FortiGate Clustering Protocol (FGCP) is a proprietary HA solution whereby FortiGates can find other member FortiGates to negotiate and create a cluster. A FortiGate HA cluster consists of at least two FortiGates (members) configured for HA operation. All FortiGates in the cluster must be the same model and have the same firmware installed. Cluster members must also have the same hardware configuration (such as the same number of hard disks). All cluster members share the same configurations except for their host name and priority in the HA settings. The cluster works like a device but always has a hot backup device.



Critical cluster components

The following are critical components in an HA cluster:

- Heartbeat connections: members will use this to communicate with each other. In general, a two-member cluster is most common. We recommend double back-to-back heartbeat connections.
- Identical connections for internal and external interfaces: as demonstrated in the topology, we recommend similar connections from each member to the switches for the cluster to function properly.

General operation

The following are best practices for general cluster operation:

- Ensure that heartbeat communication is present.
- Enable the session synchronization option in daily operation (see FGSP basic peer setup on page 2088).
- Monitor traffic flowing in and out of the interfaces.

Failover

FGCP provides failover protection in the following scenarios:

- The active device loses power.
- A monitored interface loses a connection.

After failover occurs, the user will not notice any difference, except that the active device has changed. See Failover protection on page 2033 for more information.

Synchronizing the configuration

FGCP uses a combination of incremental and periodic synchronization to make sure that the configuration of all cluster units is synchronized to that of the primary unit.

The following settings are not synchronized between cluster units:

- The FortiGate host name
- GUI Dashboard widgets
- HA override
- HA device priority
- The virtual cluster priority
- The HA priority setting for a ping server (or dead gateway detection) configuration
- The system interface settings of the HA reserved management interface
- The HA default route for the reserved management interface, set using the `ha-mgmt-interface-gateway` option of the `config system ha` command

Most subscriptions and licenses are not synchronized, as each FortiGate must be licensed individually. FortiToken Mobile is an exception; they are registered to the primary unit and synchronized to the secondary units.

The primary unit synchronizes all other configuration settings, including the other HA configuration settings.

All synchronization activity takes place over the HA heartbeat link using TCP/703 and UDP/703 packets.

The following topics provide more information about FGCP:

- Failover protection on page 2033
- HA active-passive cluster setup on page 2035
- HA active-active cluster setup on page 2037
- HA virtual cluster setup on page 2039
- Check HA synchronization status on page 2044

Failover protection

The FortiGate Clustering Protocol (FGCP) provides failover protection, meaning that a cluster can provide FortiGate services even when one of the devices in the cluster encounters a problem that would result in the complete loss of connectivity for a stand-alone FortiGate unit. Failover protection provides a backup mechanism that can be used to reduce the risk of unexpected downtime, especially in mission-critical environments.

FGCP supports failover protection in four ways:

1. If a link fails.
2. If a device loses power.
3. If an SSD fails.
4. If memory utilization exceeds the threshold for a specified amount of time.

When session-pickup is enabled in the HA settings, existing TCP session are kept, and users on the network are not impacted by downtime as the traffic can be passed without reestablishing the sessions.

When and how the failover happens

1. Link fails

Before triggering a failover when a link fails, the administrator must ensure that monitor interfaces are configured. Normally, the internal interface that connects to the internal network, and an outgoing interface for traffic to the internet or outside the network, should be monitored. Any of those links going down will trigger a failover.

2. Loss of power for active unit

When an active (primary) unit loses power, a backup (secondary) unit automatically becomes the active, and the impact on traffic is minimal. There are no settings for this kind of fail over.

3. SSD failure

An HA failover can be triggered by an SSD failure.

To enable an SSD failure triggering HA fail over:

```
config system ha
    set ssd-failover enable
end
```

4. Memory utilization

An HA failover can be triggered when memory utilization exceeds the threshold for a specific amount of time.

Memory utilization is checked at the configured sample rate (`memory-failover-sample-rate`). If the utilization is above the threshold (`memory-failover-threshold`) every time that it is sampled for the entire monitor period (`memory-failover-monitor-period`), then a failover is triggered.

If the FortiGate meets the memory utilization conditions to cause failover, but the last memory triggered failover happened within the timeout period (`memory-failover-flip-timeout`), then the failover does not occur. Other HA cluster members can still trigger memory based failovers if they meet the criteria and have not already failed within the timeout period.

After a memory based failover from FortiGate A to FortiGate B, if the memory usage on FortiGate A goes down below the threshold but the memory usage on FortiGate B is still below the threshold, then a failover is not triggered, as the cluster is working normally using FortiGate B as the primary device.

When you disable memory based failover, a new HA primary selection occurs to determine the primary device.

To configure memory based HA failover:

```
config system ha
    set memory-based-failover {enable | disable}
    set memory-failover-threshold <integer>
    set memory-failover-monitor-period <integer>
    set memory-failover-sample-rate <integer>
    set memory-failover-flip-timeout <integer>
end
```

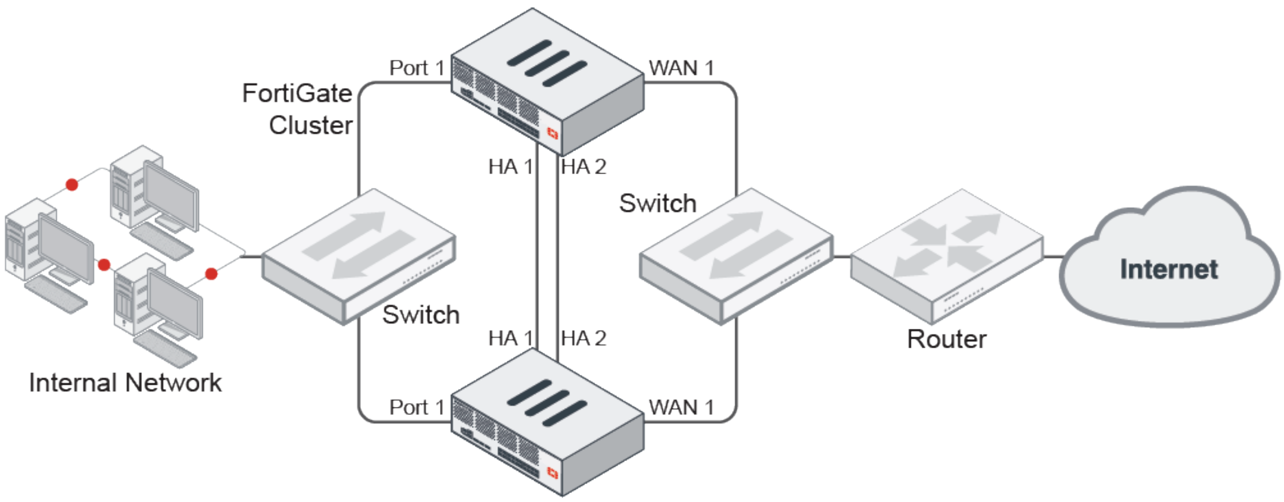
`memory-based-failover {enable | disable}`	Enable/disable memory based failover (default = disable).
`memory-failover-threshold <integer>`	The memory usage threshold to trigger a memory based failover, in percentage (0 - 95, 0 = use the conserve mode threshold, default = 0).
`memory-failover-monitor-period <integer>`	The duration of the high memory usage before a memory based failover is triggered, in seconds (1 - 300, default = 60).
`memory-failover-sample-rate <integer>`	The rate at which memory usage is sampled in order to measure memory usage, in seconds (1 - 60, default = 1).
`memory-failover-flip-timeout <integer>`	The time to wait between subsequent memory based failovers, in minutes (6 - 2147483647, default = 6).

Configuring HA failover time

On supported models, the HA heartbeat interval unit can be changed from the 100ms default to 10ms. This allows for a failover time of less than 50ms, depending on the configuration and the network.

```
config system ha
    set hb-interval-in-milliseconds {100ms | 10ms}
end
```

In this example, the HA heartbeat interval unit is changed from 100ms to 10ms. As the default heartbeat interval is two, this means that a heartbeat is sent every 20ms. The number of lost heartbeats that signal a failure is also changed to two. So, after two consecutive heartbeats are lost, a failover will be detected in 40ms.



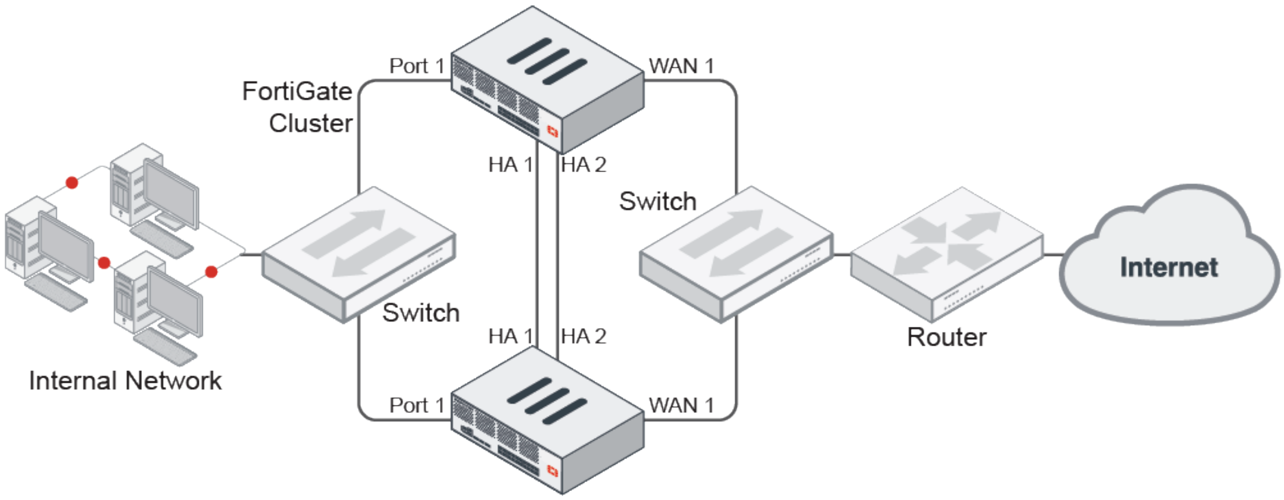
To configure the HA failover time:

```
config system ha
    set group-id 240
    set group-name "300D"
    set mode a-p
    set hbdev "port3" 50 "port5" 100
    set hb-interval 2
    set hb-interval-in-milliseconds 10ms
    set hb-lost-threshold 2
    set override enable
    set priority 200
end
```

HA active-passive cluster setup

An HA Active-Passive (A-P) cluster can be set up using the GUI or CLI.

This example uses the following network topology:

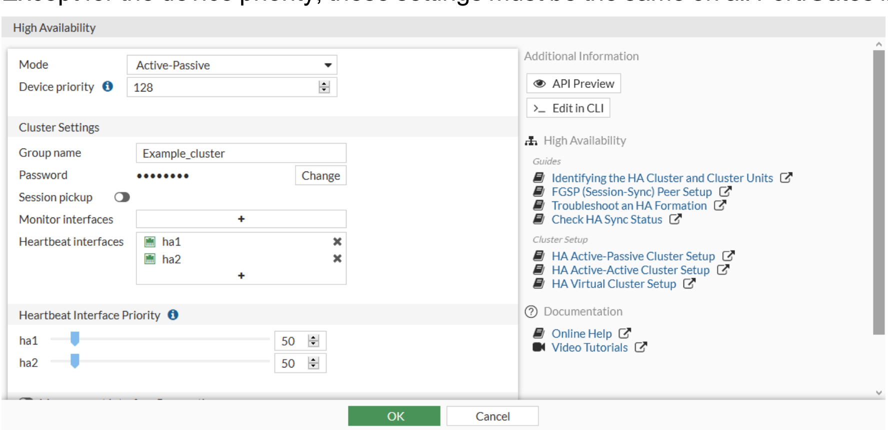


To set up an HA A-P cluster using the GUI:

1. Make all the necessary connections as shown in the topology diagram.
2. Log into one of the FortiGates.
3. Go to *System* > *HA* and set the following options:

Mode	Active-Passive
Device priority	128 or higher
Group name	Example_cluster
Heartbeat interfaces	ha1 and ha2

Except for the device priority, these settings must be the same on all FortiGates in the cluster.



4. Leave the remaining settings as their default values. They can be changed after the cluster is in operation.
5. Click *OK*.
   The FortiGate negotiates to establish an HA cluster. Connectivity with the FortiGate may be temporarily lost as the HA cluster negotiates and the FGCP changes the MAC addresses of the FortiGate's interfaces.
6. Factory reset the other FortiGate that will be in the cluster, configure GUI access, then repeat steps 1 to 5, omitting setting the device priority, to join the cluster.

To set up an HA A-P cluster using the CLI:

1. Make all the necessary connections as shown in the topology diagram.
2. Log into one of the FortiGates.
3. Change the hostname of the FortiGate:

```
config system global
    set hostname Example1_host
end
```

   Changing the host name makes it easier to identify individual cluster units in the cluster operations.
4. Enable HA:

```
config system ha
    set mode a-p
    set group-name Example_cluster
    set hbdev ha1 10 ha2 20
end
```

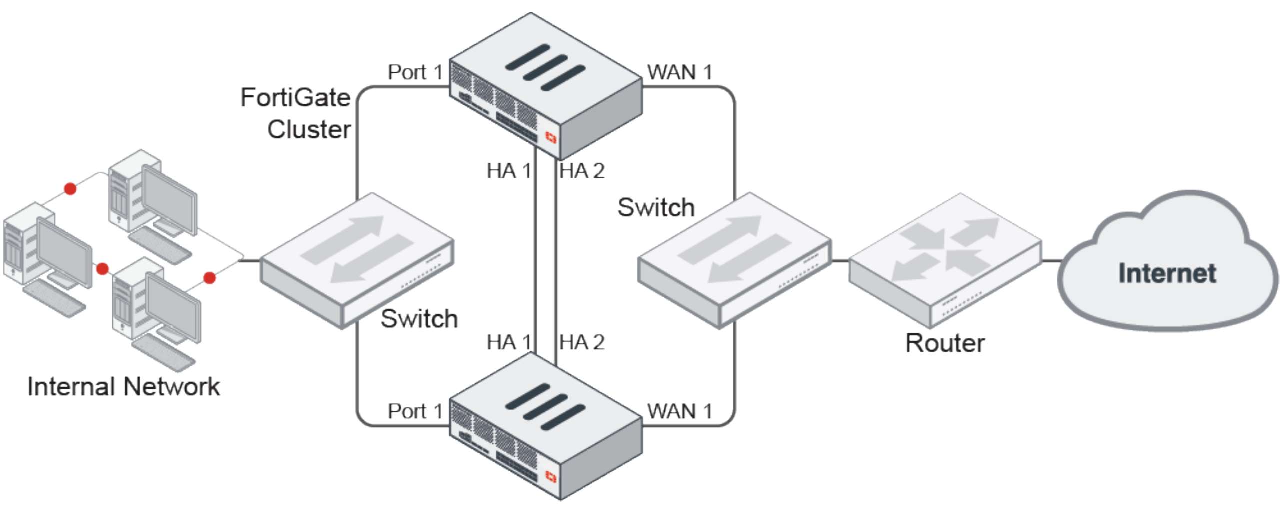
5. Leave the remaining settings as their default values. They can be changed after the cluster is in operation.
6. Repeat steps 1 to 5 on the other FortiGate devices to join the cluster, giving each device a unique hostname.

HA active-active cluster setup

An HA Active-Active (A-A) cluster can be set up using the GUI or CLI.

FGCP in Active-Active mode cannot load balance any sessions that traverse NPU VDOM links or regular VDOM links. If Active-Active session load balancing between VDOMs is required, use an external router to handle the inter-VDOM routing.

This example uses the following network topology:



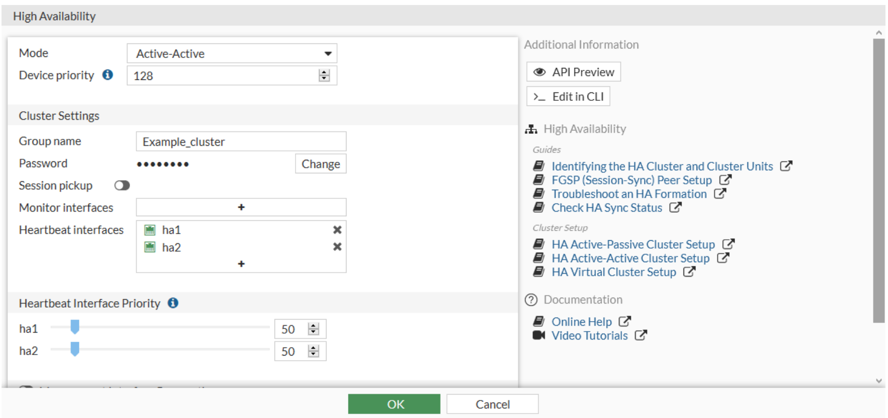
To set up an HA A-A cluster using the GUI:

1. Make all the necessary connections as shown in the topology diagram.
2. Log into one of the FortiGates.

3. Go to *System > HA* and set the following options:

Mode	Active-Active
Device priority	128 or higher
Group name	Example_cluster
Heartbeat interfaces	ha1 and ha2

Except for the device priority, these settings must be the same on all FortiGates in the cluster.



4. Leave the remaining settings as their default values. They can be changed after the cluster is in operation.
5. Click *OK*.
   The FortiGate negotiates to establish an HA cluster. Connectivity with the FortiGate may be temporarily lost as the HA cluster negotiates and the FGCP changes the MAC addresses of the FortiGate's interfaces.
6. Factory reset the other FortiGate that will be in the cluster, configure GUI access, then repeat steps 1 to 5, omitting setting the device priority, to join the cluster.

To set up an HA A-A cluster using the CLI:

1. Make all the necessary connections as shown in the topology diagram.
2. Log into one of the FortiGates.
3. Change the hostname of the FortiGate:

```
config system global
    set hostname Example1_host
end
```

   Changing the host name makes it easier to identify individual cluster units in the cluster operations.
4. Enable HA:

```
config system ha
    set mode a-a
    set group-name Example_cluster
    set hbdev ha1 10 ha2 20
end
```

5. Leave the remaining settings as their default values. They can be changed after the cluster is in operation.
6. Repeat steps 1 to 5 on the other FortiGate devices to join the cluster.

HA virtual cluster setup

Virtual clustering is an extension of FGCP HA that provides failover protection between two instances of one or more VDOMs operating on two FortiGates that are in a virtual cluster. A standard virtual cluster consists of FortiGates that are operating in active-passive HA mode with multiple VDOMs enabled. The following custom settings can be configured per cluster:

```
config system ha
    set vcluster-status enable
    config vcluster
        edit <id>
            set override {enable | disable}
            set priority <integer>
            set vdom <vdom_1>, ... [vdom_n]
            set monitor <interface_1>, ... [interface_n]
            set pingserver-monitor-interface <interface_1>, ... [interface_n]
        next
    end
end
```

`override {enable | disable}`	Enable/disable override and increase the priority of the unit that should always be the primary.
`priority <integer>`	Increase the priority to select the primary unit (0 - 255, default = 128).
`vdom <vdom_1>, ... [vdom_n]`	Set the virtual domains in the virtual cluster.
`monitor <interface_1>, ... [interface_n]`	Set the interfaces to check for port monitoring (or link failure).
`pingserver-monitor-interface <interface_1>, ... [interface_n]`	Set the interfaces to check for remote IP monitoring.

Active-passive virtual clustering uses VDOM partitioning to send traffic for some VDOMs to the primary FortiGate and traffic for other VDOMs to the secondary FortiGates. Traffic distribution between FortiGates can potentially improve throughput. If a failure occurs and only one FortiGate continues to operate, all traffic fails over to that FortiGate, similar to normal HA. If the failed FortiGates rejoin the cluster, the configured traffic distribution is restored.

In an active-passive virtual cluster of two FortiGates, the primary and secondary FortiGates share traffic processing according to the VDOM partitioning configuration. If you add a third or fourth FortiGate, the primary and first secondary FortiGate process all traffic and the other one or two FortiGates operate in standby mode. If the primary or first secondary FortiGate fails, one of the other FortiGates becomes the new primary or secondary FortiGate and begins processing traffic.

Separation of VDOM traffic

Virtual clustering creates a cluster between instances of each VDOM on the two FortiGates in the virtual cluster. All traffic to and from a given VDOM is sent to one of the FortiGates where it stays within its VDOM and is only processed by that VDOM. One FortiGate is the primary FortiGate for each VDOM and one FortiGate is the secondary FortiGate for each VDOM. The primary FortiGate processes all traffic for its VDOMs; the secondary FortiGate processes all traffic for its VDOMs.

Virtual clustering and heartbeat interfaces

The HA heartbeat provides the same HA services in a virtual clustering configuration as in a standard HA configuration. One set of HA heartbeat interfaces provides HA heartbeat services for all of the VDOMs in the cluster. You do not have to add a heartbeat interface for each VDOM.

Special considerations for NPU-based VLANs in a virtual cluster

In an FGCP cluster, the primary FortiGate uses virtual MAC addresses when forwarding traffic, and the secondary uses the physical MAC addresses when forwarding traffic. In a virtual cluster, packets are sent with the cluster's virtual MAC addresses. However, in the case of NPU offloading on a non-root VDOM, traffic that leaves an NPU-based VLAN will use the physical MAC address of its parent interface rather than the virtual MAC address. If this behavior is not desired, disable *auto-asic-offload* in the firewall policy where the VLAN interface is used.

Support up to 30 virtual clusters

FortiOS supports up to 30 virtual clusters, which allows more VDOMs to be spread across different virtual clusters without overlapping. Each virtual cluster supports its own failover conditions. Prior to 7.2.0, only two virtual clusters were supported.

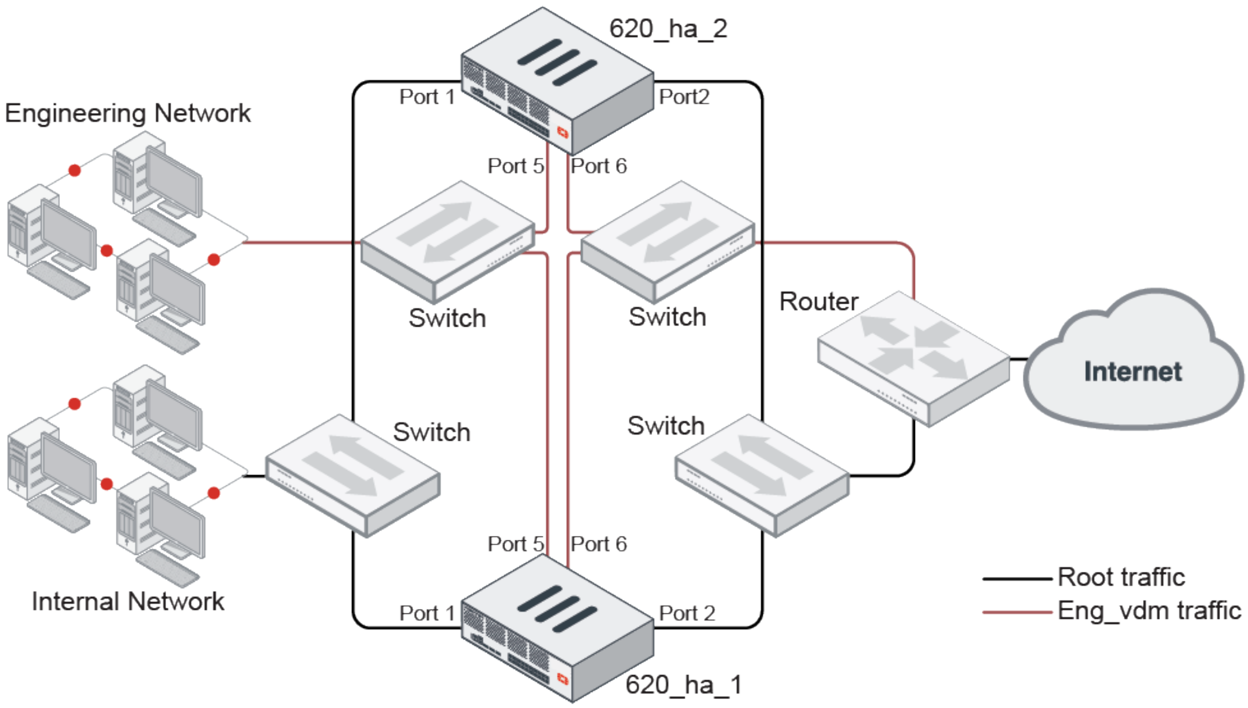
When configuring virtual clusters, the `group-id` is limited to a value from 0 to 7. If the HA `group-id` is greater than 7, use the command line first to change the `group-id` before enabling virtual clusters.

```
config system ha
    set group-id <integer>
end
```

When upgrading from 7.0 or earlier, old virtual clusters will be lost if the `group-id` is larger than 7.

Basic configuration

This example shows a virtual cluster configuration consisting of two FortiGates. The virtual cluster has two VDOMs, root and eng_vdm.



The root VDOM can only be associated with virtual cluster 1.

The VDOM that is assigned as the management VDOM can also only be associated with virtual cluster 1.

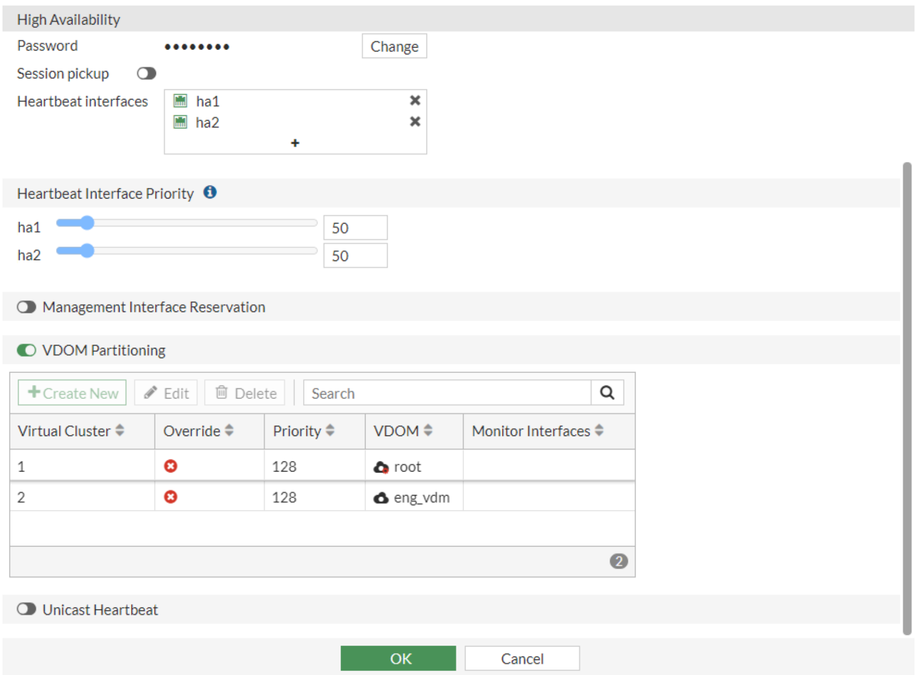
To set up an HA virtual cluster using the GUI:

1. Make all the necessary connections as shown in the topology diagram.
2. Configure a regular A-P cluster:
   a. Log in to one of the FortiGates.
   b. Go to *System* > *HA* and set the following options:

Mode	Active-Passive
Device priority	128 or higher
Group name	Example_cluster
Heartbeat interfaces	ha1 and ha2

   Except for the device priority, these settings must be the same on all FortiGates in the cluster.
   c. Leave the remaining settings as their default values. They can be changed after the cluster is in operation.
   d. Click *OK*.
   The FortiGate negotiates to establish an HA cluster. Connectivity with the FortiGate may be temporarily lost as the HA cluster negotiates and the FGCP changes the MAC addresses of the FortiGate's interfaces.
   e. Factory reset the other FortiGate that will be in the cluster, configure GUI access, then repeat step 2 (omitting setting the device priority) to join the cluster.
3. On the primary FortiGate, go to *System* > *Settings* and enable *Virtual Domains*.
4. Click *Apply*. You will be logged out of the FortiGate.

5. Log back in to the FortiGate, and ensure that you are in the global VDOM.
6. Create the eng_vdm VDOM:
   a. Go to *System > VDOM* and click *Create New*. The *New Virtual Domain* pane opens.
   b. Enter the name in the *Virtual Domain* field, then click *OK*.
7. Implement a virtual cluster by moving the new VDOM to virtual cluster 2:
   a. Go to *System > HA* and enable *VDOM Partitioning*.
   b. In the table, click *Create New*. The *New Virtual Cluster* pane opens.
   c. Click the + and add the *eng_vdm* VDOM.
   d. Click *OK* to save the virtual cluster.



   e. Click *OK* to save the HA configuration.

To set up an HA virtual cluster using the CLI:

1. Make all the necessary connections as shown in the topology diagram.
2. Configure a regular A-P cluster. See HA active-passive cluster setup on page 2035.
3. Enable VDOMs:

```
config system global
    set vdom-mode multi-vdom
end
```

   You will be logged out of the FortiGate.

4. Create the eng_vdm VDOM:

```
config vdom
    edit eng_vdm
    next
end
```

5. Reconfigure the HA settings to be a virtual cluster:

```
config system ha
    set vcluster-status enable
    config vcluster
        edit 1
            set vdom root
            set override disable
        next
        edit 2
            set vdom eng_vdm
            set override disable
        next
    end
end
```

Configuration with 30 virtual clusters

In this example, there are 30 customers managed by an MSSP on an HA cluster, and each customer VDOM needs to failover independently of other customer VDOMs. Each customer is assigned to a different virtual cluster with its own virtual cluster configurations. This may include different monitored interfaces, ping servers, and priority for the primary and secondary cluster members. Each virtual cluster will fail over according to their own virtual cluster configurations.

This example assumes an A-P cluster and VDOMs have already been configured. See HA active-passive cluster setup on page 2035 and Virtual Domains on page 1994 for more information.

For each virtual cluster, this example assumes that unit 1 has an HA priority of 200, while unit 2 has an HA priority of 100. By default, unit 1 will be the primary cluster member of all the virtual clusters.

To configure multiple virtual clusters in the GUI:

1. Go to *System > HA* and enable *VDOM Partitioning*.
2. Create a virtual cluster:
   a. In the table, click *Create New*. The *New Virtual Cluster* pane opens.
   b. Set the *Device priority* to *200*.
   c. Click the + and add the *Virtual domains*.
   d. Optionally, click the + and add the *Monitor interfaces*.
   e. Click *OK*.
3. Repeat step 2 to create the remaining virtual clusters.
4. Click *OK* to save the HA configuration. The *HA* page summary displays the multiple virtual clusters, each with a *Primary* and *Secondary* HA member.
5. Edit the priority settings for the secondary members to be 100:
   a. Select the *Secondary* member in the table, and click *Edit*.
   b. Set the *Priority* to *100*.
   c. Click *OK*.
6. Repeat step 5 for the remaining secondary members.

To configure multiple virtual clusters in the CLI:

1. Configure the primary FortiGate:

```
config system ha
    set vcluster-status enable
    config vcluster
        edit 1
            set override disable
            set priority 200
            set vdom "vdom1"
        next
        edit 2
            set override disable
            set priority 200
            set vdom "vdom2"
        next
        ...
        edit 30
            set override disable
            set priority 200
            set vdom "vdom30"
        next
    end
end
```

2. Configure the secondary FortiGate:

```
config system ha
    set vcluster-status enable
    config vcluster
        edit 1
            set override disable
            set priority 100
            set vdom "vdom1"
        next
        edit 2
            set override disable
            set priority 100
            set vdom "vdom2"
        next
        ...
        edit 30
            set override disable
            set priority 100
            set vdom "vdom30"
        next
    end
end
```

Check HA synchronization status

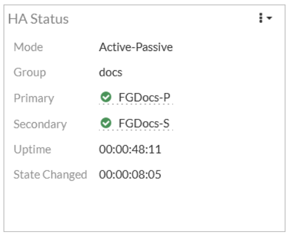
The HA synchronization status can be viewed in the GUI through either a widget on the *Dashboard* or on the *System > HA* page. It can also be confirmed through the CLI. When a cluster is out of synchronization, administrators should correct the issue as soon as possible as it affects the configuration integrity and can cause issues to occur.

When units are out of synchronization in an HA cluster, the GUI will compare the HA checksums and display the tables that caused HA to be out of synchronization. This can be visualized on the HA monitor page and in the HA status widget.

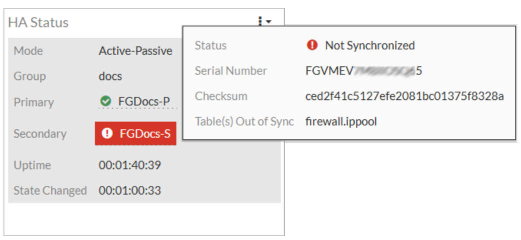
HA synchronization status in the GUI

Following HA setup, the *HA Status* widget can be added to the *Dashboard* that shows the HA synchronization statuses of the members.

A green checkmark is shown next to each member that is in synchronization.

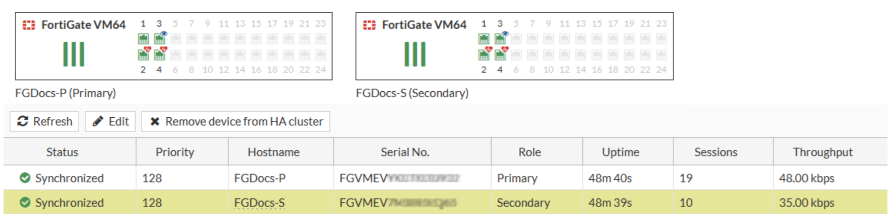


A member that is out of synchronization is highlighted in red. Hover the cursor over the unsynchronized device to see the tables that are out of synchronization and the checksum values.

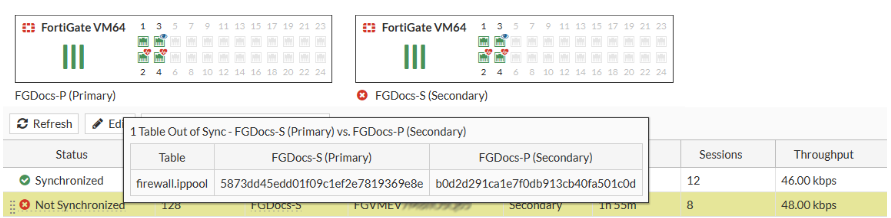


You can also go to *System > HA* to see the synchronization statuses of the members. A member that is out of synchronization will have a red icon next to its name. Hover the cursor over the unsynchronized device to see the tables that are out of synchronization and the checksum values.

Synchronized:



Unsynchronized:



HA synchronization status in the CLI

In the CLI, run the `get system ha status` command to see if the cluster is in synchronization . The synchronization status is reported under *Configuration Status*.

When both members are in synchronization:

```
# get system ha status
HA Health Status: OK
Model: FortiGate-VM64
Mode: HA A-P
Group: 0
Debug: 0
Cluster Uptime: 0 days 0:52:39
Cluster state change time: 2021-04-29 13:17:03
Primary selected using:
    <2021/04/29 13:17:03> FGVMEV0000000002 is selected as the primary because its uptime is
larger than peer member FGVMEV7000000005.
    <2021/04/29 12:37:17> FGVMEV0000000002 is selected as the primary because it's the only
member in the cluster.
ses_pickup: disable
override: disable
Configuration Status:
    FGVMEV0000000002(updated 3 seconds ago): in-sync
    FGVMEV7000000005(updated 2 seconds ago): in-sync
System Usage stats:
    FGVMEV0000000002(updated 3 seconds ago):
        sessions=9, average-cpu-user/nice/system/idle=1%/0%/0%/99%, memory=66%
    FGVMEV7000000005(updated 2 seconds ago):
        sessions=0, average-cpu-user/nice/system/idle=0%/0%/0%/100%, memory=65%
HBDEV stats:
    FGVMEV0000000002(updated 3 seconds ago):
        port2: physical/1000auto, up, rx-bytes/packets/dropped/errors=7698164/22719/0/0,
tx=7815947/23756/0/0
        port4: physical/1000auto, up, rx-bytes/packets/dropped/errors=714501/1749/0/0,
tx=724254/1763/0/0
    FGVMEV7000000005(updated 2 seconds ago):
        port2: physical/1000auto, up, rx-bytes/packets/dropped/errors=7819515/23764/0/0,
tx=7697305/22724/0/0
        port4: physical/1000auto, up, rx-bytes/packets/dropped/errors=726500/1766/0/0,
tx=714129/1751/0/0
MONDEV stats:
    FGVMEVYKXTDJN932(updated 3 seconds ago):
        port3: physical/1000auto, up, rx-bytes/packets/dropped/errors=4610/15/0/0,
tx=1224/21/0/0
    FGVMEV7000000005(updated 2 seconds ago):
        port3: physical/1000auto, up, rx-bytes/packets/dropped/errors=1200/20/0/0,
tx=630/10/0/0
Primary     : FGDocs-P        , FGVMEV0000000002, HA cluster index = 0
Secondary   : FGDocs-S        , FGVMEV7000000005, HA cluster index = 1
number of vcluster: 1
vcluster 1: work 169.254.0.1
Primary: FGVMEV0000000002, HA operating index = 0
Secondary: FGVMEV7000000005, HA operating index = 1
```

When one of the members is out of synchronization:

```
# get system ha status
HA Health Status: OK
Model: FortiGate-VM64
Mode: HA A-P
Group: 0
Debug: 0
Cluster Uptime: 0 days 2:24:46
Cluster state change time: 2021-04-29 13:17:03
Primary selected using:
    <2021/04/29 13:17:03> FGVMEV0000000002 is selected as the primary because its uptime is 
larger than peer member FGVMEV7000000005.
    <2021/04/29 12:37:17> FGVMEV0000000002 is selected as the primary because it's the only 
member in the cluster.
ses_pickup: disable
override: disable
Configuration Status:
    FGVMEV0000000002(updated 0 seconds ago): in-sync
    FGVMEV7000000005(updated 3 seconds ago): out-of-sync
System Usage stats:
    FGVMEV0000000002(updated 0 seconds ago):
        sessions=11, average-cpu-user/nice/system/idle=1%/0%/0%/99%, memory=67%
    FGVMEV7000000005(updated 3 seconds ago):
        sessions=0, average-cpu-user/nice/system/idle=0%/0%/0%/100%, memory=65%
HBDEV stats:
    FGVMEV0000000002(updated 0 seconds ago):
        port2: physical/1000auto, up, rx-bytes/packets/dropped/errors=22257271/64684/0/0, 
tx=24404848/69893/0/0
        port4: physical/1000auto, up, rx-bytes/packets/dropped/errors=12026623/29407/0/0, 
tx=12200664/29417/0/0
    FGVMEV7000000005(updated 3 seconds ago):
        port2: physical/1000auto, up, rx-bytes/packets/dropped/errors=24401109/69877/0/0, 
tx=22245634/64666/0/0
        port4: physical/1000auto, up, rx-bytes/packets/dropped/errors=12195025/29401/0/0, 
tx=12018480/29390/0/0
MONDEV stats:
    FGVMEV0000000002(updated 0 seconds ago):
        port3: physical/1000auto, up, rx-bytes/packets/dropped/errors=4610/15/0/0, 
tx=1224/21/0/0
    FGVMEV7000000005(updated 3 seconds ago):
        port3: physical/1000auto, up, rx-bytes/packets/dropped/errors=1200/20/0/0, 
tx=630/10/0/0
Primary     : FGDocs-P        , FGVMEV0000000002, HA cluster index = 0
Secondary   : FGDocs-S        , FGVMEV7000000005, HA cluster index = 1
number of vcluster: 1
vcluster 1: work 169.254.0.1
Primary: FGVMEV0000000002, HA operating index = 0
Secondary: FGVMEV7000000005, HA operating index = 1
```

Out-of-band management with reserved management interfaces

As part of an HA configuration, you can reserve up to four management interfaces to provide direct management access to all cluster units. For each reserved management interface, you can configure a different IP address, administrative access, and other interface settings, for each cluster unit. By connecting these interfaces to your network, you can separately manage each cluster unit from different IP addresses.

- Reserved management interfaces provide direct management access to each cluster unit, and give each cluster unit a different identity on your network. This simplifies using external services, such as SNMP, to monitor separate cluster units.
- Reserved management interfaces are not assigned HA virtual MAC addresses. They retain the permanent hardware address of the physical interface, unless you manually change it using the `config system interface` command.
- Reserved management interfaces and their IP addresses should not be used for managing a cluster using FortiManager. To manage a FortiGate HA cluster with FortiManager, use the IP address of one of the cluster unit interfaces.
- Configuration changes to a reserved management interface are not synchronized to other cluster units. Other configuration changes are automatically synchronized to all cluster units.

You can configure an in-band management interface for a cluster unit. See In-band management on page 2053 for information. In-band management does not reserve the interface exclusively for HA management.

Management interface

Enable HTTPS or HTTP administrative access on the reserved management interfaces to connect to the GUI of each cluster unit. On secondary units, the GUI has the same features as the primary unit, except for unit specific information, for example:

- The System Information widget on the Status dashboard shows the secondary unit's serial number.
- In the cluster members list at *System > HA*, you can change the HA configuration of the unit that you are logged into. You can only change the host name and device priority of the primary and other secondary units.
- The system events logs show logs for the device that you are logged into. Use the HA device drop down to view the log messages for other cluster units, including the primary unit.

Enable SSH administrative access on the reserved management interfaces to connect to the CLI of each cluster unit. The CLI prompt includes the host of the cluster unit that you are connected to. Use the `execute ha manage` command to connect to other cluster unit CLIs.

Enable SNMP administrative access on a reserved management interface to use SNMP to monitor each cluster unit using the interface's IP address. Direct management of cluster members must also be enabled, see Configuration examples on page 2049.

Reserved management interfaces are available in both NAT and transparent mode, and when the cluster is operating with multiple VDOMs.

FortiCloud, FortiSandbox, and other management services

By default, management services such as FortiCloud, FortiSandbox, SNMP, remote logging, and remote authentication, use a cluster interface. This means that communication from each cluster unit will come from a cluster interface of the primary unit, and not from the individual cluster unit's interface.

You can configure HA reserved management interfaces to be used for communication with management services by enabling the `ha-direct` option. This separates management traffic for each cluster unit, and allows each unit to be individually managed. This is especially useful when cluster units are in different physical locations.

The following management features will then use the HA reserved management interface:

- Remote logging, including syslog, FortiAnalyzer, and FortiCloud
- Remote authentication and certificate verification
- Communication with FortiSandbox
- Netflow and sflow, see Routing NetFlow data over the HA management interface on page 2067 for information.
- SNMP queries and traps

Syntax for HA reserved management interfaces is as follows:

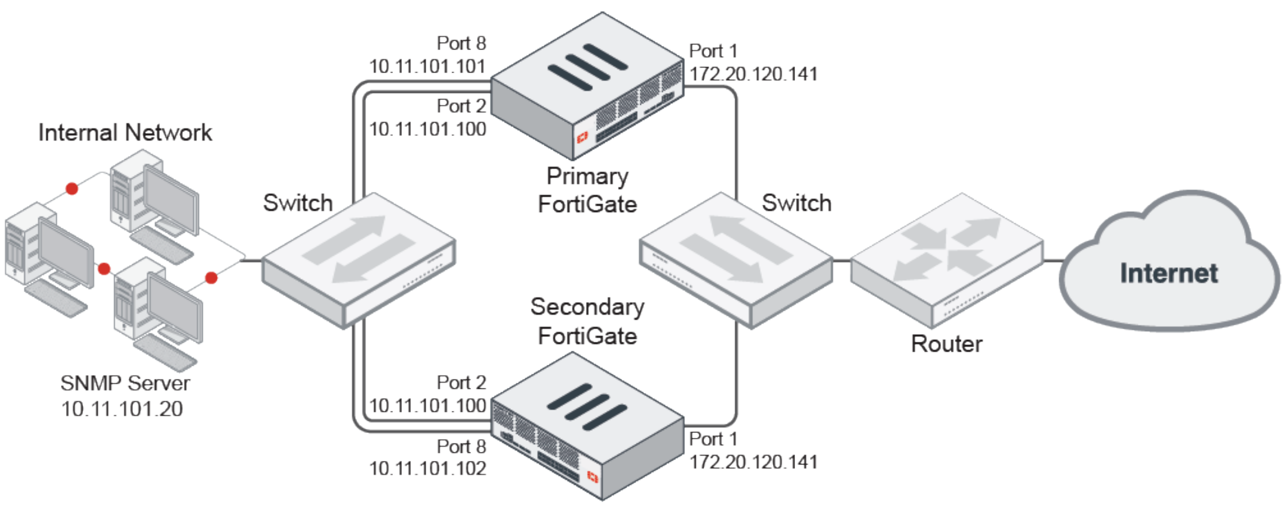
```
config system ha
    set ha-direct enable
    set ha-mgmt-status enable
    config ha-mgmt-interfaces
        edit 1
            set interface <interface>
            set dst <destination IP>
            set gateway <IPv4 gateway>
            set gateway6 <IPv6 gateway>
        next
    end
end
```

The `ha-direct` option is a pre-requisite for allowing communication on each HA reserved management interface for various management services listed above. Once enabled, all `source-ip` settings will be unset from log related, netflow and sflow management services.

SNMP requires `ha-direct` to be configured under SNMP settings only. See below for more configuration options.

Configuration examples

The configuration examples below will use the following topology:



Two FortiGate units are already operating in a cluster. On each unit, port8 is connected to the internal network through a switch and configured as an out-of-band reserved management interface.

Configuration changes to the reserved management interface are not synchronized to other cluster units.

Administrative access and default route for HA management interface

To configure the primary unit's reserved management interface, configure an IP address and management access on port8. Then, configure the necessary HA settings to enable the HA reserved management interface and its route. To configure the secondary unit's reserved management interface, access the unit's CLI through the primary unit, and configure an IP address, management access on port8, and the necessary HA settings. Configuration changes to the reserved management interface are not synchronized to other cluster units.

To configure the primary unit reserved management interface to allow HTTPS, SSH, and ICMP access:

1. From a computer on the internal network, connect to the CLI at 10.11.101.100 on port2.
2. Change the port8 IP address and management access:

```
config system interface
    edit port8
        set ip 10.11.101.101/24
        set allowaccess https ping ssh
    next
end
```

3. Configure the HA settings for the HA reserved management interface by defining a default route to route to the gateway 10.11.101.2:

```
config system ha
    set ha-mgmt-status enable
    config ha-mgmt-interfaces
        edit 1
            set interface port8
            set gateway 10.11.101.2
        next
    end
end
```

   You can now log into the primary unit's GUI by browsing to https://10.11.101.101. You can also log into the primary unit's CLI by using an SSH client to connect to 10.11.101.101.

To configure secondary unit reserved management interfaces to allow HTTPS, SSH, and ICMP access:

1. From a computer on the internal network, connect to the primary unit's CLI.
2. Connect to the secondary unit with the following command:

```
execute ha manage <unit id> <username> <password>
```

3. Change the port8 IP address and management access:

```
config system interface
    edit port8
        set ip 10.11.101.102/24
        set allowaccess https ping ssh
```

```
    next
end
```

```
exit
```

4. Configure the HA settings for the HA reserved management interface by defining a default route to route to the gateway 10.11.101.2:

```
config system ha
    set ha-mgmt-status enable
    config ha-mgmt-interfaces
        edit 1
            set interface port8
            set gateway 10.11.101.2
        next
    end
end
```

You can now log into the secondary unit's GUI by browsing to https://10.11.101.102. You can also log into the secondary unit's CLI by using an SSH client to connect to 10.11.101.102.

SNMP monitoring

The SNMP server can get status information from the cluster members. To use the reserved management interfaces, you must add at least one HA direct management host to an SNMP community. If the SNMP configuration includes SNMP users with user names and passwords, HA direct management must be enabled for the users.

To configure the cluster for SNMP management using the reserved management interfaces in the CLI:

1. Allow SNMP on port8 on both primary and secondary units:

```
config system interface
    edit port8
        append allowaccess snmp
    next
end
```

2. Add an SNMP community with a host for the reserved management interface of each cluster member. The host includes the IP address of the SNMP server.

```
config system snmp community
    edit 1
        set name "Community"
        config hosts
            edit 1
                set ip 10.11.101.20 255.255.255.255
                set ha-direct enable
            next
        end
    next
end
```

Enabling `ha-direct` in a non-HA environment will make SNMP unusable.

3. Add an SNMP user for the reserved management interface:

```
config system snmp user
    edit "1"
        set notify-hosts 10.11.101.20
        set ha-direct enable
    next
end
```

The SNMP configuration is synchronized to all cluster units.

To get CPU, memory, and network usage information from the SNMP manager for each cluster unit using the reserved management IP addresses:

1. Connect to the SNMP manager CLI.
2. Get resource usage information for the primary unit using the MIB fields:

```
snmpget -v2c -c Community 10.11.101.101 fgHaStatsCpuUsage
snmpget -v2c -c Community 10.11.101.101 fgHaStatsMemUsage
snmpget -v2c -c Community 10.11.101.101 fgHaStatsNetUsage
```

3. Get resource usage information for the primary unit using the OIDs:

```
snmpget -v2c -c Community 10.11.101.101 1.3.6.1.4.1.12356.101.13.2.1.1.3.1
snmpget -v2c -c Community 10.11.101.101 1.3.6.1.4.1.12356.101.13.2.1.1.4.1
snmpget -v2c -c Community 10.11.101.101 1.3.6.1.4.1.12356.101.13.2.1.1.5.1
```

4. Get resource usage information for the secondary unit using the MIB fields:

```
snmpget -v2c -c Community 10.11.101.102 fgHaStatsCpuUsage
snmpget -v2c -c Community 10.11.101.102 fgHaStatsMemUsage
snmpget -v2c -c Community 10.11.101.102 fgHaStatsNetUsage
```

5. Get resource usage information for the primary unit using the OIDs:

```
snmpget -v2c -c Community 10.11.101.102 1.3.6.1.4.1.12356.101.13.2.1.1.3.1
snmpget -v2c -c Community 10.11.101.102 1.3.6.1.4.1.12356.101.13.2.1.1.4.1
snmpget -v2c -c Community 10.11.101.102 1.3.6.1.4.1.12356.101.13.2.1.1.5.1
```

Firewall local-in policies for the reserved management interface

Enabling `ha-mgmt-intf-only` applies the local-in policy only to the VDOM that contains the reserved management interface. The incoming interface is set to match any interface in the VDOM.

To add local-in policies for the reserved management interface:

```
config firewall local-in-policy
    edit 0
        set ha-mgmt-intf-only enable
        set intf any
        set srcaddr internal-net
        set dstaddr mgmt-int
        set action accept
        set service HTTPS
```

```
        set schedule weekdays
    next
end
```

NTP over reserved management interfaces

When NTP is enabled in an HA cluster, the primary unit will always be the unit to contact the NTP server and synchronize system time to the secondary units over the HA heartbeat interface. However, in the event that the primary should contact the NTP server over the HA reserved management interface, then the `ha-direct` option should be enabled under the `config system ha` settings.

```
config system interface
    edit port5
        set ip 172.16.79.46 255.255.255.0
    next
end
```

```
config system ha
    set group-name FGT-HA
    set mode a-p
    set ha-mgmt-status enable
    config ha-mgmt-interfaces
        edit 1
            set interface port5
            set gateway 172.16.79.1
        next
    end
    set ha-direct enable
end
```

```
config system ntp
    set ntpsync enable
    set syncinterval 5
end
```

In-band management

In-band management IP addresses are an alternative to reserved HA management interfaces, and do not require reserving an interface exclusively for management access. They can be added to multiple interfaces on each cluster unit.

The in-band management IP address is accessible from the network that the cluster interface is connected to. It should be in the same subnet as the interface that you are adding it to. It cannot be in the same subnet as other interface IP addresses.

In-band management interfaces support ping, HTTP, HTTPS, and SNMP administrative access options.

Primary and secondary units send packets differently from an interface with a management IP address configured:

- On the primary unit, packets are sent to destinations based on routing information.
- On secondary units, packets can only be sent to destinations with the same management IP address segment.

In-band management IP address configuration is not synchronized to other cluster units.

To add an in-band management IP address to port23 with HTTPS, SSH, and SNMP access:

```
config system interface
    edit port23
        set management-ip 172.25.12.5/24
        set allowaccess https ssh snmp
    next
end
```

Upgrading FortiGates in an HA cluster

You can upgrade the firmware on an HA cluster in the same way as on a standalone FortiGate. During a firmware upgrade, the cluster upgrades the primary unit and all of the subordinate units to the new firmware image.

Before upgrading a cluster, back up your configuration (Configuration backups on page 65), schedule a maintenance window, and make sure that you are using a supported upgrade path (https://docs.fortinet.com/upgrade-tool).

Uninterrupted upgrade

An uninterrupted upgrade occurs without interrupting communication in the cluster.

To upgrade the cluster firmware without interrupting communication, the following steps are followed. These steps are transparent to the user and the network, and might result in the cluster selecting a new primary unit.

1. The administrator uploads a new firmware image using the GUI or CLI. See Upgrading individual device firmware on page 1955 for details.
2. The firmware is upgraded on all of the subordinate units.
3. A new primary unit is selected from the upgraded subordinates.
4. The firmware is upgraded on the former primary unit.
5. Primary unit selection occurs, according to the standard primary unit selection process.

If all of the subordinate units crash or otherwise stop responding during the upgrade process, the primary unit will continue to operate normally, and will not be upgraded until at least one subordinate rejoins the cluster.

Interrupted upgrade

An interrupted upgrade upgrades all cluster members at the same time. This takes less time than an uninterrupted upgrade, but it interrupts communication in the cluster. Interrupted upgrade is disabled by default.

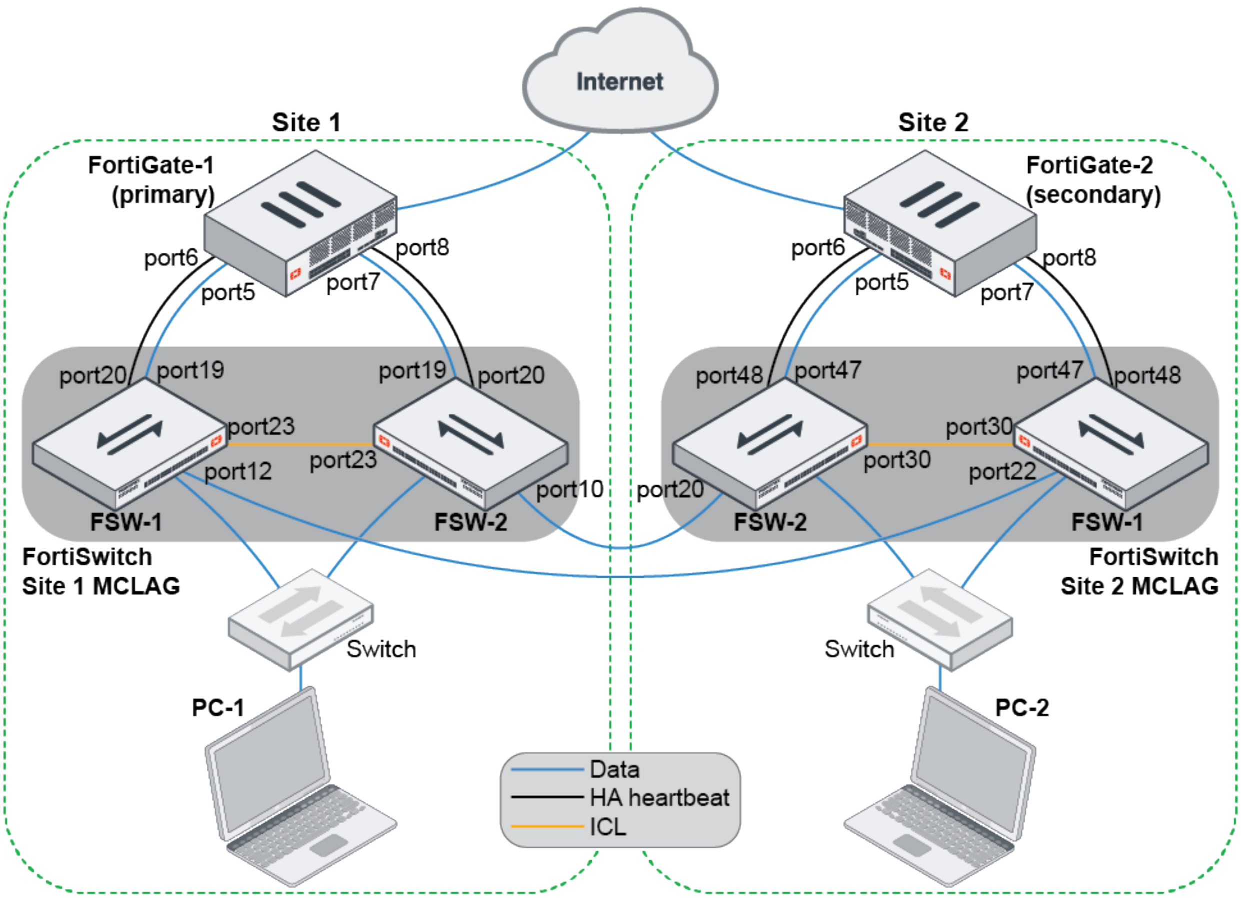
To enable interrupted upgrade:

```
config system ha
    set uninterruptible-upgrade disable
end
```

HA between remote sites over managed FortiSwitches

In a multi-site FortiGate HA topology that uses managed FortiSwitches in a multi-chassis link aggregation group (MCLAG) to connect between sites, HA heartbeat signals can be sent through the switch layer of the FortiSwitches, instead of through back-to-back links between the heartbeat interfaces. This means that two fiber connections can be used, instead of four (two back-to-back heartbeat fiber connections and two connections for the FortiSwitches). The FortiSwitches can be different models, but must all support MCLAG and be running version 6.4.2 or later.

This example shows how to configure heartbeat VLANs to assign to the access ports that the heartbeat interfaces connect to, passing over the trunk between the FortiSwitches on the two sites.

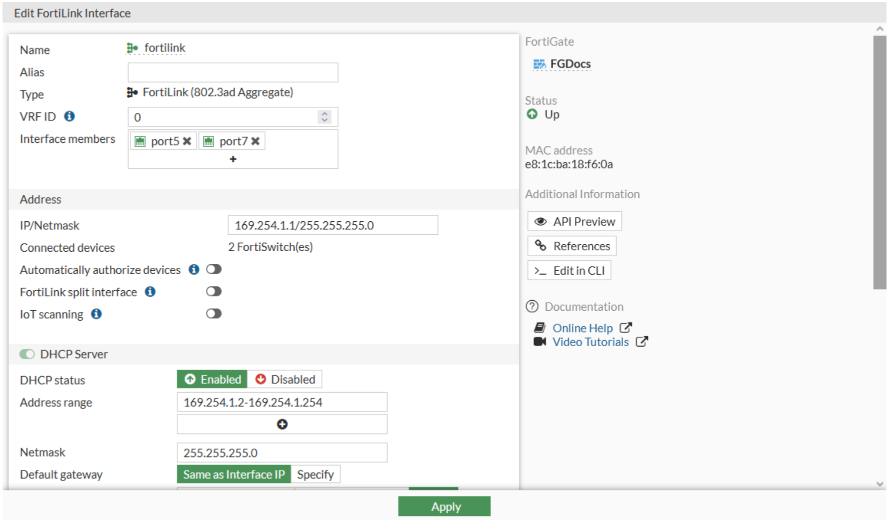


FortiGate HA is with two FortiGates in separate locations and the switch layer connection between the FortiSwitches is used for the heartbeat signal.

To configure the example:

1. Disconnect the physical connections between Site 1 and Site 2:
   - Disconnect the cable on Site 1 FSW-1 port 12.
   - Disconnect the cable on Site 1 FSW-2 port 10.
2. Configure Site 1:

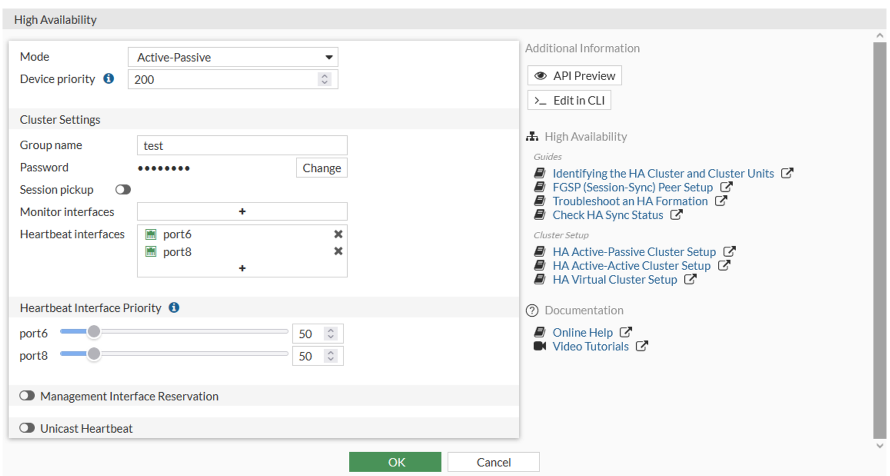
a. On the FortiGate, go to *WiFi & Switch Controller > FortiLink Interface* and configure FortiLink:



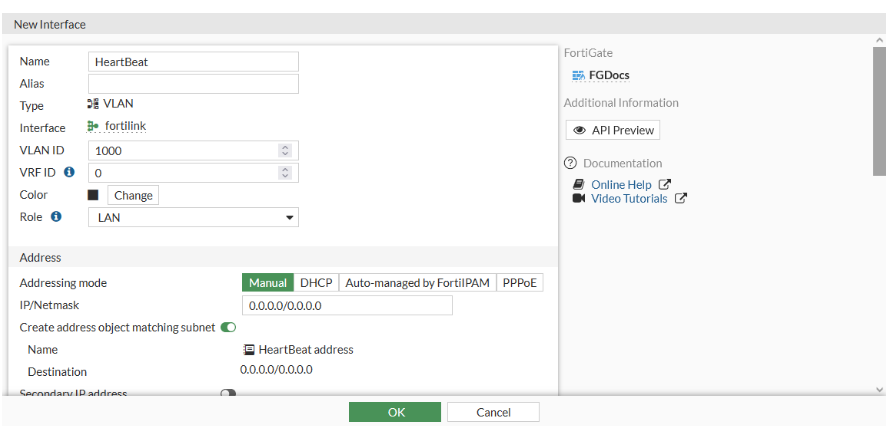
b. Go to *System > HA* and configure HA:

i. Set the heartbeat ports to the ports that are connected to FortiSwitch.

ii. Adjust the priority and enable override so that this FortiGate becomes the primary.



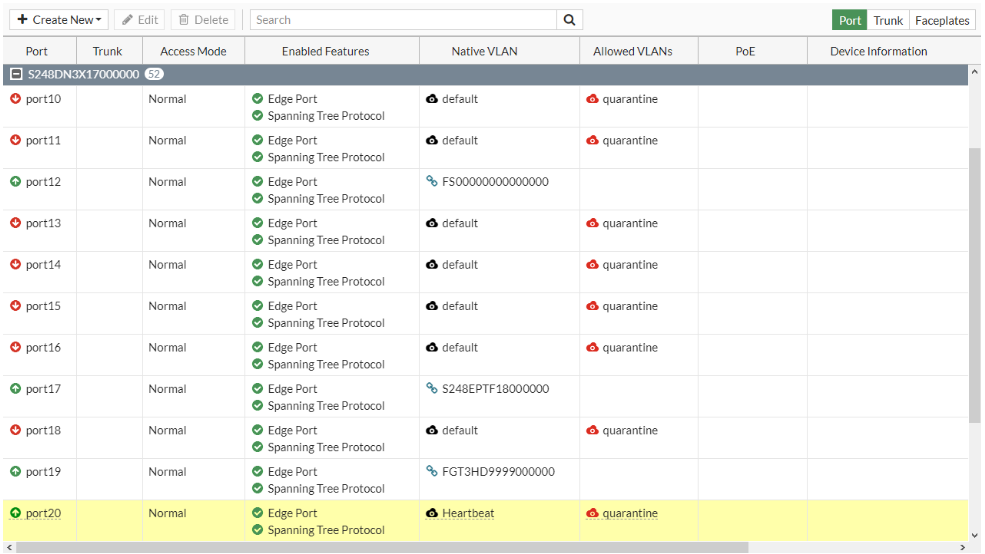
c. Go to *WiFi & Switch Controller > FortiSwitch VLANs* and create switch VLANs that are dedicated to each FortiGate HA heartbeat interface between the two FortiGates: Heartbeat VLAN 1000 and Heartbeat VLAN 1100.



d. Assign the native VLAN of the switch ports that are connected to the heartbeat ports to the created VLAN. Each HA heartbeat should be in its own VLAN.

i. Go to *WiFi & Switch Controller > FortiSwitch Ports*.

ii. In the *Native VLAN* column for the heartbeat port that is connected to FSW-1, click the edit icon and select the *Heartbeat* VLAN.



iii. In the *Native VLAN* column for the heartbeat port that is connected to FSW-2, click the edit icon and select the *Heartbeat2* VLAN.

e. On each FortiSwitch, enable MCLAG-ICL on the trunk port:

```
config switch trunk
    edit D243Z17000032-0
        set mclag-icl enable
    next
end
```

3. Configure Site 2 the same as Site 1, except set the HA priority so that the FortiGate becomes the secondary.
4. Disconnect the physical connections for FortiGate HA and FortiLink interfaces on Site 2:
   - Disconnect the cable on Site 2 FSW-1 ports 47 and 48.
   - Disconnect the cable on Site 2 FSW-2 ports 47 and 48.
5. Connect cables between the FortiSwitch MCLAG in Site 1 and Site 2:
   - Connect a cable from Site 1 FSW-1 port 12 to Site 2 FSW-1 port 22.
   - Connect a cable from Site 1 FSW-2 port 10 to Site 2 FSW-2 port 20.
6. On all of the FortiSwitches, configure the `auto-isl-port-group`. The group must match on both sides.

a. Site 1 FSW-1:

Set `members` to the port that is connected to Site 2 FSW-1:

```
config switch auto-isl-port-group
    edit 1
        set members port12
    next
end
```

b. Site 1 FSW-2:

Set `members` to the port that is connected to Site 1 FSW-1:

```
config switch auto-isl-port-group
    edit 1
        set members port22
    next
end
```

c. Site 2 FSW-1:

Set `members` to the port that is connected to Site 2 FSW-2:

```
config switch auto-isl-port-group
    edit 1
        set members port10
    next
end
```

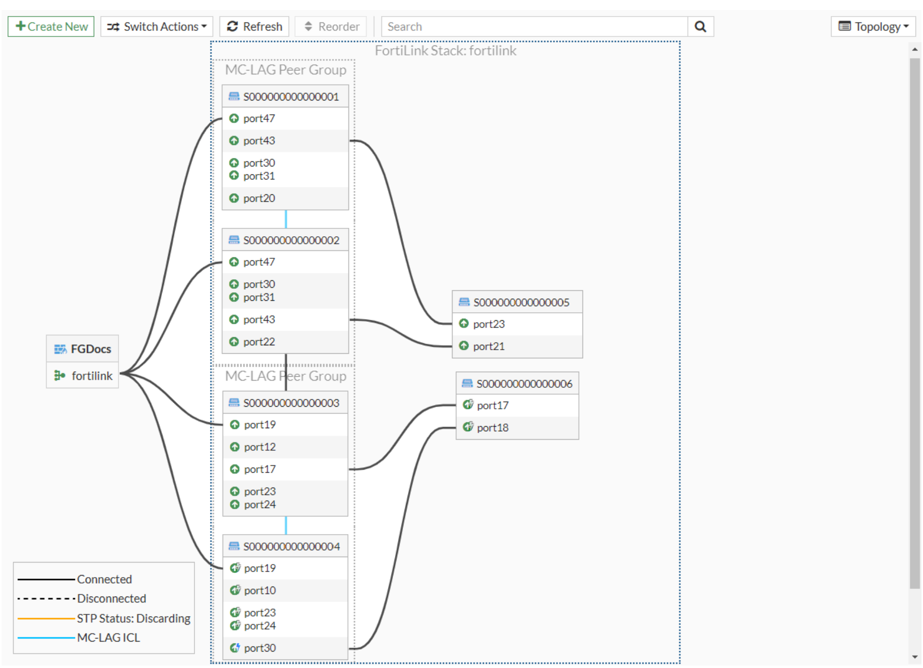
d. Site 2 FSW-2:

Set `members` to the port that is connected to Site 1 FSW-2:

```
config switch auto-isl-port-group
    edit 1
        set members port20
    next
end
```

7. Connect the FortiGate HA and FortiLink interface connections on Site 2.
8. Configure a firewall policy and route for traffic so that the client can reach the internet.
9. Wait for HA to finish synchronizing and for all of the FortiSwitches to come online, then on FortiGate-1, go to *WiFi & Switch Controller* > *Managed FortiSwitches* and select the Topology view from the drop-down on the right.

   The page should look similar to the following:



To test the configuration to confirm what happens when there is a failover:

1. On both PC-1 and PC-2, access the internet and monitor traffic. The traffic should be going through the primary FortiGate.
2. Perform a continuous ping to an outside IP address, then reboot any one of the FortiSwitches.
   Traffic from both Site 1 and Site 2 to the internet should be recovered in approximately five seconds.
3. Perform a continuous ping to an outside IP address, then force an HA failover (see Force HA failover for testing and demonstrations on page 2069).
   Traffic from both Site 1 and Site 2 to the internet should be recovered in approximately five seconds.
4. After an HA failover, on the new primary FortiGate, go to *WiFi & Switch Controller > Managed FortiSwitch*.
   The switch layer tiering will be changed so that the directly connected FortiSwitches are at the top of the topology.

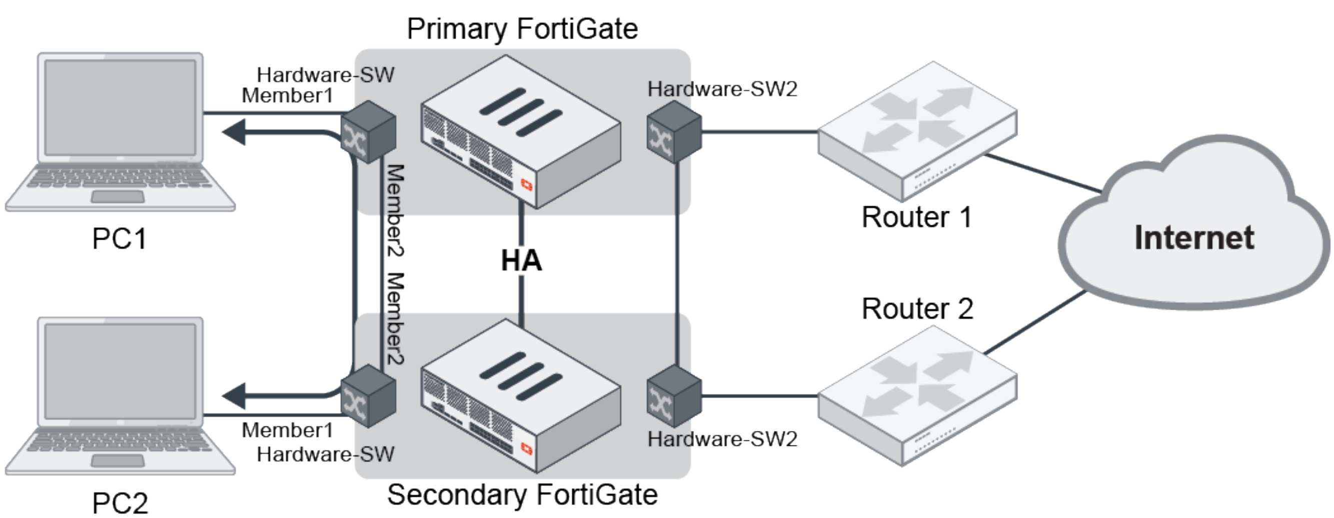
HA using a hardware switch to replace a physical switch

Using a hardware switch to replace a physical switch is not recommended, as it offers no redundancy or interface monitoring.

- If one FortiGate loses power, all of the clients connected to that FortiGate device cannot go to another device until that FortiGate recovers.
- A hardware switch cannot be used as a monitor interface in HA. Any incoming or outgoing link failures on hardware member interfaces will not trigger failover; this can affect traffic.

Examples

The examples use the following topology:



Traffic between hardware switches

When using Hardware switch in HA environment, a client device connected to the hardware switch on the primary FortiGate can communicate with client devices connected to the hardware switch on secondary FortiGates as long as there is a direct connection between the two switches.

No configuration is required after setting up the hardware switches. If a client connected to both of the hardware switches needs to reach destinations outside of the cluster, the firewall must be configured for it.

To configure the FortiGate devices:

1. Connect the devices as shown in the topology diagram.
2. On each FortiGate, configure HA:

```
config system ha
    set mode a-a
    set group-name Example_cluster
    set hbdev ha1 10 ha2 20
end
```

3. On the primary FortiGate, configure the hardware switch:

```
config system virtual-switch
    edit Hardware-SW
        set physical-switch sw0
        config port
            edit port3
            next
            edit port5
            next
        end
    next
end
```

4. On each FortiGate, configure the IP addresses on the hardware switches:

```
config system interface
    edit Hardware-SW
        set ip 6.6.6.1 255.255.255.0
        set allowaccess ping ssh http https
    next
end
```

After configuring the hardware switches, PC1 and PC2 can now communicate with each other.

Traffic passes through FortiGate

If client device needs to send traffic through the FortiGate, additional firewall configuration on the FortiGate is required.

All traffic from the hardware switches on either the primary or secondary FortiGate reaches the primary FortiGate first. The traffic is then directed according to the HA mode and firewall configuration.

To configure the FortiGate devices:

1. Connect the devices as shown in the topology diagram.
2. On each FortiGate, configure HA:

```
config system ha
    set mode a-a
    set group-name Example_cluster
    set hbdev ha1 10 ha2 20
end
```

3. On the primary FortiGate, configure the hardware switch:

```
config system virtual-switch
    edit Hardware-SW
```

```
        set physical-switch sw0
        config port
            edit port3
            next
            edit port5
            next
        end
    next
    edit Hardware-SW2
        set physical-switch sw0
        config port
            edit port1
            next
        end
    next
end
```

4. On each FortiGate, configure the IP addresses on the hardware switch:

```
config system interface
    edit Hardware-SW
        set ip 6.6.6.1 255.255.255.0
        set allowaccess ping ssh http https
    next
    edit Hardware-SW2
        set ip 172.16.200.1 255.255.255.0
        set allowaccess ping ssh http https
    next
end
```

5. On each FortiGate, configure a firewall policy:

```
config firewall policy
    edit 1
        set srcintf Hardware-SW
        set dstintf Hardware-SW2
        set srcaddr all
        set dstaddr all
        set service ALL
        set action accept
        set schedule always
        set nat enable
    next
end
```

6. On each FortiGate, configure a static route:

```
config router static
    edit 1
        set device Hardware-SW2
        set gateway 172.16.200.254
    next
end
```

Traffic from PC1 and PC2 can now reach destinations outside of the FortiGate cluster.

VDOM exceptions

VDOM exceptions are settings that can be selected for specific VDOMs or all VDOMs that are not synchronized to other HA members. This can be required when cluster members are not in the same physical location, subnets, or availability zones in a cloud environment.

Some examples of possible use cases include:

- You use different source IP addresses for FortiAnalyzer logging from each cluster member. See Override FortiAnalyzer and syslog server settings on page 2063 for more information.
- You need to keep management interfaces that have specific VIPs or local subnets that cannot transfer from being synchronized.
- In a unicast HA cluster in the cloud, you use NAT with different IP pools in different subnets, so IP pools must be exempt.

When a VDOM exception is configured, the object will not be synchronized between the primary and secondary devices when the HA forms. Different options can be configured for every object.

When VDOM mode is disabled, the configured object is excluded for the entire device. To define a scope, VDOM mode must be enabled and the object must be configurable in a VDOM.

VDOM exceptions are synchronized to other HA cluster members.

To configure VDOM exceptions:

```
config global
    config system vdom-exception
        edit 1
            set object <object name>
            set scope {all* | inclusive | exclusive}
            set vdom <vdom name>
        next
    end
end
```

`object`	The name of the configuration object that can be configured independently for some or all of the VDOMs. See Objects on page 2062 for a list of available settings and resources.
`scope`	Determine if the specified object is configured independently for all VDOMs or a subset of VDOMs. • `all`: Configure the object independently on all VDOMs. • `inclusive`: Configure the object independently only on the specified VDOMs. • `exclusive`: Configure the object independently on all of the VDOMs that are not specified.
`vdom`	The names of the VDOMs that are included or excluded.

Objects

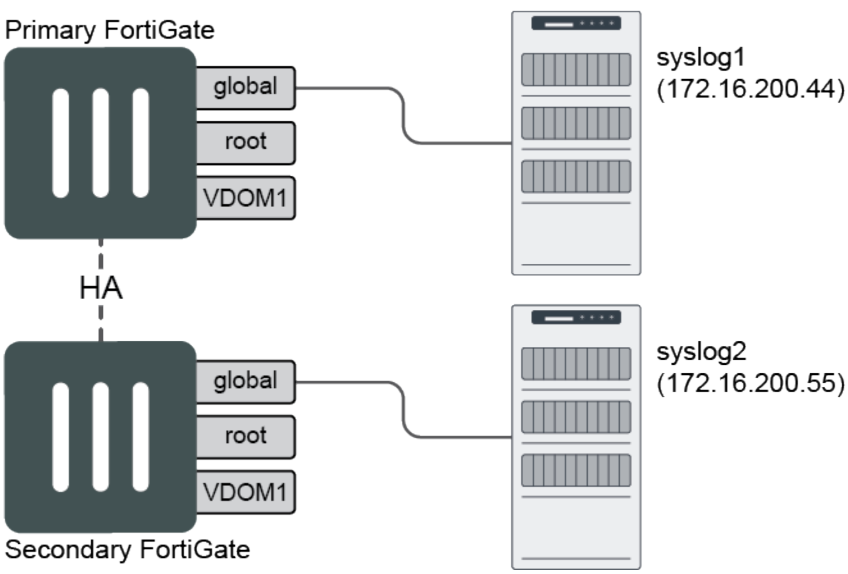
The following settings and resources can be exempt from synchronization in an HA cluster:

log.fortianalyzer.setting	user.radius
log.fortianalyzer.override-setting	system.interface
log.fortianalyzer2.setting	vpn.ipsec.phase1-interface
log.fortianalyzer2.override-setting	vpn.ipsec.phase2-interface
log.fortianalyzer3.setting	router.bgp
log.fortianalyzer3.override-setting	router.route-map
log.fortianalyzer-cloud.setting	router.prefix-list
log.fortianalyzer-cloud.override-setting	firewall.ippool
log.syslogd.setting	firewall.ippool6
log.syslogd.override-setting	router.static
log.syslogd2.setting	router.static6
log.syslogd2.override-setting	firewall.vip
log.syslogd3.setting	firewall.vip6
log.syslogd3.override-setting	system.sdwan
log.syslogd4.setting	system.saml
log.syslogd4.override-setting	router.policy
system.central-management	router.policy6
system.csf	

Override FortiAnalyzer and syslog server settings

In an HA cluster, secondary devices can be configured to use different FortiAnalyzer devices and syslog servers than the primary device. VDOMs can also override global syslog server settings.

Configure a different syslog server on a secondary HA device



To configure the primary HA device:

1. Configure a global syslog server:

```
config global
    config log syslog setting
        set status enable
        set server 172.16.200.44
        set facility local6
        set format default
    end
end
```

2. Set up a VDOM exception to enable setting the global syslog server on the secondary HA device:

```
config global
    config system vdom-exception
        edit 1
            set object log.syslogd.setting
        next
    end
end
```

To configure the secondary HA device:

1. Configure a global syslog server:

```
config global
    config log syslogd setting
        set status enable
        set server 172.16.200.55
        set facility local5
    end
end
```

2. After the primary and secondary device synchronize, generate logs on the secondary device.

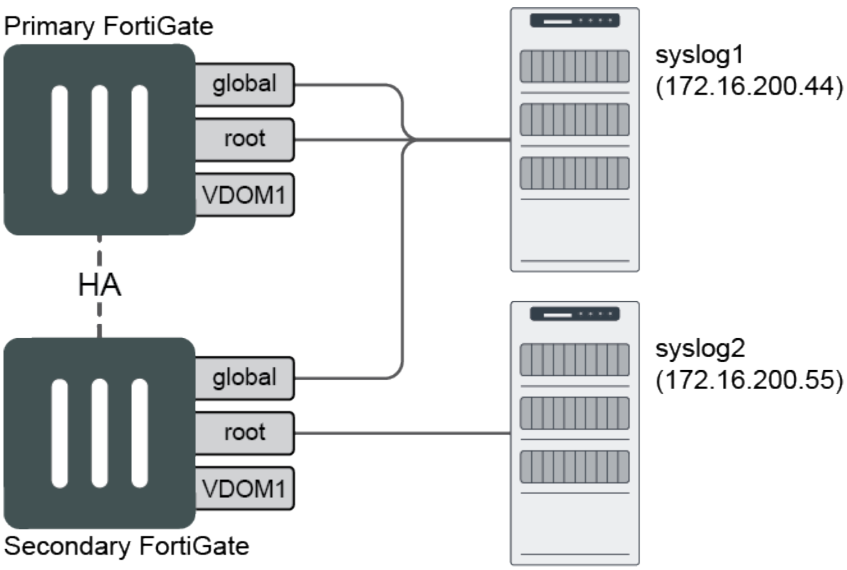
To confirm that logs are been sent to the syslog server configured on the secondary device:

1. On the primary device, retrieve the following packet capture from the secondary device's syslog server:

```
# diagnose sniffer packet any "host 172.16.200.55" 6
interfaces=[any]
filters=[host 172.16.200.55]

266.859494 port2 out 172.16.200.2.7434 -> 172.16.200.55.514: udp 278
0x0000   0000 0000 0000 0009 0f09 0004 0800 4500        ..............E.
0x0010   0132 f3c7 0000 4011 9d98 ac10 c802 ac10        .2....@.........
0x0020   c837 1d0a 0202 011e 4b05 3c31 3734 3e64        .7......K.<174>d
0x0030   6174 653d 3230 3230 2d30 332d 3134 2074        ate=2020-03-14.t
0x0040   696d 653d 3132 3a30 303a 3035 2064 6576        ime=12:00:05.dev
0x0050   6e61 6d65 3d22 466f 7274 6947 6174 652d        name="FGT-81E-Sl
0x0060   3831 455f 4122 2064 6576 6964 3d22 4647        ave-A".devid="FG
0x0070   5438 3145 3451 3136 3030 3030 3438 2220        T81E4Q16000048".
0x0080   6c6f 6769 643d 2230 3130 3030 3230 3032        logid="010002002
0x0090   3722 2074 7970 653d 2265 7665 6e74 2220        7".type="event".
0x00a0   7375 6274 7970 653d 2273 7973 7465 6d22        subtype="system"
0x00b0   206c 6576 656c 3d22 696e 666f 726d 6174        .level="informat
```

```
0x00c0   696f 6e22 2076 643d 2276 646f 6d31 2220        ion".vd="vdom1".
0x00d0   6576 656e 7474 696d 653d 3135 3834 3231        eventtime=158421
0x00e0   3234 3035 3835 3938 3335 3639 3120 747a        2405859835691.tz
0x00f0   3d22 2d30 3730 3022 206c 6f67 6465 7363        ="-0700".logdesc
0x0100   3d22 4f75 7464 6174 6564 2072 6570 6f72        ="Outdated.repor
0x0110   7420 6669 6c65 7320 6465 6c65 7465 6422        t.files.deleted"
0x0120   206d 7367 3d22 4465 6c65 7465 2031 206f        .msg="Delete.1.o
0x0130   6c64 2072 6570 6f72 7420 6669 6c65 7322        ld.report.files"
```

Configure a different syslog server in the root VDOM on a secondary HA device



To configure the primary HA device:

1. Configure a global syslog server:

```
config global
    config log syslog setting
        set status enable
        set server 172.16.200.44
        set facility local6
        set format default
    end
end
```

2. Set up a VDOM exception to enable `syslog-override` in the secondary HA device root VDOM:

```
config global
    config system vdom-exception
        edit 1
            set object log.syslogd.override-setting
            set scope inclusive
            set vdom root
        next
    end
end
```

3. In the VDOM, enable `syslog-override` in the log settings, and set up the override syslog server:

```
config root
    config log setting
        set syslog-override enable
    end
    config log syslog override-setting
        set status enable
        set server 172.16.200.44
        set facility local6
        set format default
    end
end
```

After `syslog-override` is enabled, an override syslog server must be configured, as logs will not be sent to the global syslog server.

To configure the secondary HA device:

1. Configure an override syslog server in the root VDOM:

```
config root
    config log syslogd override-setting
        set status enable
        set server 172.16.200.55
        set facility local5
        set format default
    end
end
```

2. After the primary and secondary device synchronize, generate logs in the root VDOM on the secondary device.

To confirm that logs are been sent to the syslog server configured for the root VDOM on the secondary device:

1. On the primary device, retrieve the following packet capture from the syslog server configured in the root VDOM on the secondary device:

```
# diagnose sniffer packet any "host 172.16.200.55" 6
interfaces=[any]
filters=[host 172.16.200.55]

156.759696 port2 out 172.16.200.2.1165 -> 172.16.200.55.514: udp 277
0x0000   0000 0000 0000 0009 0f09 0004 0800 4500        ..............E.
0x0010   0131 f398 0000 4011 9dc8 ac10 c802 ac10        .1....@.........
0x0020   c837 048d 0202 011d af5f 3c31 3734 3e64        .7......._<174>d
0x0030   6174 653d 3230 3230 2d30 332d 3134 2074        ate=2020-03-14.t
0x0040   696d 653d 3131 3a33 353a 3035 2064 6576        ime=11:35:05.dev
0x0050   6e61 6d65 3d22 466f 7274 6947 6174 652d        name="FGT-81E-Sl
0x0060   3831 455f 4122 2064 6576 6964 3d22 4647        ave-A".devid="FG
0x0070   5438 3145 3451 3136 3030 3030 3438 2220        T81E4Q16000048".
0x0080   6c6f 6769 643d 2230 3130 3030 3230 3032        logid="010002002
0x0090   3722 2074 7970 653d 2265 7665 6e74 2220        7".type="event".
0x00a0   7375 6274 7970 653d 2273 7973 7465 6d22        subtype="system"
0x00b0   206c 6576 656c 3d22 696e 666f 726d 6174        .level="informat
0x00c0   696f 6e22 2076 643d 2272 6f6f 7422 2065        ion".vd="root".e
0x00d0   7665 6e74 7469 6d65 3d31 3538 3432 3130        venttime=1584210
```

```
0x00e0   3930 3537 3539 3334 3132 3632 2074 7a3d        905759341262.tz=
0x00f0   222d 3037 3030 2220 6c6f 6764 6573 633d        "-0700".logdesc=
0x0100   224f 7574 6461 7465 6420 7265 706f 7274        "Outdated.report
0x0110   2066 696c 6573 2064 656c 6574 6564 2220        .files.deleted".
0x0120   6d73 673d 2244 656c 6574 6520 3220 6f6c        msg="Delete.2.ol
0x0130   6420 7265 706f 7274 2066 696c 6573 22          d.report.files"
```

Routing NetFlow data over the HA management interface

In an HA environment, the `ha-direct` option allows data from services such as syslog, FortiAnalyzer, SNMP, and NetFlow to be routed over the outgoing interface.

The following example shows how NetFlow data can be routed over the HA management interface mgmt1.

To route NetFlow data over the HA management interface:

1. On the primary unit (FortiGate A), configure the HA and mgmt1 interface settings:

```
(global) # config system ha
    set group-name "test-ha"
    set mode a-p
    set password *********
    set hbdev "port6" 50
    set hb-interval 4
    set hb-lost-threshold 10
    set session-pickup enable
    set ha-mgmt-status enable
    config ha-mgmt-interfaces
        edit 1
            set interface "mgmt1"
        next
    end
    set override enable
    set priority 200
    set ha-direct enable
end
```

```
(global) # config system interface
    edit "mgmt1"
        set ip 10.6.30.111 255.255.255.0
        set allowaccess ping https ssh http telnet fgfm
        set type physical
        set dedicated-to management
        set role lan
        set snmp-index 1
    next
end
```

2. On the secondary unit (FortiGate B), configure the HA and mgmt1 interface settings:

```
(global) # config system ha
    set group-name "test-ha"
    set mode a-p
    set password *********
    set hbdev "port6" 50
    set hb-interval 4
```

```
        set hb-lost-threshold 10
        set session-pickup enable
        set ha-mgmt-status enable
        config ha-mgmt-interfaces
            edit 1
                set interface "mgmt1"
            next
        end
        set override enable
        set priority 100
        set ha-direct enable
    end
```

```
(global) # config system interface
    edit "mgmt1"
        set ip 10.6.30.112 255.255.255.0
        set allowaccess ping https ssh http telnet fgfm
        set type physical
        set dedicated-to management
        set role lan
        set snmp-index 1
    next
end
```

3. On the primary unit (FortiGate A), configure the NetFlow setting:

```
(global) # config system netflow
    set collector-ip 10.6.30.59
end
```

4. Verify that NetFlow uses the mgmt1 IP:

```
(global) # diagnose test application sflowd 3
```

5. Verify that the NetFlow packets are being sent by the mgmt1 IP:

```
(vdom1) # diagnose sniffer packet any 'udp and port 2055' 4
interfaces=[any]
filters=[udp and port 2055]
8.397265 mgmt1 out 10.6.30.111.1992 -> 10.6.30.59.2055: udp 60
23.392175 mgmt1 out 10.6.30.111.1992 -> 10.6.30.59.2055: udp 188
23.392189 mgmt1 out 10.6.30.111.1992 -> 10.6.30.59.2055: udp 60
...
3 packets received by filter
0 packets dropped by kernel
```

6. On the secondary device (FortiGate B), change the priority so that it becomes the primary:

```
(global) # config system ha
    set priority 250
end
```

7. Verify the NetFlow status on FortiGate A, which is using the new primary's mgmt1 IP:

```
(global) # diagnose test application sflowd 3
```

8. Verify that the NetFlow packets use the new source IP on FortiGate B:

```
(vdom1) # diagnose sniffer packet any 'udp and port 2055' 4
interfaces=[any]
filters=[udp and port 2055]
```

```
7.579574 mgmt1 out 10.6.30.112.3579 -> 10.6.30.59.2055: udp 60
22.581830 mgmt1 out 10.6.30.112.3579 -> 10.6.30.59.2055: udp 60
29.038336 mgmt1 out 10.6.30.112.3579 -> 10.6.30.59.2055: udp 1140
^C
3 packets received by filter
0 packets dropped by kernel
```

Force HA failover for testing and demonstrations

This command should only be used for testing, troubleshooting, maintenance, and demonstrations.

Do not use it in a live production environment outside of an active maintenance window.

HA failover can be forced on an HA primary device. The device will stay in a failover state regardless of the conditions. The only way to remove the failover status is by manually turning it off.

Syntax

```
execute ha failover set <cluster_id>
execute ha failover unset <cluster_id>
```

Variable	Description
<cluster_id>	The cluster ID is `1` for any cluster that is not in virtual cluster mode, and can be `1` or `2` if virtual cluster mode is enabled.

Example

To manually force an HA failover:

```
# execute ha failover set 1
Caution: This command will trigger an HA failover.
It is intended for testing purposes.
Do you want to continue? (y/n)y
```

To view the failover status:

```
# execute ha failover status
failover status: set
```

To view the system status of a device in forced HA failover:

```
# get system ha status
HA Health Status: OK
Model: FortiGate-300D
Mode: HA A-P
Group: 240
Debug: 0
Cluster Uptime: 0 days 2:11:46
Cluster state change time: 2020-03-12 17:38:04
Primary selected using:
```

```
    <2020/03/12 17:38:04> FGT3HD3914800153 is selected as the primary because it has EXE_
FAIL_OVER flag set.
    <2020/03/12 15:27:26> FGT3HD3914800069 is selected as the primary because it has the
largest value of override priority.
ses_pickup: disable
override: enable
Configuration Status:
    FGT3HD3914800069(updated 4 seconds ago): in-sync
    FGT3HD3914800153(updated 3 seconds ago): in-sync
System Usage stats:
    FGT3HD3914800069(updated 4 seconds ago):
        sessions=5, average-cpu-user/nice/system/idle=0%/0%/0%/100%, memory=30%
    FGT3HD3914800153(updated 3 seconds ago):
        sessions=41, average-cpu-user/nice/system/idle=0%/0%/0%/99%, memory=30%
HBDEV stats:
    FGT3HD3914800069(updated 4 seconds ago):
        port3: physical/1000auto, up, rx-bytes/packets/dropped/errors=15914162/42929/0/0,
tx=15681840/39505/0/0
        port5: physical/1000auto, up, rx-bytes/packets/dropped/errors=17670346/52854/0/0,
tx=20198409/54692/0/0
    FGT3HD3914800153(updated 3 seconds ago):
        port3: physical/1000auto, up, rx-bytes/packets/dropped/errors=16636700/45544/0/0,
tx=15529791/39512/0/0
        port5: physical/1000auto, up, rx-bytes/packets/dropped/errors=20199928/54699/0/0,
tx=17672146/52862/0/0
Secondary: FortiGate-300D  , FGT3HD3914800069, HA cluster index = 1
Primary: FortiGate-300D  , FGT3HD3914800153, HA cluster index = 0
number of vcluster: 1
vcluster 1: standby 169.254.0.1
Secondary: FGT3HD3914800069, HA operating index = 1
Primary: FGT3HD3914800153, HA operating index = 0
```

To stop the failover status:

```
# execute ha failover unset 1
Caution: This command may trigger an HA failover.
It is intended for testing purposes.
Do you want to continue? (y/n)y
```

To view the system status of a device after forced HA failover is disabled:

```
# get system ha status
HA Health Status: OK
Model: FortiGate-300D
Mode: HA A-P
Group: 240
Debug: 0
Cluster Uptime: 0 days 2:14:55
Cluster state change time: 2020-03-12 17:42:17
Primary selected using:
    <2020/03/12 17:42:17> FGT3HD3914800069 is selected as the primary because it has the
largest value of override priority.
    <2020/03/12 17:38:04> FGT3HD3914800153 is selected as the primary because it has EXE_
FAIL_OVER flag set.
    <2020/03/12 15:27:26> FGT3HD3914800069 is selected as the primary because it has the
largest value of override priority.
```

```
ses_pickup: disable
override: enable
Configuration Status:
    FGT3HD3914800069(updated 3 seconds ago): in-sync
    FGT3HD3914800153(updated 2 seconds ago): in-sync
System Usage stats:
    FGT3HD3914800069(updated 3 seconds ago):
        sessions=0, average-cpu-user/nice/system/idle=0%/0%/0%/100%, memory=30%
    FGT3HD3914800153(updated 2 seconds ago):
        sessions=38, average-cpu-user/nice/system/idle=0%/0%/0%/100%, memory=30%
HBDEV stats:
    FGT3HD3914800069(updated 3 seconds ago):
        port3: physical/1000auto, up, rx-bytes/packets/dropped/errors=16302442/43964/0/0,
tx=16053848/40454/0/0
        port5: physical/1000auto, up, rx-bytes/packets/dropped/errors=18161941/54088/0/0,
tx=20615650/55877/0/0
    FGT3HD3914800153(updated 2 seconds ago):
        port3: physical/1000auto, up, rx-bytes/packets/dropped/errors=17033009/46641/0/0,
tx=15907891/40462/0/0
        port5: physical/1000auto, up, rx-bytes/packets/dropped/errors=20617180/55881/0/0,
tx=18163135/54091/0/0
Primary: FortiGate-300D  , FGT3HD3914800069, HA cluster index = 1
Secondary: FortiGate-300D  , FGT3HD3914800153, HA cluster index = 0
number of vcluster: 1
vcluster 1: work 169.254.0.2
Primary: FGT3HD3914800069, HA operating index = 0
Secondary: FGT3HD3914800153, HA operating index = 1
```

Disabling stateful SCTP inspection

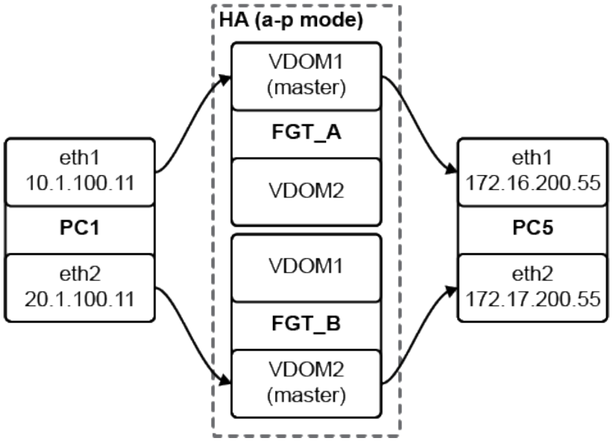
There is an option in FortiOS to disable stateful SCTP inspection. This option is useful when FortiGates are deployed in a high availability (HA) cluster that uses the FortiGate Clustering Protocol (FGCP) and virtual clustering in a multihoming topology. In this configuration, the primary stream control transmission protocol (SCTP) path traverses the primary FortiGate node by using its active VDOM (for example, VDOM1), and the backup SCTP path traverses the other passive FortiGate node by using its active VDOM (for example, VDOM2).

When stateful SCTP inspection is enabled, SCTP heartbeat traffic fails by means of the backup path because the primary path goes through a different platform and VDOM. Since there is no state sharing between VDOMs, the passive FortiGate is unaware of the original SCTP session and drops the heartbeats because of no associated sessions. When stateful SCTP inspection is disabled, the passive node permits the SCTP heartbeats to pass.

When set to `enable`, SCTP session creation without SCTP INIT is enabled. When set to `disable`, SCTP session creation without SCTP INIT is disabled (this is the default setting):

```
config system settings
    set sctp-session-without-init {enable | disable}
end
```

The following is an example topology and scenario:



In this example, FGT_A and FGT_B are in HA a-p mode with two virtual clusters. Two primaries exist on different FortiGate units. PC1 eth1 can access PC5 eth1 through VDOM1, and PC1 eth2 can access PC5 eth2 through VDOM2.

On PC5, to listen for an SCTP connection:

```
sctp_darn -H 172.16.200.55  -B 172.17.200.55 -P 2500 -l
```

On PC1, to start an SCTP connection:

```
sctp_darn -H 10.1.100.11 -B 20.1.100.11 -P 2600 -c 172.16.200.55 -c 172.17.200.55 -p 2500 -s
```

An SCTP four-way handshake is on one VDOM, and a session is created on that VDOM. With the default configuration, there is no session on any other VDOM, and the heartbeat on another path (another VDOM) is dropped. After enabling `sctp-session-without-init`, the other VDOM creates the session when it receives the heartbeat, and the heartbeat is forwarded:

```
config system settings
  set sctp-session-without-init enable
end
```

Resume IPS scanning of ICCP traffic after HA failover

After HA failover occurs, the IPS engine will resume processing ICCP sessions and keep the traffic going on the new primary unit. `session-pickup` must be enabled in an active-passive cluster to pick up the ICCP sessions.

Example

The following example uses an active-passive cluster. See HA active-passive cluster setup on page 2035 for more information.

To configure HA:

```
config system ha
    set group-name "HA-APP"
    set mode a-p
    set password ************
    set hbdev "port3" 100
    set session-pickup enable
    set override enable
end
```

Session states before failover

When HA is working, the ICCP session information is stored in the HA session cache on the secondary FortiGate.

To verify the HA session cache on the secondary FortiGate:

```
# diagnose ips share list
 HA Session Cache
  client=10.1.100.178:57218 server=172.16.200.177:102
    service=39, ignore_app_after=0, last_app=76919, buffer_len=32
    stock tags: nr=981, hash=e68dc8120970448
    custom tags: nr=0, hash=1a49b996b6a42aa2
    tags [count=2]: s-737, s-828,
```

The ICCP session information can be found in the IPS session list and the session table on the primary FortiGate.

To verify the IPS session information on the primary FortiGate:

```
# diagnose ips session list
SESSION id:1 serial:35487 proto:6 group:6 age:134 idle:1 flag:0x800012a6
        feature:0x4 encap:0 ignore:0,0 ignore_after:204800,0
        tunnel:0 children:0 flag:..s.-....-....
  C-10.1.100.178:57218, S-172.16.200.177:102
  state: C-ESTABLISHED/13749/0/0/0/0, S-ESTABLISHED/48951/0/0/0/0 pause:0, paws:0
  expire: 3599
  app: unknown:0 last:44684 unknown-size:0
  cnfm: cotp
  set: cotp
  asm: cotp
```

To verify the system information on the primary FortiGate:

```
# diagnose sys session list
session info: proto=6 proto_state=11 duration=209 expire=3585 timeout=3600 flags=00000000
socktype=0 sockport=0 av_idx=0 use=5
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/255
state=log may_dirty ndr npu syn_ses app_valid
statistic(bytes/packets/allow_err): org=11980/104/1 reply=57028/164/1 tuples=3
tx speed(Bps/kbps): 0/0 rx speed(Bps/kbps): 0/0
orgin->sink: org pre->post, reply pre->post dev=10->9/9->10 gwy=172.16.200.177/10.1.100.178
hook=post dir=org act=snat 10.1.100.178:57218->172.16.200.177:102(172.16.200.4:57218)
hook=pre dir=reply act=dnat 172.16.200.177:102->172.16.200.4:57218(10.1.100.178:57218)
hook=post dir=reply act=noop 172.16.200.177:102->10.1.100.178:57218(0.0.0.0:0)
pos/(before,after) 0/(0,0), 0/(0,0)
misc=0 policy_id=2 auth_info=0 chk_client_info=0 vd=1
serial=00008a9f tos=ff/ff app_list=2003 app=44684 url_cat=0
sdwan_mbr_seq=0 sdwan_service_id=0
rpdb_link_id=00000000 rpdb_svc_id=0 ngfwid=n/a
npu_state=0x003c94 ips_offload
npu info: flag=0x81/0x81, offload=8/8, ips_offload=1/1, epid=71/71, ipid=134/132,
vlan=0x0000/0x0000
vlifid=134/132, vtag_in=0x0000/0x0000 in_npu=1/1, out_npu=1/1, fwd_en=0/0, qid=10/10
```

Sample log on current primary FortiGate:

```
# execute log display
304 logs found.
10 logs returned.
28.8% of logs has been searched.

1: date=2021-06-04 time=16:54:40 eventtime=1622850881110547135 tz="-0700" logid="1059028704"
type="utm" subtype="app-ctrl" eventtype="signature" level="information" vd="vd1" appid=44684
srcip=10.1.100.178 dstip=172.16.200.177 srcport=57218 dstport=102 srcintf="port2"
srcintfrole="undefined" dstintf="port1" dstintfrole="undefined" proto=6 service="tcp/102"
direction="incoming" policyid=2 sessionid=35487 applist="test" action="pass"
appcat="Industrial" app="ICCP_Transfer.Reporting" incidentserialno=61868187 msg="Industrial:
ICCP_Transfer.Reporting," apprisk="elevated"
```

Session states after failover

After HA failover, the IPS engine on the new primary picks up the related ICCP sessions and continues passing the traffic. The HA session cache disappears on the new primary. The ICCP session now appears on the IPS session list and session table on the new primary.

To verify the IPS session information on the new primary FortiGate:

```
# diagnose ips session list
SESSION id:1 serial:35487 proto:6 group:6 age:90 idle:2 flag:0x820012a3
        feature:0x4 encap:0 ignore:1,0 ignore_after:204800,0
        tunnel:0 children:0 flag:....-....-..i.
  C-10.1.100.178:57218, S-172.16.200.177:102
  state: C-ESTABLISHED/9114/0/0/0/0, S-ESTABLISHED/0/0/0/0/0 pause:0, paws:0
  expire: 28
  app: unknown:0 last:44684 unknown-size:0
```

The server and client IPs, ports, and protocols remain the same.

To verify the system information on the primary FortiGate:

```
# diagnose sys session list
session info: proto=6 proto_state=11 duration=569 expire=3577 timeout=3600 flags=00000000
socktype=0 sockport=0 av_idx=0 use=5
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/255
state=log may_dirty ndr npu syn_ses app_valid
statistic(bytes/packets/allow_err): org=38629/308/1 reply=160484/483/1 tuples=3
tx speed(Bps/kbps): 158/1 rx speed(Bps/kbps): 1139/9
orgin->sink: org pre->post, reply pre->post dev=10->9/9->10 gwy=172.16.200.177/10.1.100.178
hook=post dir=org act=snat 10.1.100.178:57218->172.16.200.177:102(172.16.200.4:57218)
hook=pre dir=reply act=dnat 172.16.200.177:102->172.16.200.4:57218(10.1.100.178:57218)
hook=post dir=reply act=noop 172.16.200.177:102->10.1.100.178:57218(0.0.0.0:0)
pos/(before,after) 0/(0,0), 0/(0,0)
misc=0 policy_id=2 auth_info=0 chk_client_info=0 vd=1
serial=00008a9f tos=ff/ff app_list=2003 app=44684 url_cat=0
sdwan_mbr_seq=0 sdwan_service_id=0
rpdb_link_id=00000000 rpdb_svc_id=0 ngfwid=n/a
```

```
npu_state=0x003c94 ips_offload
npu info: flag=0x81/0x81, offload=8/8, ips_offload=1/1, epid=71/71, ipid=134/132,
vlan=0x0000/0x0000
vlifid=134/132, vtag_in=0x0000/0x0000 in_npu=1/1, out_npu=1/1, fwd_en=0/0, qid=10/10
```

The server and client IPs, ports, and NPU state remain the same.

Sample log on new primary FortiGate:

```
# execute log display
653 logs found.
10 logs returned.
65.8% of logs has been searched.

1: date=2021-06-04 time=17:05:20 eventtime=1622851521364635480 tz="-0700" logid="1059028704"
type="utm" subtype="app-ctrl" eventtype="signature" level="information" vd="vd1" appid=44684
srcip=10.1.100.178 dstip=172.16.200.177 srcport=57218 dstport=102 srcintf="port2"
srcintfrole="undefined" dstintf="port1" dstintfrole="undefined" proto=6 service="tcp/102"
direction="incoming" policyid=2 sessionid=35487 applist="test" action="pass"
appcat="Industrial" app="ICCP_Transfer.Reporting" incidentserialno=198181218
msg="Industrial: ICCP_Transfer.Reporting," apprisk="elevated"
```

Querying autoscale clusters for FortiGate VM

When a FortiGate VM secondary device is added to a cluster, the new secondary member can query the cluster about its autoscale environment. FortiManager can then run this query on the new secondary member to update its autoscale record.

To view cluster information from a secondary member:

```
# diagnose sys ha checksum autoscale-cluster
```

Cluster information sample

Sample cloud topology:

```
FGT_BYOL; primary; 10.0.0.6; FGVM04TM00000066
FGT_BYOL; secondary; 10.0.0.7; FGVM00000000056
FGT_PAYG; secondary; 10.0.0.4; FGTAZ000000000CD
FGT_PAYG; secondary; 10.0.0.5; FGTAZ0000000003D
```

From the secondary device, you can see cluster checksums and the primary device:

```
# diagnose sys ha checksum autoscale-cluster
================== FGTAZ000000000CD ==================
is_autoscale_master()=0
debugzone
global: 56 49 b3 02 f2 b7 5b 82 ec 2d c2 1a ff 80 8c 79
root: bf 18 cf 83 1e 04 c3 04 4c e4 66 bc 38 fe 3a dc
all: 77 06 d0 89 6e 06 c0 86 17 98 53 72 33 85 ae ff
checksum
global: 56 49 b3 02 f2 b7 5b 82 ec 2d c2 1a ff 80 8c 79
root: bf 18 cf 83 1e 04 c3 04 4c e4 66 bc 38 fe 3a dc
all: 77 06 d0 89 6e 06 c0 86 17 98 53 72 33 85 ae ff
================== FGVM04TM00000066 ==================
```

```
is_autoscale_master()=1
debugzone
global: 56 49 b3 02 f2 b7 5b 82 ec 2d c2 1a ff 80 8c 79
root: bf 18 cf 83 1e 04 c3 04 4c e4 66 bc 38 fe 3a dc
all: 77 06 d0 89 6e 06 c0 86 17 98 53 72 33 85 ae ff
checksum
global: 56 49 b3 02 f2 b7 5b 82 ec 2d c2 1a ff 80 8c 79
root: bf 18 cf 83 1e 04 c3 04 4c e4 66 bc 38 fe 3a dc
all: 77 06 d0 89 6e 06 c0 86 17 98 53 72 33 85 ae ff
================== FGVM00000000056 ==================
is_autoscale_master()=0
debugzone
global: 56 49 b3 02 f2 b7 5b 82 ec 2d c2 1a ff 80 8c 79
root: bf 18 cf 83 1e 04 c3 04 4c e4 66 bc 38 fe 3a dc
all: 77 06 d0 89 6e 06 c0 86 17 98 53 72 33 85 ae ff
checksum
global: 56 49 b3 02 f2 b7 5b 82 ec 2d c2 1a ff 80 8c 79
root: bf 18 cf 83 1e 04 c3 04 4c e4 66 bc 38 fe 3a dc
all: 77 06 d0 89 6e 06 c0 86 17 98 53 72 33 85 ae ff
================== FGTAZ0000000003D ==================
is_autoscale_master()=0
debugzone
global: 56 49 b3 02 f2 b7 5b 82 ec 2d c2 1a ff 80 8c 79
root: bf 18 cf 83 1e 04 c3 04 4c e4 66 bc 38 fe 3a dc
all: 77 06 d0 89 6e 06 c0 86 17 98 53 72 33 85 ae ff
checksum
global: 56 49 b3 02 f2 b7 5b 82 ec 2d c2 1a ff 80 8c 79
root: bf 18 cf 83 1e 04 c3 04 4c e4 66 bc 38 fe 3a dc
all: 77 06 d0 89 6e 06 c0 86 17 98 53 72 33 85 ae ff
```

To get ha sync information from the secondary device:

```
# get test hasync 50
autoscale_count=69. current_jiffies=41235125
    10.0.0.6, timeo=31430, serial_no=FGVM04TM19001766
    10.0.0.7, timeo=31430, serial_no=FGVM04TM19008156
    10.0.0.5, timeo=31430, serial_no=FGTAZR7UZRKKNR3D
```

Cluster virtual MAC addresses

In a cluster, the FGCP assigns virtual MAC addresses (VMACs) to each primary device interface. HA uses VMAC addresses so that if a failover occurs, the new primary device interfaces will have the same VMAC addresses and IP addresses as the failed primary device. As a result, most network equipment will identify the new primary device as the same device as the failed primary device and still be able to communicate with the cluster.

If a cluster is operating in NAT mode, the FGCP assigns a different VMAC address to each primary device interface. VLAN subinterfaces are assigned the same VMAC address as the physical interface that the VLAN subinterface is added to. Redundant or 802.3ad aggregate interfaces are assigned the VMAC address of the first interface in the redundant or aggregate list.

If a cluster is operating in transparent mode, the FGCP assigns a VMAC address to the primary device's management IP address. Since you can connect to the management IP address from any interface, all FortiGate interfaces appear to have the same VMAC address.

The MAC address of a reserved management interface does not change to a VMAC address; it keeps its original MAC address.

Subordinate device MAC addresses do not change. Use `diagnose hardware deviceinfo nic <interface>` on the subordinate device to display the MAC addresses of each interface.

A MAC address conflict can occur when two clusters are operating on the same network using the same group ID (see Diagnosing packet loss). It is recommended that each cluster in the same network and broadcast domain uses a unique group ID.

Failover

When the new primary device is selected after a failover, the primary device sends gratuitous ARP packets to update the devices connected to the cluster interfaces (usually layer 2 switches) with the VMAC addresses. This is sometimes called using gratuitous ARP packets (or GARP packets) to train the network. The gratuitous ARP packets sent from the primary unit are intended to make sure that the layer 2 switch forwarding databases (FDBs) are updated as quickly as possible.

Sending gratuitous ARP packets is not a requirement because connected devices will eventually learn of the new ports to forward the packets to. However, many network switches will update their FDBs more quickly after a failover if the new primary device sends gratuitous ARP packets.

Configuring ARP packet settings

The following settings can be configured.

```
config system ha
    set arps <integer>
    set arps-interval <integer>
    set gratuitous-arps {enable | disable}
    set link-failed-signal {enable | disable}
end
```

`arps <integer>`	Set the number of gratuitous ARPs; lower the value to reduce traffic, and increase the value to reduce failover time (1 - 60, default = 5).
`arps-interval <integer>`	Set the time between gratuitous ARPs; lower the value to reduce failover time, and increase the value to reduce traffic, in seconds (1 - 20, default = 8).
`gratuitous-arps {enable | disable}`	Enable/disable gratuitous ARPs (default = enable).
`link-failed-signal {enable | disable}`	Enable/disable shutting down all interfaces for one second after a failover. Use if gratuitous ARPs do not update the network (default = disable).

If you disable sending gratuitous ARP packets, it is recommended to enable the `link-failed-signal` setting. The `linked-fail-signal` alerts the connected switches of a failed link, which triggers them to react immediately to the changes.

For more information about gratuitous ARP packets see RFC 826 and RFC 3927.

Determining VMAC addresses

A VMAC address is determined based on following formula:

`<group-prefix>:<group-id_hex>:(<vcluster_integer> + <idx>)`

The `<group-prefix>` is determined by the following set of group IDs:

- Set 1: group IDs 0 - 255: group prefix 00:09:0f:09
- Set 2: group IDs 256 - 511: group prefix e0:23:ff:fc
- Set 3: group IDs 512 - 767: group prefix e0:23:ff:fd
- Set 4: group IDs 768 - 1023: group prefix e0:23:ff:fe

The `<group-id_hex>` is determined by the group ID % 256, converted to hexadecimal. For example:

Group ID	Hexadecimal ID
0: 0 % 256 = 0	00
255: 255 % 256 = 255	ff
256: 256 % 256 = 0	00
511: 511 % 256 = 255	ff
512: 512 % 256 = 0	00
...	...

The `<vcluster_integer>` is 00 for virtual cluster 1, and 20 for virtual cluster 2. If VDOMs are not enabled, HA sets the virtual cluster to 1 and by default all interfaces are in the root VDOM. Including virtual cluster and VDOM factors in the VMAC address formula means that the same formula can be used whether or not VDOMs and virtual clustering are enabled.

The `<idx>` is the index number of the interface. Interfaces are numbered from 0 to x (where x is the number of interfaces). Interfaces are numbered according to their map order. The first interface has an index of 0. The second interface in the list has an index of 1, and so on.

The following table compares the VMAC addresses for interfaces with an unchanged HA group ID (0) with VDOMs not enabled and interfaces when the group ID is changed to 34:

Interface	VMAC address with unchanged group ID (0)	VMAC address with changed group ID (34)
port5	00-09-0f-09-00-0a	00-09-0f-09-22-0a
port6	00-09-0f-09-00-0b	00-09-0f-09-22-0b
port7	00-09-0f-09-00-0c	00-09-0f-09-22-0c
port8	00-09-0f-09-00-0d	00-09-0f-09-22-0d

Using the same interfaces, a cluster with VDOMs is enabled and the group ID changes to 35. The root VDOM contains port5 and port6 (virtual cluster 1), and vdom_1 contains port7 and port8 (virtual cluster 2). The interfaces have the following VMAC addresses:

Interface	VMAC address with group ID 35
port5	00-09-0f-09-23-0a
port6	00-09-0f-09-23-0b
port7	00-09-0f-09-23-2c
port8	00-09-0f-09-23-2d

Displaying VMAC addresses

Each FortiGate physical interface has two MAC addresses: the permanent and current hardware addresses. The permanent hardware address cannot be changed, as it is the actual MAC address of the interface hardware. The current hardware address can be changed, as it is the address seen by the network.

To change the current hardware address on a FortiGate not operating in HA:

```
config system interface
    edit <name>
        set macaddr <address>
    next
end
```

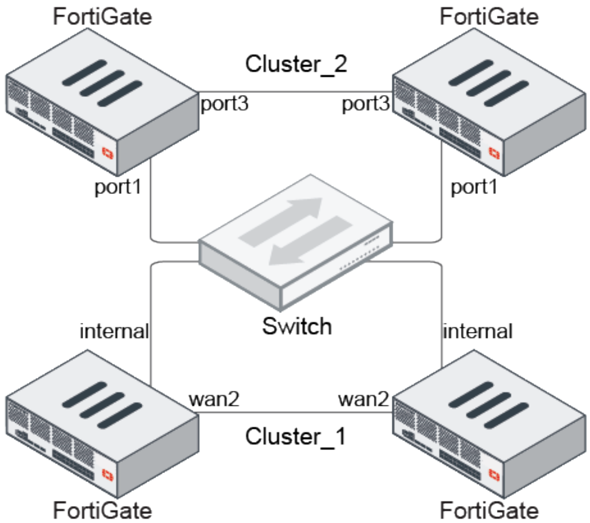
In an operating cluster, the current hardware address of each cluster device interface is changed to the HA virtual MAC address by the FGCP. The `macaddr` option is not available for a functioning cluster.

To display MAC addresses on a FortiGate operating in HA:

```
# diagnose hardware deviceinfo nic port1
...
Current_HWaddr 00:09:0f:09:ff:02
Permanent_HWaddr 08:5b:0e:72:3b:b2
```

Diagnosing packet loss

A network can experience packet loss when two FortiGate HA clusters are deployed in the same broadcast domain due to MAC address conflicts. You can resolve the MAC address conflict by changing the HA group ID (or cluster ID) configuration of the two clusters.



You can diagnose packet loss by pinging from one cluster to the other, or by pinging both of the clusters from a device within the broadcast domain.

To check for a MAC address conflict in a HA cluster:

1. On Cluster_1 and Cluster_2, check the VMAC address (`Current_HWaddr`) used in an interface on the primary device:

   ```
   # diagnose hardware deviceinfo nic <interface>
   ```

   If the group prefix and group hexadecimal ID are identical, there will be MAC address conflicts.
2. Change one of the clusters to use a different group ID:

   ```
   config system ha
       set group-id <integer>
   end
   ```

Abbreviated TLS handshake after HA failover

TLS sessions that pass through an HA A-A or A-P cluster can use an abbreviated TLS handshake instead of a full TLS handshake upon failover from a primary HA unit to a secondary HA unit. This reduces session pickup delays by reducing the time needed to renegotiate the TLS session, given that the TLS session ticket can be re-used.

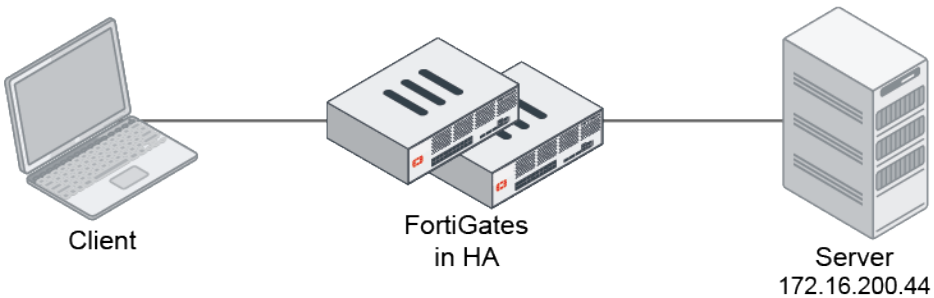
To accomplish this, FortiOS uses the web proxy global `ssl-ca-cert` to generate the key used in the TLS session ticket:

```
config web-proxy global
    set ssl-ca-cert "Fortinet_CA_SSL"
end
```

The certificate can be synchronized to the secondary HA unit, which allows the secondary unit to generate the same session key for a TLS session. When a TLS session reconnects after HA failover using the same session ticket as the first session, the new primary unit is able to generate the same key matching that session ticket and allow an abbreviated handshake.

Example

In this example, OpenSSL is used to create a TLS session between the client and the server through the primary FortiGate. The session ticket is outputted and saved. Upon failover, the same session ticket is reused to create a TLS session through the new primary unit. Because the new primary unit uses the same certificate to generate the key for the TLS session ticket, it allows the connection to be made using an abbreviated TLS handshake.



> This example is for demonstration purposes only. In a normal failover, TLS sessions from clients will automatically be able to re-establish using an abbreviated handshake through the new primary unit.

To verify if an abbreviated TLS handshake is used after HA failover:

1. On the client using OpenSSL, open a new session to 172.16.200.44:443 and output the session ticket to a file called aaa.txt. This session will pass through the current HA primary unit:

   ```
   # openssl s_client -connect 172.16.200.44:443 -sess_out aaa.txt
   ```

2. Fail over the primary unit to the secondary unit. The HA secondary unit starts handling the traffic.
3. On the client, try connecting to 172.16.200.44:443 using the same saved session ticket as before (aaa.txt):

   ```
   # openssl s_client -connect 172.16.200.44:443 -sess_in aaa.txt
   ```

4. Verify whether the session succeeds in using the original session ticket:

   ```
   Reused, TLSv1.3, Cipher is TLS_AES_256_GCM_SHA384
   Server public key is 4096 bit
   Secure Renegotiation IS NOT supported
   Compression: NONE
   Expansion: NONE
   No ALPN negotiated
   Early data was not sent
   Verify return code: 19 (self signed certificate in certificate chain)
   ...
   ```

   If the session is established using the same ticket, Reused, TLSv1.3, Cipher is <name> is displayed. If session is established using a new ticket, New, TLSv1.3, Cipher is <name> is displayed.

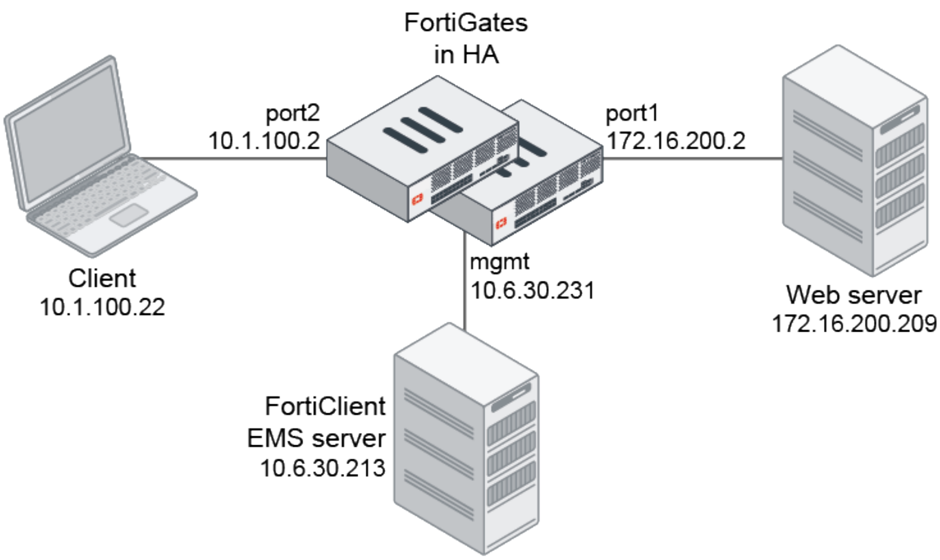
   The new primary is able to use the web proxy global ssl-ca-cert to generate the same key as the old primary that was used in the session ticket. So, the second TLS connection that reuses the TLS session ticket from the first session can complete an abbreviated TLS handshake.

Session synchronization during HA failover for ZTNA proxy sessions

User information and TLS sessions are synchronized between HA members for ZTNA proxy sessions. When a failover occurs, the new primary unit will continue allowing sessions from the logged in users without asking for the client certificate and re-authentication again.

Example

In this example, a FortiGate HA pair is acting as a ZTNA access proxy. Clients that are trying to access the web server on qa.test.com are proxied by the ZTNA access proxy. Remote clients must be registered to the EMS server, and pass a client certificate check and user authentication in order to connect. Upon HA member failure, a failover occurs and the new primary unit will continue to allow connections without requesting client certificate check and user authentication for existing users and devices.



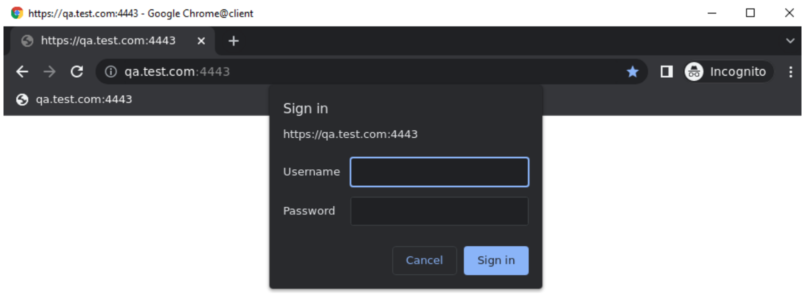
This example assumes ZTNA and EMS server settings are already configured.

To configure the HA settings:

```
config system ha
    set group-name "501E"
    set mode a-p
    set password **********
    set hbdev "ha" 0
    set session-pickup enable
    set override disable
    set monitor "port1" "port2"
end
```

To verify that the proxy sessions are synchronized between HA members:

1. On the client, access the web server. The ZTNA access proxy challenges the user for a client certificate and user authentication.



2. On the primary FortiGate, verify that the user information and TLS sessions are synchronized between HA members.
   a. Verify the list of proxy users:

```
501E-primary # diagnose wad user list
ID: 1, VDOM: root, IPv4: 10.1.100.22
  user name   : localuser1
  worker      : 0
  duration    : 8
  auth_type   : IP
  auth_method : Basic
  pol_id      : 1
  g_id        : 0
  user_based  : 1
  expire      : 597
  LAN:
    bytes_in=2093 bytes_out=5753
  WAN:
    bytes_in=2024 bytes_out=1235
```

   b. Apply a filter to WAD debug to diagnose the wad informer process:

```
501E-primary # diagnose test application wad 2400
Set diagnosis process: type=informer index=0 pid=305
```

   c. Show the user cache from the WAD informer. Verify that the localuser1 entry exists:

```
501E-primary # diagnose test application wad 110
users:
[1]     localuser1@10.1.100.22:0 upn_domain= from:worker worker:6 vf:0 ref:1 stale=0
ntlm:0, has_fsae:0, guest:0
                user_node:(0x7fe18dcf0048) user:1[max=65536](0x7fe18dd08048) ip:1
(0x7fe18dd00048) scheme:0 outofsync:0(0) id:1
...
```

   d. Verify using WAD real-time debugs on the secondary FortiGate. The user information is synchronized to the secondary FortiGate:

```
501E-secondary # diagnose wad debug enable category all
501E-secondary # diagnose wad debug enable level verbose
501E-secondary # diagnose debug enable
[I][p:296]              wad_proc_informer_ha_dgram_on_read:2811  Got HA msg: type=0,
sizeof(msg)=8, dlen=80, sz=88
[I][p:296]              wad_proc_informer_on_ha_user_add   :1493  reader:
ip=10.1.100.22:45852 vf=0 seq=0 grp_type=0 scheme=0 is_ntlm=0 has_fsae=0 concur_
user=65536 domain=''
[I][p:296]              wad_informer_update_user_ext      :782
ip=10.1.100.22:45852 name=localuser1 from=worker
```

```
[I][p:296]              wad_informer_find_user_ip_entries :621   find=false(1) vf=0
ip=10.1.100.22:45852 pr=(nil)
mapping user_node:0x7fc1c84dd048, user_ip:0x7fc1c84ed048(0), user:0x7fc1c84f5048(0).
```

3. Verify the user cache from the WAD informer:

```
501E-secondary # diagnose test application wad 110
users:
[1]     localuser1@10.1.100.22:0 upn_domain= from:worker worker:-126 vf:0 ref:1 stale=0
ntlm:0, has_fsae:0, guest:0
                user_node:(0x7fa3eb07d048) user:1[max=65536](0x7fa3eb095048) ip:1
(0x7fa3eb08d048) scheme:0 outofsync:0(0) id:7
...
```

If the client tries to access the web server again after failover occurs, the client certificate check and authentication prompt does not appear. ZTNA allows the traffic to pass.

The ZTNA logs for both FortiGates contain the same user information.

Primary FortiGate log:

```
1: date=2022-03-23 time=11:49:57 eventtime=1648061397548444970 tz="-0700" logid="0005000024"
type="traffic" subtype="ztna" level="notice" vd="root" srcip=10.1.100.22 srcname="client"
srcport=45826 srcintf="port2" srcintfrole="lan" dstcountry="Reserved" srccountry="Reserved"
dstip=172.16.200.209 dstport=443 dstintf="port1" dstintfrole="lan" sessionid=4786
service="HTTPS" proto=6 action="accept" policyid=1 policytype="proxy-policy"
poluuid="ea7a8a04-a56e-51ec-9d7b-90d24b3a28e9" policyname="ztna" duration=5
user="localuser1" gatewayid=1 vip="ztna" accessproxy="ztna"
clientdeviceid="EF73C831C3FE4FF195A5B2030B******" clientdevicetags="FCTEMS8821000000_all_
registered_clients/MAC_FCTEMS8821000000_all_registered_clients/FCTEMS8821000000_ZT_FILE_
CERTFILE" wanin=2024 rcvdbyte=2024 wanout=1325 lanin=1511 sentbyte=1511 lanout=1075
fctuid="EF73C831C3FE4FF195A5B2030B******" unauthuser="fosqa" unauthusersource="forticlient"
appcat="unscanned"
```

Secondary FortiGate log:

```
1: date=2022-03-23 time=11:55:01 eventtime=1648061701628425041 tz="-0700" logid="0005000024"
type="traffic" subtype="ztna" level="notice" vd="root" srcip=10.1.100.22 srcname="client"
srcport=45830 srcintf="port2" srcintfrole="lan" dstcountry="Reserved" srccountry="Reserved"
dstip=172.16.200.209 dstport=443 dstintf="port1" dstintfrole="lan" sessionid=676
service="HTTPS" proto=6 action="accept" policyid=1 policytype="proxy-policy"
poluuid="ea7a8a04-a56e-51ec-9d7b-90d24b3a28e9" policyname="ztna" duration=5
user="localuser1" gatewayid=1 vip="ztna" accessproxy="ztna"
clientdeviceid="EF73C831C3FE4FF195A5B2030B******" clientdevicetags="FCTEMS8821000000_all_
registered_clients/MAC_FCTEMS8821000000_all_registered_clients/FCTEMS8821000000_ZT_FILE_
CERTFILE" wanin=2024 rcvdbyte=2024 wanout=1325 lanin=1511 sentbyte=1511 lanout=1075
fctuid="EF73C831C3FE4FF195A5B2030B******" unauthuser="fosqa" unauthusersource="forticlient"
appcat="unscanned"
```

Troubleshoot an HA formation

The following are requirements for setting up an HA cluster or FGSP peers.

Cluster members must have:

- The same model.
- The same hardware configuration.

- The same connections.
- The same generation.

The requirement to have the same generation is done as a best practice as it avoids issues that can occur later on. If you are unsure if the FortiGates are from the same generation, please contact customer service.

Troubleshooting common HA formation errors

One member keeps shutting down during HA setup (hard drive failure):

If one member has a hard drive failure but the other does not, the one with the hard drive failure will be shut down during HA setup. In this case, RMA the member to resolve the issue.

Split brain scenario:

A split brain scenario occurs when two or more members of a cluster cannot communicate with each other on the heartbeat interface, causing each member to think it is the primary. As a result, each member assumes the primary HA role and applies the same IP and virtual MAC addresses on its interfaces. This causes IP and MAC conflicts on the network, and causes flapping on L2 devices when they learn the same MAC address on ports connected to different FortiGates.

A split brain scenario is usually caused by a complete lost of the heartbeat link or links. This can be a physical connectivity issue, or less commonly, something blocking the heartbeat packets between the HA members. Another cause is congestion and latency in the heartbeat links that exceeds the heartbeat lost intervals and thresholds.

The following are common symptoms of a split brain scenario:

- The connections to the FortiGates in the cluster work intermittently when trying to connect with administrative access.
- Sessions cannot be established through the FortiGate, and the traffic drops.
- When logging in to the FortiGates using the console, `get system ha status` shows each FortiGate as the primary.

To resolve a split brain scenario:

- Be physically on-site with the FortiGates (recommended). If this is not possible, connect to the FortiGates using console access.
- Identify the heartbeat ports, and verify that they are physically connected and up.
- Verify that heartbeat packets are being sent and received on the heartbeat ports.
- Verify that the HA configurations match between the HA members. The HA `mode`, `group-name`, `group-id`, and `password` settings should be the same. Different `group-id` values will result in different virtual MAC addresses, which might not cause a MAC conflict. However, an IP conflict can still occur.
- If everything seems to be in working order, run `get system ha status` to verify that HA has formed successfully.

To avoid a split brain scenario:

- In a two-member HA configuration, use back-to-back links for heartbeat interface instead of connecting through a switch.
- Use redundant HA heartbeat interfaces.

- In a configuration where members are in different locations, ensure the heartbeat lost intervals and thresholds are longer than the possible latency in the links.

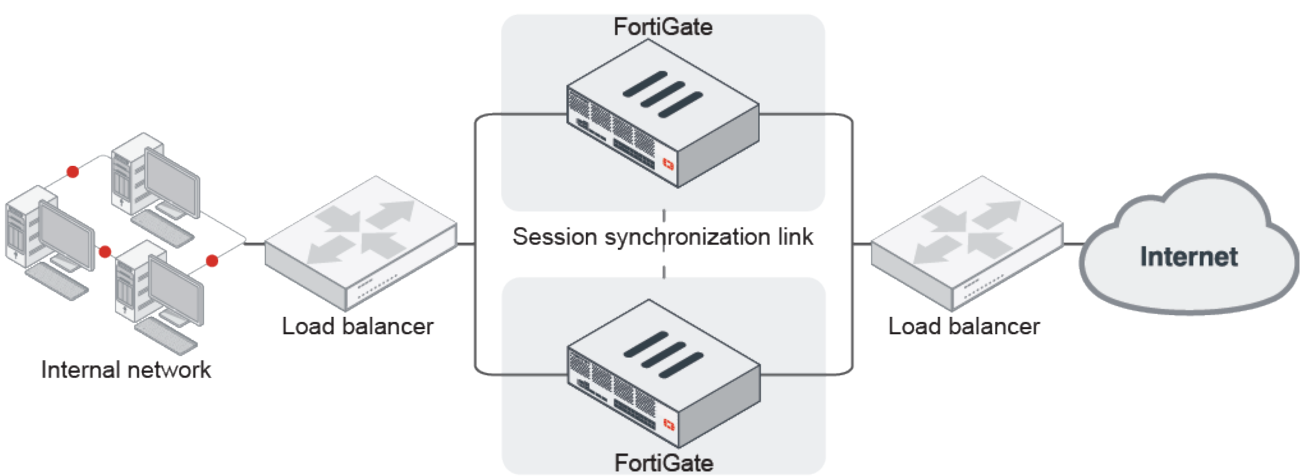
FGSP

Standalone FortiGates or FGCP clusters can be integrated into the load balancing configuration using the FortiGate Session Life Support Protocol (FGSP) in a network where traffic is load balanced by an upstream load balancer and scanned by downstream FortiGates. FGSP can perform session synchronization of IPv4 and IPv6 TCP, SCTP, UDP, ICMP, expectation, and NAT sessions to keep the session tables synchronized on all entities. If one of the FortiGates fails, the upstream load balancer should detect the failed member and stop distributing sessions to it. Session failover occurs and active sessions fail over to the peers that are still operating. Traffic continues to flow on the new peer without data loss because the sessions are synchronized.

The FortiGates in FGSP operate as peers that process traffic and synchronize sessions. An FGSP deployment can include two to 16 standalone FortiGates, or two to 16 FortiGate FGCP clusters of two members each. Adding more FortiGates increases the CPU and memory required to keep all of the FortiGates synchronized, and it increases network synchronization traffic. Exceeding the numbers of members is not recommended and may reduce overall performance. By default, FGSP synchronizes all IPv4 and IPv6 TCP sessions, and IPsec tunnels. You can optionally add filters to control which sessions are synchronized, such as synchronizing packets from specific source and destination addresses, source and destination interfaces, or services.

FGSP is also compatible with FortiGate VRRP.

FGSP is primarily used instead of FGCP when external load balancers are part of the topology, and they are responsible for distributing traffic amongst the downstream FortiGates. FGSP provides the means to synchronize sessions between the FortiGate peers without needing a primary member to distribute the sessions like in FGCP active-active mode. If the external load balancers direct all sessions to one peer, the effect is similar to active-passive FGCP HA. If external load balancers balance traffic to both peers, the effect is similar to active-active FGCP HA. The load balancers should be configured so that all packets for any given session are processed by the same peer, including return packets whenever possible.



Session pickup

Session pickup is an optional setting that can be enabled to synchronize connectionless (UDP and ICMP) sessions, expectation sessions, and NAT sessions. If session pickup is not enabled, the FGSP does not share session tables for the particular session type, and sessions do not resume after a failover. All sessions are interrupted by the failover and must be re-established at the application level. Many protocols can successfully restart sessions with little, or no, loss of data. Others may not recover as easily. Enable session pickup for sessions that may be difficult to reestablish. Since session pickup requires FortiGate memory and CPU resources, only enable this feature for sessions that need to synchronize.

Session synchronization link

The session synchronization link is an optional configuration that allows peers to synchronize sessions over a dedicated interface instead of the interface in which the peer IP is routed. In this configuration, communications occur over L2 instead of L3. Configuring session synchronization links is recommended when you want to minimize traffic over the peering interface when there are many sessions that need to be synchronized.

Expectation sessions

FortiOS session helpers keep track of the communication of layer 7 protocols, such as FTP and SIP, that have control sessions and expectation sessions. The control sessions establish the link between the server and client, and negotiate the ports and protocols that will be used for data communications. The session helpers then create expectation sessions through the FortiGate for the ports and protocols negotiated by the control session.

The expectation sessions are the sessions that actually communicate data. For FTP, the expectation sessions transmit files being uploaded or downloaded. For SIP, the expectation sessions transmit voice and video data. Expectation sessions usually have a timeout value of 30 seconds. If the communication from the server is not initiated within 30 seconds, the expectation session times out and traffic will be denied.

By default, FGSP does not synchronize expectation sessions; if a failover occurs, the sessions will have to be restarted.

To synchronize expectation sessions so they continue after a failover:

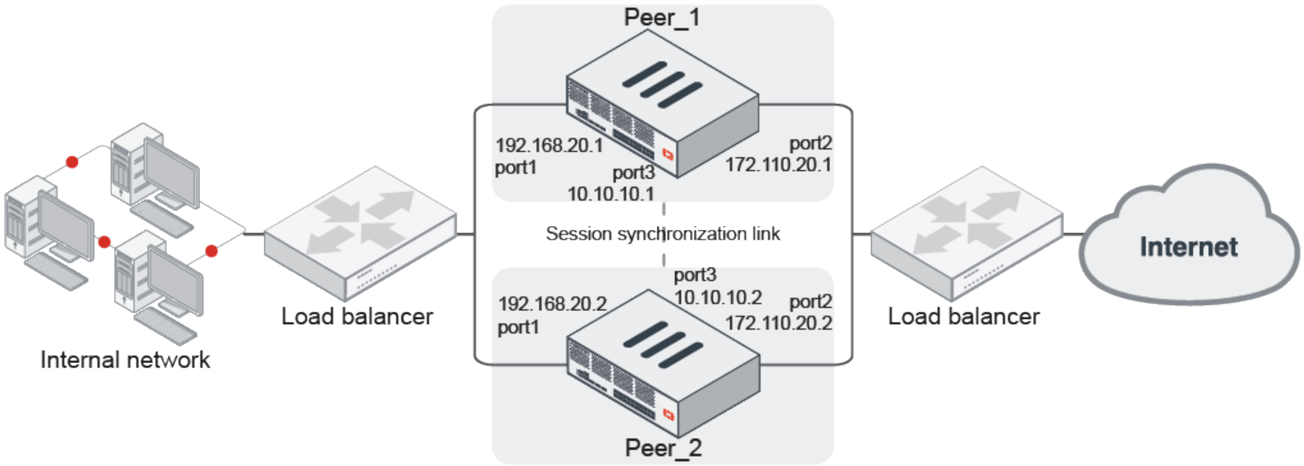
```
config system ha
    set session-pickup enable
    set session-pickup-expectation enable
end
```

The following topics provide more information about FGSP:

- FGSP basic peer setup on page 2088
- Synchronizing sessions between FGCP clusters on page 2091
- Session synchronization interfaces in FGSP on page 2092
- UTM inspection on asymmetric traffic in FGSP on page 2094
- UTM inspection on asymmetric traffic on L3 on page 2096
- Encryption for L3 on asymmetric traffic in FGSP on page 2098
- Optimizing FGSP session synchronization and redundancy on page 2099
- IKE monitor for FGSP on page 2104
- Firmware upgrades in FGSP on page 2106
- FGSP session synchronization between different FortiGate models or firmware versions on page 2106

FGSP basic peer setup

The FortiGate Session Life Support Protocol (FGSP) is a proprietary HA solution for only sharing sessions between entities based on peer-to-peer communications. The entities could be standalone FortiGates or an FGCP cluster. This example uses two peer FortiGates. The load balancer is configured to send all sessions to Peer_1, and if Peer_1 fails, all traffic is sent to Peer_2.



To configure a basic FGSP peer setup:

These instructions assume that all FortiGates have been factory reset.

1. Make all the necessary connections as shown in the topology diagram.
2. On Peer_1, configure the peer IP in which this device will peer with:

```
config system cluster-sync
    edit 1
        set peerip 10.10.10.2
    next
end
```

   If there are multiple peer IPs from the same peer, enter them as separate entries. If there are multiple peers, enter the IP of each peer in separate entries. See Optimizing FGSP session synchronization and redundancy on page 2099 for an example.

   Sessions by default will be synchronized over layer 3 on the interface in which the current unit connects to the peer's IP.
3. On Peer_2, configure session synchronization:

```
config system cluster-sync
    edit 1
        set peerip 10.10.10.1
    next
end
```

4. Configure identical firewall policies on each peer, such as for traffic going from the same incoming interface (port1) to the outgoing interface (port2).

To test the FGSP peer setup:

1. Initiate TCP traffic (like HTTP access) to go through Peer_1.
2. Check the session information:

   ```
   # diagnose sys session filter src <IP_address>
   ```

   ```
   # diagnose sys session list
   ```

3. Enter the same commands on Peer_2 to verify if the same session information appears.

Optional filters

Filters can be added to synchronize certain types of sessions that meet the filter criteria.

To add filters for session synchronization:

```
config system cluster-sync
    edit <id>
        config session-sync-filter
            set srcintf <interface>
            set dstintf <interface>
            set srcaddr <IPv4_address>
            set dstaddr <IPv4_address>
            set srcaddr6 <IPv6_address>
            set dstaddr6 <IPv6_address>
        end
    next
end
```

Filter examples

To synchronize only sessions with a particular source subnet:

```
config system cluster-sync
    edit 1
        config session-sync-filter
            set srcaddr 192.168.20.0/24
        end
    next
end
```

To synchronize only sessions with a particular source address range:

```
config system cluster-sync
    edit 1
        config session-sync-filter
            set srcaddr 192.168.20.10 192.168.20.20
        end
    next
end
```

To synchronize only sessions with a particular destination address range:

```
config system cluster-sync
    edit 1
        config session-sync-filter
            set dstaddr6 2001:db8:0:2::/64
        end
    next
end
```

Session pickup

You can enable this setting to synchronize connectionless (UDP and ICMP) sessions, expectation sessions, and NAT sessions. If session pickup is not enabled, the FGSP does not share session tables for the particular session type, and sessions do not resume after a failover.

To enable UDP and ICMP session synchronization:

```
config system ha
    set session-pickup enable
    set session-pickup-connectionless enable
end
```

Session synchronization

You can specify interfaces used to synchronize sessions in L2 instead of L3 using the `session-sync-dev` setting. For more information about using session synchronization, see Session synchronization interfaces in FGSP on page 2092.

To configure session synchronization over redundant L2 connections:

```
config system standalone-cluster
    set session-sync-dev <interface 1> [<interface 2>] ... [<interface n>]
end
```

VDOM synchronization

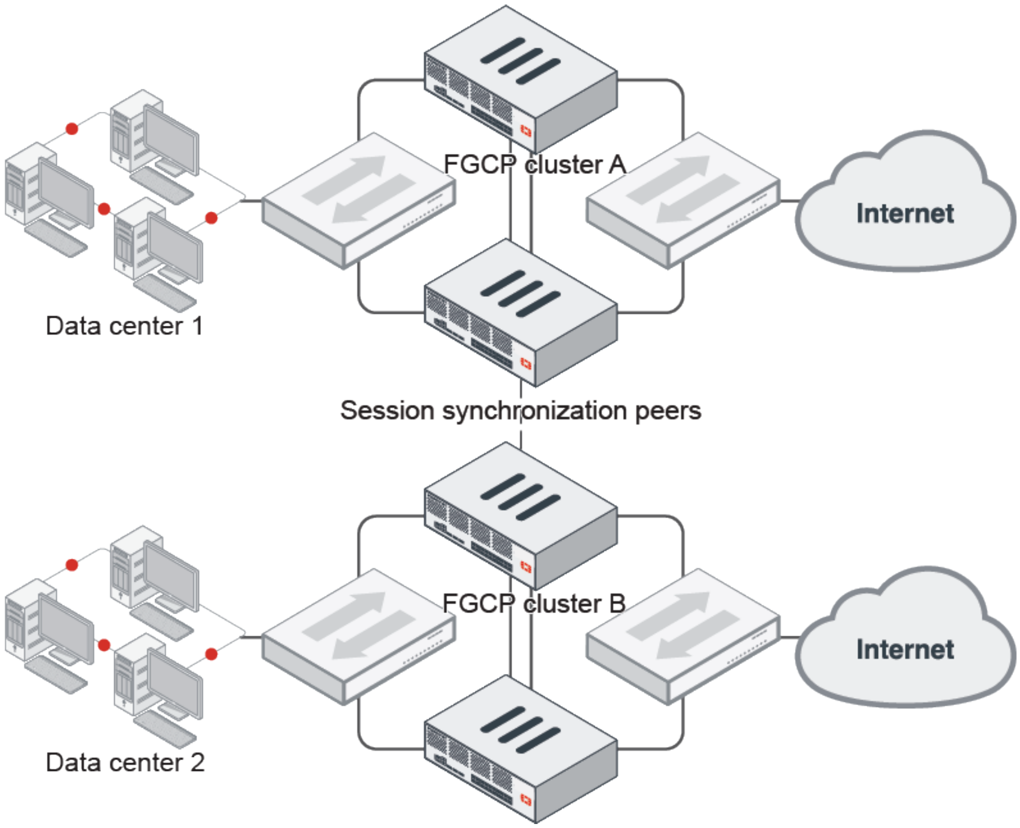
When multi-VDOM mode is enabled, you can specify the peer VDOM and the synchronized VDOMs. The peer VDOM contains the session synchronization link interface on the peer unit. The synchronized VDOMs' sessions are synchronized using this session synchronization configuration.

To synchronize between VDOMs:

```
config system cluster-sync
    edit 1
        set peerip <IP address>
        set peervd <vdom>
        set syncvd <vdom 1> [<vdom 2>] ... [<vdom n>]
    next
end
```

Synchronizing sessions between FGCP clusters

Synchronizing sessions between FGCP clusters is useful when data centers in different locations are used for load balancing, and traffic must be shared and flow freely based on demand.



There are some limitations when synchronizing sessions between FGCP clusters:

- All FortiGates must have the same model and generation, hardware configuration, and FortiOS version.
- A total of 16 clusters can share sessions.

To configure session synchronization between two clusters:

1. Configure the two clusters (see HA active-passive cluster setup on page 2035 or HA active-active cluster setup on page 2037).
2. On cluster A, configure the peer IP for the interface:

```
config system interface
    edit "port5"
        set vdom "root"
        set ip 10.10.10.1 255.255.255.0
        set allowaccess ping https ssh snmp http telnet
    next
end
```

   In this example, cluster A uses port5 and its IP address, 10.10.10.1, is reachable from another cluster.

3. On cluster A, configure cluster and session synchronization:

```
config system cluster-sync
    edit 1
        set peerip 10.10.10.2
    next
end
```

4. On cluster A, configure additional FGSP attributes as needed:

```
config system standalone-cluster
    set standalone-group-id 1
    set group-member-id 0
    set session-sync-dev <interface>
end
```

   The `standalone-group-id` must match between FGSP members. The `group-member-id` is unique for each FGCP cluster. `session-sync-dev` is an optional command to specify the interfaces to sync sessions.

5. On cluster B, configure the peer IP for the interface:

```
config system interface
    edit "port5"
        set vdom "root"
        set ip 10.10.10.2 255.255.255.0
        set allowaccess ping https ssh snmp http telnet
    next
end
```

   In this example, cluster B uses port5 and its IP address, 10.10.10.2, is reachable from another cluster.

6. On cluster B, configure cluster and session synchronization:

```
config system cluster-sync
    edit 1
        set peerip 10.10.10.1
    next
end
```

7. On cluster B, configure additional FGSP attributes as needed:

```
config system standalone-cluster
    set standalone-group-id 1
    set group-member-id 1
    set session-sync-dev <interface>
end
```

Session synchronization interfaces in FGSP

When peering over FGSP, by default, the FortiGates or FGCP clusters share information over L3 between the interfaces that are configured with Peer IP addresses. When a session synchronization interface is configured and FGSP peers are directly connected on this interface, then session synchronization is done over L2, only falling back to L3 if the session synchronization interface becomes unavailable.

When using a session synchronization interface, the synchronization process is offloaded to the kernel. A fast, dedicated, and stable L2 connection should be used for the session synchronization interface between the FGSP peers. For redundancy, multiple synchronization interfaces can be configured.

To provide full redundancy, FGCP clusters can be used in FGSP peering. This is called FGCP over FGSP.

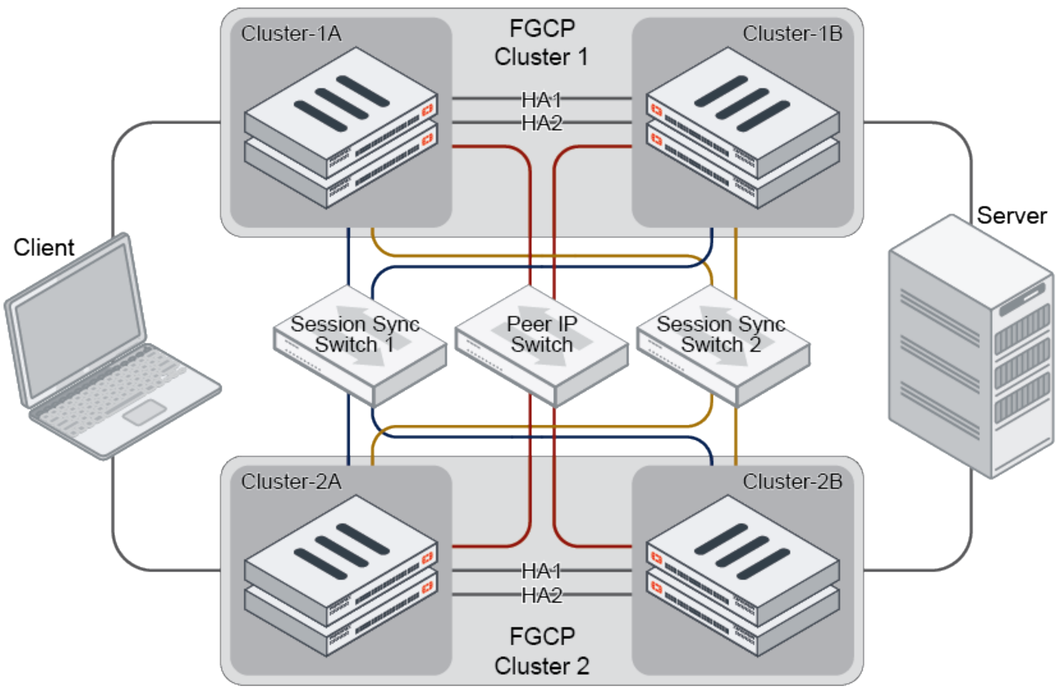
To configure session-sync interfaces:

```
config system standalone-cluster
    set session-sync-dev <interface 1> [<interface 2>] ... [<interface n>]
    set layer2-connection {available | unavailable}
    set encryption {enable | disable}
end
```

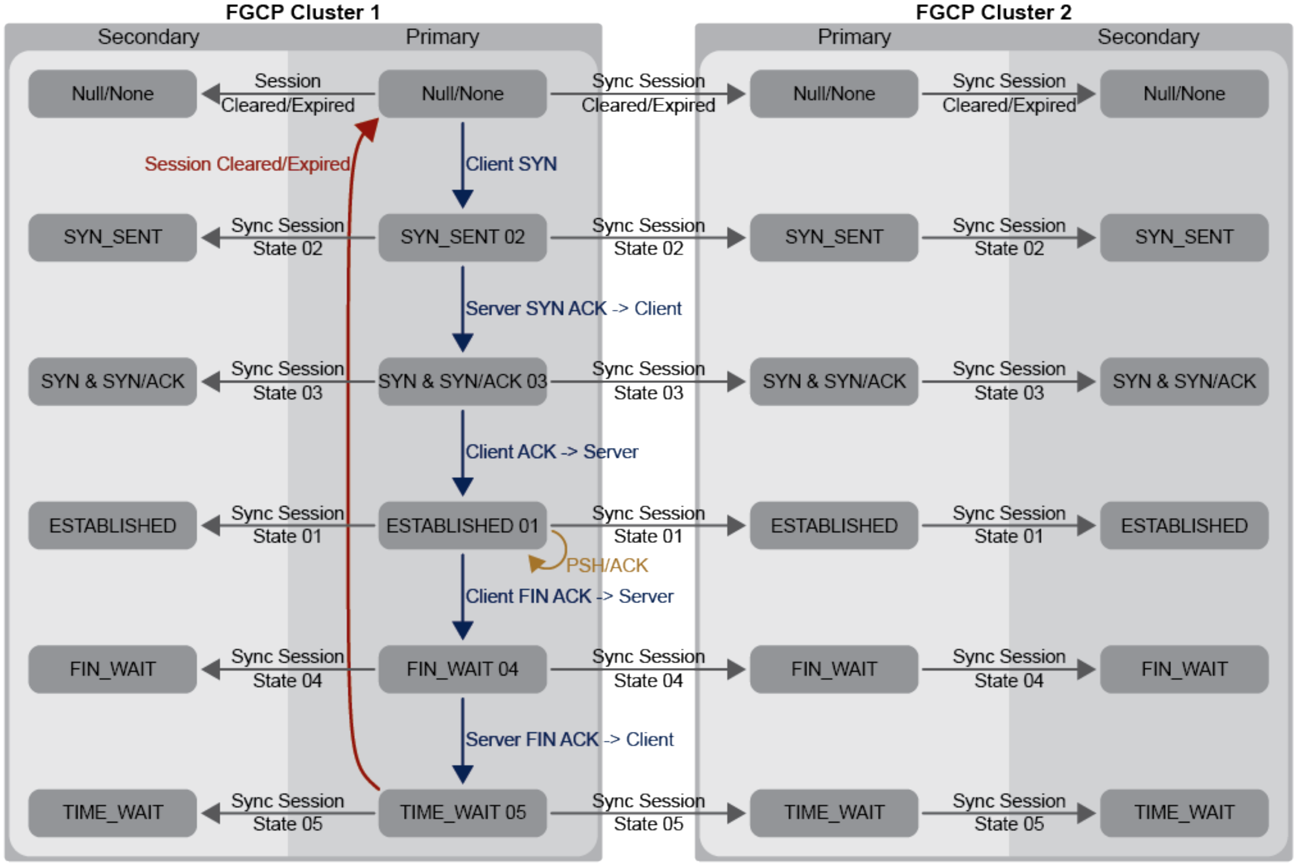
The `layer2-connection` setting is for forwarded traffic between FGSP peers. Set it to `available` if the peer interface user for traffic forwarding is directly connected and supports L2 forwarding. See UTM inspection on asymmetric traffic in FGSP on page 2094 for more information.

Session synchronization in FGCP over FGSP

The following topology uses multiple session synchronization interfaces with a full mesh backbone to prevent any single point of failure.



The state diagram summarizes the session synchronization of a TCP session. It assumes that the session is connected over FGCP Cluster 1 and processed entirely by the primary unit, Cluster-1A.



1. The session starts with the Client SYN packet.
2. As the session is established, Cluster-1A synchronizes the session with Cluster-1B over the heartbeat interface, and with Cluster-2A over the session synchronization interface.
3. Cluster-2A then synchronizes the session with Cluster-2B over its heartbeat interface.
4. The process then repeats as it transitions to different states.

Session synchronization if links fail

In the previous topology, if any single session synchronization link fails on the primary member of each cluster, session synchronization will continue on the second link from the pair of session of session synchronization interfaces.

If the second link on the primary member of the same cluster then fails, L2 session synchronization over the session synchronization interface stops, and synchronization fails over to L3 between the peer IP links.

If the Peer IP link then fails, the FGSP peers are effectively disconnected, and no session synchronization will occur.

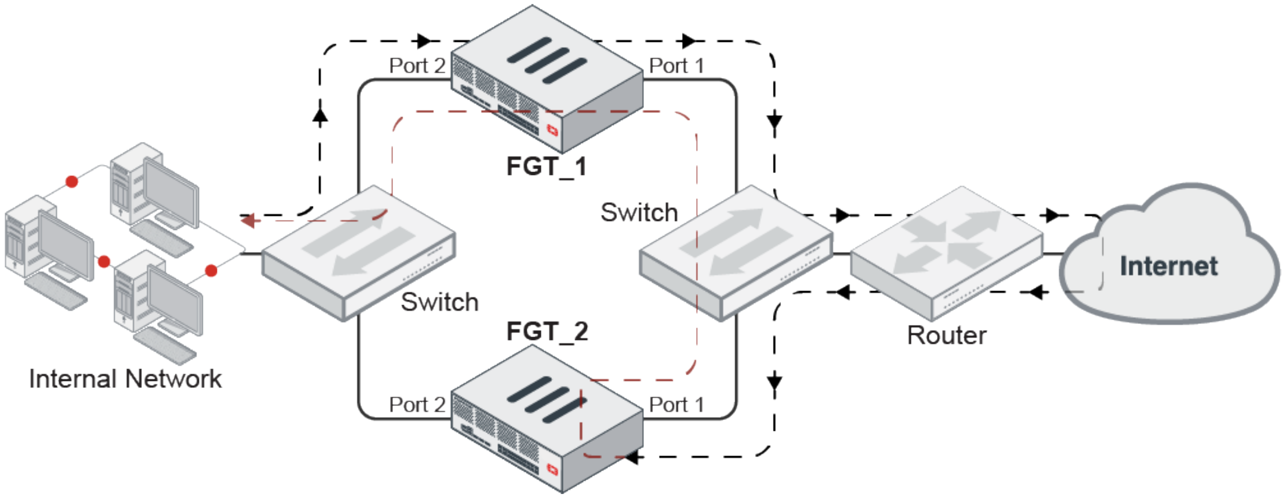
UTM inspection on asymmetric traffic in FGSP

When traffic passes asymmetrically through FGSP peers, UTM inspection can be supported by always forwarding traffic back to the session owner for processing. The session owner is the FortiGate that receives the first packet of the session.

In this example, traffic from the internal network first hits FGT_1, but the return traffic is routed to FGT_2. Consequently, traffic bounces from FGT_2 port1 to FGT_1 port1 using FGT_1’s MAC address. Traffic is then inspected by FGT_1.

This example requires the following settings:

- The internal and outgoing interfaces of both FortiGates in the FGSP pair are in the same subnet.
- Both peers have layer 2 access with each other.



To configure FTG_1:

1. Configure the cluster, setting the peer IP to the IP address of FGT_2:

```
config system cluster-sync
    edit 1
        set peerip 10.2.2.2
    next
end
```

2. Configure FGSP cluster attributes:

```
config system standalone-cluster
    set standalone-group-id 1
    set group-member-id 0
    set layer2-connection available
    unset session-sync-dev
end
```

3. Configure the firewall policy:

```
config firewall policy
    edit 1
        set srcintf "port2"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set av-profile "default"
        set logtraffic all
        set nat enable
```

```
    next
end
```

To configure FTG_2:

1. Configure the cluster, setting the peer IP to the IP address of FGT_1:

```
config system cluster-sync
    edit 1
        set peerip 10.2.2.1
    next
end
```

2. Configure FGSP cluster attributes:

```
config system standalone-cluster
    set standalone-group-id 1
    set group-member-id 1
    set layer2-connection available
    unset session-sync-dev
end
```

3. Configure the firewall policy:

```
config firewall policy
    edit 1
        set srcintf "port2"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set av-profile "default"
        set logtraffic all
        set nat enable
    next
end
```

Results

Capture packets on FGT_2 to see that traffic bounced from FGT_2 to FGT_1 over the traffic interface.

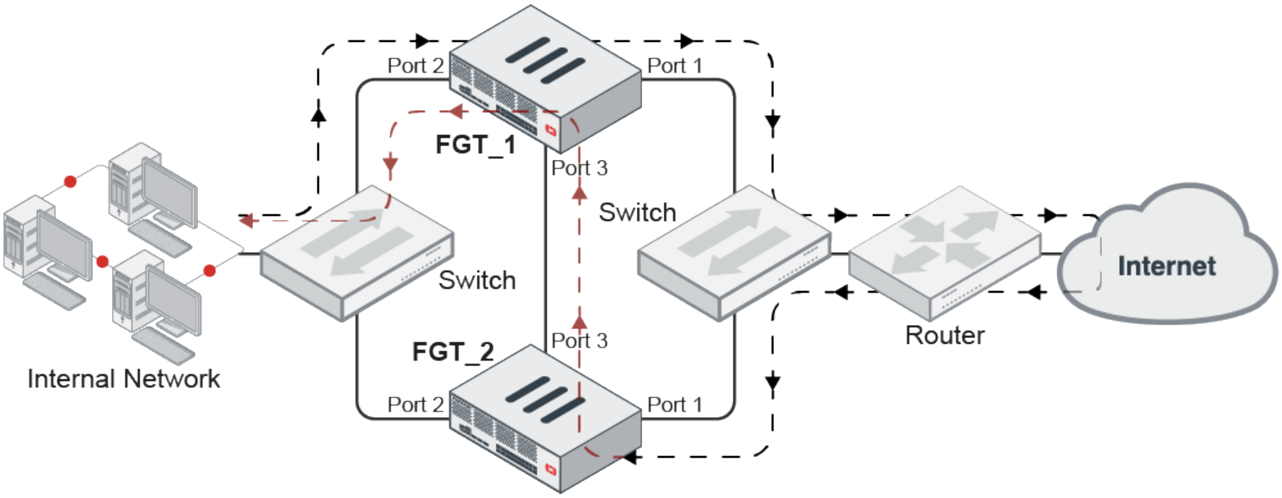
```
FGT_2 # diagnose sniffer packet any 'host 10.1.100.15 and host 172.6.200.55' 4
interfaces=[any]
filters=[host 10.1.100.15 and host 172.16.200.55]
91.803816 port1 in 172.16.200.55.80 -> 10.1.100.15.40008: syn 2572073713 ack 261949279
92.800480 port1 in 172.16.200.55.80 -> 10.1.100.15.40008: syn 2572073713 ack 261949279
92.800486 port1 out 172.16.200.55.80 -> 10.1.100.15.40008: syn 2572073713 ack 261949279
92.800816 port1 in 172.16.200.55.80 -> 10.1.100.15.40008: syn 2572073713 ack 261949279
92.800818 port1 out 172.16.200.55.80 -> 10.1.100.15.40008: syn 2572073713 ack 261949279
```

UTM inspection on asymmetric traffic on L3

When traffic passes asymmetrically through FGSP peers, UTM inspection can be supported by always forwarding traffic back to the session owner for processing. The session owner is the FortiGate that receives the first packet of the session.

For networks where L2 connectivity is not available, such as cloud environments, traffic bound for the session owner are forwarded through the peer interface using a UDP connection.

In this example, traffic from the internal network first hits FGT_1, but the return traffic is routed to FGT_2. Consequently, return traffic is packed and sent from FGT_2 to FGT_1 using UDP encapsulation between two peer interfaces (port 3). Traffic is then inspected by FGT_1.



To configure FTG_1:

1. Configure the cluster, setting the peer IP to the IP address of FGT_2:

```
config system cluster-sync
    edit 1
        set peerip 10.2.2.2
    next
end
```

2. Configure FGSP cluster attributes:

```
config system standalone-cluster
    set standalone-group-id 1
    set group-member-id 0
    set layer2-connection unavailable
    unset session-sync-dev
end
```

3. Configure the firewall policy:

```
config firewall policy
    edit 1
        set srcintf "port2"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set av-profile "default"
```

```
        set logtraffic all
        set nat enable
    next
end
```

To configure FTG_2:

1. Configure the cluster, setting the peer IP to the IP address of FGT_1:

```
config system cluster-sync
    edit 1
        set peerip 10.2.2.1
    next
end
```

2. Configure FGSP cluster attributes:

```
config system standalone-cluster
    set standalone-group-id 1
    set group-member-id 1
    set layer2-connection unavailable
    unset session-sync-dev
end
```

3. Configure the firewall policy:

```
config firewall policy
    edit 1
        set srcintf "port2"
        set dstintf "port1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set av-profile "default"
        set logtraffic all
        set nat enable
    next
end
```

Encryption for L3 on asymmetric traffic in FGSP

In scenarios where asymmetric routing between FGSP members occurs, the return traffic can be encrypted and routed back to the session owner on Layer 3 (L3).

To encrypt L3 traffic in FGSP:

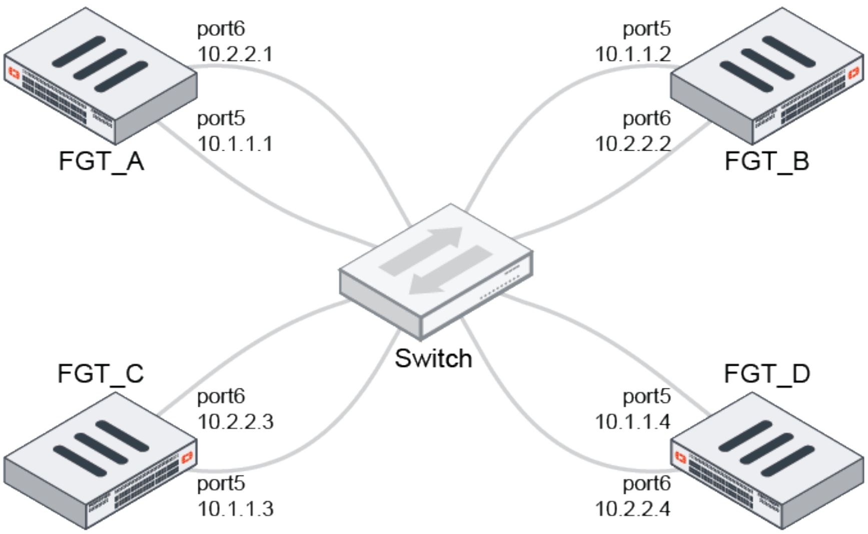
1. Run the following on both FortiGates:

```
config system standalone-cluster
    set encryption enable
    set psksecret xxxxxxxxx
end
```

Optimizing FGSP session synchronization and redundancy

By using `session-sync-dev` to offload session synchronization processing to the kernel, FGSP session synchronization can be supported to handle heavy loads.

Topology



In this topology, there are three FGSP peer groups for each FortiGate. Sessions are synchronized between each FortiGate and its peer groups. Redundancy is achieved by using two dedicated session sync device links for each peer setup. There are a total of six peer IPs for each session synchronization device link in each FGSP peer. When one link is fails, session synchronization is not affected.

For optimization, `sync-packet-balance` is enabled to distribute synchronization packets processing to multiple CPUs. The session synchronization process is offloaded to the kernel, and sessions are synchronized over layer 2 over the connected interfaces (`set session-sync-dev "port5" "port6"`). Jumbo frame MTU 9216 is configured on each session synchronization device link to reduce the number of packets; however, setting MTU to 9216 is entirely optional.

To configure FGT_A:

1. Configure HA:

```
config system ha
    set sync-packet-balance enable
    set session-pickup enable
    set session-pickup-connectionless enable
    set session-pickup-expectation enable
    set session-pickup-nat enable
end
```

2. Configure the layer 2 session synchronization links:

```
config system standalone-cluster
    set session-sync-dev "port5" "port6"
end
```

3. Configure the session TTL default timeout:

```
config system session-ttl
    set default 300
end
```

4. Configure the interfaces:

```
config system interface
    edit port5
        set ip 10.1.1.1/24
        set mtu-override enable
        set mtu 9216
    next
    edit port6
        set ip 10.2.2.1/24
        set mtu-override enable
        set mtu 9216
    next
end
```

5. Configure FGSP session synchronization:

```
config system cluster-sync
    edit 1
        set peerip 10.1.1.2
    next
    edit 2
        set peerip 10.2.2.2
    next
    edit 3
        set peerip 10.1.1.3
    next
    edit 4
        set peerip 10.2.2.3
    next
    edit 5
        set peerip 10.1.1.4
    next
    edit 6
        set peerip 10.2.2.4
    next
end
```

To configure FGT_B:

1. Configure HA:

```
config system ha
    set sync-packet-balance enable
    set session-pickup enable
    set session-pickup-connectionless enable
    set session-pickup-expectation enable
    set session-pickup-nat enable
end
```

2. Configure the layer 2 session synchronization links:

```
config system standalone-cluster
    set session-sync-dev "port5" "port6"
end
```

3. Configure the session TTL default timeout:

```
config system session-ttl
    set default 300
end
```

4. Configure the interfaces:

```
config system interface
    edit port5
        set ip 10.1.1.2/24
        set mtu-override enable
        set mtu 9216
    next
    edit port6
        set ip 10.2.2.2/24
        set mtu-override enable
        set mtu 9216
    next
end
```

5. Configure FGSP session synchronization:

```
config system cluster-sync
    edit 1
        set peerip 10.1.1.1
    next
    edit 2
        set peerip 10.2.2.1
    next
    edit 3
        set peerip 10.1.1.3
    next
    edit 4
        set peerip 10.2.2.3
    next
    edit 5
        set peerip 10.1.1.4
    next
    edit 6
        set peerip 10.2.2.4
    next
end
```

To configure FGT_C:

1. Configure HA:

```
config system ha
    set sync-packet-balance enable
    set session-pickup enable
    set session-pickup-connectionless enable
    set session-pickup-expectation enable
```

```
    set session-pickup-nat enable
end
```

2. Configure the layer 2 session synchronization links:

```
config system standalone-cluster
    set session-sync-dev "port5" "port6"
end
```

3. Configure the session TTL default timeout:

```
config system session-ttl
    set default 300
end
```

4. Configure the interfaces:

```
config system interface
    edit port5
        set ip 10.1.1.3/24
        set mtu-override enable
    set mtu 9216
    next
    edit port6
        set ip 10.2.2.3/24
        set mtu-override enable
        set mtu 9216
    next
end
```

5. Configure FGSP session synchronization:

```
config system cluster-sync
    edit 1
        set peerip 10.1.1.1
    next
    edit 2
        set peerip 10.2.2.1
    next
    edit 3
        set peerip 10.1.1.2
    next
    edit 4
        set peerip 10.2.2.2
    next
    edit 5
        set peerip 10.1.1.4
    next
    edit 6
        set peerip 10.2.2.4
    next
end
```

To configure FGT_D:

1. Configure HA:

```
config system ha
    set sync-packet-balance enable
```

```
    set session-pickup enable
    set session-pickup-connectionless enable
    set session-pickup-expectation enable
    set session-pickup-nat enable
end
```

2. Configure the layer 2 session synchronization links:

```
config system standalone-cluster
    set session-sync-dev "port5" "port6"
end
```

3. Configure the session TTL default timeout:

```
config system session-ttl
    set default 300
end
```

4. Configure the interfaces:

```
config system interface
    edit port5
        set ip 10.1.1.4/24
        set mtu-override enable
        set mtu 9216
    next
    edit port6
        set ip 10.2.2.4/24
        set mtu-override enable
        set mtu 9216
    next
end
```

5. Configure FGSP session synchronization:

```
config system cluster-sync
    edit 1
        set peerip 10.1.1.1
    next
    edit 2
        set peerip 10.2.2.1
    next
    edit 3
        set peerip 10.1.1.2
    next
    edit 4
        set peerip 10.2.2.2
    next
    edit 5
        set peerip 10.1.1.3
    next
    edit 6
        set peerip 10.2.2.3
    next
end
```

IKE monitor for FGSP

Split-brain situations occur in a scenario where session synchronization is down between two FGSP peers. This can have an effect if IKE fails over from one unit to another, causing the tunnel to be invalid due to the IKE session and role being out of sync, and ESP anti-replay detection. In split-brain situations, the IKE monitor provides a mechanism to maintain the integrity of the state tables and primary/secondary roles for each VPN gateway. It continues to provide fault tolerance by keeping track of the timestamp of the latest received traffic, and it uses the ESP sequence number jump ahead value to preserve the sequence number per gateway. Once the link is up, the cluster resolves the role and synchronizes the session and IKE data. During this process, if the IKE fails over from one unit to another, the tunnel will remain valid and traffic continues to flow.

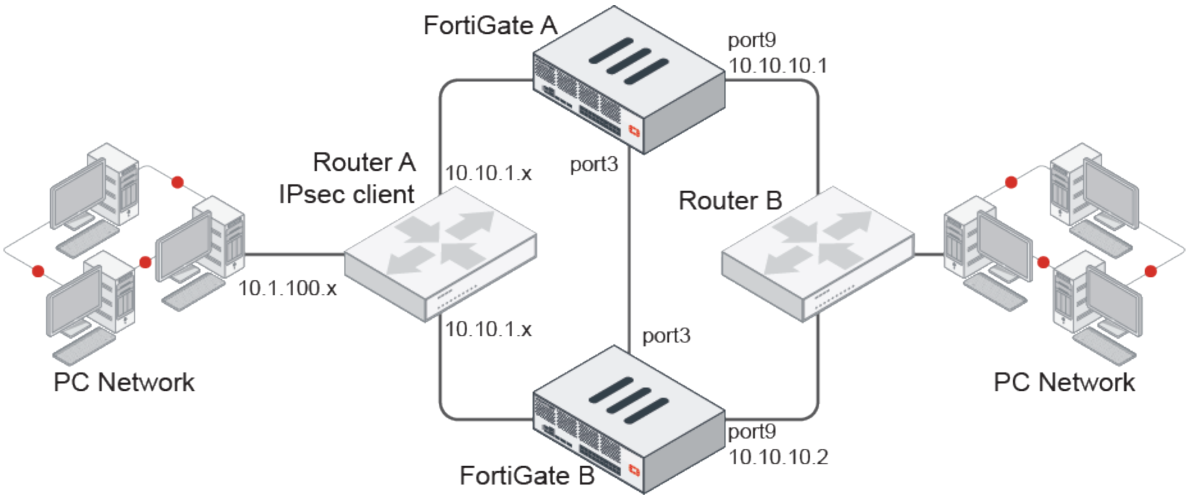
The IKE monitor only works with 2 peers in FGSP.

To configure the IKE monitor:

```
config system cluster-sync
    edit <id>
        set peerip <address>
        set ike-monitor {enable | disable}
        set ike-monitor-interval <integer>
        set ike-heartbeat-interval <integer>
        set ike-seqjump-speed <integer>
    next
end
```

`ike-monitor {enable | disable}`	Enable/disable IKE HA monitor (default = disable).
`ike-monitor-interval <integer>`	Set the monitoring interval for determining how fast the cluster members detect split-brain mode, in seconds (10 - 300, default = 15).
`ike-heartbeat-interval <integer>`	Set the heartbeat message interval for sending the heartbeat per gateway to the other peers, in seconds (1 - 60, default = 3).
`ike-seqjump-speed <integer>`	Set the ESP jump ahead factor, in packets per second equivalent (1 - 10, default = 10). A value of 10 means it is the factor for a 10G interface.

Example



In this example, FortiGate A and FortiGate B are FGSP peers with port3 as the session synchronization link. The FortiGates act as IPsec dial-up servers and PCs on the 10.1.100.0 subnet are the IPsec dial-up clients. Router A acts as the external load balancer for IKE sessions between the FortiGates. Dynamic routing OSPF is configured for the FortiGates and routers.

When PC2 and other clients form IPsec dial-up tunnels to the FGSP peers, these tunnels terminate on either FortiGate A or FortiGate B, not both. For each tunnel, one FortiGate is the primary and the other is the secondary.

When the session synchronization link goes down, the FGSP split-brain scenario occurs. Without using the IKE monitor mechanism, the IKE and ESP information becomes out of sync between the two FortiGates. The secondary FortiGate for a tunnel does not receive any information about updated tunnel status. If there is a failover and tunnel traffic begins to flow to the secondary FortiGate, the tunnel will be invalidated because its state tables for that session are out of sync.

By using the IKE monitor when a split-brain scenario occurs, each unit starts periodically monitoring traffic flows and managing the sequence number jump ahead on standby units. Using a combination of timers with ESP sequence number jump ahead lets the units maintain integrity of the shared SA runtime state table, including ESP anti-replay sequence numbers.

Once the session synchronization link is up, the FGSP peers synchronize the state tables and resume regular operations.

To configure the IKE monitor:

```
config system cluster-sync
    edit 1
        set peerip 10.10.10.2
        set ike-monitor enable
        set ike-monitor-interval 12
        set ike-heartbeat-interval 2
        set ike-seqjump-speed 2
    next
end
```

Firmware upgrades in FGSP

The following steps are recommended to upgrade the firmware of FortiGates in an FGSP deployment. Follow these steps whether or not you have enabled standalone configuration synchronization.

This example FGSP deployment has two FortiGates, FGT-1 and FGT-2.

To upgrade the firmware in an FGSP deployment:

1. Switch all traffic to FGT-1:
   a. Configure the load balancer or router that distributes traffic between the FortiGates to send all traffic to FGT-1.
2. Disconnect FGT-2 from the network.
   Make sure to also disconnect the interfaces that allow heartbeat and synchronization communication with FGT-1. This is to prevent FGT-2 from communicating with FGT-1.
3. Upgrade the firmware on FGT-2.
4. Reconnect the traffic interfaces on FGT-2, but not the interfaces used for heartbeat and synchronization communication with FGT-1.
5. Switch all traffic to the newly upgraded FGT-2:
   a. Configure the load balancer or router that distributes traffic between the FortiGates to send all traffic to FGT-2.
6. Upgrade the firmware on FGT-1 (while heartbeat and synchronization communication with FGT-2 remains disconnected).
7. Reconnect the FGT-2 interfaces that allow heartbeat and synchronization communication between FGT-1 and FGT-2.
8. Restore the original traffic distribution between FGT-1 and FGT-2:
   a. Configure the load balancer or router to distribute traffic to both FortiGates in the FGSP deployment.

FGSP session synchronization between different FortiGate models or firmware versions

FGSP HA deployments are generally meant for interoperating between FortiGates with the same model and firmware version. However, situations may arise where individual members or FGCP clusters running over FGSP use different models or firmware versions. For example, to avoid downtime while upgrading the members, some FGSP members or clusters may be upgraded first and then re-join the FGSP peers after a successful upgrade. Or while performing maintenance, sessions may need to be offloaded to a temporary member or FGCP cluster of a different model.

Being able to perform FGSP session synchronization between members of different models or firmware versions is helpful to transition the traffic smoothly and causes minimal disruptions. This topic outlines requirements to be aware of before assessing whether FGSP session synchronization may work between members with different models or firmware versions.

Different FortiGate models

The general guideline is to only use FortiGate models in a similar tier and family. Vastly different models have different performance and capabilities, which may not be compatible. The goal is for two models to have similar capabilities so that data structures used in session synchronization will match, and are capable of delivering similar performance.

When considering FGSP session synchronization between two FortiGates, ensure that:

- The FortiGates use the same 32-bit kernel or 64-bit kernel.
- The FortiGates use the same type of CPU (such as ARM or x86).

- For network interfaces:
  - The same type of physical interface should be used on each member.
  - The physical interfaces should be capable of the same speeds.
- The device memory should be similar in size. If the FortiGates have vastly different memory sizes, their performance may be different if one device supports more sessions than the other.
- The configurations related to session tables should match. For example, the logical names used in firewall policies, IPsec interface names, VDOM names, firewall policy tables, and so on.

Virtual clusters and asymmetric routing are not supported.

Different firmware versions

When operating in FGSP, the firmware needs to have compatible data structures and session synchronization packet headers. The firmware is generally able to handle different data structures between old and new FortiOS sessions. Session synchronization packets are typically the same between versions.

Note the following exceptions and guidelines when assessing FGSP session synchronization compatibility between different firmware versions:

- FortiOS 7.0.2 added support for widening the HA virtual MAC address range. This change updated the session synchronization packet header structure.
  - FortiGates running 7.0.2 or later, and FortiGates running 7.0.1 or earlier will not accept session synchronization packets from each other.
- If the traffic uses a new feature only available in a newer FortiOS version, it may not work when synchronized to an older FortiOS version.
  - For example, PFCP (Packet Forwarding Control Protocol) support was added in 7.0.1, and a PFCP profile name was added to the sessions. When the sessions are synchronized to an older firmware version, the PFCP profile name will be lost and the sessions will not be able to handle the traffic as they would in 7.0.1.

Session synchronization interfaces

Session synchronization between FGSP members uses an L3 connection over the peer IP by default.

Session synchronization between FGSP members uses an L2 connection when a session synchronization interface (`session-sync-dev`) is used. The synchronization process is also offloaded to the kernel.

FGSP is also compatible with FortiGate VRRP.

Standalone configuration synchronization

You can configure synchronization from one standalone FortiGate to another standalone FortiGate (`standalone-config-sync`). With the exception of some configurations that do not sync (settings that identify the FortiGate to the network), the rest of the configurations are synced, such as firewall policies, firewall addresses, and UTM profiles.

This option is useful in situations when you need to set up FGSP peers, or when you want to quickly deploy several FortiGates with the same configurations. You can set up `standalone-config-sync` for multiple members.

> `standalone-config-sync` is an independent feature and should be used with caution as there are some limitations. We recommend disabling it once the configurations have been synced over.

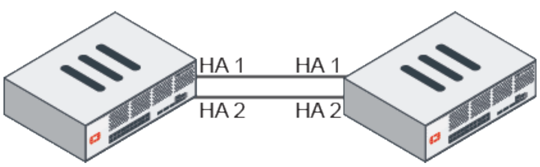
Limitations

When standalone configuration synchronization is enabled, there are some limitations, including but not limited to the following:

- Network interruptions occur during firmware upgrades: when upgrading the firmware, all members in the `standalone-config-sync` group are upgraded simultaneously. This creates downtime if the FortiGates are the only outgoing gateway in the network. We recommend disabling the option before upgrading firmware.
- Some unwanted configurations might be synced: the current design and implementation of `standalone-config-sync` is based on requirements from specific customers. Thus, some users may find that unwanted parts of the configurations are synced. Should this occur, we recommend disabling the option and modifying those configurations manually.
- The wrong primary device might be selected accidentally: `standalone-config-sync` is derived from the HA primary unit selection mechanism. All members in the group will join the selection process in the same way as a the HA cluster selection process. It is important to select the correct device as the primary, otherwise the wrong device could be selected and existing configurations could be overwritten.

Setting up standalone configuration synchronization

Two or more standalone FortiGates should be connected to each other with one or more heartbeat interfaces, either back-to-back or via a switch. In the following example, the device supplying the configurations is called "conf-prim," and the devices receiving the configurations are called "conf-secos."



To set up standalone configuration synchronization:

1. Configure the conf-prim device for the group:

```
config system ha
    set hbdev ha1 50 ha2 100
    set priority 255
    set override enable
    set standalone-config-sync enable
end
```

2. Configure the conf-prim device as needed to be functional.
3. Configure the other group members as conf-secos:

```
config system ha
    set standalone-config-sync enable
end
```

4. Wait 10–15 minutes for the configurations to sync over.
5. Verify the synchronization status:

```
# get system ha status
path=system, objname=ha, tablename=(null), size=5912
HA Health Status:
  WARNING: FG201E4Q17900771 has hbdev down;
  WARNING: FG201ETK19900991 has hbdev down;
Model: FortiGate-201E
Mode: ConfigSync
Group: 0
Debug: 0
Cluster Uptime: 0 days 0:0:51
Cluster state change time: 2019-09-03 17:46:07
Primary selected using:
  <2019/09/03 17:46:07> FG201ETK19900991 is selected as the primary because it has the
largest value of override priority.
ses_pickup: disable
override: disable
Configuration Status:
  FG201E4Q17900771(updated 3 seconds ago): out-of-sync
  FG201ETK19900991(updated 1 seconds ago): in-sync
System Usage stats:
  FG201E4Q17900771(updated 3 seconds ago):
    sessions=1, average-cpu-user/nice/system/idle=0%/0%/0%/100%, memory=16%
  FG201ETK19900991(updated 1 seconds ago):
    sessions=1, average-cpu-user/nice/system/idle=0%/0%/0%/100%, memory=16%
HBDEV stats:
  FG201E4Q17900771(updated 3 seconds ago):
    wan2: physical/1000auto, up, rx-bytes/packets/dropped/errors=114918/266/0/0,
tx=76752/178/0/0
    ha: physical/00, down, rx-bytes/packets/dropped/errors=0/0/0/0, tx=0/0/0/0
  FG201ETK19900991(updated 1 seconds ago):
    wan2: physical/1000auto, up, rx-bytes/packets/dropped/errors=83024/192/0/0,
tx=120216/278/0/0
    ha: physical/00, down, rx-bytes/packets/dropped/errors=0/0/0/0, tx=0/0/0/0
Secondary: FortiGate-201E, FG201E4Q17900771, HA cluster index = 1
Primary: FortiGate-201E, FG201ETK19900991, HA cluster index = 0
number of vcluster: 1
vcluster 1: work 169.254.0.1
Secondary: FG201E4Q17900771, HA operating index = 1
Primary: FG201ETK19900991, HA operating index = 0
```

If all members are `in-sync`, this means all members share the same configurations, except those that should not be synced. If any members are `out-of-sync`, this means the member failed to sync with the primary device.

Debugging is similar when a cluster is out of sync.

The following topic provides more information about standalone configuration synchronization:

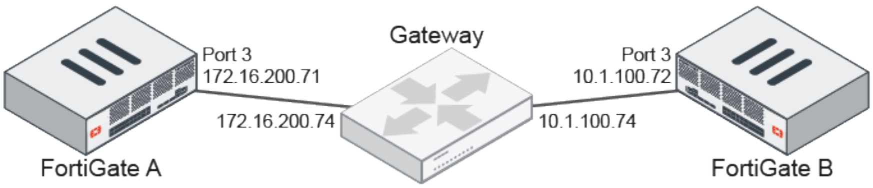
- Layer 3 unicast standalone configuration synchronization on page 2110

Layer 3 unicast standalone configuration synchronization

Unicast standalone configuration synchronization is supported on layer 3, allowing peers to be synchronized in cloud environments that do not support layer 2 networking. Configuring a unicast gateway allows peers to be in different subnets.

Example

In this example, two FortiGates in different subnets are connected through a unicast gateway. Both cluster members use the same port for the heartbeat interface.



To configure unicast synchronization between peers:

1. Configure FortiGate A:

```
config system ha
    set group-name "testcs"
    set hbdev "port3" 50
    set standalone-config-sync enable
    config unicast-peers
        edit 1
            set peer-ip 10.1.100.72
        next
    end
    set override enable
    set priority 200
    set unicast-status enable
    set unicast-gateway 172.16.200.74
end
```

2. Configure FortiGate B:

```
config system ha
    set group-name "testcs"
    set hbdev "port3" 50
    set standalone-config-sync enable
    config unicast-peers
        edit 1
            set peer-ip 172.16.200.71
        next
    end
    set override enable
    set priority 100
    set unicast-status enable
    set unicast-gateway 10.1.100.74
end
```

3. Check the HA status on FortiGate A:

```
# get system ha status
HA Health Status: OK
Model: FortiGate-VM64
Mode: ConfigSync
Group: 0
Debug: 0
Cluster Uptime: 2 days 3:40:25
Cluster state change time: 2021-03-08 12:00:38
Primary selected using:
    <2021/03/08 12:00:38> FGVMSLTM00000001 is selected as the primary because its
override priority is larger than peer member FGVMSLTM00000002.
    <2021/03/06 11:50:35> FGVMSLTM00000001 is selected as the primary because it's the
only member in the cluster.
ses_pickup: disable
override: enable
Configuration Status:
    FGVMSLTM21000151(updated 5 seconds ago): in-sync
    FGVMSLTM21000152(updated 5 seconds ago): in-sync
System Usage stats:
    FGVMSLTM21000151(updated 5 seconds ago):
        sessions=7, average-cpu-user/nice/system/idle=0%/0%/0%/100%, memory=24%
    FGVMSLTM21000152(updated 5 seconds ago):
        sessions=5, average-cpu-user/nice/system/idle=0%/0%/0%/100%, memory=23%
HBDEV stats:
    FGVMSLTM21000151(updated 5 seconds ago):
        port3: physical/1000auto, up, rx-
bytes/packets/dropped/errors=466060007/1049137/0/0, tx=429538329/953028/0/0
    FGVMSLTM21000152(updated 5 seconds ago):
        port3: physical/1000auto, up, rx-
bytes/packets/dropped/errors=48805199/85441/0/0, tx=33470286/81425/0/0
Primary     : FGT-71          , FGVMSLTM00000001, HA cluster index = 1
Secondary   : FGT-72          , FGVMSLTM00000002, HA cluster index = 0
number of vcluster: 1
vcluster 1: work 0.0.0.0
Primary: FGVMSLTM00000001, HA operating index = 0
Secondary: FGVMSLTM00000002, HA operating index = 1
```

4. Check the HA checksums on FortiGate A:

```
# diagnose sys ha checksum cluster

================== FGVMSLTM00000001 ==================

is_manage_primary()=1, is_root_primary()=1
debugzone
global: 4f 2c a2 04 07 57 46 c4 47 28 ca d2 5a c5 98 ee
root: 16 af 5d a4 ac cf a5 4b b7 22 93 ce f9 02 68 bc
all: 6e 28 7f 8a 74 f7 37 43 8f 32 73 68 1e d6 ca cd

checksum
global: 4f 2c a2 04 07 57 46 c4 47 28 ca d2 5a c5 98 ee
root: 16 af 5d a4 ac cf a5 4b b7 22 93 ce f9 02 68 bc
all: 6e 28 7f 8a 74 f7 37 43 8f 32 73 68 1e d6 ca cd

================== FGVMSLTM00000002 ==================

is_manage_primary()=0, is_root_primary()=1
```

```
debugzone
global: 4f 2c a2 04 07 57 46 c4 47 28 ca d2 5a c5 98 ee
root: 16 af 5d a4 ac cf a5 4b b7 22 93 ce f9 02 68 bc
all: 6e 28 7f 8a 74 f7 37 43 8f 32 73 68 1e d6 ca cd

checksum
global: 4f 2c a2 04 07 57 46 c4 47 28 ca d2 5a c5 98 ee
root: 16 af 5d a4 ac cf a5 4b b7 22 93 ce f9 02 68 bc
all: 6e 28 7f 8a 74 f7 37 43 8f 32 73 68 1e d6 ca cd
```

5. Verify that configuration changes on the primary FortiGate are synchronized to the secondary FortiGate:
   a. Adjust the administrator timeout value on FortiGate A:

   ```
   config system global
       set admintimeout 100
   end
   ```

   b. Check the debug messages on FortiGate B:

   ```
   # diagnose debug cli 7
   Debug messages will be on for 30 minutes.

   # diagnose debug enable

   create pid=15639, clictxno=0, last=1615246288
   0: conf sys global
   0: set admintimeout 100
   0: end
   ```

VRRP

A Virtual Router Redundancy Protocol (VRRP) configuration can be used as a high availability solution to ensure that a network maintains connectivity with the internet (or with other networks) even if the default router for the network fails. If a router or a FortiGate fails, all traffic to this device transparently fails over to another router or FortiGate that takes over the role of the failed device. If the failed device is restored, it will take over processing the network traffic.

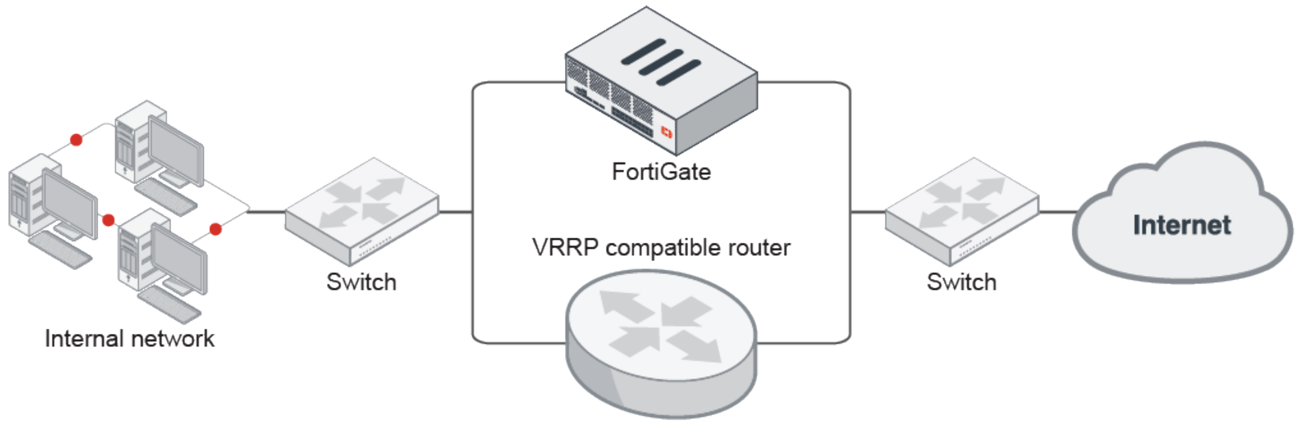
FortiOS supports VRRP versions 2 and 3. VRRP domains can be created, which can include multiple FortiGates and other VRRP-compatible routers. Different FortiGate models can be added to the same VRRP domain.

FortiOS supports IPv4 and IPv6 VRRP, so IPv4 and IPv6 VRRP virtual routers can be added to the same interface. FortiGates can quickly and easily integrate into a network that has already deployed VRRP.

Basic VRRP configuration

The most common VRRP application is to provide redundant default routers between an internal network and the internet. The default routers can be FortiGates or any routers that support VRRP.

Two or more FortiGate interfaces or routers must be configured with the same virtual router ID and IP address so they can automatically join the same VRRP domain. Priorities must be assigned to each FortiGate interface or router in the VRRP domain. All of the routers in the VRRP domain should have different priorities. One FortiGate interface or router must have the highest priority to become the primary router. The other FortiGates or routers in the domain are assigned lower priorities and become backups. If the primary router fails, VRRP automatically fails over to the router in the domain with the next highest priority.



To configure VRRP:

1. Add a virtual VRRP router to the internal interface of each FortiGate and/or router. This adds the FortiGates and routers to the same VRRP domain.
2. Set the VRRP IP address of the domain to the internal network default gateway IP address.
3. Set the priorities.

See Adding IPv4 and IPv6 virtual routers to an interface on page 2113 Single-domain VRRP example on page 2120, and Multi-domain VRRP example on page 2121 for configuration examples.

During normal operations, all traffic from the internal network to the internet passes through the primary VRRP router. The primary router also sends VRRP advertisement messages to the backup routers. A backup router will not attempt to become a primary router while receiving these messages. If the primary router fails, the backup router with the highest priority becomes the new primary router after a short delay. All packets sent to the default route are now sent to the new primary router. If the new primary router is a FortiGate, the network continues to benefit from FortiOS security features. If the new primary router is just a router, traffic continues to flow, but FortiOS security features are unavailable until the FortiGate is back online.

If the backup router is a FortiGate, during a VRRP failover as the FortiGate begins operating as the new primary router, it will not have session information for all of the failed over in-progress sessions. So, it would normally not be able to forward in-progress session traffic.

Adding IPv4 and IPv6 virtual routers to an interface

This topic describes to how to add IPv4 and IPv6 virtual routers to an interface. VRRP can only be configured on physical or VLAN interfaces. VRRP cannot be configured on hardware switch interfaces where multiple physical interfaces are combined into a hardware switch interface.

IPv4 virtual router

In this example, an IPv4 VRRP router is added to port10 on the FortiGate. The VRRP virtual router has a virtual router ID of 200, uses IP address 10.31.101.200, and has a priority of 255. Since this is the highest priority in the configuration, this interface is configured to be the primary router of the VRRP domain.

To configure the interface settings:

```
config system interface
    edit port10
        config vrrp
            edit 200
                set vrip 10.31.101.200
                set priority 255
            next
        end
    next
end
```

IPv6 virtual router

In this example, an IPv6 VRRP router is added to port20 on the FortiGate. The VRRP virtual router has a virtual router ID of 220, uses IP address 2001:db8:1::12, and has a priority of 255. Since this is the highest priority in the configuration, this interface is configured to be the primary router of the VRRP domain.

To configure the interface settings:

```
config system interface
    edit port20
        config ipv6
            set vrip6_link_local <IPv6_address>
            config vrrp6
                edit 220
                    set vrip 2001:db8:1::12
                    set priority 255
                next
            end
        end
    next
end
```

VRRP failover

VRRP routers in a VRRP domain periodically send VRRP advertisement messages to all routers in the domain to maintain one router as the primary router and the others as backup routers. The primary router has the highest priority. If the backup routers stop receiving these packets from the primary router, the backup router with the highest priority becomes the new primary router.

The primary router stops sending VRRP advertisement messages if it fails or becomes disconnected. Up to two VRRP destination addresses can be configured to be monitored by the primary router. As a best practice, the destination addresses should be remote addresses. If the primary router is unable to connect to these destination addresses, it stops sending VRRP advertisement messages, and the backup router with the highest priority becomes the primary router.

To configure IPv4 VRRP with two destination addresses for monitoring:

```
config system interface
    edit port14
        config vrrp
```

```
            edit 12
                set vrdst 10.10.10.20 10.20.20.10
            next
        end
    next
end
```

To configure IPv6 VRRP with one destination address for monitoring:

```
config system interface
    edit port23
        config ipv6
            config vrrp6
                edit 223
                    set vrdst 2001:db8:1::12
                next
            end
        end
    next
end
```

IPv4 VRRP active failover

The `vrdst-priority` option can be used to reduce IPv4 VRRP failover times. This option causes the primary router to actively signal to the backup routers when the primary router cannot reach its configured destination addresses. The primary router sends a lower priority for itself in the VRRP advertisement messages. The backup router with the highest priority becomes the new primary router and takes over traffic processing.

In this example, the primary router is configured to have a priority of 255, so it should always become the primary router. The `vrdst-priority` is set to 10. If the primary router cannot connect to the 10.10.10.1 destination address, then the primary router informs the VRRP group that its priority is now 10.

To set the priority of the virtual router when the destination address is unreachable:

```
config system interface
    edit port10
        config vrrp
            edit 12
                set vrip 10.31.101.200
                set priority 255
                set vrdst 10.10.10.1
                set vrdst-priority 10
            next
        end
    next
end
```

IPv4 VIP and IP pool failover

The `proxy-arp` option can be used to map VIPs and IP pool address ranges to each router's VMAC (virtual MAC). After failover, the IP or ranges configured in the VRRP settings are routed to the new primary router's VMAC. In this example, a single IP and an address range are added for proxy ARP.

To configure the IP addresses for proxy ARP:

```
config system interface
    edit port5
        set vrrp-virtual-mac enable
        config vrrp
            edit 1
                config proxy-arp
                    edit 1
                        set ip 192.168.62.100-192.168.62.200
                    next
                    edit 2
                        set ip 192.168.62.225
                    next
                end
            next
        end
    next
end
```

Changing the advertisement message interval

By default, VRRP advertisement messages are sent once every second. The frequency can be changed with the `adv-interval` option to change the frequency of sending these messages (1 - 255 seconds).

The `adv-interval` also affects the period of time that a backup VRRP router waits before assuming the primary router has failed. The waiting period is three times the `adv-interval`. For example, if the `adv-interval` is set to 5, then the backup router waits for up to 15 seconds to receive a VRRP advertisement from the current primary router before taking over the role as the primary router.

To configure IPv4 VRRP to send advertisement messages every 10 seconds:

```
config system interface
    edit port14
        config vrrp
            edit 12
                set adv-interval 10
            next
        end
    next
end
```

To configure IPv6 VRRP to send advertisement messages every 20 seconds:

```
config system interface
    edit port23
        config ipv6
            config vrrp6
                edit 223
                set adv-interval 20
            next
        end
    next
end
```

Changing the VRRP startup time

The VRRP startup time is the time a backup or primary VRRP router waits before sending or receiving VRRP advertisements before potentially changing state (`start-time` in seconds, 1 - 255, default = 3). This timer is mainly visible when VRRP-monitored interfaces become up after previously been down. When this occurs, the device will wait for the time period before considering, and potentially changing its status.

There are some instances when the advertisement messages might be delayed. For example, some switches with spanning tree enabled may delay some of the advertisement message packets. If backup routers are attempting to become primary routers even though the primary router has not failed, extend the start time to ensure that the backup routers wait long enough for the advertisement messages.

To configure the IPv4 VRRP startup time to 10 seconds:

```
config system interface
    edit port14
        config vrrp
            edit 12
                set start-time 10
            next
        end
    next
end
```

To configure the IPv6 VRRP startup time to 15 seconds:

```
config system interface
    edit port23
        config ipv6
            config vrrp6
                edit 223
                set start-time 15
            next
        end
    next
end
```

VRRP groups

If VRRP routers are added to multiple interfaces of the same FortiGate, each router will be in a different VRRP domain. If one of the VRRP routers fails, it is useful if all of the VRRP routers added to the FortiGate also fail.

VRRP can only check the routers' status in a single VRRP domain and cannot track the status of routers in other domains. For multiple VRRP domains on a single FortiGate, only one can switch to being a backup, and the others remain operating normally. Using VRRP groups resolves this issue.

All the VRRP virtual routers on the FortiGate can be added to a VRRP group. If one of the virtual routers in a VRRP group switches to the backup, the VRRP group forces all members to switch to backups. All VRRP traffic being processed by the FortiGate fails over to other devices in the network.

The status of the virtual routers in a VRRP group only changes when one or more of the virtual routers in the group changes status. A VRRP group should not be used to manually change the status of the virtual routers in the group.

To configure two IPv4 VRRP routers in a VRRP group:

```
config system interface
    edit port10
        config vrrp
            edit 200
                set vrip 10.31.101.200
                set priority 255
                set vrgrp 10
            next
        end
    next
    edit port20
        config vrrp
            edit 100
                set vrip 10.23.1.223
                set priority 20
                set vrgrp 10
            next
        end
    next
end
```

To configure two IPv6 VRRP routers in a VRRP group:

```
config system interface
    edit port11
        config ipv6
            set vrip6_link_local <IPv6_address>
            config vrrp6
                edit 220
                    set vrip 2001:db8:1::12
                    set priority 255
                    set vrgrp 90
                next
            end
        end
    next
    edit port12
        config ipv6
            set vrip6_link_local <IPv6_address>
            config vrrp6
                edit 220
                    set vrip 2001:db8:1::14
                    set priority 100
                    set vrgrp 90
                next
            end
        end
    next
end
```

VRRP virtual MACs

The VRRP virtual MAC address (or virtual router MAC address) is a shared MAC address adopted by the primary router. If the primary router fails, the same virtual MAC address is picked up by the new primary router, allowing all devices on the network to transparently connect to the default route using the same virtual MAC address. This feature must be enabled on all members in a VRRP domain.

Each VRRP router has its own virtual MAC address. The last part octet is based on the VRRP router ID using the following format:

`00-00-5E-00-01-<VRID_hex>`

Where `<VRID_hex>` is the VRRP router ID in hexadecimal format in internet standard bit-order. For more information about virtual MAC formatting, see RFC 3768.

For example:

- If the VRRP router ID is 10, then the virtual MAC is `00-00-5E-00-01-0a`.
- If the VRRP router ID is 200, then the virtual MAC is `00-00-5E-00-01-c8`.

If the VRRP virtual MAC address feature is disabled (the default setting), the VRRP domain uses the MAC address of the primary router. On a FortiGate VRRP virtual router, this is the MAC address of the FortiGate interface that the VRRP router is added to. If the primary fails, when the new primary takes over, it sends gratuitous ARPs to associate the VRRP router IP address with the MAC address of the new primary (or the FortiGate interface that became the new primary).

When a VRRP virtual MAC address is enabled, the new primary uses the same MAC address as the old primary.

Since devices on the LAN do not have to learn a new MAC address for a new VRRP router in the event of a failover, this feature can improve network efficiency, especially in large and complex networks.

To enable virtual MAC addresses in IPv4 VRRP:

```
config system interface
    edit <name>
        set vrrp-virtual-mac enable
    next
end
```

To enable virtual MAC addresses in IPv6 VRRP:

```
config system interface
    edit <name>
        config ipv6
            set vrrp-virtual-mac6 enable
        end
    next
end
```

Preempt mode

When preempt mode is enabled (the default setting), a higher priority backup router can preempt a lower priority primary router. This can happen if the primary router fails, the backup router becomes the primary router, and the failed primary router restarts. Since the restarted router has a higher priority, if preempt mode is enabled, the restarted router replaces the current primary router becoming the new primary router. If preempt mode is disabled, a restarted router that has a higher priority would not take over as the primary router.

To configure preempt mode in IPv4 VRRP:

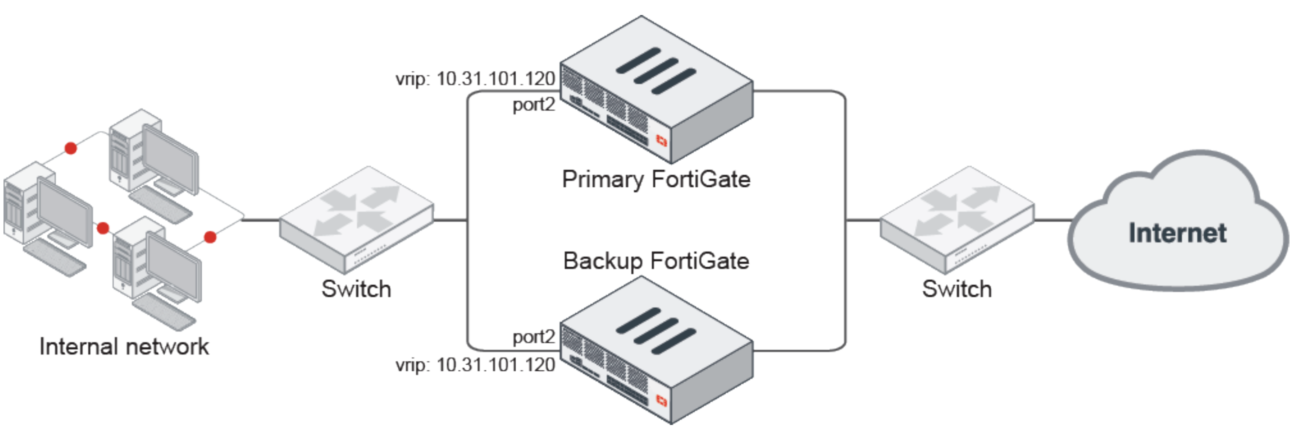
```
config system interface
    edit <name>
        config vrrp
            edit <vrid>
                set preempt {enable | disable}
            next
        end
    next
end
```

To configure preempt mode in IPv6 VRRP:

```
config system interface
    edit <name>
        config ipv6
            config vrrp6
                edit <vrid>
                    set preempt {enable | disable}
                next
            end
        end
    next
end
```

Single-domain VRRP example

This example consists of a VRRP domain with two FortiGates that connect an internal network to the internet. The FortiGate port2 interfaces connect to the internal network, and a VRRP virtual router is added to each port2 interface with VRRP virtual MAC addresses enabled. The internal network default route is 10.31.101.120. Each FortiGate port2 interface has an IP address that is different from the virtual router IP address. Since `vrrp-virtual-mac` is enabled, upon failover, the new primary VRRP router will use the same VMAC as the previous router.



To configure the primary FortiGate:

```
config system interface
    edit port2
        set vrrp-virtual-mac enable
```

```
        config vrrp
            edit 5
                set vrip 10.31.101.120
                set priority 255
            next
        end
    next
end
```

To configure the backup FortiGate:

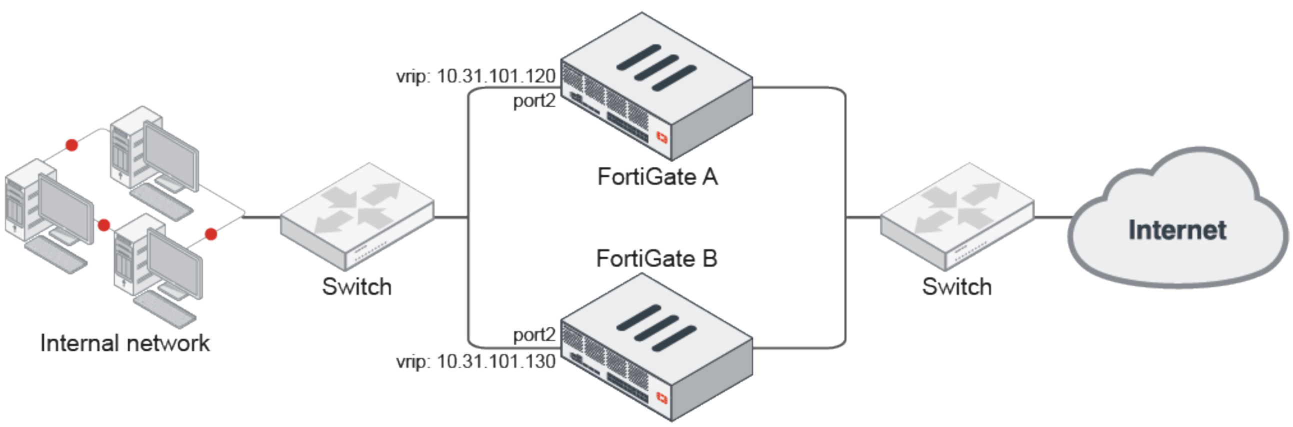
```
config system interface
    edit port2
        set vrrp-virtual-mac enable
        config vrrp
            edit 5
                set vrip 10.31.101.120
                set priority 50
            next
        end
    next
end
```

Multi-domain VRRP example

This example consists of two VRRP domains, and both FortiGates participate in the domains that connect an internal network to the internet. One FortiGate is the primary router of one domain and the other FortiGate is the primary router of the other domain. The network distributes traffic between two different default routes (10.31.101.120 and 10.31.101.130). One VRRP domain is configured with one of the default route IP addresses and the other VRRP domain gets the other default route IP address. During normal operation, both FortiGates process traffic, and the VRRP domains are used to load balance the traffic between the two FortiGates.

If one of the FortiGates fails, the remaining FortiGate becomes the primary router of both VRRP domains. The network sends all traffic for both default routes to this FortiGate. The result is a configuration that (under normal operational load) balances traffic between two FortiGates, but if one of the FortiGates fails, all traffic fails over to the FortiGate that is still operating.

VRRP virtual MAC address are enabled on both FortiGates' port2 interfaces so that the VRRP domains use their VRRP virtual MAC addresses.



Device	VRRP primary			VRRP backup		
	Virtual router IP	ID	Priority	Virtual router IP	ID	Priority
FortiGate A	10.31.101.120	50	255	10.31.101.130	100	50
FortiGate B	10.31.101.130	100	255	10.31.101.120	50	50

To configure FortiGate A:

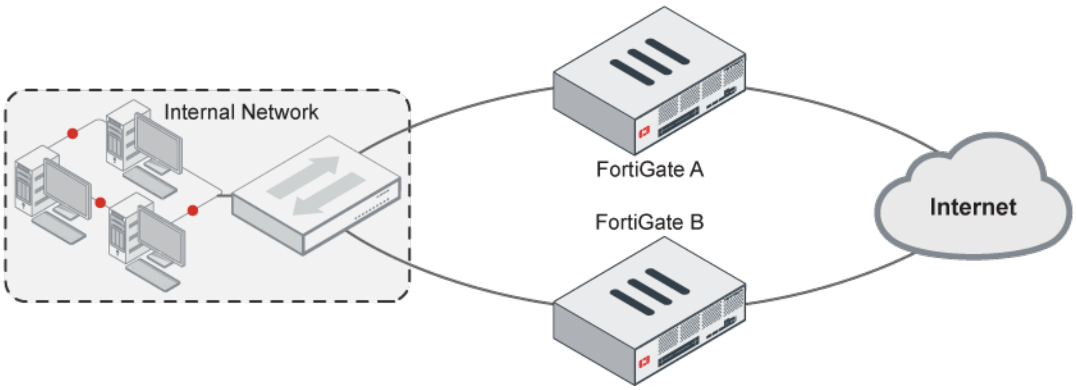
```
config system interface
    edit port2
        set vrrp-virtual-mac enable
        config vrrp
            edit 50
                set vrip 10.31.101.120
                set priority 255
            next
            edit 100
                set vrip 10.31.101.130
                set priority 50
            next
        end
    next
end
```

To configure FortiGate B:

```
config system interface
    edit port2
        set vrrp-virtual-mac enable
        config vrrp
            edit 50
                set vrip 10.31.101.120
                set priority 50
            next
            edit 100
                set vrip 10.31.101.130
                set priority 255
            next
        end
    next
end
```

VRRP on EMAC-VLAN interfaces

Virtual Router Redundancy Protocol (VRRP) can be configured on EMAC-VLAN interfaces.



To configure the interfaces:

1. Configure FortiGate A:

```
config system interface
    edit "emac"
        set vdom "root"
        set ip 172.16.209.1 255.255.255.0
        set allowaccess ping https ssh snmp http telnet fgfm
        set type emac-vlan
        set vrrp-virtual-mac enable
        config vrrp
            edit 1
                set vrip 172.16.209.111
                set priority 200
            next
        end
        set snmp-index 61
        set interface "port1"
    next
end
```

2. Configure FortiGate B:

```
config system interface
    edit "emac"
        set vdom "root"
        set ip 172.16.209.2 255.255.255.0
        set allowaccess ping https ssh snmp http telnet fgfm
        set type emac-vlan
        set vrrp-virtual-mac enable
        config vrrp
            edit 1
                set vrip 172.16.209.111
                set priority 222
            next
        end
        set snmp-index 32
        set interface "port1"
    next
end
```

Check the VRRP information on the FortiGates:

Because FortiGate B has a higher priority, it is the primary device and FortiGate A is the backup.

1. FortiGate A:

```
# get router info vrrp
Interface: emac, primary IP address: 172.16.209.1
  UseVMAC: 1, SoftSW: 0, EmacVlan: 1 BrPortIdx: 0, PromiscCount: 0
  HA mode: primary (0:0:1) VRRP master number: 0
  VRID: 1 verion: 2
    vrip: 172.16.209.111, priority: 200 (200,0), state: BACKUP
    adv_interval: 1, preempt: 1, ignore_dft: 0 start_time: 3
    master_adv_interval: 100, accept: 1
    vrmac: 00:00:5e:00:01:01
    vrdst:
    vrgrp: 0
```

2. FortiGate B:

```
# get router info vrrp
Interface: emac, primary IP address: 172.16.209.2
  UseVMAC: 1, SoftSW: 0, EmacVlan: 1 BrPortIdx: 0, PromiscCount: 1
  HA mode: primary (0:0:1) VRRP master number: 1
  VRID: 1 verion: 2
    vrip: 172.16.209.111, priority: 222 (222,0), state: PRIMARY
    adv_interval: 1, preempt: 1, ignore_dft: 0 start_time: 3
    master_adv_interval: 100, accept: 1
    vrmac: 00:00:5e:00:01:01
    vrdst:
    vrgrp: 0
```

SNMP

SNMP enables you to monitor hardware on your network. You can configure the hardware, such as the FortiGate SNMP agent, to report system information and send traps (alarms or event messages) to SNMP managers. SNMP traps alert you to events that happen, such as when a log disk is full or a virus is detected.

The FortiGate SNMP implementation is read-only. SNMP v1/v2c, and v3 compliant SNMP managers have read-only access to FortiGate system information through queries, and can receive trap messages from the FortiGate unit.

- Interface access on page 2125
- MIB files on page 2125
- SNMP agent on page 2126
- SNMP v1/v2c communities on page 2126
- SNMP v3 users on page 2128
- Access control for SNMP on page 2129
- Important SNMP traps on page 2131
- SNMP traps and query for monitoring DHCP pool on page 2133

Interface access

Before a remote SNMP manager can connect to the FortiGate SNMP agent, you must configure one or more FortiGate interfaces to accept SNMP connections.

To configure a FortiGate interface to accept SNMP connections in the GUI:

1. Go to *Network > Interfaces*.
2. Edit the interface.
3. In the *Administrative Access* options, enable *SNMP*.
4. Click *OK*.

To configure a FortiGate interface to accept SNMP connections in the CLI:

```
config system interface
    edit <interface>
        append allowaccess snmp
        set snmp-index <integer>
        config ipv6
            append ip6-allowaccess snmp
        end
    next
end
```

MIB files

The FortiGate SNMP agent supports Fortinet proprietary MIBs, as well as the parts of RFC 2665 and RFC 1213 that apply to FortiGate unit configuration.

Your SNMP manager may already include standard and private MIBs in a compiled database that is ready to use. You must add the Fortinet proprietary MIBs to this database to have access to Fortinet specific information.

MIB file or RFC	Description
FORTINET-CORE-MIB.mib	The Fortinet core MIB includes all system configuration and trap information that is common to all Fortinet products. Your SNMP manager requires this information to monitor Fortinet device settings and receive traps from the FortiGate SNMP agent.
FORTINET-FORTIGATE-MIB.mib	The FortiGate MIB includes all system configuration information and trap information that is specific to FortiGate units. Your SNMP manager requires this information to monitor FortiGate settings and receive traps from the FortiGate SNMP agent.
RFC-1213 (MIB II)	The FortiGate SNMP agent supports MIB II groups with the following exceptions: • No support for the EGP group from MIB II (RFC 1213, section 3.11 and 6.10). • Protocol statistics returned for MIB II groups (IP/ICMP/TCP/UDP/etc.) do not accurately capture all Fortinet traffic activity. More accurate information can be obtained from the information reported by the Fortinet MIB.
RFC-2665 (Ethernet-like MIB)	The FortiGate SNMP agent supports Ethernet-like MIB information.

MIB file or RFC	Description
	FortiGate SNMP does not support for the dot3Tests and dot3Errors groups.

To download the MIB files:

1. Go to *System > SNMP*.
2. Click *Download FortiGate MIB File* and save the file to the management computer.
3. Click *Download Fortinet Core MIB File* and save the file to the management computer.

SNMP agent

The SNMP agent sends SNMP traps originating on the FortiGate to an external monitoring SNMP manager defined in a SNMP community. The SNMP manager can monitor the FortiGate system to determine if it is operating properly, or if any critical events occurring.

The description, location, and contact information for this FortiGate system will be part of the information that the SNMP manager receives. This information is useful if the SNMP manager is monitoring many devices, and enables faster responses when the FortiGate system requires attention.

To configure the SNMP agent in the GUI:

1. Go to *System > SNMP*.
2. Enable *SNMP Agent*.
3. Enter a description of the agent.
4. Enter the location of the FortiGate unit.
5. Enter a contact or administrator for the SNMP Agent or FortiGate unit.
6. Click *Apply*.

To configure the SNMP agent in the CLI:

```
config system snmp sysinfo
    set status enable
    set description <string>
    set contact-info <string>
    set location <string>
end
```

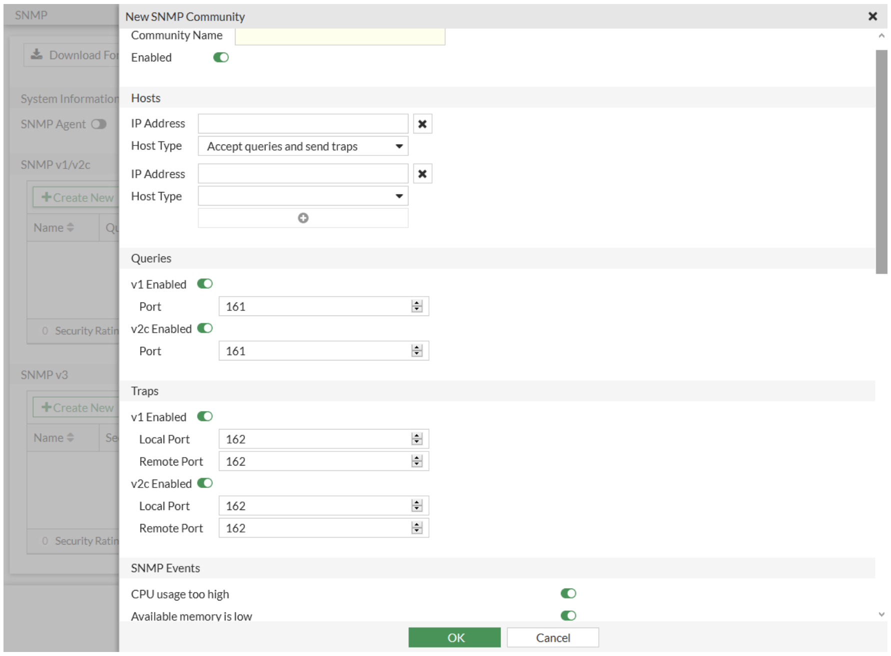
SNMP v1/v2c communities

An SNMP community is a grouping of equipment for network administration purposes. A single device can belong to multiple communities.

You must add an SNMP community to the FortiGate so that the SNMP manager can receive traps and system information. Up to three communities can be added.

To create a n SNMP v1/v2c community in the GUI:

1. Go to *System* > *SNMP*.
2. In the *SNMP v1/v2c* table, click *Create New*.



3. Enter a *Community Name* and enable the community.
4. In the *Hosts* section, enter the *IP Address* and select the *Host Type* for each SNMP manager.
5. In the *Queries* section, enable or disable v1 and v2c queries, then enter the port numbers that the SNMP managers in this community use for them.
6. In the *Traps* section, enable or disable v1 and v2c traps, then enter the local and remote port numbers that the SNMP managers in this community use for them.
7. In the *SNMP Events* section, enable or disable the events that activate traps in this community.
8. Click *OK*.

To create a n SNMP v1/v2c community in the CLI:

```
config system snmp community
    edit 2
        set name <string>
        set status {enable | disable}
        config hosts
            edit <host_id>
                set ip <ip/mask>
                set source-ip <class_ip>
                set ha-direct {enable | disable}
                set host-type {any | query | trap}
            next
        end
        set query-v1-port <port_number>
        set query-v1-status {enable | disable}
        set query-v2c-port <port_number>
        set query-v2c-status {enable | disable}
        set trap-v1-lport <port_number>
```

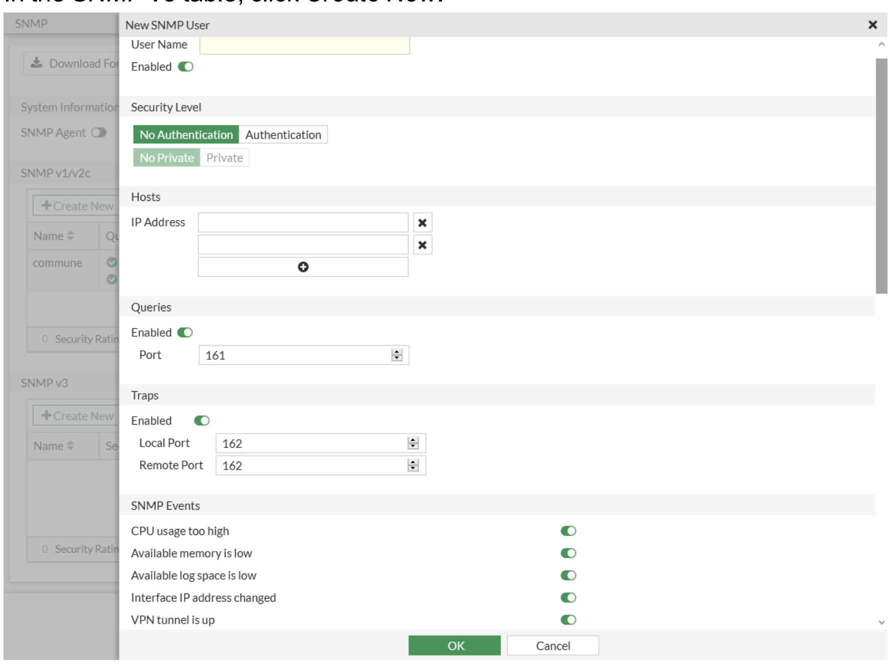
```
        set trap-v1-rport <port_number>
        set trap-v1-status {enable | disable}
        set trap-v2c-lport <port_number>
        set trap-v2c-rport <port_number>
        set trap-v2c-status {enable | disable}
        set events <events>
    next
end
```

SNMP v3 users

Authentication is used to ensure the identity of users. Privacy allows for encryption of SNMP v3 messages to ensure confidentiality of data. These protocols provide a higher level of security than is available in SNMP v1 and v2c, which use community strings for security. Both authentication and privacy are optional.

To create a n SNMP v3 user in the GUI:

1. Go to *System* > *SNMP*.
2. In the *SNMP v3* table, click *Create New*.



3. Enter a *User Name* and enable the user.
4. In the *Security Level* section, configure the security level:
   - *No Authentication*: No authentication or encryption.
   - *Authentication*: Select the authentication algorithm and password.
   - *Authentication and Private*: Select both the authentication and encryption algorithms and password.
5. In the *Hosts* section, enter the *IP Address* for each SNMP manager.
6. In the *Queries* section, enable or disable queries, then enter the port number that the SNMP managers use for them.
7. In the *Traps* section, enable or disable traps, then enter the local and remote port numbers that the SNMP managers use for them.

8. In the *SNMP Events* section, enable or disable the events that activate traps.
9. Click *OK*.

To create an SNMP v3 user in the CLI:

```
config system snmp user
    edit <user>
        set status {enable | disable}
        set trap-status {enable | disable}
        set trap-lport <port_number>
        set trap-rport <port_number>
        set queries {enable | disable}
        set query-port <port_number>
        set notify-hosts <class_ip> ... <class_ip>
        set source-ip <class_ip>
        set ha-direct {enable | disable}
        set events <events>
        set security-level {no-auth-no-priv | auth-no-priv | auth-priv}
        set auth-proto {md5 | sha | sha224 | sha256 | sha384 | sha512}
        set auth-pwd <password>
        set priv-proto {aes | des | aes256 | aes256cisco}
        set priv-pwd <password>
    next
end
```

Access control for SNMP

Administrators can provide access control to SNMP users and communities based on restricting a MIB view to specific OID subtrees. They can also define access based on the VDOM. This allows multi-tenant FortiGate deployments to provide restricted access per VDOM.

- MIB view access control allows the SNMP clients to query specific OIDs that are filtered by the MIB view settings.
- VDOM access control allows the SNMP clients to query data from specific VDOMs that are filtered by the VDOM settings.

When access control is enabled, the users can only access the information that is allowed by the access control, and all other information is inaccessible. Administrators have granular control, and can easily restrict specific information based on access control.

To configure MIB views:

```
config system snmp mib-view
    edit <name>
        set include <OIDs>
        set exclude <OIDs>
    next
end
```

`include <OIDs>`	Enter the OID subtrees to be included in the view. A maximum of 16 subtrees can be added.
`exclude <OIDs>`	Enter the OID subtrees to be excluded in the view. A maximum of 64 subtrees can be added.

To configure access control based on MIB views and VDOMs for SNMP users and communities:

```
config system snmp user
    edit <user>
        set mib-view <view>
        set vdoms <vdoms>
    next
end
```

```
config system snmp community
    edit <community>
        set mib-view <view>
        set vdoms <vdoms>
    next
end
```

`mib-view <view>`	Set the SNMP access control MIB view.
`vdoms <vdoms>`	Set the SNMP access control VDOMs.

Example

In this example, two MIB views are created and, with VDOMs, used to control access for SNMP users and communities.

To configure access control for SNMP users and communities:

1. Configure two MIB views:

```
config system snmp mib-view
    edit "view1"
        set include "1.3.6.1.2"
    next
    edit "view2"
        set include "1.3.6.1.2.1"
        set exclude "1.3.6.1.2.1.2.1" "1.3.6.1.2.1.4.31" "1.3.6.1.2.1.1.9.1"
    next
end
```

2. Add the MIB view and VDOM restrictions to SNMP users:

```
config system snmp user
    edit "v3user"
        set mib-view "view1"
    next
    edit "v3user1"
        set vdom "vdom1"
    next
    edit "v3user2"
        set mib-view "view1"
        set vdoms "root" "vdom1"
    next
end
```

3. Add the MIB view and VDOM restrictions to SNMP communities:

```
config system snmp community
    edit 1
        set name "REGR-SYS"
        set vdoms "vdom1"
    next
    edit 2
        set name "REGR-SYS1"
        set mib-view "view2"
    next
    edit 3
       set name "REGR-SYS2"
       set mib-view "view1"
       set vdoms "root" "vdom1"
    next
end
```

Important SNMP traps

Link Down and Link Up traps

This trap is sent when a FortiGate port either goes down or is brought up.

For example, the following traps are generated when the state of port34 is set to down using `set status down`, and then brought up using `set status up`:

```
NET-SNMP version 5.7.3 2019-01-31 14:11:48 10.1.100.1(via UDP: [10.1.100.1]:162->
[10.1.100.11]:162) TRAP, SNMP v1, community REGR-SYS SNMPv2-MIB::snmpTraps Link Down Trap
(0) Uptime: 0:14:44.95 IF-MIB::ifIndex.42 = INTEGER: 42 IF-MIB::ifAdminStatus.42 = INTEGER:
down(2) IF-MIB::ifOperStatus.42 = INTEGER: down(2) FORTINET-CORE-MIB::fnSysSerial.0 =
STRING: FG140P3G15800330 SNMPv2-MIB::sysName.0 = STRING: FortiGate-140D-POE
```

```
2019-01-31 14:11:48 <UNKNOWN> [UDP: [10.1.100.1]:162->[10.1.100.11]:162]: DISMAN-EVENT-
MIB::sysUpTimeInstance = Timeticks: (88495) 0:14:44.95 SNMPv2-MIB::snmpTrapOID.0 = OID: IF-
MIB::linkDown IF-MIB::ifIndex.42 = INTEGER: 42 IF-MIB::ifAdminStatus.42 = INTEGER: down(2)
IF-MIB::ifOperStatus.42 = INTEGER: down(2) FORTINET-CORE-MIB::fnSysSerial.0 = STRING:
FG140P3G15800330 SNMPv2-MIB::sysName.0 = STRING: FortiGate-140D-POE 2019-01-31 14:12:01
10.1.100.1(via UDP: [10.1.100.1]:162->[10.1.100.11]:162) TRAP, SNMP v1, community REGR-SYS
SNMPv2-MIB::snmpTraps Link Up Trap (0) Uptime: 0:14:57.98 IF-MIB::ifIndex.42 = INTEGER: 42
IF-MIB::ifAdminStatus.42 = INTEGER: up(1) IF-MIB::ifOperStatus.42 = INTEGER: up(1) FORTINET-
CORE-MIB::fnSysSerial.0 = STRING: FG140P3G15800330 SNMPv2-MIB::sysName.0 = STRING:
FortiGate-140D-POE
```

```
2019-01-31 14:12:01 <UNKNOWN> [UDP: [10.1.100.1]:162->[10.1.100.11]:162]: DISMAN-EVENT-
MIB::sysUpTimeInstance = Timeticks: (89798) 0:14:57.98 SNMPv2-MIB::snmpTrapOID.0 = OID: IF-
MIB::linkUp IF-MIB::ifIndex.42 = INTEGER: 42 IF-MIB::ifAdminStatus.42 = INTEGER: up(1) IF-
MIB::ifOperStatus.42 = INTEGER: up(1) FORTINET-CORE-MIB::fnSysSerial.0 = STRING:
FG140P3G15800330 SNMPv2-MIB::sysName.0 = STRING: FortiGate-140D-POE
```

fgFmTrapIfChange trap

This trap is sent when any changes are detected on the interface. The change can be very simple, such as giving an IPV4 address.

For example, the user has given the IP address of 1.2.3.4/24 to port 1 and the EMS Manager has detected the following trap:

```
DISMAN-EXPRESSION-MIB::sysUpTimeInstance = Timeticks: (7975058) 22:09:10.58 SNMPv2-
MIB::snmpTrapOID.0 = OID: FORTINET-FORTIGATE-MIB::fgFmTrapIfChange FORTINET-CORE-
MIB::fnSysSerial.0 = STRING: FG140P3G15800330 IF-MIB::ifName.45 = STRING: port1 FORTINET-
FORTIGATE-MIB::fgManIfIp.0 = IpAddress: 1.2.3.4 FORTINET-FORTIGATE-MIB::fgManIfMask.0 =
IpAddress: 255.255.255.0 FORTINET-FORTIGATE-MIB::fgManIfIp6.0 = STRING: 0:0:0:0:0:0:0:0
```

entConfigChange trap

The change to the interface in the previous example has also triggered the *ConfChange Trap* which is sent along with the *fgFmTrapIfChange* trap:

```
2018-11-15 09:30:23 FGT_A [UDP: [172.16.200.1]:162->[172.16.200.55]:162]: DISMAN-EXPRESSION-
MIB::sysUpTimeInstance = Timeticks: (8035097) 22:19:10.97 SNMPv2-MIB::snmpTrapOID.0 = OID:
ENTITY-MIB::entConfigChange
```

fgTrapDeviceNew trap

This trap is triggered when a new device, like a FortiSwitch, is connected to the FortiGate.

For example, the following scenario has given the device a new trap for adding FortiAP on a PoE interface a FortiGate 140D-POE. The trap has important information about the device name, device MAC address, and when it was last seen.

```
2018-11-15 11:17:43 UDP/IPv6: [2000:172:16:200::1]:162 [UDP/IPv6: [2000:172:16:200::1]:162]:
DISMAN-EXPRESSION-MIB::sysUpTimeInstance = Timeticks: (520817) 1:26:48.17 SNMPv2-
MIB::snmpTrapOID.0 = OID: FORTINET-FORTIGATE-MIB::fgTrapDeviceNew FORTINET-CORE-
MIB::fnSysSerial.0 = STRING: FG140P3G15800330 SNMPv2-MIB::sysName.0 = STRING: FGT_A IF-
MIB::ifIndex.0 = INTEGER: 0 FORTINET-FORTIGATE-MIB::fgVdEntIndex.0 = INTEGER: 0 FORTINET-
FORTIGATE-MIB::fgDeviceCreated.0 = Gauge32: 5 FORTINET-FORTIGATE-MIB::fgDeviceLastSeen.0 =
Gauge32: 5 FORTINET-FORTIGATE-MIB::fgDeviceMacAddress.0 = STRING: 90:6c:ac:f9:97:a0
```

```
2018-11-15 11:17:43 FGT_A [UDP: [172.16.200.1]:162->[172.16.200.55]:162]: DISMAN-EXPRESSION-
MIB::sysUpTimeInstance = Timeticks: (520817) 1:26:48.17 SNMPv2-MIB::snmpTrapOID.0 = OID:
FORTINET-FORTIGATE-MIB::fgTrapDeviceNew FORTINET-CORE-MIB::fnSysSerial.0 = STRING:
FG140P3G15800330 SNMPv2-MIB::sysName.0 = STRING: FGT_A IF-MIB::ifIndex.0 = INTEGER: 0
FORTINET-FORTIGATE-MIB::fgVdEntIndex.0 = INTEGER: 0 FORTINET-FORTIGATE-
MIB::fgDeviceCreated.0 = Gauge32: 5 FORTINET-FORTIGATE-MIB::fgDeviceLastSeen.0 = Gauge32: 5
FORTINET-FORTIGATE-MIB::fgDeviceMacAddress.0 = STRING: 90:6c:ac:f9:97:a0
```

fgTrapAvOversize trap

The `fgTrapAvOversize` trap is generated when the antivirus scanner detects an oversized file:

```
019-01-31 13:22:04 10.1.100.1(via UDP: [10.1.100.1]:162->[10.1.100.11]:162) TRAP, SNMP v1,
community REGR-SYS FORTINET-FORTIGATE-MIB::fgt140P Enterprise Specific Trap (602) Uptime: 1
day, 3:41:10.31 FORTINET-CORE-MIB::fnSysSerial.0 = STRING: FG140P3G15800330 SNMPv2-
MIB::sysName.0 = STRING: FortiGate-140D-POE 2019-01-31 13:22:29 <UNKNOWN> [UDP:
[10.1.100.1]:162->[10.1.100.11]:162]: DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks:
(9967031) 1 day, 3:41:10.31 SNMPv2-MIB::snmpTrapOID.0 = OID: FORTINET-FORTIGATE-
MIB::fgTrapAvOversize FORTINET-CORE-MIB::fnSysSerial.0 = STRING: FG140P3G15800330 SNMPv2-
MIB::sysName.0 = STRING: FortiGate-140D-POE
```

SNMP traps and query for monitoring DHCP pool

The SNMP DHCP event contains three traps and one query.

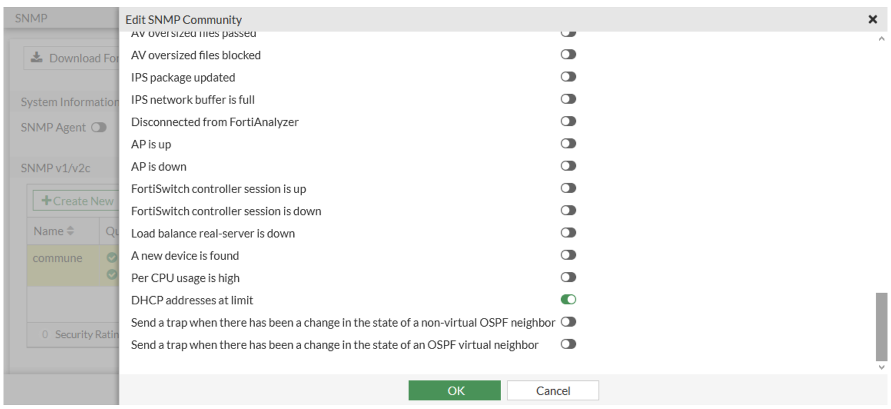
Traps are sent when:

- DHCP server IP pool usage reaches 90%
- DHCP server detect an IP address that is already in use
- DHCP client receives DHCP NAK

SNMP queries are accepted for DHCP lease usage information (OID = 1.3.6.1.4.1.12356.101.23). The query result is based on the leased out percentage.

To enable the SNMP DHCP event in the GUI:

1. Go to *System* > *SNMP*.
2. Click *Create New* in either the *SNMP v1/v2c* table or *SNMP v3* table, or edit an existing community or user.
3. Configure the settings as required.
4. In the *SNMP Events* list, enable *DHCP addresses at limit*.



5. Click *OK*.

To enable the SNMP DHCP event in the CLI:

```
config system snmp community
    edit 1
        set name "REGR-SYS"
        config hosts
            edit 1
                set ip 10.1.100.11 255.255.255.255
            next
            edit 2
                set ip 172.16.200.55 255.255.255.255
            next
        end
        set events dhcp
    next
end
```

```
config system snmp user
    edit "1"
        set notify-hosts 172.10.1.0 172.20.1.0
```

```
        set events dhcp
        set security-level auth-priv
        set auth-proto sha384
        set auth-pwd ********************
        set priv-proto aes256
        set priv-pwd *********************
    next
end
```

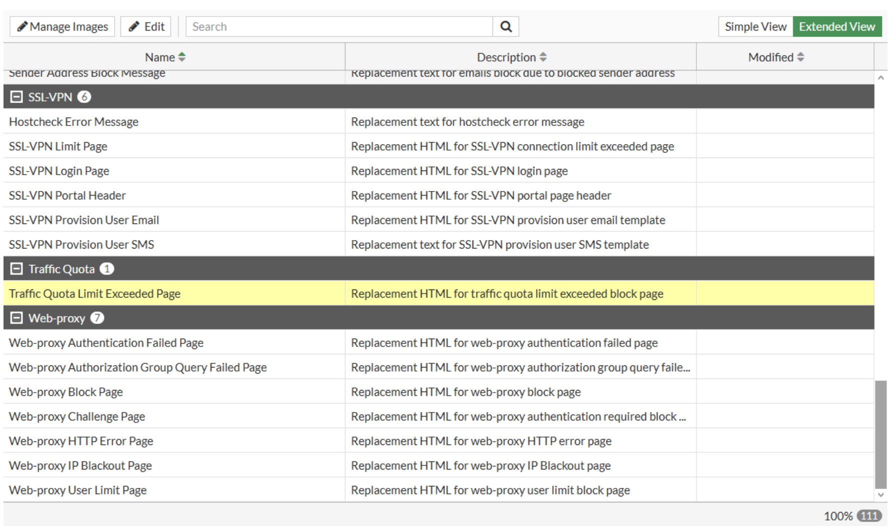
Replacement messages

FortiOS has replacement messages that are HTML and text files. These messages can be customized to meet user requirements. The content can be modified, and images can be added.

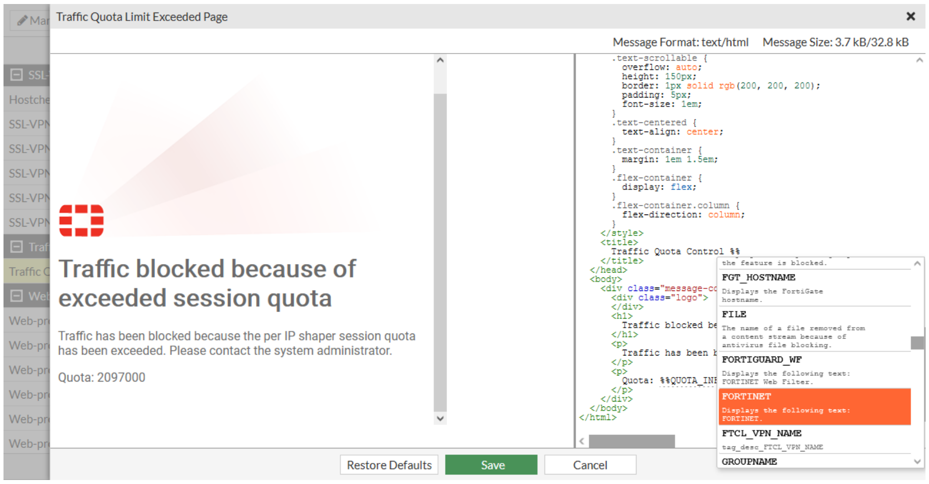
Modifying replacement messages

The *Replacement Messages* page has two views. *Simple View* (the default view) shows the most commonly used replacement messages. *Extended View* shows the entire list and all replacement message categories.

To modify a replacement message in the GUI:

1. Go to *System > Replacement Messages*.
2. Select a replacement message and click *Edit*.
   If the message you want to edit is not visible, click *Extended View* in the upper right-hand corner of the top menu.
   

3. Edit the HTML code.
   The message is visible on the left alongside the HTML code on the right. The message view updates in real-time as you edit the content.
   When adding a variable to the code, right-click and select *Insert Tag* or type `%%` to view a list of the available variables, or start typing the variable name then press *Enter* or *TAB* to auto-complete the variable name.



4. Click *Save*.

Click *Restore Defaults* to return to the original message and code base.

To modify a replacement message in the CLI:

For example, to modify the *Traffic Quota Limit Exceeded Page* message:

```
config system replacemsg traffic-quota "per-ip-shaper-block"
    set buffer "<html>
  <head>
    <title>
      Traffic Quota Control
    </title>
  </head>
  <body>
    <font size=2>
      <table width=\"100%\">
        <tr>
          <td bgcolor=#3300cc align=\"center\" colspan=2>
            <font color=#ffffff>
              <b>
                Traffic blocked because exceeded session quota
              </b>
            </font>
          </td>
        </tr>
      </table>
      <br>
      <br>
      Traffic blocked because it exceeded the per IP shaper session quota. Please contact
the system administrator.
      <br>
      %%QUOTA_INFO%%
      <br>
      <br>
      <hr>
```

```
    </font>
  </body>
</html>"
    set header http
    set format html
end
```

Replacement message images

Images can be added to replacement messages on:

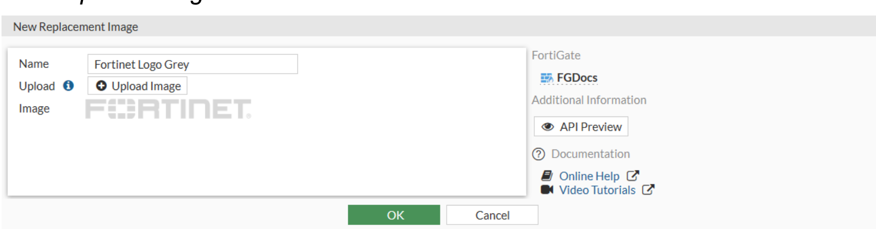
- Disclaimer pages
- Login pages
- Declined disclaimer pages
- Login failed pages
- Login challenge pages
- Keepalive pages

The supported image formats are GIF, JPEG, TIFF, and PNG. The maximum file size supported is 24 KB.

Adding images to replacement messages

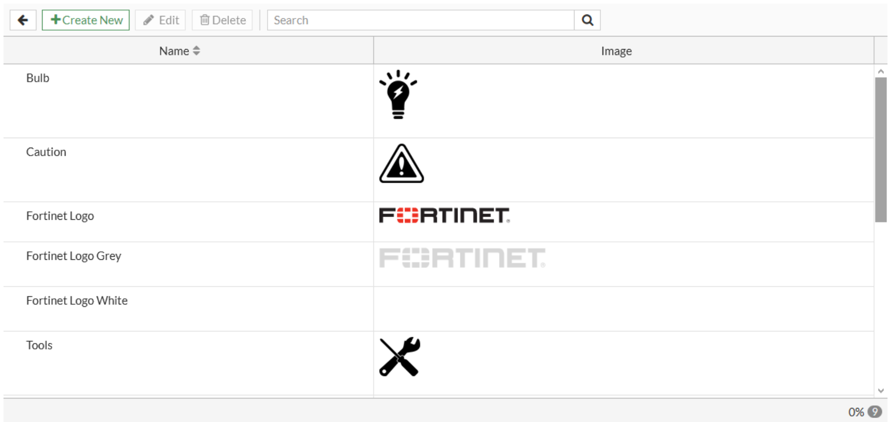
To add images to replacement messages in the GUI:

1. Go to *System > Replacement Messages*.
2. In the top menu, click *Manage Images*.
3. Click *Create New*.
4. Enter a name for the image.
5. Click *Upload Image* and locate the file.



6. Click *OK*.

   The file is now visible in the list.



7. Return to the replacement message list and edit a message.
8. Right-click in the message code where you want to add the image, and select *Insert Image*.
9. Select the image from the list then press *Enter*, or double-click on the image to add it to the message.
10. Click *Save*.

To add images to replacement messages in the CLI:

1. Add the image to the FortiGate:

```
config system replacemsg-image
    edit <image_name>
        set image-type {gif | jpg | tiff | png}
        set image-base64 <string>
    next
end
```

2. Edit the replacement message, and include `%%IMAGE:<image name>%%` in the code to add the image.

Replacement message groups

Replacement message groups allow users to customize replacement messages for individual policies and profiles.

There are two types of replacement message groups:

Type	Usage	Customizable categories
utm	Used with UTM settings in firewall policies.	• admin • alertmail • custom-message • fortiguard-wf • ftp • http • icap • mail • nac-quar • spam

Type	Usage	Customizable categories
		• sslvpn • traffic-quota • utm • webproxy
auth	Used with authentication pages in firewall policies.	• auth • webproxy

The messages added to a group do not need to be customized. The message body content, header type, and format will use the default values if not customized.

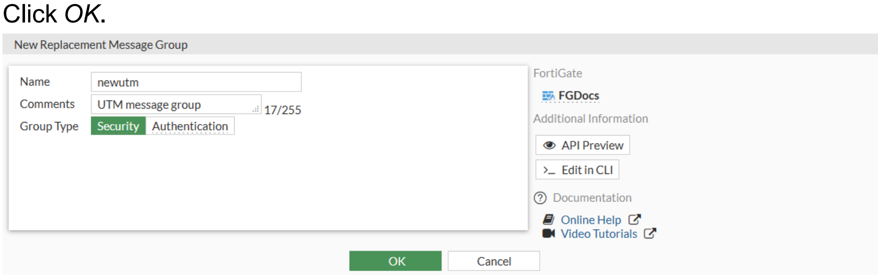
To make replacement message groups visible in the GUI:

```
config system global
    set gui-replacement-message-groups enable
end
```

In the following example, two replacement message groups are created. The UTM message group includes custom mail-related messages and is assigned to an email filter profile. The authentication message group has a custom authentication success message that is applied to a proxy-based firewall policy that has an assigned email filter profile.

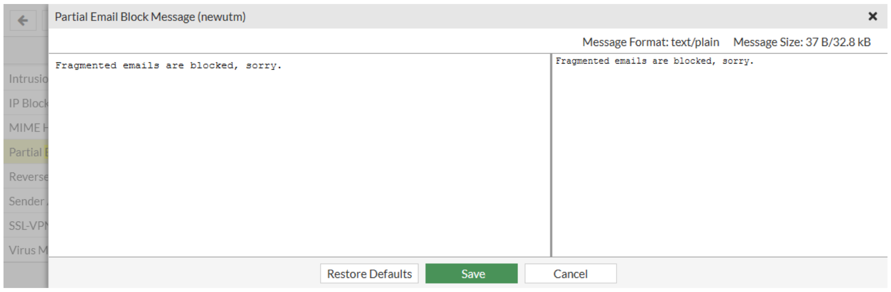
To create replacement message groups in the GUI:

1. Create the *Security* replacement message group:
   a. Go to *System > Replacement Message Groups*.
   b. Click *Create New*.
   c. For *Name*, enter *newutm*.
   d. In the *Comments* field, enter *UTM message group*.
   e. For *Group Type*, select *Security*.
   f. Click *OK*.



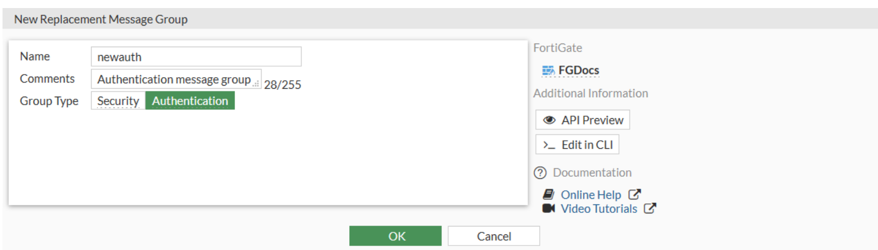
2. Customize the replacement messages in the *newutm* group:
   a. Go to *System > Replacement Message Groups*.
   b. Edit the *newutm* group.

c. Select the *Partial Email Block Message*.



d. Edit the message and click *Save*.
e. Select the *ASE Block Message*.
f. Edit the message and click *Save*.

3. Create the *Authentication* replacement message group:
   a. Go to *System > Replacement Message Groups*.
   b. Click *Create New*.
   c. For *Name*, enter *newauth*.
   d. In the *Comments* field, enter *Authentication message group*.
   e. For *Group Type*, select *Authentication*.
   f. Click *OK*.



4. Apply the *newutm* replacement message group to an email filter profile in the CLI:

```
config emailfilter profile
    edit "newmsgs"
        set replacemsg-group "newutm"
    next
end
```

5. Apply the *newauth* replacement message group and the email filter profile to a firewall policy in the CLI:

```
config firewall policy
    edit 1
        ...
        set replacemsg-override-group "newauth"
        set inspection-mode proxy
        set emailfilter-profile "newmsgs"
        ...
    next
end
```

To create replacement message groups in the CLI:

1. Create the replacement message groups:

```
config system replacemsg-group
    edit "newutm"
        set comment "UTM message group"
        set group-type utm
        config mail
            edit "partial"
                set buffer "Fragmented emails are blocked, sorry."
            next
        end
        config spam
            edit "smtp-spam-ase"
                set buffer "This message has been blocked because ASE reports it as
spam. You\'re welcome."
            next
        end
    next
    edit "newauth"
        set comment 'Authentication message group'
        set group-type auth
        config auth
            edit "auth-success-msg"
                set buffer "Welcome to the firewall. Your authentication has been
accepted, please reconnect."
            next
        end
    next
end
```

2. Apply the message group to the email filter:

```
config emailfilter profile
    edit "newmsgs"
        set replacemsg-group "newutm"
    next
end
```

3. Apply the email filter and message group to the policy:

```
config firewall policy
    edit 1
        ...
        set replacemsg-override-group "newauth"
        set inspection-mode proxy
        set emailfilter-profile "newmsgs"
        ...
    next
end
```

FortiGuard

FortiGuard services can be purchased and registered to your FortiGate unit. The FortiGate must be connected to the Internet in order to automatically connect to the FortiGuard Distribution Network (FDN) to validate the license and download FDN updates.

The FortiGuard subscription update services include:

- Antivirus (AV)
- Intrusion Protection Service (IPS)
- Application Control
- Antispam
- Web Filtering
- Web Application Firewall (WAF)

AV and IPS packages are signed by the Fortinet CA, and automatic updates will validate whether a package is signed.

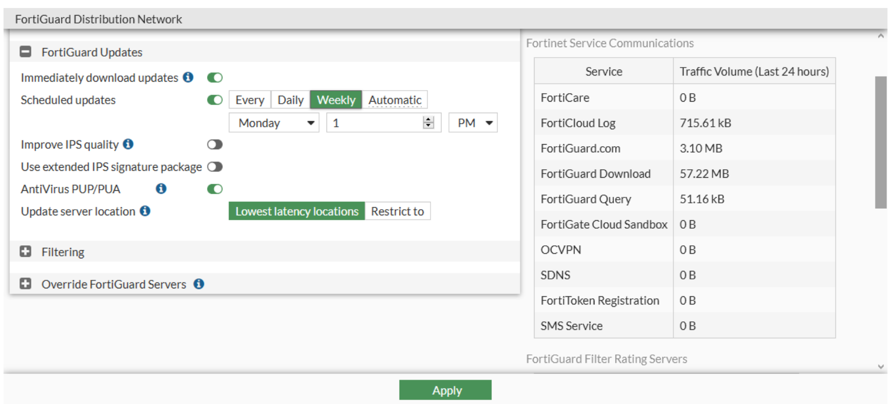
To view FDN support contract information, go to *System > FortiGuard*. The *License Information* table shows the status of your FortiGate's support contract.

Configuring FortiGuard updates

To configure FortiGuard updates:

1. Go to *System > FortiGuard*
2. Scroll down to the *FortiGuard Updates* section.
3. Configure the options for connecting and downloading definition files:

Immediately download updates	The option can be enabled on 2U and larger hardware models when the FortiGuard are servers are connected in anycast mode. The FortiGate forms a secure, persistent connection with FortiGuard to get notifications of new updates through an HTTPS connection. The FortiGate uses the fds_notify daemon to wait for the notification, then makes another connection to the FortiGuard server to download the updates.
Scheduled Updates	Enable to schedule updates to be sent to the FortiGate at the specified time or automatically. See Scheduled updates on page 2146 and Automatic updates on page 2145.
Improve IPS quality	Enable to send information to the FortiGuard servers when an attack occurs. This can help keep the FortiGuard database current as attacks evolve, and improve IPS signatures.
Use extended IPS signature package	Enable to use the extended IPS database, that includes protection from legacy attacks, along with the regular IPS database that protects against the latest common and in-the-wild attacks.
AntiVirus PUP/PUA	Enable antivirus grayware checks for potentially unwanted applications.
Update server location	The FortiGuard update server location. See Update server location on page 2147 for details.



4. Click *Apply*.

Configuring a proxy server for FortiGuard updates

You can configure FortiOS to use a proxy server to connect to the FortiGuard Distribution Network (FDN).

Proxy tunneling is supported only for registration, AV, and IPS updates. For FortiGate virtual machines, proxy tunneling can also be used for license validation. For web filtering or spam filtering, UDP protocol is used on ports 53 or 8888. UDP protocol traffic cannot be directed over a proxy server, even if you are using versions of FortiOS that support web filtering over port 443.

Consider the following before configuring FortiOS to use a proxy server to connect to FDN:

- FortiOS connects to the proxy server using the HTTP CONNECT method. For information about the HTTP CONNECT method, see RFC 2616.
- The proxy server must not inspect the HTTPS traffic used for FortiOS communication.
- FortiOS sends to the proxy server an HTTP CONNECT request that specifies the IP address and port required for the FDN connection. Authentication information is optional for the request.
- FortiOS must be configured to use DNS servers that resolve the addresses of FDN servers to support AV and IPS updates.
- The proxy server establishes the connection to FDN and passes information between FortiOS and FDN.

Use the following syntax to configure a proxy server in the CLI:

```
config system autoupdate tunneling
    set address <proxy_address>
    set port <proxy_port>
    set username <username>
    set password <password>
    set status {enable | disable}
end
```

In the following example, a proxy server with IP address 10.1.1.1 is configured to listen on port TCP/3128 without authentication.

To configure a proxy server:

```
config system autoupdate tunneling
    set address 10.1.1.1
    set port 3128
    set status enable
end
```

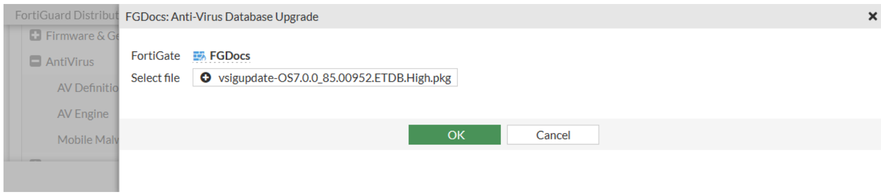
In a closed network without direct internet connection for web filtering or spam filtering, you can use FortiManager as a FortiGuard server. FortiManager supports proxy for both updates and rating, and FortiOS retrieves its updates and ratings through FortiManager. See Using FortiManager as a local FortiGuard server on page 2153.

Manual updates

When needed, FortiGuard Distribution Network (FDN) updates can be manually uploaded.

To manually update the signature definitions files:

1. Log in to the Fortinet Support website.
2. Go to *Support > Service Updates*.
3. Select your *OS Version* from the dropdown list.
4. Locate your device in the table, and download the signature definitions files.
5. On the FortiGate, go to *System > FortiGuard*.
6. In the *License Information* table, locate and expand the definitions that you are updating.
7. From the *Actions* menu in the rightmost column, select *Upgrade Database*.
8. In the pane that opens, click *Upload*, locate the downloaded definitions file on your computer, then click *Open*. The download may take a few minutes to complete.



9. Click *OK*.

AV and IPS manual updates

AV and IPS packages are signed by the Fortinet CA to ensure authenticity of the packages. During manual package updates, signed and validated packages will be accepted. If a package is not signed, the following applies:

- Level-0: accept the new package even if it is unsigned.
- Level-1: display a warning and request a user confirmation to accept.
- Level-2: display an error and reject the image.
- If no level is configured, apply Level-1.

Security levels are pre-configured on the BIOS.

To execute the update:

```
# execute restore ips tftp nids-720-19.261.pkg 172.16.200.55
```

To verify the manual AV and IPS package updates:

```
# diagnose debug app updated -1
# diagnose debug enable
```

Manual update of an unsigned package with level-1 configured

A warning message is displayed in the console, and requests a user confirmation to accept the update of an unsigned package.

To execute the update:

```
# execute restore ips tftp nids-720-19.261.pkg 172.16.200.55
This operation will overwrite the current IPS package!
Do you want to continue? (y/n)y

Please wait...

Connect to tftp server 172.16.200.55 ...
##

Get IPS database from tftp server OK.
******WARNING: This package file has no signature for validation.******
Fortinet cannot verify the authenticity of this package and therefore
```

```
there may be a risk that the package contains code unknown to Fortinet.
In short, Fortinet cannot validate the package and makes no warranties
or representations concerning the package.
Please continue only if you understand and are willing to accept the risks.
Do you want to continue? (y/n)y
```

Manual update of an unsigned package with level-2 configured

A warning message is displayed in the console, and the image is rejected.

To execute the update:

```
# execute restore ips tftp nids-720-19.261.pkg 172.16.200.55
This operation will overwrite the current IPS package!
Do you want to continue? (y/n)y

Please wait...

Connect to tftp server 172.16.200.55 ...
##

Get IPS database from tftp server OK.
```

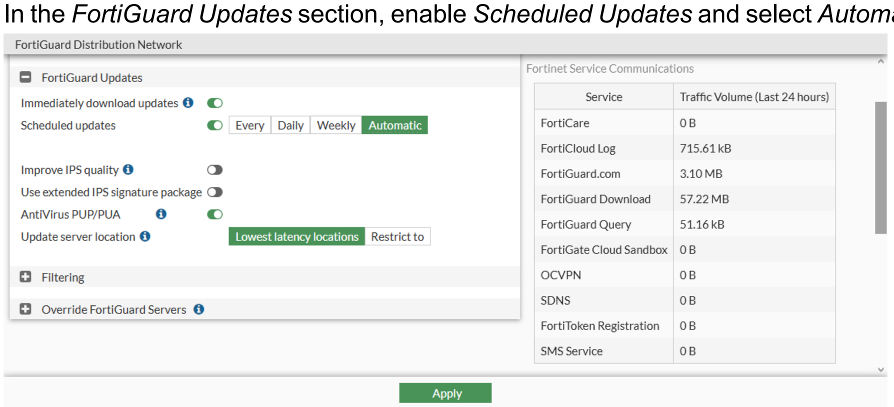
Automatic updates

The default auto-update schedule for FortiGuard packages is automatic. The update interval is calculated based on the model and percentage of valid subscriptions, within one hour.

For example, if a FortiGate 501E has 78% valid contracts, then based on this device model, the update schedule is calculated to be every 10 minutes. If you verify the system event logs (ID 0100041000), they are generated approximately every 10 minutes.

To configure automatic updates in the GUI:

1. Go to *System > FortiGuard*
2. In the *FortiGuard Updates* section, enable *Scheduled Updates* and select *Automatic*.



3. Click *Apply*.

To configure scheduled updates in the CLI:

```
config system autoupdate schedule
    set status enable
    set frequency automatic
end
```

Scheduled updates

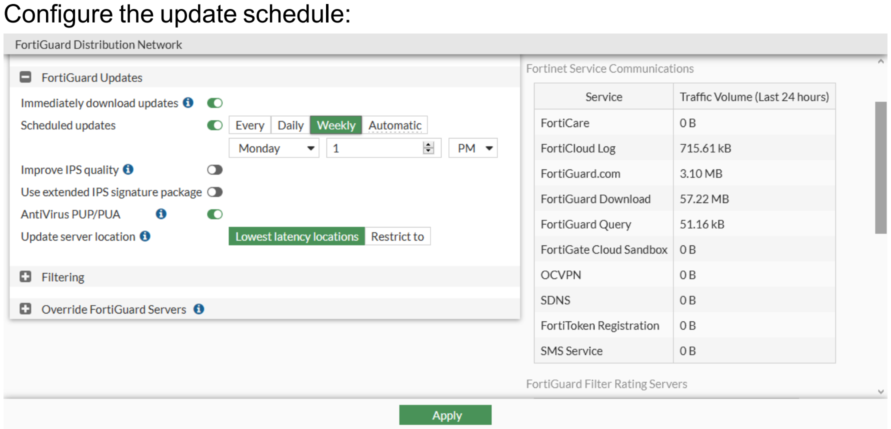
Scheduling updates ensures that the virus and IPS definitions are downloaded to your FortiGate on a regular basis.

Updating definitions can cause a brief disruption in traffic that is currently being scanned while the FortiGate unit applies the new signature database. Updates should be scheduled during off-peak hours when network usage is at a minimum to ensure that network activity will not be affected by downloading the definitions files.

A schedule of once a week means any urgent updates will not be pushed until the scheduled time. If an urgent update is required, click the *Update Licenses & Definitions Now* button to manually update the definitions.

To configure scheduled updates in the GUI:

1. Go to *System > FortiGuard*
2. In the *FortiGuard Updates* section, enable *Scheduled Updates*.
3. Configure the update schedule:



4. Click *Apply*.

To configure scheduled updates in the CLI:

```
config system autoupdate schedule
    set status enable
    set frequency {every | daily | weekly}
    set time <hh:mm>
    set day <day_of_week>
end
```

Sending malware statistics to FortiGuard

FortiGate devices periodically send encrypted antivirus, IPS, botnet IP list, and application control statistics to FortiGuard. Included with these data is the IP address and serial number of the FortiGate, and the country that it is in. This information is never shared with external parties, Fortinet Privacy Policy.

The malware statistics are used to improve various aspects of FortiGate malware protection. For example, antivirus data allow FortiGuard to determine what viruses are currently active. Signatures for those viruses are kept in the Active AV Signature Database that is used by multiple Fortinet products.Inactive virus signatures are moved to the Extended AV Signature Database (see Configuring FortiGuard updates on page 2141). When events for inactive viruses start appearing in the malware data, the signatures are moved back into the AV Signature Database.

The FortiGate and FortiGuard servers go through a 2-way SSL/TLS 1.2 authentication before any data is transmitted. The certificates used in this process must be trusted by each other and signed by the Fortinet CA server.

The FortiGate only accepts data from authorized FortiGuard severs. Fortinet products use DNS to find FortiGuard servers and periodically update their FortiGate server list. All other servers are provided by a list that is updated through the encrypted channel.

Malware statistics are accumulated and sent every 60 minutes by default.

To configure sharing this information, use the following CLI command:

```
config system global
    set fds-statistics {enable | disable}
    set fds-statistics-period <minutes>
end
```

The submission of malware data is in accordance with the Fortinet Privacy Policy.

There is no sensitive or personal information included in these submissions. Only malware statistics are sent.

Fortinet uses the malware statistics collected in this manner to improve the performance of the FortiGate services and to display statistics on the Fortinet Support website for customers registered FortiGate devices.

Fortinet may also publish or share statistics or results derived from this malware data with various audiences. The malware statistics shared in this way do not include any customer data.

Update server location

Administrators can specify the location of the FortiGuard update server used by FortiGate. You can can set the location to only servers in the USA, only servers in the European Union (EU), or to servers with the lowest latency.

In EU locations, it can be required that certain traffic is only handled by servers located in the EU. By setting the update server location to EU only, the FortiGate will use EU domains to resolve to EU servers for FortiGuard traffic to update, URL rating, and IoT servers.

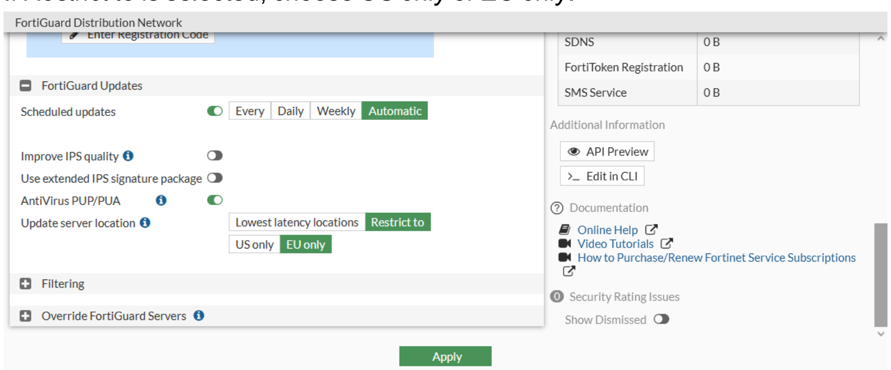
Server location	Anycast domain name	Non-Anycast FQDN addresses
EU only	euupdate.fortinet.net euguardservice.fortinet.net	

Server location	Anycast domain name	Non-Anycast FQDN addresses
US only	usupdate.fortinet.net usguardservice.fortinet.net	usupdate.fortiguard.net UDP: usservice.fortiguard.net HTTPS: ussecurewf.fortiguard.net
Lowest latency (automatic)	globalupdate.fortinet.net globalguardservice.fortinet.net	update.fortiguard.net UDP: service.fortiguard.net HTTPS: securewf.fortiguard.net

On hardware FortiGate devices, the default is *Lowest latency locations*. On VM devices, the default is *US only*.

To configure the update server location in the GUI:

1. Go to *System > FortiGuard*
2. In the *FortiGuard Updates* section, set *Update server location* to *Lowest latency locations* or *Restrict to*.
3. If *Restrict to* is selected, choose *US only* or *EU only*.



4. Click *Apply*.

To configure the update server location in the CLI:

```
config system fortiguard
    set update-server-location {automatic | usa | eu}
end
```

Filtering

Web filtering is used to block access to harmful, inappropriate, and dangerous web sites (see FortiGuard filter on page 1128).

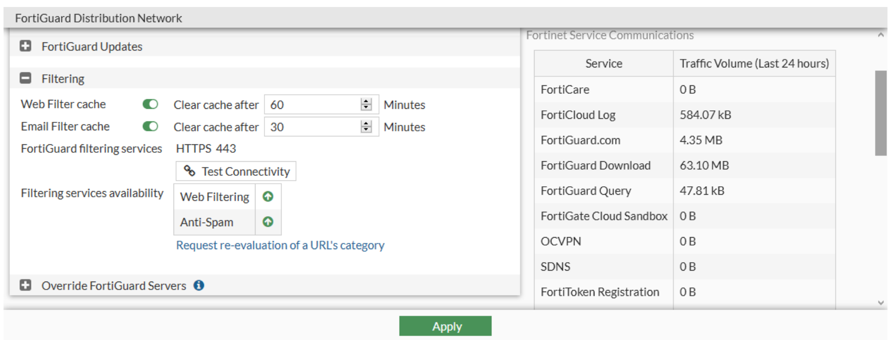
Email filtering is used to detect and block spam messages (see FortiGuard-based filters on page 1240).

To configure filtering in the GUI:

1. Go to *System > FortiGuard*
2. Scroll down to the *Filtering* section.

3. Configure the settings as needed:

Web Filter Cache	Enable/disable web filter cache, and set the amount of time that the FortiGate will store a blocked IP address or URL locally. After the time expires, the FortiGate contacts the FDN to verify the address.
Email Filter Cache	Enable/disable email filter cache, and set the amount of time that the FortiGate will store an email address locally.
FortiGuard filtering services	The protocol and port used to contact the FortiGuard servers. These options can be changed in the CLI.
Filtering service availability	The status of the filtering service. Click *Test Connectivity* if the filtering service is not available.
Request re-evaluation of a URL's category	Click to re-evaluate a URL category rating on the FortiGuard web filter service.



4. Click *Apply*.

To configure filtering in the CLI:

```
config system fortiguard
    set protocol {https | udp}
    set port {443 | 53 | 8888}
    set antispam-force-off {enable | disable}
    set antispam-cache {enable | disable}
    set antispam-cache-ttl <integer>
    set antispam-cache-mpercent <percent>
    set antispam-timeout <integer>
    set webfilter-force-off {enable | disable}
    set webfilter-cache {enable | disable}
    set webfilter-cache-ttl <integer>
    set webfilter-timeout <integer>
end
```

When anycast is enabled (by default) the protocol is HTTPS and the port is 443.

Online security tools

FortiGuard Labs provides a number of online security tools, including but not limited to:

- **URL lookup**
  Enter a website address to see if it has been rated and what category and classification it is filed as. If you find a site that has been wrongly categorized, use this page to request that the site be re-evaluated: https://www.fortiguard.com/webfilter
- **Threat Encyclopedia**
  Browse FortiGuard Labs extensive encyclopedia of threats. Search for viruses, botnet C&C, IPS, endpoint vulnerabilities, and mobile malware: https://www.fortiguard.com/encyclopedia
- **Application Control**
  Browse FortiGuard Labs extensive encyclopedia of applications: https://www.fortiguard.com/appcontrol

FortiGuard anycast and third-party SSL validation

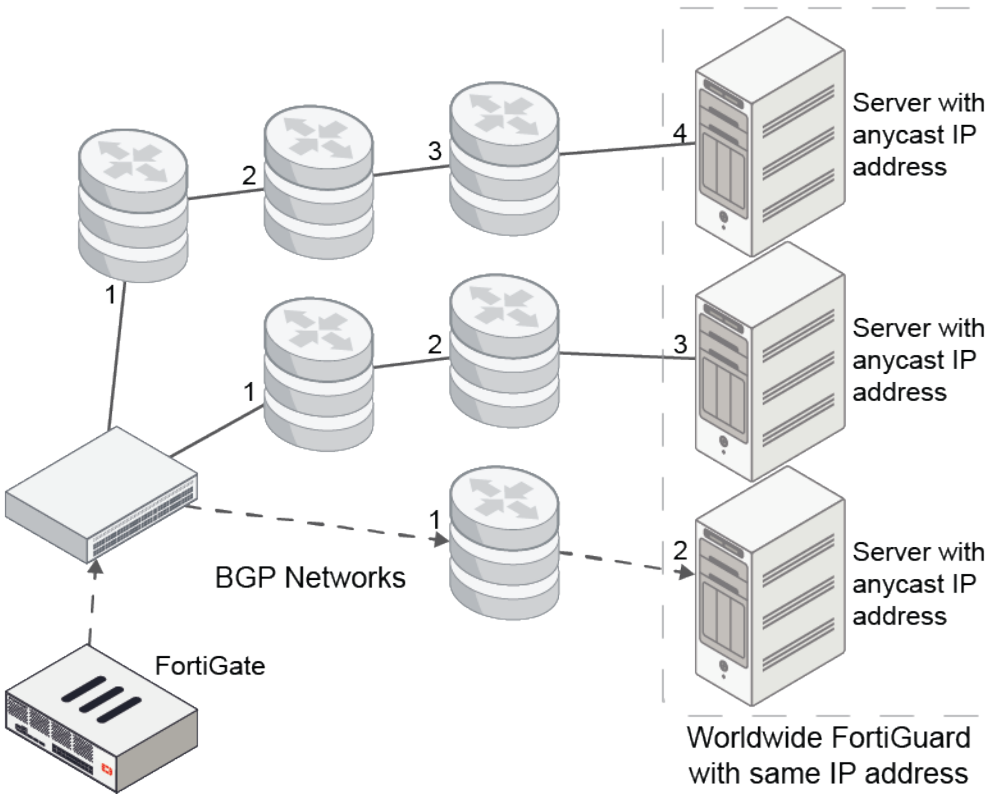
Anycast optimizes routing performance to FortiGuard servers. It is the default FortiGuard access mode.

Using Fortinet DNS servers, the FortiGate receives a single IP address for the domain name of each FortiGuard service. BGP routing optimization is transparent to the FortiGate. The domain name of each FortiGuard service is the common name in that service's certificate, which is signed by a third-party intermediate CA. The FortiGuard server uses third-party certificate verification and the Online Certificate Status Protocol (OCSP) stapling check, so that the FortiGate can always validate the FortiGuard server certificate efficiently.

FortiGate will only complete the TLS handshake with an anycast server that has a good OCSP status for its certificate. Any other status will result in a failed SSL connection. OCSP stapling is reflected on the signature interval so that *good* means that the certificate is not revoked at that timestamp. The FortiGuard servers query the CA's OCSP responder every four hours and update its OCSP status. If the FortiGuard is unable to reach the OCSP responder, it will keep the last known OCSP status for up to seven days. This cached OCSP status will be sent out immediately when a client connection request is made, optimizing the response time.

FortiGuard represents all cloud based servers; see Anycast and unicast services for details.

The anycast server has one IP address to match its domain name. The FortiGate connects with a single server address, using HTTPS and port 443, regardless of where the FortiGate is located.

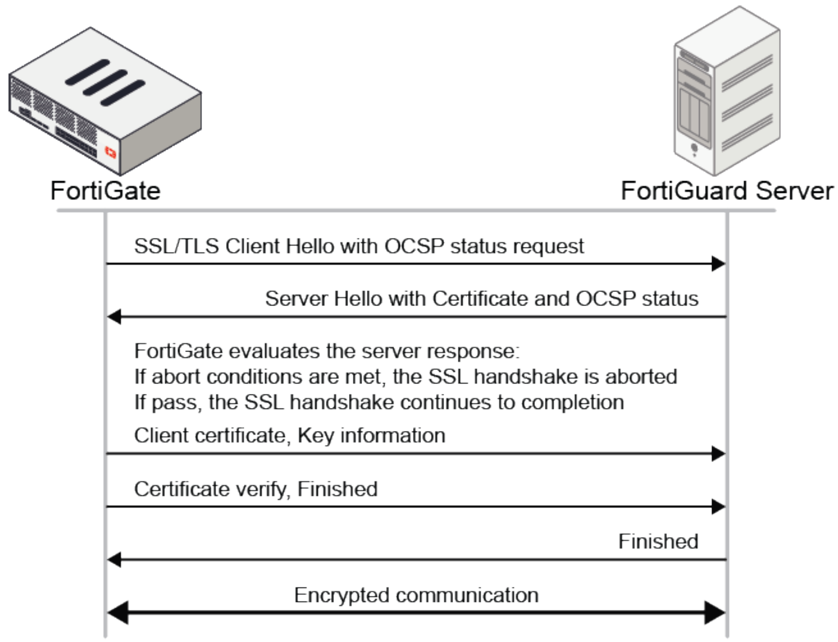


To configure the anycast FortiGuard access mode:

```
config system fortiguard
    set fortiguard-anycast {enable | disable}
    set fortiguard-anycast-source {fortinet | aws}
end
```

Connection process

The following process is used to connect to an anycast server:

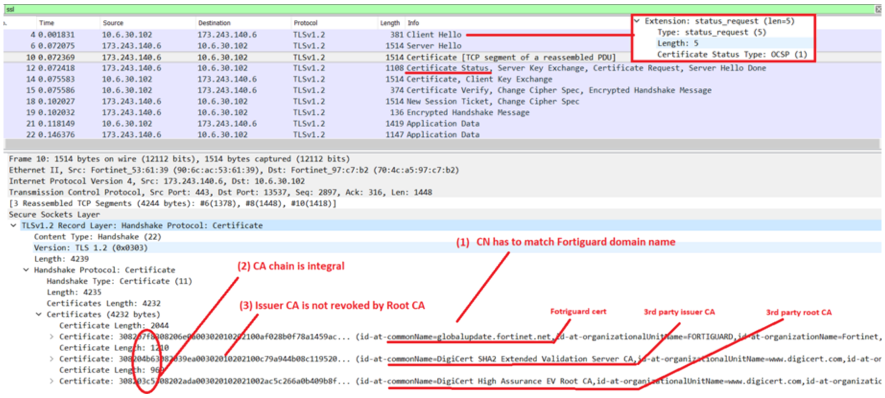


1. The FortiGate embeds the CA_bundle certificate, which includes the root CA with CRL list and third-party intermediate CA, in the root CA level.
2. The FortiGate finds the FortiGuard IP address from its domain name from DNS.
3. The FortiGate starts a TLS handshake with the FortiGuard IP address. The client hello includes an extension of the *status request*.
4. The FortiGuard servers provide a certificate with its OCSP status: *good*, *revoked*, or *unknown*.
5. The FortiGate verifies the CA chain against the root CA in the CA_bundle.
6. The FortiGate verifies the intermediate CA's revoke status against the root CA's CRL.
7. The FortiGate verifies the FortiGuard certificate's OCSP status:

```
OCSP Response Data:
    OCSP Response Status: successful (0x0)
    Response Type: Basic OCSP Response
    Version: 1 (0x0)
    Responder Id: 3DD350A5D6A0ADEEF34A600A65D321D4F8F8D60F
    Produced At: Aug 20 07:50:58 2019 GMT
    Responses:
    Certificate ID:
      Hash Algorithm: sha1
      Issuer Name Hash: 49F4BD8A18BF760698C5DE402D683B716AE4E686
      Issuer Key Hash: 3DD350A5D6A0ADEEF34A600A65D321D4F8F8D60F
      Serial Number: 02555C9F3901B799DF1873402FA9392D
    Cert Status: good
    This Update: Aug 20 07:50:58 2019 GMT
    Next Update: Aug 27 07:05:58 2019 GMT
```

   Abort conditions include:
   - The CN in the server's certificate does not match the domain name resolved from the DNS.
   - The OCSP status is not good.
   - The issuer-CA is revoked by the root-CA.
8. Once the SSL handshake is established, the FortiGate can engage the server.

Example Wireshark PCAP:



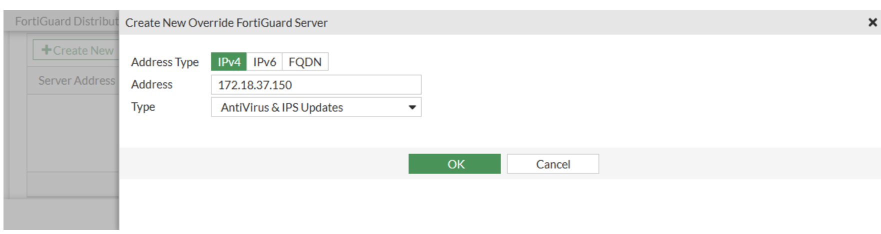
Using FortiManager as a local FortiGuard server

FortiManager can provide a local FortiGuard server with port 443 access.

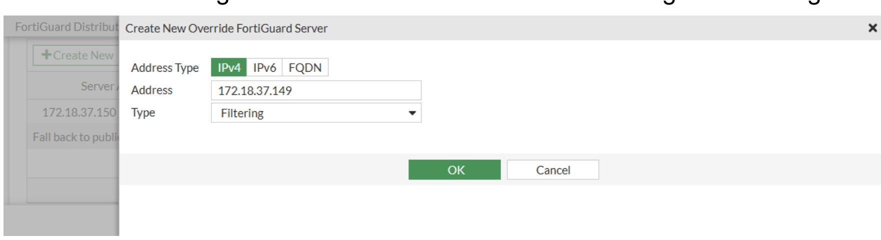
Anycast FortiGuard settings force the rating process to use port 443, even with an override server. Using a unique address in the same subnet as the FortiManager access IP address, the FortiManager can provide local FortiGuard updates and rating access with a dedicated IP address and port 443.

To use a FortiManager as a local FortiGuard server in the GUI:

1. Go to *System > FortiGuard*
2. In the *Override FortiGuard Servers* table, click *Create New*. The *Create New Override FortiGuard Server* pane opens.
3. Select the server address type: *IPv4*, *IPv6*, or *FQDN*.
4. Enter the FortiManager address in the *Address* field.
5. Select the type of server: *AntiVirus & IPS Updates*, *Filtering*, or *Both*.



6. Click *OK*.
7. Click *Create New* again to add a second override FortiManager for filtering.



8. Click *OK*, then click *Apply*.

To use a FortiManager as a local FortiGuard server in the CLI:

```
config system central-management
    set type fortimanager
    set fmg "172.18.37.148"
    config server-list
        edit 1
            set server-type update
            set server-address 172.18.37.150
        next
        edit 2
            set server-type rating
            set server-address 172.18.37.149
        next
    end
    set fmg-update-port 443
    set include-default-servers enable
end
```

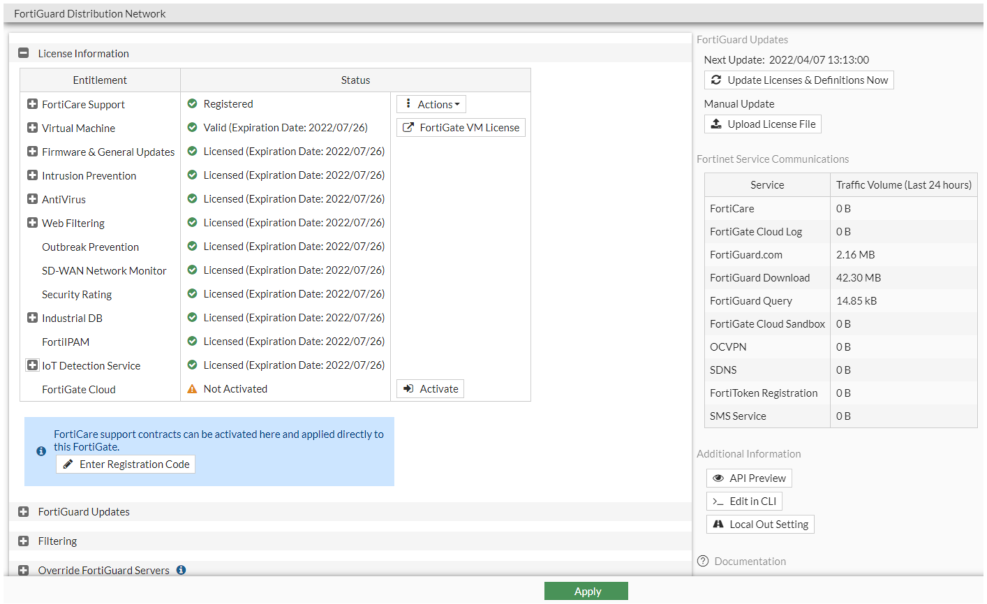
When `fmg-update-port` is set to `443`, the update process will use port 443 to connect to the override update server, which is the local FortiGuard server in the FortiManager. If this is not set, the update process will use port 8890, and the server address setting has to be the FortiManager access IP address. Override FortiGuard services come from the server list that is the local FortiGuard server in the FortiManager, and use the traditional, non-OCSP TLS handshake. If override servers in the FortiManager are not available, the default FortiGuard servers are connected, and the anycast OCSP TLS handshake is used.

Cloud service communication statistics

Fortinet service communications statistics are displayed on the *FortiGuard* page. The statistics correspond with the output from `diagnose sys service-communication`. The traffic volume values in the GUI are the sums of data from the last 24 hours.

To view Fortinet service communications statistics:

1. Go to *System* > *FortiGuard*.
   The *Fortinet Service Communications* statistics are displayed on the right side of the screen:



2. Enter the following CLI command:

```
# diagnose sys service-communication
FortiCare:
The last 1 hour(in bytes):  0 0 0 0 0 0 0 0 0 0 0 0
The last 24 hours(in bytes):  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
The last 7 days(in bytes):  0 0 0 0 0 0 0
FortiGuard Download:
The last 1 hour(in bytes):  0 0 0 336 1992 0 0 0 0 0 0 0
The last 24 hours(in bytes):  0 2328 6752 4450632 0 33696 0 5666528 0 49712 0 28840 0
29840 0 4185832 0 31488 0 76424 0 4226808 0 173880
The last 7 days(in bytes):  14454160 14985496 9532184 0 0 0 0
FortiGuard Query:
The last 1 hour(in bytes):  0 0 0 372 1107 0 0 0 0 0 0 0
The last 24 hours(in bytes):  0 1479 4828 929 0 929 0 929 0 929 0 929 0 929 0 1858 0 929
0 1858 0 1858 0 929
The last 7 days(in bytes):  13739 15793 13624 0 0 0 0
FortiCloud Log:
The last 1 hour(in bytes):  0 343 563 899 1014 405 0 0 0 570 405 0
The last 24 hours(in bytes):  0 4535 6004 2184 684 1906 1938 680 861 1933 685 1020 687
1772 693 978 1023 1574 1195 697 1035 1323 1020 678
The last 7 days(in bytes):  26560 26136 0 0 0 0 0
FortiSandbox Cloud:
The last 1 hour(in bytes):  0 0 0 0 0 0 0 0 0 0 0 0
The last 24 hours(in bytes):  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
The last 7 days(in bytes):  0 0 0 0 0 0 0
FortiGuard.com:
The last 1 hour(in bytes):  0 0 122162 123544 122162 244324 0 0 0 0 0 0
The last 24 hours(in bytes):  0 612192 532887 1939 1143 122162 44924 5039 0 125091 43096
1939 0 123305 43090 1939 0 123305 43096 1939 0 122162 42478 4930
The last 7 days(in bytes):  1658746 1347340 1421746 0 0 0 0
OCVPN Service:
The last 1 hour(in bytes):  1044 9382 0 0 0 0 0 0 0 0 0 0
The last 24 hours(in bytes):  1044 9382 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
The last 7 days(in bytes):  10426 0 0 0 0 0 0
SDNS Service:
```

```
The last 1 hour(in bytes):  0 0 0 0 0 0 0 0 0 0 0 0
The last 24 hours(in bytes):  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
The last 7 days(in bytes):  0 0 0 0 0 0 0
FortiToken Registration:
The last 1 hour(in bytes):  0 0 0 0 0 0 0 0 0 0 0 0
The last 24 hours(in bytes):  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
The last 7 days(in bytes):  0 0 0 0 0 0 0
SMS Service:
The last 1 hour(in bytes):  0 0 0 0 0 0 0 0 0 0 0 0
The last 24 hours(in bytes):  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
The last 7 days(in bytes):  0 0 0 0 0 0 0
```

IoT detection service

Internet of Things (IoT) detection is a subscription service that allows FortiGate to detect unknown devices in FortiGuard that are not detected by the local Device Database (CIDB). When the service is activated, FortiGate can send device information to the FortiGuard collection server. When a new device is detected, FortiGate queries the results from the FortiGuard query for more information about the device.

This feature requires an IoT Detection Service license.

FortiGate device requirements:

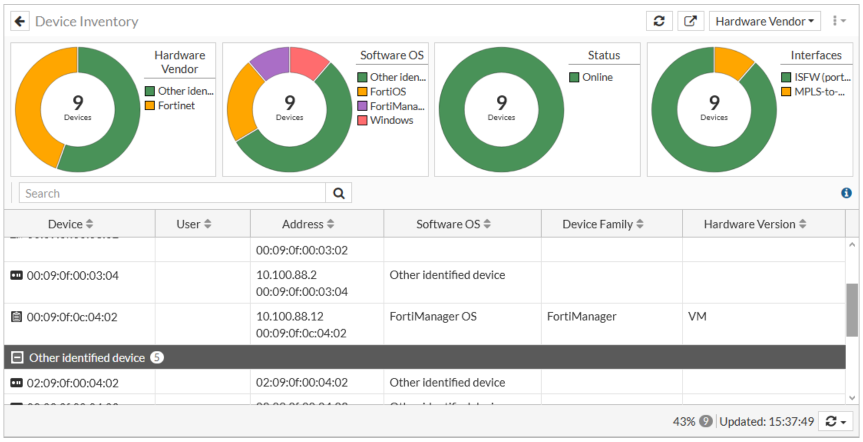
The FortiGate device must be:

- Registered with FortiCare
- Connected to an anycast FortiGuard server

How the service works:

1. Enable Device Detection on an interface.
2. FortiGate uses the interface to detect device traffic flow.
3. Upon detecting traffic from an unknown device, FortiGate sends the device data to the FortiGuard collection server.
4. The collection server returns data about the new device to the FortiGuard query server.
5. If the device signature does not appear in the local Device Database (CIDB) or some fields are not complete, FortiGate queries FortiGuard for more information about the device.

To view the latest device information in the GUI, go to *Dashboard > Users & Devices* and expand the *Device Inventory* widget.



To debug the daemon in the CLI:

1. Disable the local device database in order to force all queries to go to FortiGuard.

```
# diagnose cid sigs disable
```

2. Enable iotd debugs.

```
# diagnose debug application iotd -1
# diagnose debug enable
```

FortiGate sends the device data to the FortiGuard collection server.

```
FortiWiFi-60E # [iotd] recv request from caller size:61
[iotd] service:collect hostname: ip: fd:-1 request tlv_len:41
[iotd] txt(.....y...w.....Jasons-iPhone6....579=23..)
[iotd] hex
      (02010007017903060f77fc0203000e4a61736f6e732d6950686f6e653602040008353739 3d32330cf
      f)
[iotd] service:collect hostname:qadevcollect.fortinet.net ip: fd:-1 got server hostname
[iotd] service:collect hostname:qadevcollect.fortinet.net ip:192.168.100.133 fd:-1 got
      server ip
[iotd] service:collect hostname:qadevcollect.fortinet.net ip:192.168.100.133 fd:13
      socket created
[iotd] service:collect hostname:qadevcollect.fortinet.net ip:192.168.100.133 fd:13
      connecting
[iotd] fd:13 monitor event:pollout
[iotd] service:collect hostname:qadevcollect.fortinet.net ip:192.168.100.133 fd:13 build
      req packet
[iotd] service:collect hostname:qadevcollect.fortinet.net ip:192.168.100.133 fd:13
      collect resp:1(pending)
```

The FortiGuard collection server returns new device data to the FortiGuard query server.

```
[iotd] service:query hostname:qadevquery.fortinet.net ip:192.168.100.248 fd:17 got query
      resp
[iotd] service:query hostname:qadevquery.fortinet.net ip:192.168.100.248 fd:17 id:0
      total_len:48 header_len:16 tlv_len:32 confidence:100 mac:f8:87:f1:1f:ab:95
[iotd] service:query hostname:qadevquery.fortinet.net ip:192.168.100.248 fd:17
      remaining_len:32 type:1 len:6
[iotd] service:query hostname:qadevquery.fortinet.net ip:192.168.100.248 fd:17 got tlv
      category:'Mobile'
[iotd] service:query hostname:qadevquery.fortinet.net ip:192.168.100.248 fd:17
      remaining_len:24 type:2 len:6
[iotd] service:query hostname:qadevquery.fortinet.net ip:192.168.100.248 fd:17 got tlv
      sub_category:'Mobile'
```

```
[iotd] service:query hostname:qadevquery.fortinet.net ip:192.168.100.248 fd:17
     remaining_len:16 type:3 len:5
[iotd] service:query hostname:qadevquery.fortinet.net ip:192.168.100.248 fd:17 got tlv
     vendor:'Apple'
[iotd] service:query hostname:qadevquery.fortinet.net ip:192.168.100.248 fd:17
     remaining_len:9 type:4 len:0
[iotd] service:query hostname:qadevquery.fortinet.net ip:192.168.100.248 fd:17
     remaining_len:7 type:5 len:3
[iotd] service:query hostname:qadevquery.fortinet.net ip:192.168.100.248 fd:17 got tlv
     os:'iOS'
[iotd] service:query hostname:qadevquery.fortinet.net ip:192.168.100.248 fd:17
     remaining_len:2 type:6 len:0
[iotd] service:query hostname:qadevquery.fortinet.net ip:192.168.100.248 fd:17 send
     query response to caller size:48
[iotd] txt(...........d0 ...Mobile..Mobile..Apple....iOS..)
[iotd] hex
     (f887f11fab950000000000006430200001064d6f62696c6502064d6f62696c6503054170706c650400
     0503694f530600)
[iotd] service:query hostname:qadevquery.fortinet.net ip:192.168.100.248 fd:17 read
     resp:0(good)
```

3. The query returns the device information including the information source (`src fortiguard`).

```
# diagnose user device list
  vd root/0  f8:87:f1:1f:ab:95  gen 26  req OUA/34
    created 503s  gen 23  seen 102s  lan  gen 7
    ip 192.168.1.110  src arp
    hardware vendor 'Apple'  src fortiguard  id 0  weight 100
    type 'Mobile'  src fortiguard  id 0  weight 100
    family 'Mobile'  src fortiguard  id 0  weight 100
    os 'iOS'  src fortiguard  id 0  weight 100
    host 'Jasons-iPhone6'  src dhcp
```

FortiAP query to FortiGuard IoT service to determine device details

A FortiAP collects packets from devices and queries FortiGuard with the help of the FortiGate. Device detection results are reported back to the FortiGate where this information is displayed. Querying the FortiGuard service requires an IoT Detection Service license.

The following attributes can be configured in `wireless-controller setting`:

Attribute	Description
device-weight <integer>	Set the device upper limit of confidence (0 - 255, default = 1, 0 = disable).
device-holdoff <integer>	Set the device lower limit of creation time, in minutes (0 - 60, default = 5).
device-idle <integer>	Set the device upper limit of idle time, in minutes (0 - 14400, default = 1440).

To query the FortiGuard IoT service:

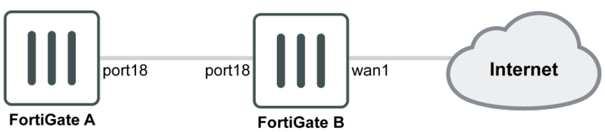
```
config wireless-controller setting
    ...
    set device-weight 1
    set device-holdoff 5
    set device-idle 1440
```

```
    ...
end

# diagnose user device list
  vd root/0  54:27:1e:e6:26:3d  gen 89  req OUA/34
    created 70s  gen 86  seen 2s  port29  gen 28
    ip 10.29.1.214  src mac
    hardware vendor 'Asustek compute'  src fortiguard  id 0  weight 21
    type 'Home & Office'  src fortiguard  id 0  weight 21
    family 'Computer'  src fortiguard  id 0  weight 21
    os 'Linux'  src dhcp  id 822  weight 128
    host 'test-wifi'  src dhcp
```

FortiGate Cloud / FDN communication through an explicit proxy

Explicit proxy communication to FortiGate Cloud and FortiGuard servers from FortiGate is enabled. A proxy server can be configured in the FortiGuard settings so that all FortiGuard connections under the `forticldd` process can be established through the proxy server.

Not all FortiGuard services are supported by these proxy settings. For example, web filter service traffic to FortiGuard will not be directed to the configured proxy.



To configure a proxy server and communicate with FortiGate Cloud though it:

1. Configure FortiGate B as a proxy server:

```
config firewall proxy-policy
    edit 1
        set proxy explicit-web
        set dstintf "wan1"
        set srcaddr "all"
        set dstaddr "all"
        set service "webproxy"
        set action accept
        set schedule "always"
        set logtraffic all
        set users "guest1"
    next
end
config user local
    edit "guest1"
        set type password
        set passwd 123456
    next
end
config authentication scheme
    edit "local-basic"
```

```
        set method basic
        set user-database "local-user-db"
    next
end
config authentication rule
    edit "local-basic-rule"
        set srcaddr "all"
        set ip-based disable
        set active-auth-method "local-basic"
    next
end
```

2. Configure a firewall policy on FortiGate B to allow FortiGate A to get DNS resolution:

```
config firewall policy
    edit 1
        set name "dns"
        set srcintf "port18"
        set dstintf "wan1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "DNS"
        set fsso disable
        set nat enable
    next
end
```

3. Configure the FortiGuard proxy settings on FortiGate A:

```
config system fortiguard
    set proxy-server-ip 10.2.2.2
    set proxy-server-port 8080
    set proxy-username "guest1"
    set proxy-password 123456
end
```

4. On FortiGate A, log in to FortiGate Cloud to activate the logging service:

```
execute fortiguard-log login <username> <password>
```

5. On FortiGate A, view the `forticldd` debug message to see the connection to the log controller through the proxy server:

```
# diagnose test application forticldd 1
```

FDS-only ISDB package in firmware images

FortiOS firmware images include Fortinet objects in the built-in Internet Service Database (ISDB).

```
# diagnose firewall internet-service list
List internet service in kernel(global):
Internet Service Database Kernel Table: size 14974 bytes, Entry size 5844 bytes, number of
index entries 165 number of IP range entries 0

Group(0): Weight(15), number of entries(162)
......
```

This lightweight ISDB package allows firewall rules and policy routes that use ISDB to access FortiGuard servers to continue working after upgrading FortiOS. For example, the following policy will work after an upgrade:

```
config firewall policy
    edit 440
        set name "Fortinet Updates"
        set srcintf "port25"
        set dstintf "port1"
        set srcaddr "FortiAnalyzer" "FortiAuthenticator" "Tesla Management Interface"
"BackupFortinet" "SipFW" "ConnectVPNMgmt"
        set internet-service enable
        set internet-service-id 1245187 1245326 1245324 1245325 1245193 1245192 1245190
1245185
        set action accept
        set schedule "always"
        set logtraffic all
        set fsso disable
    next
end
```

After the FortiGate reboots after a firmware update, an automatic update will run in five minutes so that the FortiGate can get the ISDB, whether or not scheduled update is enabled.

```
# diagnose autoupdate versions | grep Internet -A 6

Internet-service Full Database
---------
Version: 7.02217 signed
Contract Expiry Date: n/a
Last Updated using manual update on Thu Mar 10 12:06:58 2022
Last Update Attempt: Thu Mar 10 12:07:27 2022
```

Licensing in air-gap environments

In the Operational Technology industry, industrial equipment is critical and must not be connected to the internet. However, the equipment is still required to be protected by a firewall in this air-gap environment. Without a gateway to FortiGuard in air-gap environments, FortiGuard packages, such as AntiVirus and IPS, must be manually uploaded to the FortiGate. FortiGate licenses can be downloaded from FortiCloud and uploaded manually to the FortiGate.

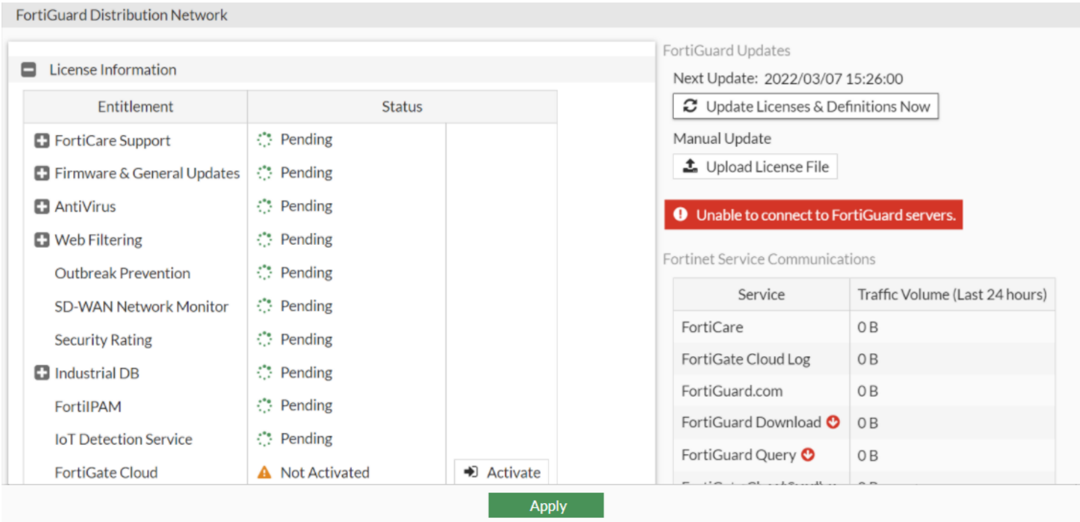
Manual licensing for air-gap environments is supported only on FortiGate hardware appliances, for both rugged and non-rugged models running FortiOS 7.2.0 or later. Manual licensing is currently not supported on FortiGate virtual machine (VM) appliances.

To manually upload FortiGate licenses in the GUI:

1. Register the FortiGuard license on FortiCloud. See Registration in the FortiOS Administration Guide for more information.
2. Download the product entitlement file in FortiCloud:
   a. Go to *Products* > *Product List*.
   b. Select the serial number of the FortiGate. The product page opens.

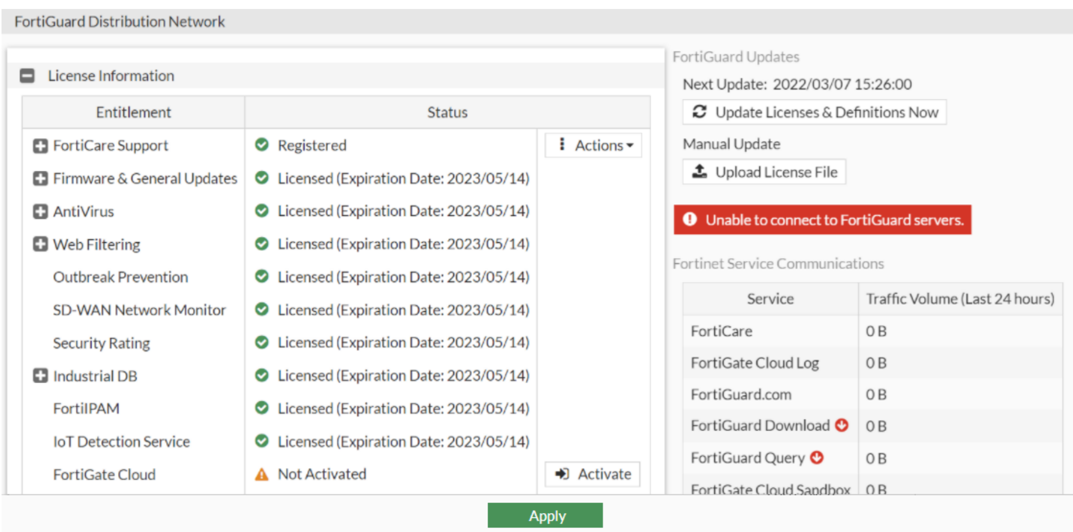
   c. In the *License & Key* section, click *Get The License File*. The file downloads to your device in the format `FG201E*********ProductEntitlement.lic`.

3. In FortiOS, go to *System > FortiGuard*. Currently, the status for all services is *Pending*.



4. Click *Upload License File*. The file explorer opens.
5. Navigate to the product entitlement file and click *Open*.

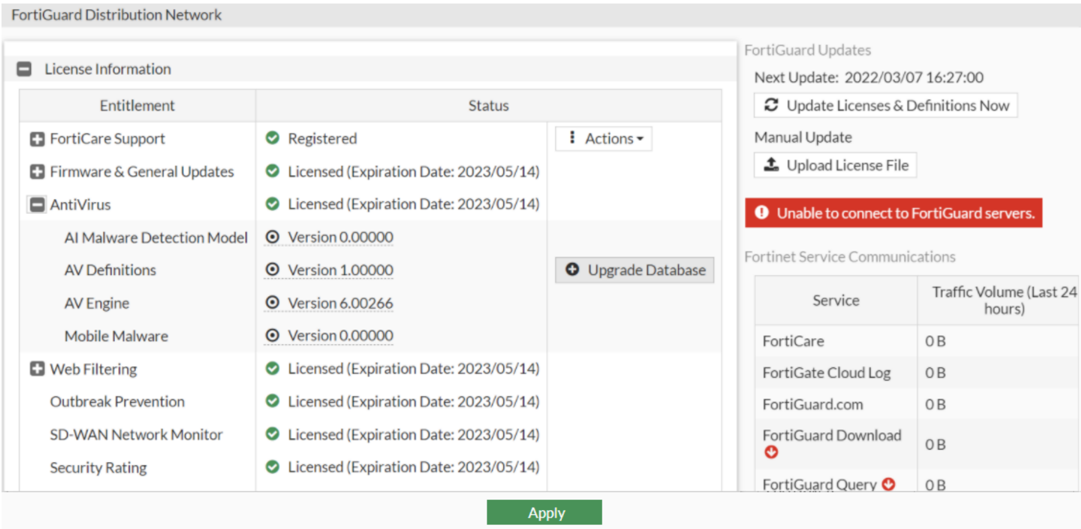
   The license file uploads to the FortiGate. Once the upload is complete, the FortiGate shows that it is registered and licensed.



6. Click *Apply*.

To manually upgrade the AntiVirus Database in the GUI:

1. Download the static upgrade file from FortiCloud:
   a. Go to support.fortinet.com.
   b. Go to *Download > Download FortiGuard Service Updates > FortiGate*.
   c. Select the FortiOS version from the *OS Version* dropdown.
   d. Select the file from the appropriate FortiGate product model section. The file downloads to your device.
2. In FortiOS, go to *System > FortiGuard* and expand the *AntiVirus* section to view the current licenses.



3. Click *Upgrade Database*. The *Anti-Virus Database Upgrade* pane opens.
4. Click *Upload*. The file explorer opens.
5. Navigate to the static upgrade file and click *Open*.
6. Click *OK*.
7. Click *Apply*.
   The AntiVirus Database is upgraded.

To manually upload FortiGate licenses in the CLI:

```
# execute restore manual-license {ftp | tftp} <license file name> <server> [args]
```

Feature visibility

Feature visibility is used to control which features are visible in the GUI. This allows features that are not in use to be hidden. Some features are also invisible by default and must be made visible before they can be configure in the GUI.

The visibility of a feature does not affect its functionality or configuration. Invisible features can still be configured using the CLI.

To change the visibility of features:

1. Go to *System > Feature Visibility*.
2. Change the visibility of the features as required.
   For information about what settings each option affects, click on the + icon to the right of the feature name.
   Changes are listed on the right side of the content pane.
3. Click *Apply*.

Certificates

FortiOS leverages certificates in multiple areas, such as VPNs, administrative access, and deep packet inspection. This section contains topics about uploading certificates and provides examples of how certificates may be used to encrypt and decrypt communications, and represent the identity of the FortiGate. This sections assumes the reader has a high level understanding of the public key infrastructure (PKI) system, particularly how entities leverage trusted certificate authorities (CAs) to verify the authenticating party, and how public and private certificate keys work to secure communications.

The certificates feature is hidden by default in FortiOS. In the GUI, go to *System* > *Feature Visibility* and enable *Certificates*.

The following topics provide an overview of how to add certificates to the FortiGate:

- Uploading a certificate using the GUI on page 2164
- Uploading a certificate using the CLI on page 2167
- Uploading a certificate using an API on page 2168

The following topics provide examples of how to use certificates:

- Site-to-site VPN with digital certificate on page 1363
- Procuring and importing a signed SSL certificate on page 2173
- Microsoft CA deep packet inspection on page 2176
- ACME certificate support on page 2181
- Administrative access using certificates on page 2185
- Creating certificates with XCA on page 2186
- Configuring FortiClient EMS on page 2234
- SSL VPN with certificate authentication on page 1658
- SSL VPN with LDAP-integrated certificate authentication on page 1663
- Configuring certificates for SAML SSO on page 2321
- Protecting an SSL server on page 1299

Uploading a certificate using the GUI

On the *System* > *Certificates* page, there are two options to add a certificate: *Generate* (use a certificate signing request) and *Import*.

Generate certificate signing request

Certificate signing requests (CSRs) are used to generate a certificate which is then signed by a CA to create a chain of trust. The CSR includes details of the FortiGate (see table below) and its public key. A CSR is not strictly necessary; some CAs allow you to provide the details of the FortiGate manually, but a CSR helps streamline the process. Selecting *Generate* takes you the *Generate Certificate Signing Request* page to enter the following information:

Certificate Name	Enter the certificate name; this is how it will appear in the *Local Certificates* list.
Subject Information	Specify an ID type: host IP address, domain name (FQDN), or email address.

Optional Information	Although listed as optional, we recommended entering the information for each field in this section. If you are generating a CSR for a third-party CA, you need to insure that these values reflect those listed for your company or organization at said certificate authority. If you are generating a certificate for a Microsoft CA, you need to check with the administrator regarding these values.
Organization Unit	Enter the name of the organizational unit under which the certificate will be issued.
Organization	Enter the overall name of the organization.
Locality(City)	Enter the city where the SSL certificate is located.
State / Province	Some issuers will reject a CSR that has an abbreviated state or province, so enter the full name of the state or province.
Country / Region	Enable the option and select the country from the dropdown.
E-Mail	Enter the email address of the technical contact for the SSL certificate that is being requested.
Subject Alternative Name	This field allows multiple domains to be used in an SSL certificate. Select from email addresses, IP addresses, URIs, DNS names, and so on.
Password for private key	If supplied, this is used as an encryption password for the private key file.
Key Type	Select *RSA* or *Elliptic Curve*.
Key Size	When *Key Type* is *RSA*, select 1024, 1536, 2048, or 4096 for bit-size/strength. We recommend using at least 2048 if your CA can issue certificates of that size.
Curve Name	When *Key Type* is *Elliptic Curve*, select the elliptic curve type: secp256r1, secp384r1, or secp521r1.

Enrollment Method	Select one of the following methods that determines how the CSR will be signed.
	• *File Based*: this will generate a certificate in the certificate menu under *Local Certificate*, which differs from the existing ones because it has no *Subject*, *Comments*, *Issuer*, or *Expires* values in the table. It will also show a *Pending* status because it is only a CSR at the moment and cannot function as a certificate just yet. You can download the CSR to provide to a CA for signing. If you open the CSR file, it should look similar to this:
	`-----BEGIN CERTIFICATE REQUEST-----`
	`MIIC7jCCAdYCAQAwgZUxCzAJBgNVBAYT (… )HEKjDX+Hg==`
	`-----END CERTIFICATE REQUEST-----`
	Next. the CSR file is supplied to a CA for signing and the returned file from the CA should be in .CER format. This file is then uploaded to the FortiGate by going to *System* > *Certificates* > *Import* > *Local Certificate* and uploading the CER file.
	• *Online SCEP*: the Simple Certificate Enrollment Protocol (SCEP) allows

	devices to enroll for a certificate by using a URL and a password. The SCEP server works as a proxy to forward the FortiGate's request to the CA and returns the result to the FortiGate (setting up an SCEP server is beyond the scope of this topic). Once the request is approved by the SCEP server, the FortiGate will have a signed certificate containing the details provided in the CSR.

Import

Although *Import* is often used in conjunction with a CSR, you may upload a certificate to the FortiGate that was generated on its own. This is typical of wildcard certificates (*.domain.tld) where the same certificate is used across multiple devices (FGT.domain.tld, FAZ.domain.tld, and so on), but may be used for individual certificates so long as the information provided to the signing CA matches that of the FortiGate.

When selecting *Import*, there are four options: *Local Certificate*, *CA Certificate*, *Remote Certificate*, and *CRL*.

Local certificate

Local certificates are used by the FortiGate to identify itself, or a service it provides, such as HTTPS administrative access, SSL VPN user portal, or virtual server load balancing where the FortiGate masquerades as the destination server. When selecting *Local Certificate*, four certificate type options appear in the *Import Certificate* pane:

Local Certificate	There is no field to upload a key with this option. Use this option when you have created a CSR on the FortiGate, as the key is generated as part of the CSR process and remains on the FortiGate. You will need to upload a .CER file.
PKCS #12 Certificate	This option takes a specific certificate file type that contains the private key. The certificate will be encrypted and a password must be supplied with the certificate file.
Certificate	This option is intended for certificates that were generated without using the FortiGate's CSR. Since the certificate private key is being uploaded, a password is required. This can be done two ways: • Certificate file and key file (typically .CER and .PEM) • Certificate and key bundle file (typically .PFX)
Automated	This option allows you to configure the Automated Certificate Management Environment (ACME), which allows you to request and use trusted certificates signed by Let's Encrypt (see ACME certificate support on page 2181 for configuration details).

CA certificate

FortiGates come with many CA certificates from well-known certificate authorities pre-installed, just as most modern operating systems like Windows and MacOS. Use this option to add private CA certificates to the FortiGate so that certificates signed by this private CA are trusted by the FortiGate.

For example, a private CA can be used when two FortiGates are establishing a site-to-site VPN tunnel using a certificate not signed by a public or trustworthy CA, or for your LDAPS connection to your corporate AD server that also uses a certificate signed with a private CA in your domain. It is very common to upload a private CA when using PKI user authentication, since most PKI user certificates will be signed by an internal CA.

When selecting *CA Certificate*, two type options appear in the *Import CA Certificate* pane:

Online SCEP	The FortiGate contacts an SCEP server to request the CA certificate.
File	The CA certificate is uploaded directly to the FortiGate.

Remote certificate

Remote certificates are public certificates and contain only the public key. They are used to identify a remote device. For example, when configuring your FortiGate for SAML authentication with the FortiGate as an identity provider (IdP), you can optionally specify the service provider (SP) certificate. However, when configuring your FortiGate as a SP, you must specify the certificate used by the IdP. Both these certificates can be uploaded to the FortiGate as a remote certificate, since the private key is not necessary for its implementation.

CRL

Since it is not possible to recall a certificate, the CRL (certificate revocation list) list details certificates signed by valid CAs that should no longer be trusted. Certificates may be revoked for many reasons, such as if the certificate was issued erroneously, or if the private key of a valid certificate has been compromised. When selecting *CRL*, two import methods are available:

File Based	CAs publish a file containing the list of certificates that should no longer be trusted.
Online Updating	This is the preferred way to keep the list of revoked certificates up to date. Three protocols are offered: HTTP, LDAP, and SCEP.

Uploading a certificate using the CLI

Generate certificate signing request

The generated CSR must be signed by a CA then loaded to the FortiGate. See Generate certificate signing request on page 2164 for more details.

To generate a CSR:

```
# execute vpn certificate local generate cmp <certificate_name> <key_size> <server> <path>
<server_certificate> <auth_certificate> <user> <password> <subject> [SANs] [options]

# execute vpn certificate local generate default-ssl-ca

# execute vpn certificate local generate default-ssl-key-certs

# execute vpn certificate local generate default-ssl-serv-key

# execute vpn certificate local generate ec <certificate_name> <curve_name> <subject>
<country> <state/province> <city> <organization> <OU> <email> [SANs] [options]

# execute vpn certificate local generate rsa <certificate_name> <key_size> <subject>
<country> <state/province> <city> <organization> <OU> <email> [SANs] [options]
```

cmp	Generate a certificate request over CMPv2.
default-ssl-ca	Generate the default CA certificate used by SSL Inspection.
default-ssl-ca-untrusted	Generate the default untrusted CA certificate used by SSL Inspection.
default-ssl-key-certs	Generate the default RSA, DSA and ECDSA key certs for ssl resign.
default-ssl-serv-key	Generate the default server key used by SSL Inspection.
ec	Generate an elliptic curve certificate request.
rsa	Generate a RSA certificate request.

Import

Any certificate uploaded to a VDOM is only accessible to that VDOM. Any certificate uploaded to the Global VDOM is globally accessible by all VDOMs.

A signed certificate that is created using a CSR that was generated by the FortiGate does not include a private key, and can be imported to the FortiGate from a TFTP file server.

To import a certificate that does not require a private key:

```
# execute vpn certificate local import tftp <file_name> <server_address> <cert_type>
[password]
```

To import a certificate that requires a private key to a VDOM, or when VDOMs are disabled:

```
config vpn certificate {local | ca | remote | ocsp-server | crl}
```

Refer to the FortiOS CLI Reference for detailed options for each certificate type (local, CA, remote, OSCP server, CRL).

To import a global certificate that requires a private key when VDOMs are enabled:

```
config certificate {local | ca | remote | crl}
```

This command is only available when VDOMs are enabled. For details, see the FortiOS CLI Reference.

Uploading a certificate using an API

There are several API methods to upload a certificate based on the type and purpose of the certificate. The parameters of each method are available options, and some methods do not require all parameters to upload the certificate.

When uploading a certificate to the FortiGate using API, the certificate must be provided to the FortiGate in Base64 encoding. You must create a REST API user to authenticate to the FortiGate and use the generated API token in the request.

api/v2/monitor/vpn-certificate/ca/import

```
{
  "import_method": "[file|scep]",
  "scep_url": "string",
```

```
  "scep_ca_id": "string",
  "scope": "[vdom*|global]",
  "file_content": "string"
}
```

api/v2/monitor/vpn-certificate/crl/import

```
{
  "scope": "[vdom*|global]",
  "file_content": "string"
}
```

api/v2/monitor/vpn-certificate/local/import

```
{
  "type": "[local|pkcs12|regular]",
  "certname": "string",
  "password": "string",
  "key_file_content": "string",
  "scope": "[vdom*|global]",
  "acme-domain": "string",
  "acme-email": "string",
  "acme-ca-url": "string",
  "acme-rsa-key-size": 0,
  "acme-renew-window": 0,
  "file_content": "string"
}
```

api/v2/monitor/vpn-certificate/remote/import

```
{
  "scope": "[vdom*|global]",
  "file_content": "string"
}
```

api/v2/monitor/vpn-certificate/csr/generate

```
{
  "certname": "string",
  "subject": "string",
  "keytype": "[rsa|ec]",
  "keysize": [1024|1536|2048|4096],
  "curvename": "[secp256r1|secp384r1|secp521r1]",
  "orgunits": [
    "string"
  ],
  "org": "string",
  "city": "string",
  "state": "string",
  "countrycode": "string",
  "email": "string",
  "sub_alt_name": "string",
  "password": "string",
  "scep_url": "string",
  "scep_password": "string",
```

```
  "scope": "[vdom*|global]"
}
```

Example

In this example, a PKCS 12 certificate is uploaded as a local certificate using Postman as the API client. PowerShell is used for the Base64 encoding.

To upload a PKCS 12 certificate using an API:

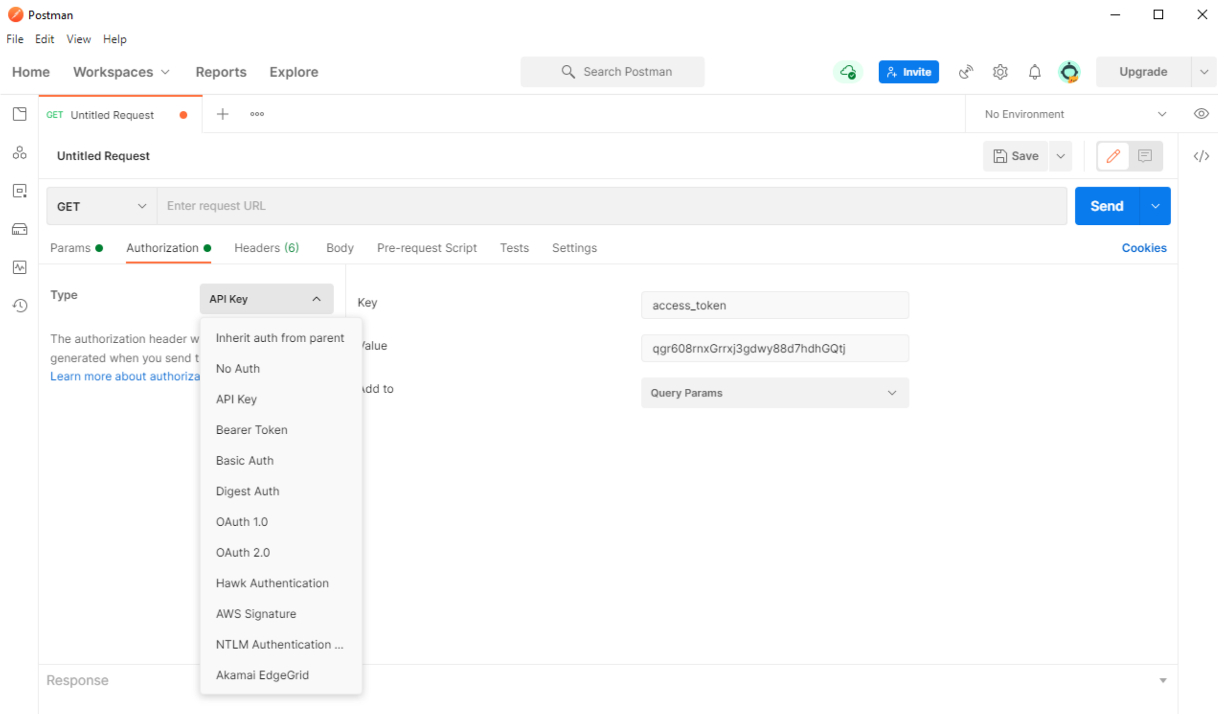
1. In PowerShell , encode the PKCS 12 certificate to Base64:

   ```
   cd C:\users\username\desktop
   $pkcs12cert = get-content 'C:\users\path\to\certificate\certificatename.p12' -Encoding
   Byte
   [System.Convert]::ToBase64String($pkcs12cert) | Out-File ‘base12encodedcert.txt’
   ```

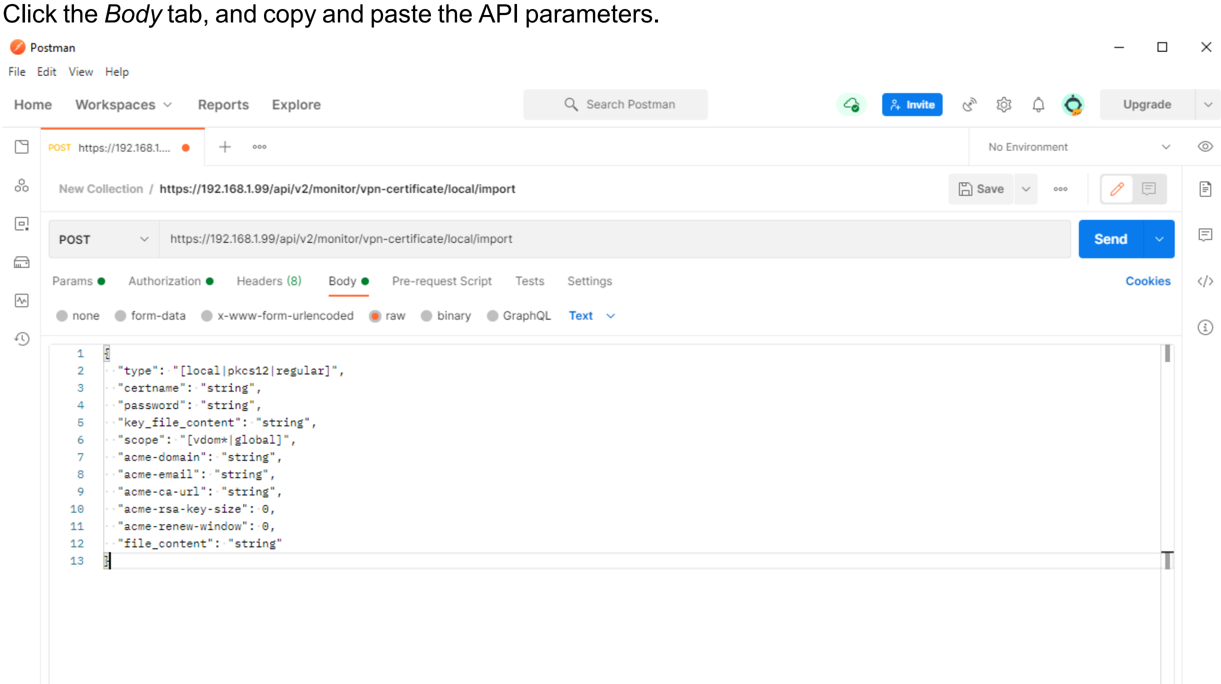
   These three lines of code do the following:

   a. Changes to working directory to the location where the encoded certificate will be created. In this example, it is the desktop.
   b. Creates a variable called `pkcs12cert` and defines it as the certificate file by specifying the full path to the certificate.
   c. Creates a text file called `base12encodedcert` at the location specified in the first step. You will copy and paste the contents of this as `file_content` later in Postman.
2. Generate an API token on the FortiGate by creating a REST API user. See Generate an API token on the Fortinet Developer Network. A subscription to the Fortinet Developer Network is required to view this topic.
3. Open Postman and create a new request:
   a. Click the +.
   b. Click the *Authorization* tab and in the *Type* dropdown, select *API Key*.
   c. For *Key*, enter *access_token* and enter the *Value* for the API user.

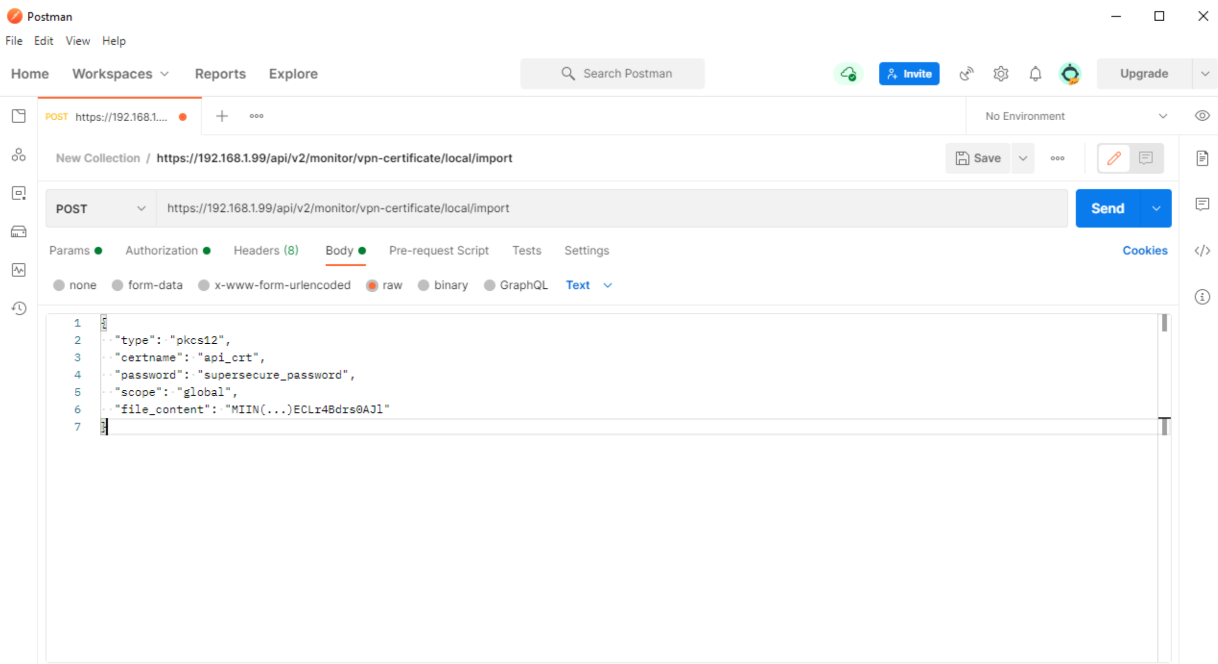
d. For *Add to*, select *Query Params*.



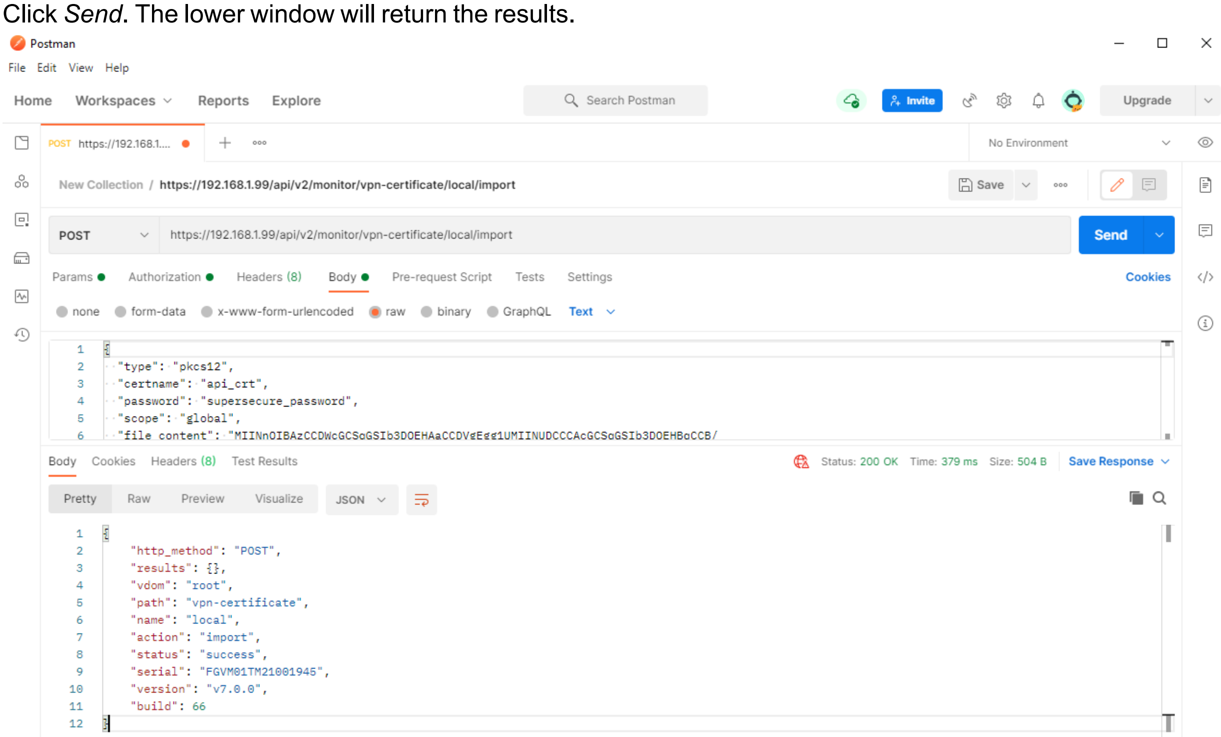
4. In the HTTP request dropdown, change the request from *GET* to *POST*, and enter the FortiGate's IP address and the URL of the API call.
5. Click the *Body* tab, and copy and paste the API parameters.



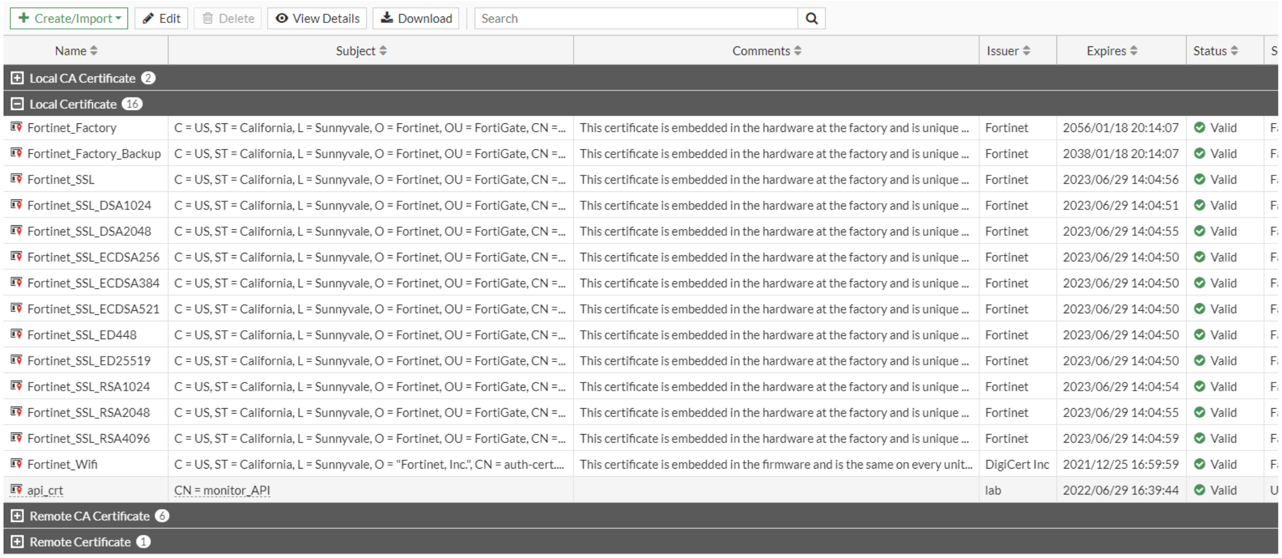
6. Remove unnecessary parameters (ACME related parameters and `key_file_content`) and enter the correct settings for your certificate. Copy and paste the contents of the file generated by PowerShell earlier into `file_content`.



7. Click *Send*. The lower window will return the results.



8. In FortiOS, go to *System > Certificates* and verify that the uploaded certificate is shown in the table (*api_crt*).



Create/Import | Edit | Delete | View Details | Download | Search

Name	Subject	Comments	Issuer	Expires	Status	S
Local CA Certificate 2						
Local Certificate 16						
Fortinet_Factory	C = US, ST = California, L = Sunnyvale, O = Fortinet, OU = FortiGate, CN =...	This certificate is embedded in the hardware at the factory and is unique ...	Fortinet	2056/01/18 20:14:07	Valid	F
Fortinet_Factory_Backup	C = US, ST = California, L = Sunnyvale, O = Fortinet, OU = FortiGate, CN =...	This certificate is embedded in the hardware at the factory and is unique ...	Fortinet	2038/01/18 20:14:07	Valid	F
Fortinet_SSL	C = US, ST = California, L = Sunnyvale, O = Fortinet, OU = FortiGate, CN =...	This certificate is embedded in the hardware at the factory and is unique ...	Fortinet	2023/06/29 14:04:56	Valid	F
Fortinet_SSL_DSA1024	C = US, ST = California, L = Sunnyvale, O = Fortinet, OU = FortiGate, CN =...	This certificate is embedded in the hardware at the factory and is unique ...	Fortinet	2023/06/29 14:04:51	Valid	F
Fortinet_SSL_DSA2048	C = US, ST = California, L = Sunnyvale, O = Fortinet, OU = FortiGate, CN =...	This certificate is embedded in the hardware at the factory and is unique ...	Fortinet	2023/06/29 14:04:55	Valid	F
Fortinet_SSL_ECDSA256	C = US, ST = California, L = Sunnyvale, O = Fortinet, OU = FortiGate, CN =...	This certificate is embedded in the hardware at the factory and is unique ...	Fortinet	2023/06/29 14:04:50	Valid	F
Fortinet_SSL_ECDSA384	C = US, ST = California, L = Sunnyvale, O = Fortinet, OU = FortiGate, CN =...	This certificate is embedded in the hardware at the factory and is unique ...	Fortinet	2023/06/29 14:04:50	Valid	F
Fortinet_SSL_ECDSA521	C = US, ST = California, L = Sunnyvale, O = Fortinet, OU = FortiGate, CN =...	This certificate is embedded in the hardware at the factory and is unique ...	Fortinet	2023/06/29 14:04:50	Valid	F
Fortinet_SSL_ED448	C = US, ST = California, L = Sunnyvale, O = Fortinet, OU = FortiGate, CN =...	This certificate is embedded in the hardware at the factory and is unique ...	Fortinet	2023/06/29 14:04:50	Valid	F
Fortinet_SSL_ED25519	C = US, ST = California, L = Sunnyvale, O = Fortinet, OU = FortiGate, CN =...	This certificate is embedded in the hardware at the factory and is unique ...	Fortinet	2023/06/29 14:04:50	Valid	F
Fortinet_SSL_RSA1024	C = US, ST = California, L = Sunnyvale, O = Fortinet, OU = FortiGate, CN =...	This certificate is embedded in the hardware at the factory and is unique ...	Fortinet	2023/06/29 14:04:54	Valid	F
Fortinet_SSL_RSA2048	C = US, ST = California, L = Sunnyvale, O = Fortinet, OU = FortiGate, CN =...	This certificate is embedded in the hardware at the factory and is unique ...	Fortinet	2023/06/29 14:04:55	Valid	F
Fortinet_SSL_RSA4096	C = US, ST = California, L = Sunnyvale, O = Fortinet, OU = FortiGate, CN =...	This certificate is embedded in the hardware at the factory and is unique ...	Fortinet	2023/06/29 14:04:59	Valid	F
Fortinet_Wifi	C = US, ST = California, L = Sunnyvale, O = "Fortinet, Inc.", CN = auth-cert....	This certificate is embedded in the firmware and is the same on every unit...	DigiCert Inc	2021/12/25 16:59:59	Valid	F
api_crt	CN = monitor_API		lab	2022/06/29 16:39:44	Valid	U
Remote CA Certificate 6						
Remote Certificate 1						

To debug using the HTTPS daemon:

```
# diagnose debug reset
# diagnose debug enable
# diagnose debug application httpsd -1
<output>
# diagnose debug disable
```

Procuring and importing a signed SSL certificate

A signed SSL certificate can be used when configuring SSL VPN, for administrator GUI access, and for other functions that require a certificate.

Before creating a certificate, you must have a registered domain. With a valid FortiGuard subscription, FortiDDNS can be used to register a domain; see DDNS on page 208 for more information.

Follow these instructions to purchase, import, and use a signed SSL certificate:

- Obtain, setup, and download an SSL certificate package from a certificate authority
- Generate a CSR
- Import the signed certificate into your FortiGate
- Configure your FortiGate to use the signed certificate

Obtain, setup, and download an SSL certificate package from a certificate authority

SSL certificate packages can be purchased from any Certificate Authority (CA), such as DigiCert, GoDaddy, or GlobalSign.

Let's Encrypt can be used to generate a free, trusted SSL certificate.

A third party CA might not sign a certificate with an intranet name or IP address. For details, see Can I request a certificate for an intranet name or IP address?

The process for purchasing, setting up, and downloading a certificate will vary depending on the CA that is used, and if a CSR must be generated on the FortiGate.

To purchase a certificate package:

1. Create an account with your chosen vendor, or use the account that you used to purchase your domain.
2. Locate the SSL Certificates page.
3. Purchase a basic SSL certificate for domain validation only. If required, a more secure SSL certificate can be purchased.
4. If required, load the CSR, either by uploaded the text file or copying and pasting the contents into the requisite text box. See Generate a CSR on page 2174 for information on generating the CSR on the FortiGate.
5. If required, set the server type to *Other*.
6. Verify the certificate per the requirements of the CA.
7. Download the signed certificate to your computer.
8. Import the signed certificate into your FortiGate; see Import the signed certificate into your FortiGate on page 2176.

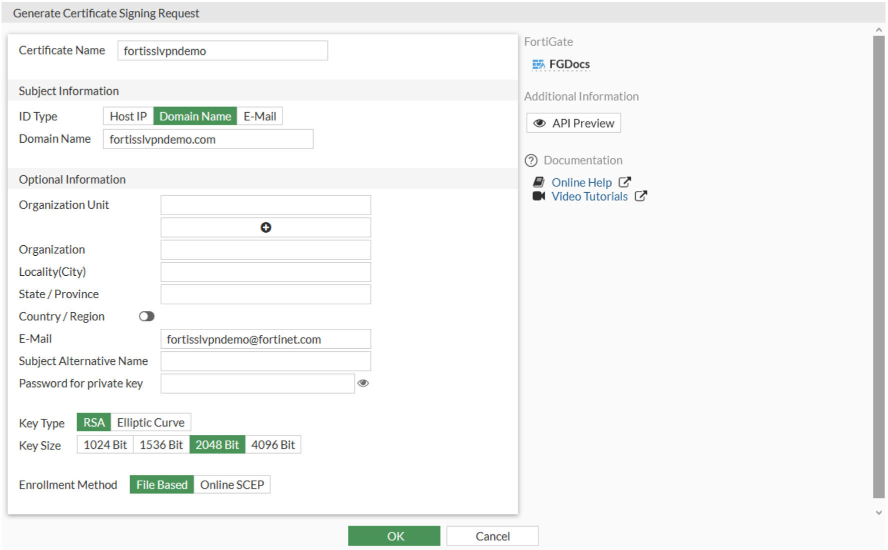
Generate a CSR

Some CAs can auto-generate the CSR during the signing process, or provide tools for creating CSRs. If necessary, a CSR can be created in your FortiGate device's GUI.

To generate a CSR on your FortiGate:

1. Go to *System > Certificates*. By default, the *Certificates* option is not visible, see Feature visibility on page 2163 for information.

2. Click *Generate*. The *Generate Certificate Signing Request* page opens.



3. Configure the CSR request:
   - Ensure that the certificate has a unique name.
   - Set the *ID Type* to *Domain Name* and enter a *Domain Name*.
   - An email address is required.
   - Ensure that the *Key Size* is set to *2048 Bit*.
   - Set the *Enrollment Method* to *File Based*.
4. Click *OK*.
   The CSR will be added to the certificate list with a status of *PENDING*.
5. In the certificate list, select the new CSR then click *Download* to save the CSR to your computer.
   The CSR file can be opened in any text editor, and will resemble the following:


```
-----BEGIN CERTIFICATE REQUEST-----
MIICuTCCAaECAQAwSzEcMBoGA1UEAxMTZm9ydGlzc2x2cG5kZW1vLmNvbTErMCkG
CSqGSIb3DQEJARYcZm9ydGlzc2x2cG5kZW1vQGZvcnRpbmV0LmNvbTCCASIwDQYJ
KoZIhvcNAQEBBQADggEPADCCAQoCggEBAMtnpNoR20NH2+UEX/NsyCmZhQqc4af3
Be1u9iOoNbo9Fk42gw47r71moAN+1jTL/Tcp3hRhXtpgoI7Zh3vjZnBbD2wwU8Ow
U7d1h5MULyMehR9r4T6OAJl4KbKPt5u90r5SpIb6mM1OIKvzMncuRS66rW1St0KP
mp/f6QjpjMrthnyJkCejgyTA1YwWNuT9BcO6PTkxBqVMLaRP6TUH6He9uhOx1Cj/
5tzvSdAozZIr2moMieQy0lNd6oQcgpdzaB9QN41+cZOlUXRCMPoH7E4KUe3/Gnis
+NMdQ8rIBijvWCXrKj20wb6sUEjAGJkcXlqVHWYCKWXl6Owejmc4ipkCAwEAAaAp
MCcGCSqGSIb3DQEJDjEaMBgwCQYDVR0TBAIwADALBgNVHQ8EBAMCBaAwDQYJKoZI
hvcNAQELBQADggEBAJKhtz2BPIKeHH9HcJKnfBKL+a6vu1l+1sW+YqnyD+3oR9ec
0eCmLnPxyyxsVel/tRsUg4DTfmooLNDhOjgfMsWxAGUQgrDH2k87cw6kiDAPCqv1
b+hFPNKZQSd09+HXAvOpXrMlrw5YdSaoRnau6Q02yUIYennKTIzFIscgh1mk4FSe
mb12DhPF+QydDCGDgtqnQbfxlDC0WmDcmxwa/0ZktoQhhhEbYgJ2O7l4TMqOxs/q
AZgwJlSNGBALLA2AxkIRUMKUteDdXz0QE8xNrvZpLTbWCNIpYJdRRqSd5C1w2VF4
CFgugTjFaJ13kYmBimeMRQsFtjLV5AxN+bUUsnQ=
-----END CERTIFICATE REQUEST-----
```

Import the signed certificate into your FortiGate

To import the signed certificate into your FortiGate:

1. Unzip the file downloaded from the CA.
   There should be two CRT files: a CA certificate with *bundle* in the file name, and a local certificate.
2. Log in to your FortiGate unit and go to *System > Certificates*.
3. Click *Import > Local Certificate*.
4. Upload the local certificate file, then click *OK*.
5. The status of the certificate will change from *PENDING* to *OK*.
6. Click *Import > CA Certificate*.
7. Set the *Type* to *File*, upload the CA certificate file, then click *OK*.
   The CA certificate will be listed in the *CA Certificates* section of the certificates list.

Configure your FortiGate to use the signed certificate

After the signed certificates have been imported, you can use it when configuring SSL VPN, for administrator GUI access, and for other functions that require a certificate.

To configure your FortiGate to use the signed certificate for SSL VPN:

1. Go to *VPN > SSL-VPN Settings*.
2. Set *Server Certificate* to the new certificate.
3. Configure other settings as needed.
4. Click *Apply*.

For more information on configuring SSL VPN, see SSL VPN on page 1617 and the Setup SSL VPN video in the Fortinet Video Library.

To configure using the certificate for administrator GUI access in the CLI:

```
config system global
    set admin-server-cert fortisslvpndemo
end
```

To change the certificate that is used for administrator GUI access in the GUI:

1. Go to *System > Settings*.
2. In the *Administration Settings* section, change *HTTPS server certificate* as needed.
3. Click *Apply*. You will be logged out of FortiOS.

Microsoft CA deep packet inspection

In most production environments, you want to use a certificate issued be your own PKI for deep packet inspection (DPI).

An existing Microsoft root CA can be used to issue a subordinate CA (sub CA) certificate that is installed as a DPI certificate on the FortiGate.

Complete the following steps to create your own sub CA certificate and use it for DPI:

1. Create a Microsoft sub CA certificate
2. Export the certificate and private key
3. Import the certificate and private key into the FortiGate
4. Configure a firewall policy for DPI
5. Verify that the sub CA certificate is being used for DPI

The FortiGate firewall uses information in the original web server certificate, then issues a new certificate signed by the Microsoft DPI certificate. The FortiGate then sends this certificate with the issuing DPI certificate to the client's web browser when the SSL session is being established.

The browser verifies that the certificate was issued by a valid CA, then looks for the issuing CA of the Microsoft DPI certificate in its loca trusted root CA store to complete the path to trusted root CA.

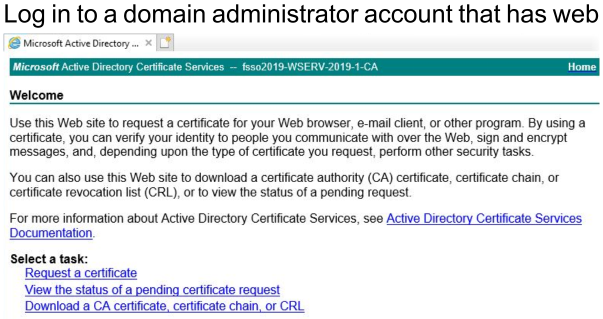
The Microsoft CA root certificate is normally deployed to all client PCs in the Windows domain, so the client can complete the certificate path up to a trusted root CA. The FortiGate now controls and can inspect the two HTTPS sessions: one with the external web server, and one with the client PC.

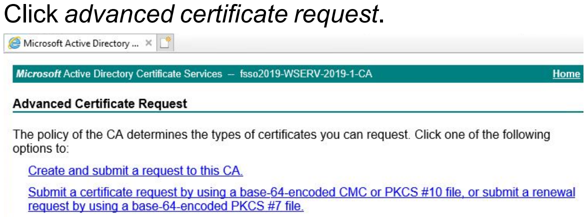
Create a Microsoft sub CA certificate

A Microsoft sub CA certificate can be created on a Microsoft CA server, or remotely using a web browser.

Creating a certificate remotely requires that the web enrollment option is configured on the Microsoft CA server. Remote certificate requests require HTTPS; requests are not allowed with HTTP.

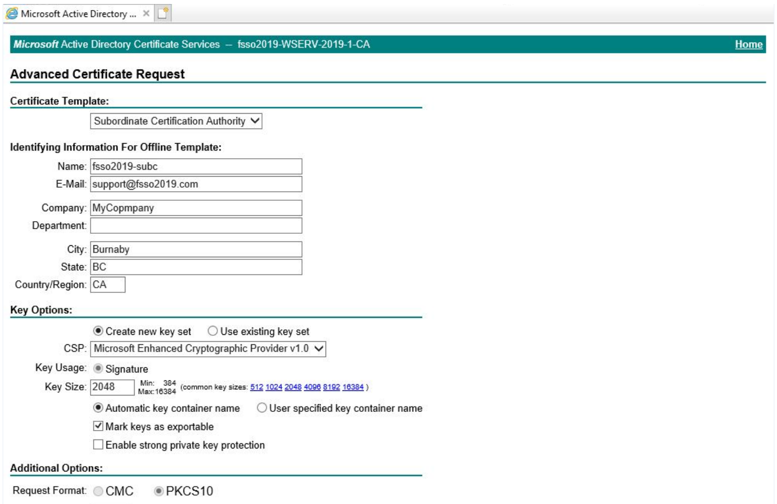
To create a Microsoft sub CA certificate remotely:

1. Open a web browser and go to one of the following URLs:
   - *https://<FQDN-CA-server>/CertSrv*
   - *https://<IP-CA-server>/CertSrv*.
2. Log in to a domain administrator account that has web enrollment rights.


3. Click *Request a certificate*.
4. Click *advanced certificate request*.


5. Click *Create and submit a request to this CA*, then click *Yes* in the *Web Access Confirmation* warning.
6. For the *Certificate Template*, select *Subordinate Certification Authority*.
7. Enable *Mark keys as exportable*.

8. Fill out the remaining information according to your security policy.

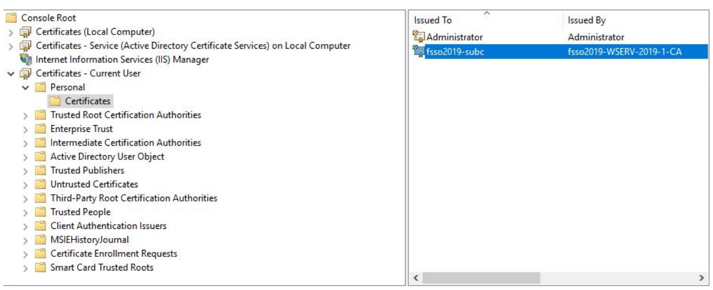


9. Submit the request.
10. Click *Yes* in the *Web Access Confirmation* warning.
11. Click *Install this certificate*.
    The certificate and private key are located in the current user's certificate store.

Export the certificate and private key

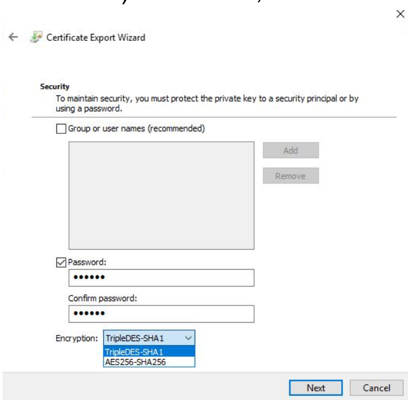
To export the certificate and private key:

1. Open the Microsoft Management Console (MMC) and add the *Certificate Snap-in*.
2. Go to the user's certificate store to locate the sub CA certificate that you just installed.



3. Right-click the certificate and select *All Tasks > Export*.
4. Click *Next* to start the *Microsoft Certificate Export Wizard*.
5. Follow the steps in the wizard:
    - When asked, select *Yes, export the private key*.
    - Only the PKCS #12 (.PFX) format is available, and it requires a password.
    - When selecting the encryption type, select *TripleDES-SHA1* if you are using an older version of FortiOS (5.6.9

and earlier). Otherwise, select *AES256-SHA256*.



6. Complete the wizard, and save the DPI certificate to a local folder.

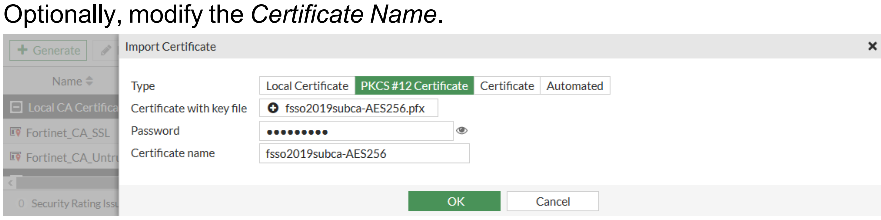
Import the certificate and private key into the FortiGate

The certificate can be imported from the local computer using the GUI, or from a TFTP server using the CLI.

After importing the certificate, you can view it in the GUI to verify that it was successfully imported.

To import the certificate and private key into the FortiGate in the GUI:

1. Go to *System > Certificates*.
2. Select *Import > Local Certificate*.
3. Set *Type* to *PKCS #12 Certificate*.
4. Click *Upload* and locate the certificate file.
5. Enter the *Password*.
6. Optionally, modify the *Certificate Name*.



7. Click *OK*.

To import the certificate and private key into the FortiGate in the CLI:

```
execute vpn certificate local import <certificate file name> <tftp ip address> <password>
```

To verify that the certificate was imported:

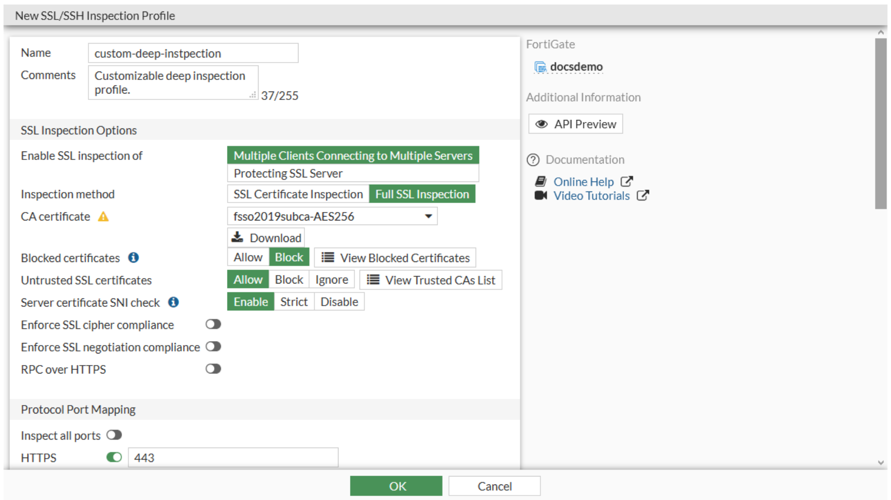
1. Go to *System > Certificates*. By default, the *Certificate* option is not visible, see Feature visibility on page 2163 for information.
2. Locate the newly imported certificate in the table.
3. Select the certificate and click *View Details* to view the certificate details.

Configure a firewall policy for DPI

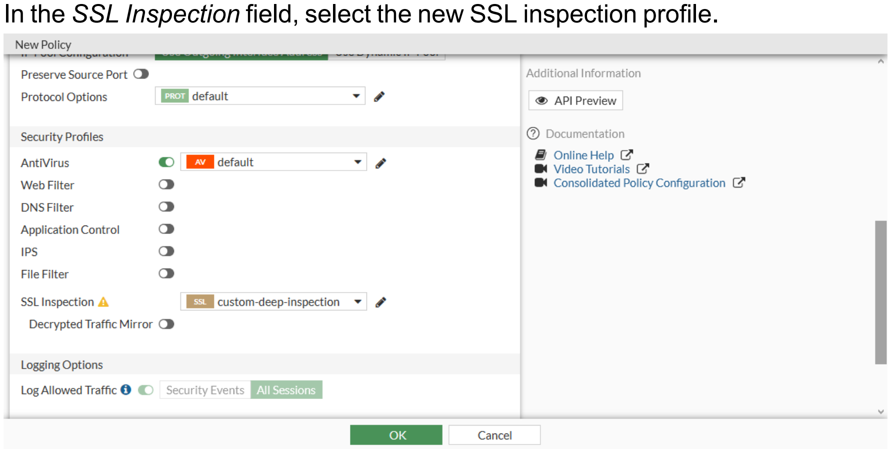
The certificate is used in an SSL/SSH inspection profile that is then used in a firewall policy.

To configure a firewall policy for DPI:

1. Go to *Security Profiles > SSL/SSH Inspection* and click *Create New*.
2. Configure the inspection profile, selecting the new certificate



3. Click *Apply*.
4. Go to *Policy & Objects > Firewall Policy*.
5. Create a new policy, or edit an existing policy.
6. In the *SSL Inspection* field, select the new SSL inspection profile.



7. Configure the remaining settings as needed.
8. Click *OK*.

Verify that the sub CA certificate is being used for DPI

You can verify that the certificate is being used for resigning web server certificates when a user connects to an external HTTPS website.

To verify that the certificate is being used:

1. On a client PC that is behind the FortiGate, go to an external HTTPS website.
   When connecting to the website, no certificate warning should be shown.
2. In your web browser, view the certificate and certificate path.
   The methods for doing this vary depending on the browser. See your browsers documentation for information.

ACME certificate support

The Automated Certificate Management Environment (ACME), as defined in RFC 8555, is used by the public Let's Encrypt certificate authority (https://letsencrypt.org) to provide free SSL server certificates. The FortiGate can be configured to use certificates that are managed by Let's Encrypt, and other certificate management services, that use the ACME protocol. The server certificates can be used for secure administrator log in to the FortiGate.

ACME certificates do not support loopback interfaces.

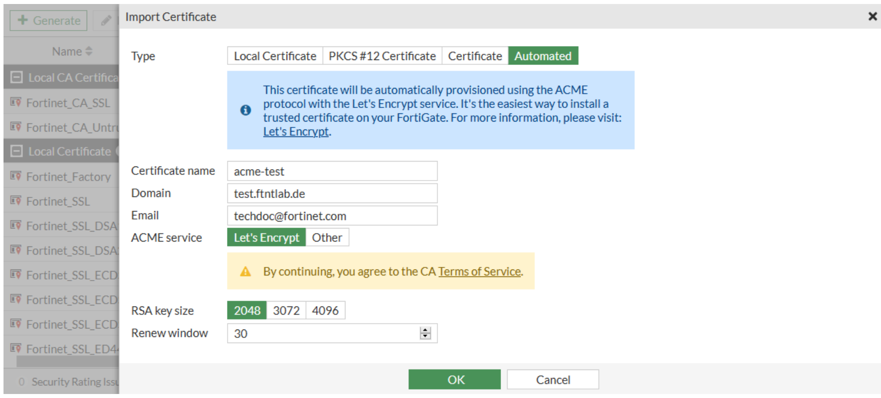
- The FortiGate must have a public IP address and a hostname in DNS (FQDN) that resolves to the public IP address.
- The configured ACME interface must be public facing so that the FortiGate can listen for ACME update requests. It must not have any VIPs, or port forwarding on port 80 (HTTP) or 443 (HTTPS).
- The Subject Alternative Name (SAN) field is automatically filled with the FortiGate DNS hostname. It cannot be edited, wildcards cannot be used, and multiple SANs cannot be added.

This example shows how to import an ACME certificate from Let's Encrypt, and use it for secured remote administrator access to the FortiGate.

To configure certificates in the GUI, go to *System* > *Feature Visibility* and enable *Certificates*.

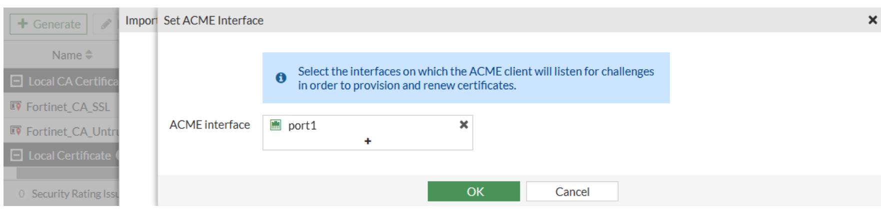
To import an ACME certificate in the GUI:

1. Go to *System* > *Certificates* and click *Import* > *Local Certificate*.
2. Set *Type* to *Automated*.
3. Set *Certificate name* to an appropriate name for the certificate.
4. Set *Domain* to the public FQDN of the FortiGate.
5. Set *Email* to a valid email address. The email is not used during the enrollment process.
6. Ensure that *ACME service* is set to *Let's Encrypt*.



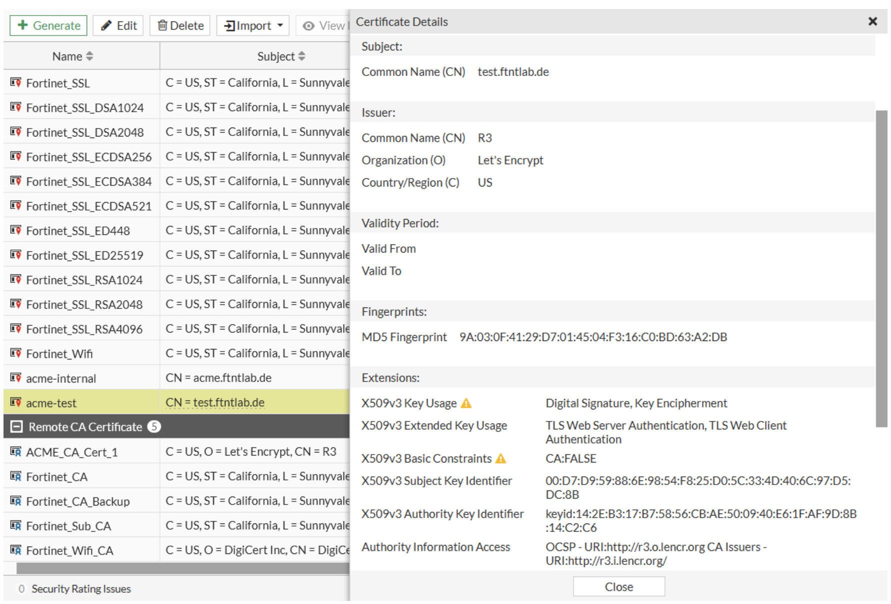
7. Configure the remaining settings as required, the click *OK*.
8. If this is the first time enrolling a server certificate with Let's Encrypt on this FortiGate, the *Set ACME Interface* pane opens.

   Select the interface that the FortiGate communicates with Let's Encrypt on, then click *OK*.

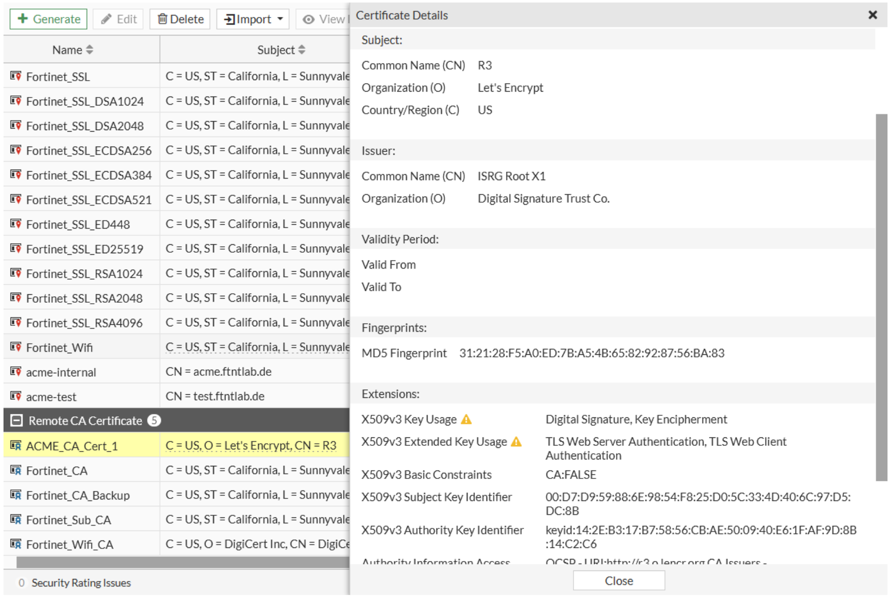


   The ACME interface can later be changed in *System > Settings*.
9. The new server certificate is added to the *Local Certificate* list.

   Click *View Details* to verify that the FortiGate's FQDN is in the certificate's *Subject: Common Name (CN)*.

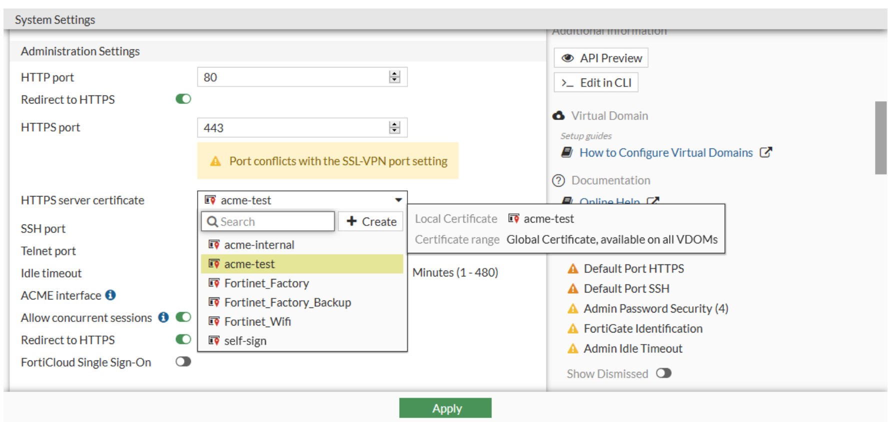


   The *Remote CA Certificate* list includes the issuing Let's Encrypt intermediate CA, issued by the public CA ISRG Root X1 from Digital Signature Trust Company.



To exchange the default FortiGate administration server certificate for the new public Let's Encrypt server certificate in the GUI:

1. Go to *System* > *Settings*.
2. Set *HTTPS server certificate* to the new certificate.



3. Click *Apply*.
4. Log in to the FortiGate using an administrator account from any internet browser. There should be no warnings related to non-trusted certificates, and the certificate path should be valid.

To import an ACME certificate in the CLI:

1. Set the interface that the FortiGate communicates with Let's Encrypt on:

```
config system acme
    set interface "port1"
end
```

2. Make sure that the FortiGate can contact the Let's Encrypt enrollment server:

```
# execute ping acme-v02.api.letsencrypt.org
PING ca80a1adb12a4fbdac5ffcbc944e9a61.pacloudflare.com (172.65.32.248): 56 data bytes
64 bytes from 172.65.32.248: icmp_seq=0 ttl=60 time=2.0 ms
64 bytes from 172.65.32.248: icmp_seq=1 ttl=60 time=1.7 ms
64 bytes from 172.65.32.248: icmp_seq=2 ttl=60 time=1.7 ms
64 bytes from 172.65.32.248: icmp_seq=3 ttl=60 time=2.1 ms
64 bytes from 172.65.32.248: icmp_seq=4 ttl=60 time=2.0 ms

--- ca80a1adb12a4fbdac5ffcbc944e9a61.pacloudflare.com ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 1.7/1.9/2.1 ms
```

3. Configure the local certificate request:

```
config vpn certificate local
    edit "acme-test"
        set enroll-protocol acme2
        set acme-domain "test.ftntlab.de"
        set acme-email "techdoc@fortinet.com"
    next
By enabling this feature you declare that you agree to the Terms of Service at
https://acme-v02.api.letsencrypt.org/directory
Do you want to continue? (y/n)y
end
```

4. Verify that the enrollment was successful:

```
# get vpn certificate local details acme-test
path=vpn.certificate, objname=local, tablename=(null), size=2632
== [ acme-test ]
        Name:        acme-test
        Subject:     CN = test.ftntlab.de
        Issuer:      C = US, O = Let's Encrypt, CN = R3
        Valid from:  2021-03-11 17:43:04  GMT
        Valid to:    2021-06-09 17:43:04  GMT
        Fingerprint: 9A:03:0F:41:29:D7:01:45:04:F3:16:C0:BD:63:A2:DB
        Serial Num:  03:d3:55:80:d2:e9:01:b4:ca:80:3f:2e:fc:24:65:ad:7c:0c
ACME details:
        Status: The certificate for the managed domain has been renewed successfully and
can be used (valid since Thu, 11 Mar 2021 17:43:04 GMT).
        Staging status: Nothing in staging
```

5. Check the ACME client full status log for the CN domain:

```
# diagnose sys acme status-full test.ftntlab.de
{
  "name": "test.ftntlab.de",
  "finished": true,
  "notified": false,
  "last-run": "Thu, 11 Mar 2021 18:43:02 GMT",
  "valid-from": "Thu, 11 Mar 2021 17:43:04 GMT",
  "errors": 0,
  "last": {
    "status": 0,
    "detail": "The certificate for the managed domain has been renewed successfully and
can be used (valid since Thu, 11 Mar 2021 17:43:04 GMT). A graceful server restart now
is recommended.",
    "valid-from": "Thu, 11 Mar 2021 17:43:04 GMT"
```

```
    },
    "log": {
      "entries": [
        {
          "when": "Thu, 11 Mar 2021 18:43:05 GMT",
          "type": "message-renewed"
        },
        ...
        {
          "when": "Thu, 11 Mar 2021 18:43:02 GMT",
          "type": "starting"
        }
      ]
    }
  }
```

To exchange the default FortiGate administration server certificate for the new public Let's Encrypt server certificate in the CLI:

```
config system global
    set admin-server-cert "acme-test"
end
```

When you log in to the FortiGate using an administrator account there should be no warnings related to non-trusted certificates, and the certificate path should be valid.

Administrative access using certificates

Certificates can be used for administrative authentication.

Generated key pairs can also be used for this authentication. See Public key SSH access on page 1948 for information about generating a key pair.

To log in to the FortiGate with a certificate private key:

1. On the PC, generate a certificate.
2. In FortiOS, import the PEM file for the remote certificate:

```
# execute vpn certificate remote import tftp certificate.pem 172.16.200.55
```

3. Display the imported remote certificate:

```
config certificate remote
    edit "REMOTE_Cert_1"
    next
end
```

4. Apply the remote certificate to the administrative user:

```
config system admin
    edit "admin1"
        set accprofile "prof_admin"
        set vdom "root"
        set ssh-certificate "REMOTE_Cert_1"
        set password ************
```

```
    next
end
```

5. On the PC, verify that the administrator can log in to the FortiGate with the SSH certificate:

```
root@PC05:~# ssh -i certificate-private.pem admin1@172.16.200.1
FortiGate-101F $ get system status
Version: FortiGate-101F v7.0.2,build0234,211019 (GA)
```

Creating certificates with XCA

This topic explains how to generate various certificates to be used in conjunction with a FortiGate, including:

- CA certificate
  - Signing server and client certificates
  - Issuing subordinate CAs for deep inspection
- Server certificate
  - SSL/TLS web administration authentication
  - VPN authentication
  - Internal SSL server protection
- Client certificate
  - End user authentication for SSL or IPsec VPN

XCA is an x509 certificate generation tool that handles RSA, DSA, and EC keys, as well as certificate signing requests (PKCS #10) and CRLs.

There are several options for generating and managing certificates. This topic covers basic certificate generation for XCA. It is not a comprehensive guide to its application and does not explore all options available when generating a certificate.

Creating the XCA database

Before creating any certificates, you must create an XCA database to group the certificates in. You should use a different database for each PKI you create.

To create the database:

1. Go to *File* > *New Database*.
2. Select a directory to store the created certificates and keys.
3. Enter a name. The provided password encrypts the private keys and is used to access the XCA database in the future.

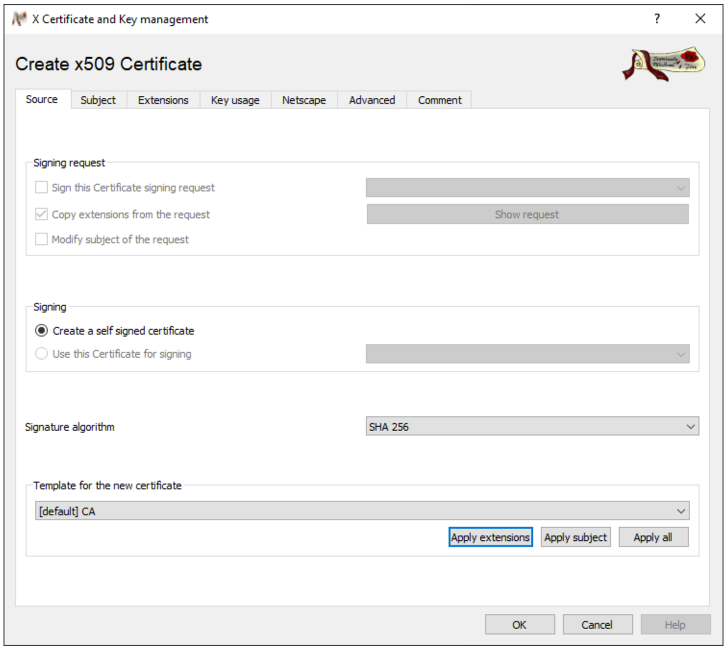
The remaining procedures in this topic assume you are using this XCA database.

Creating a CA certificate

A CA certificate marks the root of a certificate chain. If this CA certificate is trusted by an end entity, any certificates signed by the CA certificate are also trusted.

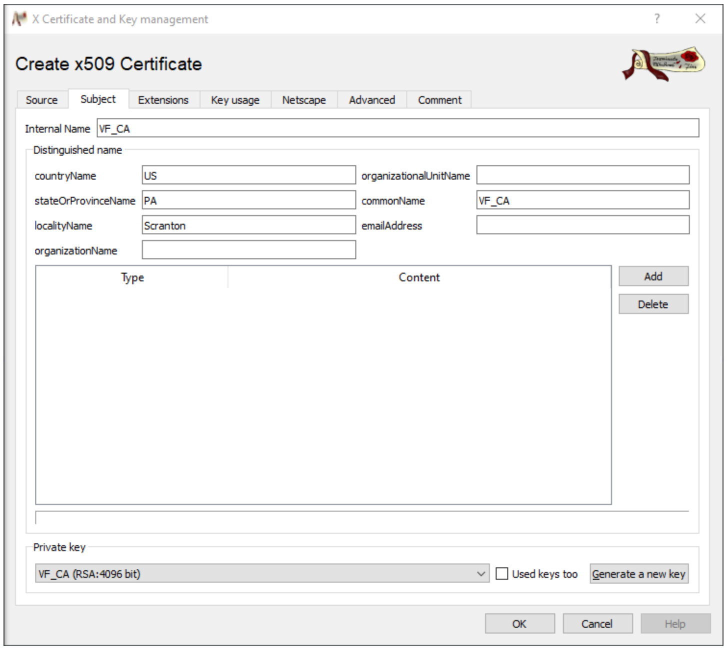
To create a CA certificate:

1. Click the *Certificates* tab, then click *New Certificate*.
2. Edit the *Source* tab:
   a. Set *Template for the new certificate* to *[default] CA*.
   b. Click *Apply extensions*.



3. Edit the *Subject* tab:
   a. Enter an *Internal Name* to reference this certificate within XCA.
   b. Enter a *commonName*.
   c. Optionally, click *Add* to add other distinguished name fields.
   d. Since this XCA database does not contain any keys yet, click *Generate a new key*. The *Private key* field is now

populated.



4. Optionally, edit the *Extensions* tab:
   a. Adjust the *Time range* if needed.
   b. Click *Apply*.
5. Click *OK*.

Issuing a subordinate CA certificate for deep inspection

Subordinate CA certificates are similar to CA certificates because they are used to sign other certificates to establish trust of the signed certificate's content. This trust of the signed certificate is only valid if the subordinate CA is also trusted by the client.

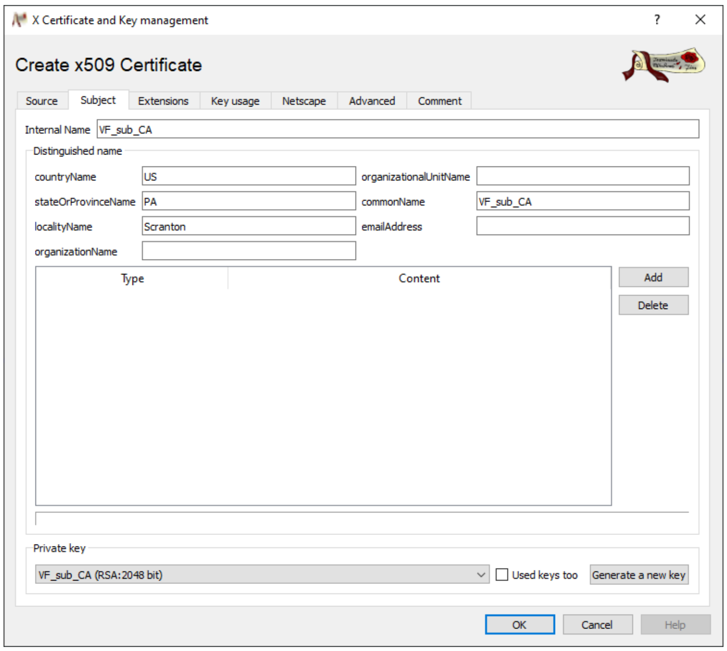
When performing deep inspection on a FortiGate, the FortiGate proxies the connection between the endpoint and the server. This is done transparently so that the end user believes they are communicating with the server, and the server with the client. To do this, when the webpage is requested by a client, the FortiGate must present a certificate that matches the requested website and is trusted by the client.

The certificate presented by the FortiGate is generated on-demand to match the requested website and is signed by this subordinate CA to establish trust with the requesting endpoint. The subordinate CA must be installed on the ForitGate (with the private key) and on the client device (without the private key).

A subordinate CA is used in place of a CA so that it may be revoked as necessary. This is critical since the subordinate CA’s private key is exported and becomes susceptible of being compromised. If the CA private key becomes compromised, you would be forced to re-create your entire PKI with a new root CA because root CAs cannot be revoked. See Microsoft CA deep packet inspection on page 2176 for more information about using subordinate CA certificates.

To issue a subordinate CA certificate for deep inspection:

1. Click the *Certificates* tab, then click *New Certificate*.
2. Edit the *Source* tab:
   a. Set *Use this Certificate for signing* to the CA created previously.
   b. Set *Template for the new certificate* to *[default] CA*.
   c. Click *Apply extensions*.
3. Edit the *Subject* tab:
   a. Enter an *Internal Name* to reference this certificate within XCA.
   b. Enter a *commonName*.
   c. Optionally, click *Add* to add other distinguished name fields.
   d. Click *Generate a new key* to create a new private key for the subordinate CA.



4. Optionally, edit the *Extensions* tab:
   a. Adjust the *Time range* if needed.
   b. Click *Apply*.
5. Click *OK*.

Creating a server host certificate

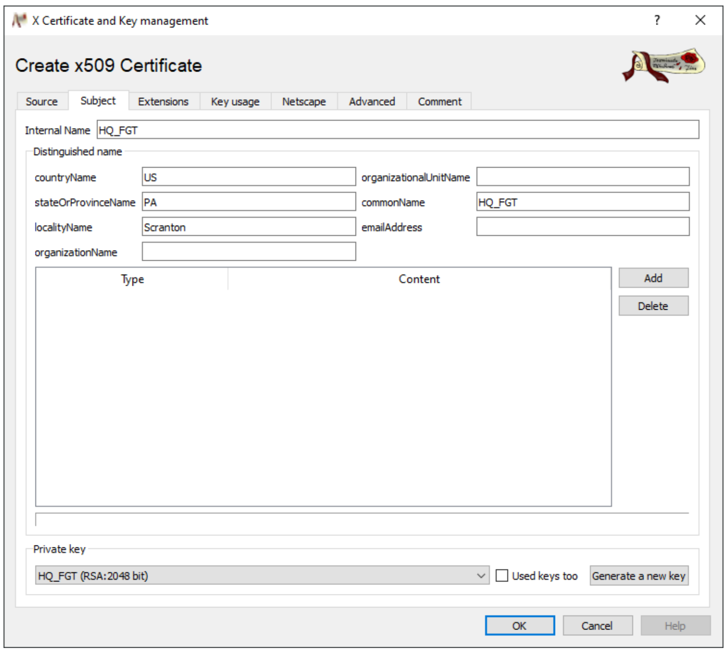
When a CA signs a host certificate, that CA is vouching for the credentials in the certificate. These credentials are what identifies the host.

Some endpoints can generate a certificate signing request (CSR). A CSR is a certificate outline that specifies the details of the endpoint, including its public key. This allows the CA to review the details and sign the request if they are true. This request is then returned or uploaded to the generating endpoint to be used.

Since some endpoints cannot generate their own CSR, you can create the certificate manually in XCA. If you already have a CSR, use the *Certificate signing requests* tab to import and then sign it.

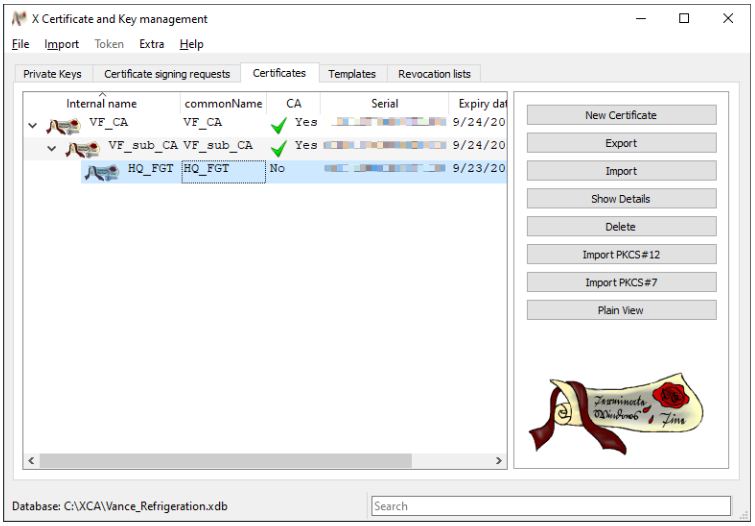
To create a server host certificate:

1. Click the *Certificates* tab, then click *New Certificate*.
2. Edit the *Source* tab:
   a. Set *Template for the new certificate* to *[default] TLS_server*.
   b. Click *Apply extensions*.
   c. In the *Signing* section, select *Use this Certificate for signing* and select the subordinate CA certificate.
3. Edit the *Subject* tab:
   a. Enter an *Internal Name* to reference this certificate within XCA.
   b. Enter the distinguished name fields as needed.
   c. Click *Generate a new key*.



4. Edit the *Extensions* tab:
   a. For *X509v3 Subject Alternative Name*, enter *email:user@domain.tld*.
5. Click *OK*.

6. Click the *Certificates* tab to view the certificate.



This certificate may be used to identify an SSL or TLS server by uploading the certificate and key pair to the server, such as when the FortiGate presents the administrative webpage or for SSL VPN authentication (see Configure your FortiGate to use the signed certificate on page 2176). Another use case for a server host certificate is to enable SSL server protection so the FortiGate simulates the real server and brokers the connection (see Protecting an SSL server on page 1299).

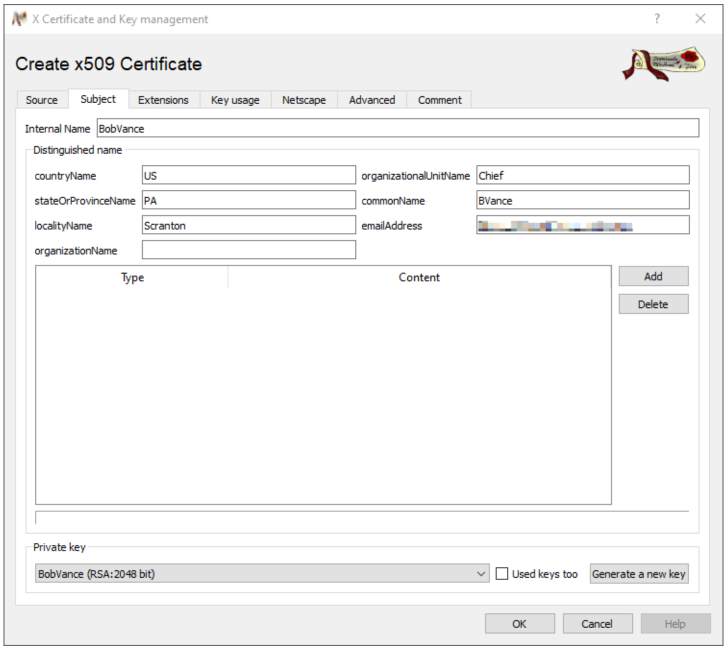
Creating a client host certificate

A client host certificate is used to identify an end entity in a more secure way than a username and password. Once the client host certificate is generated, see SSL VPN with certificate authentication on page 1658 for more information about using the certificate.

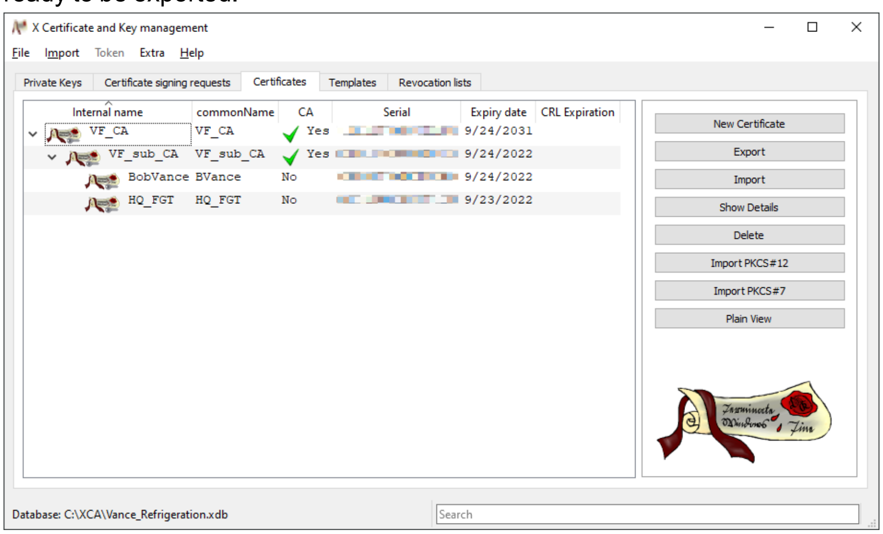
To create a client host certificate:

1. Click the *Certificates* tab, then click *New Certificate*.
2. Edit the *Source* tab:
   a. In the *Signing* section, select *Use this Certificate for signing* and select the CA or subordinate CA.
   b. Set *Template for the new certificate* to *[default] TLS_client*.
   c. Click *Apply extensions*.
3. Edit the *Subject* tab:
   a. Enter an *Internal Name* to reference this certificate within XCA.
   b. Enter the distinguished name fields as needed.

c. Click *Generate a new key*.



4. Click *OK*.
5. Click the *Certificates* tab. The FortiGate and client certificates are listed under the signing CA certificate and are ready to be exported.



6. Select a certificate and click *Export*.
7. Enter the file name and select an export format.
8. Click *OK*.

Certificate formats

Certificate file formats indicate what is contained in the file, how it is formatted, and how it is encoded. See Uploading a certificate using the GUI on page 2164 for more information about which formats the FortiGate expects for a given certificate type.

Configuration scripts

Configuration scripts are text files that contain CLI command sequences. They can be created using a text editor or copied from a CLI console, either manually or using the *Record CLI Script* function.

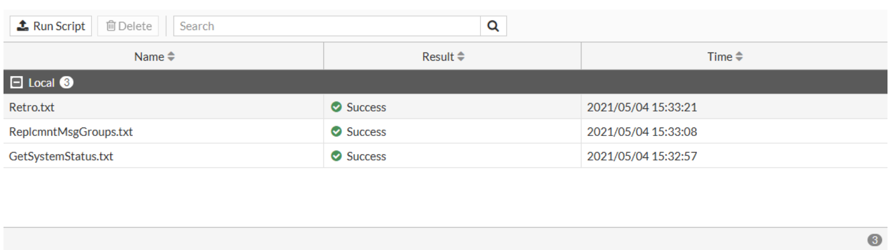
Scripts can be used to run the same task on multiple devices. For example, if your devices use the same security policies, you can enter or record the commands to create those policies in a script, and then run the script on each device. You could also create the policies in the GUI, and then copy and paste the CLI commands from the *CLI Console* using the *show* command.

If the FortiGate is managed by FortiManager, scripts can be uploaded to FortiManager and then run on any other FortiGates that are managed by that FortiManager. See Scripts in the FortiManager Administration Guide.

A comment line in a script starts with the number sign (#). Comments are not executed.

To run a script using the GUI:

1. Click on your username and select *Configuration* > *Scripts*.
2. Click *Run Script*.
3. Select the text file containing the script on your management computer, then click *OK*.
   The script runs immediately, and the *Script Execution History* table is updated, showing if the script ran successfully.



Workspace mode

Workspace mode allows administrators to make a batch of changes that are not implemented until the transaction is committed. Prior to committing, the changes can be reverted or edited as needed without impacting current operations.

When an object is edited in workspace mode it is locked, preventing other administrators from editing that object. A warning message will be shown to let the administrator know that the object is currently being configured in another transaction.

All administrators can use workspace mode; their permissions in workspace mode are the same as defined in their account profile.

A workspace mode transaction times out after five minutes if there is no activity. When a transaction times out, all changes are discarded. A warning message will be shown to let the administrator know that a timeout is imminent, or has already happened:

```
config transaction id=1 will expire in 30 seconds
config transaction id=1 will expire in 20 seconds
config transaction id=1 will expire in 10 seconds
config transaction id=1 has expired
```

The following commands are not changeable in a workspace transaction:

```
config system console
config system resource-limits
config system elbc
config system global
   set split-port
   set vdom-admin
   set management-vdom
   set wireless-mode
   set internal-switch-mode
end
config system settings
   set opmode
end
config system npu
config system np6
config system wireless
   set mode
end
config system vdom-property
config system storage
```

The `execute batch` command cannot be used in or to start workspace mode.

To use workspace mode:

1. Start workspace mode:
   `execute config-transaction`
   Once in workspace mode, the administrator can make configuration changes, all of which are made in a local CLI process that is not viewable by other processes.
2. Commit configuration changes:
   `execute config-transaction commit`
   After performing the commit, the changes are available for all other processes, and are also made in the kernel.
3. Abort configuration changes:
   `execute config-transaction abort`
   If changes are aborted, no changes are made to the current configuration or the kernel.

Diagnose commands

`diagnose sys config-transaction show txn-meta`

Show config transaction meta information. For example:

```
# diagnose sys config-transaction show txn-meta
txn_next_id=8, txn_nr=2
```

`diagnose sys config-transaction show txn-info`

Show config transaction information. For example:

```
# diagnose sys config-transaction show txn-info
current_jiffies=680372
```

```
txn_id=6, expire_jiffies=706104, clicmd_fpath='/dev/cmdb/txn/6_EiLl9G.conf'
txn_id=7, expire_jiffies=707427, clicmd_fpath='/dev/cmdb/txn/7_UXK6wY.conf'
```

`diagnose sys config-transaction show txn-entity`

Show config transaction entity. For example:

```
# diagnose sys config-transaction show txn-entity
vd='global', cli-node-oid=37(system.vdom), txn_id=7. location: fileid=0, storeid=0,
     pgnr=0, pgidx=0
vd='global', cli-node-oid=46(system.interface), txn_id=7. location: fileid=3,
     storeid=0, pgnr=0, pgidx=0
```

`diagnose sys config-transaction show txn-lock`

Show transaction lock status. For example:

```
# diagnose sys config-transaction show txn-lock
type=-1, refcnt=0, value=256, pid=128
```

`diagnose sys config-transaction status`

Show the transaction status in the current CLI.

Custom languages

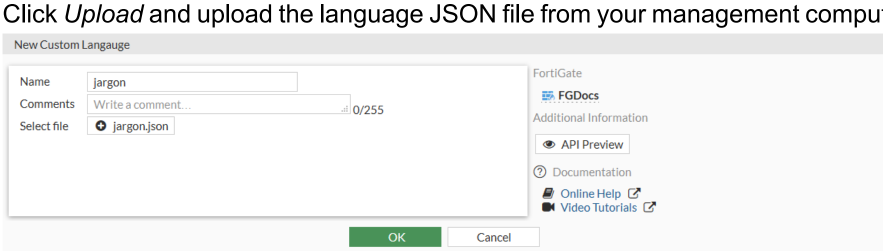
Custom languages can be uploaded and used for SSL VPN web portals. Custom languages must be enabled before they can be added in the GUI.

To enable custom languages:

```
config system global
    set gui-custom-language enable
end
```

To configure a custom language in the GUI:

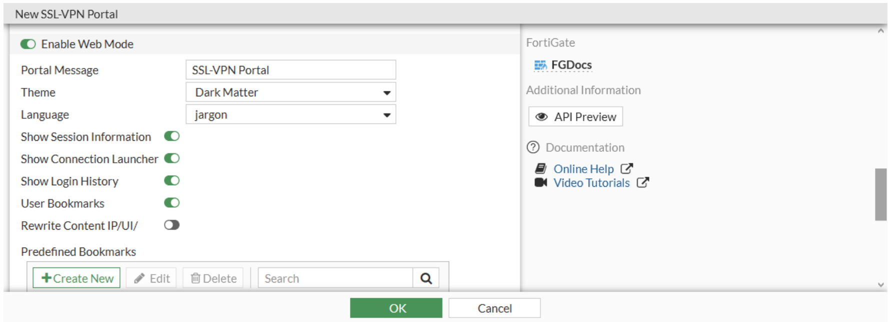
1. Go to *System > Custom Languages* and click *Create New*.
2. Enter the name of the language.
3. Optionally, enter a comment.
4. Click *Upload* and upload the language JSON file from your management computer.



5. Click *OK*.

To configure a language in an SSL VPN web portal in the GUI:

1. Go to *VPN > SSL-VPN Portals*.
2. Edit an existing portal, or click *Create New* to create a new one.
3. Enable *Enable Web Mode*, then select the language from the *Language* field.



4. Click *OK*.

To configure a custom language in the CLI:

```
config system custom-language
    edit <language>
        set filename <file>
    next
end
```

To configure a language in an SSL VPN web portal in the GUI:

```
config vpn ssl web portal
    edit <portal>
        set web-mode enable
        set custom-lang <language>
    next
end
```

RAID

Most FortiGate devices with multiple disk drives (SSD or HHD) can be configured to use RAID.

Enabling or disabling RAID, and changing the RAID level, erases all data on the log disk and reboots the device.

To verify that the FortiGate has multiple disks:

- List disk devices and partitions:

```
# execute disk list

Disk SSD1 ref: 255 223.6GiB type: SSD [ATA INTEL SSDSC2KB24] dev: /dev/sda
```

```
  partition ref: 1 220.1GiB, 219.0GiB free mounted: Y label: LOGUSEDXA707476A dev:
/dev/sda1 start: 2048

Disk SSD2 ref: 16 223.6GiB type: SSD [ATA INTEL SSDSC2KB24] dev: /dev/sdb
  partition ref: 17 62.7GiB, 62.4GiB free mounted: Y label: WANOPTXX1FEBBFA1 dev:
/dev/sdb1 start: 2048
  partition ref: 18 63.7GiB, 63.7GiB free mounted: N label: dev: /dev/sdb2 start:
133625856
  partition ref: 19 85.0GiB, 85.0GiB free mounted: N label: dev: /dev/sdb3 start:
267249664
```

- Display information about all of the disks:

```
# diagnose hardware deviceinfo disk

Disk SSD1 ref: 255 223.6GiB type: SSD [ATA INTEL SSDSC2KB24] dev: /dev/sda
  partition ref: 1 220.1GiB, 219.0GiB free mounted: Y label: LOGUSEDXA707476A dev:
/dev/sda1 start: 2048

Disk SSD2 ref: 16 223.6GiB type: SSD [ATA INTEL SSDSC2KB24] dev: /dev/sdb
  partition ref: 17 62.7GiB, 62.4GiB free mounted: Y label: WANOPTXX1FEBBFA1 dev:
/dev/sdb1 start: 2048
  partition ref: 18 63.7GiB, 63.7GiB free mounted: N label: dev: /dev/sdb2 start:
133625856
  partition ref: 19 85.0GiB, 85.0GiB free mounted: N label: dev: /dev/sdb3 start:
267249664

Disk SYSTEM(boot) 14.9GiB type: SSD [ATA 16GB SATA Flash] dev: /dev/sdc
  partition 247.0MiB, 155.0MiB free mounted: N label: dev: /dev/sdc1(boot) start: 1
  partition 247.0MiB, 154.0MiB free mounted: Y label: dev: /dev/sdc2(boot) start: 524289
  partition ref: 35 14.2GiB, 14.0GiB free mounted: Y label: dev: /dev/sdc3 start:
1048577

Disk USB-6(user-usb) ref: 48 28.6GiB type: USB [SanDisk Ultra] dev: /dev/sdd
<<<<<<===this info for usb disk because i have usb disk on FGT301E
  partition ref: 49 28.6GiB, 28.6GiB free mounted: Y label: dev: /dev/sdd1 start: 0

Total available disks: 4
Max SSD disks: 2 Available storage disks: 2
```

To check the RAID status:

- RAID enabled:

```
# execute disk raid status
RAID Level: Raid-1
RAID Status: OK (Background-Synchronizing) (9%)
RAID Size: 239GB

Disk 1: OK Used 228GB
Disk 2: OK Used 228GB
```

- RAID disabled:

```
# execute disk raid status
RAID Level: Unavailable
RAID Status: Unavailable
RAID Size: 0GB
```

```
Disk 1: OK Not-Used 228GB
Disk 2: OK Not-Used 228GB
```

To enable RAID:

```
# execute disk raid enable
This will erase all data on the log disk, and system will reboot!
Do you want to continue? (y/n)y

Dependent storage SSD2 removed.
Dependent storage SSD1 removed.
Raid-0 created with 2 disks.

Performing raid on the requested disk(s) and rebooting, please wait.. .

Configuring raid...
- unmounting /data2 : ok
- unmounting /var/log : ok
- unmounting /usb : ok
- unmounting /var/storage/SSD2-WANOPTXX0EA0EF17 : ok

Formatting the disk...
- unmounting /usb : ok
Formatting /dev/md0 ... done

The system is going down NOW !!

Please stand by while rebooting the system.
Restarting system.
```

To rebuild the RAID:

```
# execute disk raid rebuild
```

To rebuild the RAID to another level:

1. Check the supported RAID levels:

```
# execute disk raid rebuild-level
<RAID level> supported: Raid-0, Raid-1
```

2. Rebuild the RAID to the required level:

```
# execute disk raid rebuild-level Raid-1
This will erase all data on the log disk, and system will reboot!
Do you want to continue? (y/n)y

Dependent storage RAID removed.
Raid-1 created with 2 disks.

Performing raid on the requested disk(s) and rebooting, please wait...

Configuring raid...
- unmounting /data2 : ok
- unmounting /var/log : ok
```

```
- unmounting /usb : ok

Formatting the disk...
- unmounting /usb : ok
Formatting /dev/md0 ... done


The system is going down NOW !!

Please stand by while rebooting the system.
Restarting system.
```

To disable RAID:

```
# execute disk raid disable
This will erase all data on the log disk, and system will reboot!
Do you want to continue? (y/n)y

Dependent storage RAID removed.

Performing format on the requested disk(s) and rebooting, please wait...

Configuring raid...
- unmounting /data2 : ok
- unmounting /var/log : ok
- unmounting /usb : ok

Formatting the disk...
Partitioning and formatting /dev/sda label LOGUSEDX3D36836D ... done
Partitioning and formatting /dev/sdb label WANOPTXX1FEBBFA1 ...
Sending request for partno=0 start=2048 stop=133624230
Sending request for partno=1 start=133625856 stop=267248460
Sending request for partno=2 start=267249664 stop=445414150
done

The system is going down NOW !!

Please stand by while rebooting the system.
Restarting system.
FortiGate-301E (11:11-04.30.2018)
.
Reading boot image 3017355 bytes.
Initializing firewall...
System is starting...
```

FortiGate encryption algorithm cipher suites

FortiGates use SSL/TLS encryption for HTTPS and SSH administrative access, and SSL VPN remote access. When establishing an SSL/TLS or SSH connection, you can control the encryption level and the ciphers that are used in order to control the security level.

HTTPS access

HTTP administrative access encryption is controlled using the following commands:

```
config system global
    set strong-crypto {enable | disable}
    set admin-https-ssl-versions {tlsv1-1 tlsv1-2 tlsv1-3}
    set admin-https-ssl-ciphersuites {<cipher_1> ... <cipher_n>}
    set admin-https-ssl-banned-ciphers {<cipher_1> ... <cipher_n>}
end
```

When strong encryption is enabled, only TLS 1.2 and TLS 1.3 are allowed. If strong encryption is then disabled, TLS 1.1 has to be manually enabled.

Setting `admin-https-ssl-ciphersuites` controls which cipher suites are offered in TLS 1.3. TLS 1.2 and lower are not affected by this command. To disable all TLS 1.3 cipher suites, remove TLS1-3 from `admin-https-ssl-versions`.

Setting `admin-https-ssl-banned-ciphers` controls which cipher technologies will not be offered for TLS 1.2 and lower.

Specific cipher suites are supported by each TLS version:

TLS version	Supported Cipher Suites	
TLS 1.1[1]	ECDHE-RSA-AES256-SHA[1]	AES256-SHA[1]
	ECDHE-RSA-AES128-SHA[1]	AES128-SHA[1]
TLS 1.2	ECDHE-RSA-AES256-GCM-SHA384	AES256-GCM-SHA384[1]
	ECDHE-RSA-AES128-GCM-SHA256	AES128-GCM-SHA256[1]
	ECDHE-RSA-CHACHA20-POLY1305	AES256-SHA256
	ECDHE-RSA-AES256-SHA384	AES128-SHA256
	ECDHE-RSA-AES128-SHA256	AES256-SHA[1]
	ECDHE-RSA-AES256-SHA[1]	AES128-SHA[1]
	ECDHE-RSA-AES128-SHA[1]	
TLS 1.3	TLS-AES-128-GCM-SHA256	TLS-AES-128-CCM-8-SHA256
	TLS-AES-256-GCM-SHA384	TLS-CHACHA20-POLY1305-SHA256
	TLS-AES-128-CCM-SHA256	

[1] Disabled if strong encryption (`strong-crypto`) is enabled.

SSH access

SSH access encryption is controlled using the following command:

```
config system global
    set admin-ssh-v1 {enable | disable}
```

```
    set strong-crypto {enable | disable}
    set ssh-enc-algo <algo_1> [<algo_2> ... <algo_n>]
end
```

The algorithms available when configuring `set ssh-enc-algo` are affected by `set strong-crypto` as follows:

Encryption	Supported Ciphers	
Strong	chacha20-poly1305@openssh.com	aes256-ctr
	aes128-ctr	aes128-gcm@openssh.com
	aes192-ctr	aes256-gcm@openssh.com
Weak	arcfour256	cast128-cbc
	arcfour128	aes192-cbc
	aes128-cbc	aes256-cbc
	3des-cbc	arcfour
	blowfish-cbc	rijndael-cbc@lysator.liu.se

SSL VPN

For SSL VPN connections, the TLS versions and cipher suites are controlled using the following commands:

```
config vpn ssl setting
    set algorithm {high | medium | low}
    set ssl-max-proto-ver {tls1-0 | tls1-1 | tls1-2 | tls1-3}
    set ssl-min-proto-ver {tls1-0 | tls1-1 | tls1-2 | tls1-3}
    set ciphersuite {TLS-AES-128-GCM-SHA256 TLS-AES-256-GCM-SHA384 TLS-CHACHA20-POLY1305-
SHA256 TLS-AES-128-CCM-SHA256 TLS-AES-128-CCM-8-SHA256}
end
```

Cipher suites (`ciphersuite`) can only be selected when the SSL maximum version is TLS 1.3.

When the SSL VPN security level (`algorithm`) is set to high, only high levels are allowed. When it is set to medium, high and medium levels are allowed. When it is set to low, any level is allowed.

The strong encryption (`strong-crypto`) command has no effect on the SSL VPN encryption level or ciphers.

Specific cipher suites are supported by each TLS version:

TLS version	Supported Cipher Suites	
TLS 1.0	ECDHE-RSA-AES256-SHA	DHE-RSA-CAMELLIA128-SHA
	DHE-RSA-AES256-SHA	AES128-SHA
	DHE-RSA-CAMELLIA256-SHA	SEED-SHA[1]
	AES256-SHA	CAMELLIA128-SHA
	CAMELLIA256-SHA	ECDHE-RSA-DES-CBC3-SHA[1]
	ECDHE-RSA-AES128-SHA	EDH-RSA-DES-CBC3-SHA[1]
	DHE-RSA-AES128-SHA[1]	DES-CBC3-SHA[1]
	DHE-RSA-SEED-SHA	
TLS 1.1	ECDHE-RSA-AES256-SHA	DHE-RSA-CAMELLIA128-SHA
	DHE-RSA-AES256-SHA	AES128-SHA
	DHE-RSA-CAMELLIA256-SHA	SEED-SHA[1]
	AES256-SHA	CAMELLIA128-SHA
	CAMELLIA256-SHA	ECDHE-RSA-DES-CBC3-SHA[1]
	ECDHE-RSA-AES128-SHA	EDH-RSA-DES-CBC3-SHA[1]
	DHE-RSA-AES128-SHA	DES-CBC3-SHA[1]
	DHE-RSA-SEED-SHA[1]	

TLS version	Supported Cipher Suites	
TLS 1.2	ECDHE-RSA-AES256-GCM-SHA384	ECDHE-RSA-AES128-SHA
	ECDHE-RSA-AES256-SHA384	DHE-RSA-AES128-GCM-SHA256
	ECDHE-RSA-AES256-SHA	DHE-RSA-AES128-CCM8
	DHE-RSA-AES256-GCM-SHA384	DHE-RSA-AES128-CCM
	ECDHE-RSA-CHACHA20-POLY1305	AES128-CCM8
	DHE-RSA-CHACHA20-POLY1305	AES128-CCM
	DHE-RSA-AES256-CCM8	DHE-RSA-AES128-SHA256
	DHE-RSA-AES256-CCM	DHE-RSA-AES128-SHA
	DHE-RSA-AES256-SHA256	ECDHE-RSA-CAMELLIA128-SHA256
	DHE-RSA-AES256-SHA	DHE-RSA-CAMELLIA128-SHA256
	ECDHE-RSA-CAMELLIA256-SHA384	DHE-RSA-SEED-SHA[1]
	DHE-RSA-CAMELLIA256-SHA256	DHE-RSA-CAMELLIA128-SHA
	DHE-RSA-CAMELLIA256-SHA	AES128-GCM-SHA256
	AES256-GCM-SHA384	AES128-SHA256
	AES256-CCM8	AES128-SHA
	AES256-CCM	CAMELLIA128-SHA256
	AES256-SHA256	SEED-SHA[1]
	AES256-SHA	CAMELLIA128-SHA
	CAMELLIA256-SHA256	ARIA128-GCM-SHA256
	CAMELLIA256-SHA	DHE-RSA-ARIA128-GCM-SHA256
	ARIA256-GCM-SHA384	ECDHE-ARIA128-GCM-SHA256
	DHE-RSA-ARIA256-GCM-SHA384	ECDHE-RSA-AES256-GCM-SHA384
	ECDHE-ARIA256-GCM-SHA384	ECDHE-RSA-DES-CBC3-SHA[1]
	ECDHE-RSA-AES128-GCM-SHA256	EDH-RSA-DES-CBC3-SHA[1]
	ECDHE-RSA-AES128-SHA256	DES-CBC3-SHA[1]
TLS 1.3	TLS_AES_256_GCM_SHA384	TLS_AES_128_CCM_SHA256
	TLS_CHACHA20_POLY1305_SHA256	TLS_AES_128_CCM_8_SHA256
	TLS_AES_128_GCM_SHA256	

[1] This cipher is not available when the SSL VPN security level (`algorithm`) is set to high.

Fortinet Security Fabric

The Fortinet Security Fabric provides an intelligent architecture that interconnects discrete security solutions into an integrated whole to detect, monitor, block, and remediate attacks across the entire attack surface. It delivers broad protection and visibility into every network segment and device, be they hardware, virtual, or cloud based.

- The physical topology view shows all connected devices, including access layer devices. The logical topology view shows information about the interfaces that each device is connected to.
- Security rating checks analyze the Security Fabric deployment to identify potential vulnerabilities and highlight best practices to improve the network configuration, deploy new hardware and software, and increase visibility and control of the network.
- Fabric connectors provide integration with multiple SDN, cloud, and partner technology platforms to automate the process of managing dynamic security updates without manual intervention.
- Automation pairs an event trigger with one or more actions to monitor the network and take the designated actions automatically when the Security Fabric detects a threat.

Security Fabric settings and usage

This section contains information about how to configure the following devices as part of the Fortinet Security Fabric:

- Components on page 2205
- Configuring the root FortiGate and downstream FortiGates
- Configuring FortiAnalyzer
- Configuring FortiGate Cloud on page 2217
- Configuring FortiAnalyzer Cloud service on page 2221
- Configuring FortiManager on page 2225
- Configuring FortiManager Cloud service on page 2227
- Configuring Sandboxing on page 2228
- Configuring FortiClient EMS on page 2234
- Synchronizing FortiClient ZTNA tags on page 2247
- Configuring FortiNAC on page 2250
- Configuring FortiAP and FortiSwitch on page 2252
- Configuring FortiMail on page 2253
- Configuring FortiNDR on page 2255
- Configuring FortiDeceptor on page 2259
- Configuring FortiWeb on page 2262
- Configuring FortiTester on page 2264
- Configuring FortiMonitor on page 2267
- Configuring FortiVoice on page 2269
- Using the Security Fabric
- Deploying the Security Fabric on page 2288
- Deploying the Security Fabric in a multi-VDOM environment on page 2296
- Synchronizing objects across the Security Fabric on page 2301

- Security Fabric over IPsec VPN on page 2308
- Leveraging LLDP to simplify Security Fabric negotiation on page 2314

System requirements

To set up the Security Fabric, the devices that you want to include must meet the Product Integration and Support requirements in the FortiOS Release Notes.

Some features of the Security Fabric are only available in certain firmware versions and models. Not all FortiGate models can run the FortiGuard Security Rating Service if they are the root FortiGate in a Security Fabric. For more information, see the Special Notices in the FortiOS Release Notes.

Prerequisites

- If devices are not already installed in your network, complete basic installation and configuration tasks by following the instructions in the device documentation.
- FortiGate devices must be operating in NAT mode.

Components

The Fortinet Security Fabric consists of different components that work together to secure you network.

The following devices are required to create a Security Fabric:

Device	Description
FortiGate	FortiGates are the core of the Security Fabric and can have one of the following roles: • **Root**: the root FortiGate is the main component in the Security Fabric. It is typically located on the edge of the network and connects the internal devices and networks to the internet through your ISP. From the root FortiGate, you can see information about the entire Security Fabric on the *Physical* and *Logical Topology* pages in the GUI. • **Downstream**: after a root FortiGate is installed, all other FortiGate devices in the Security Fabric act as Internal Segmentation Firewalls (ISFWs), located at strategic points in your internal network, rather than on the network edge. This allows extra security measures to be taken around key network components, such as servers that contain valuable intellectual property. ISFW FortiGates create network visibility by sending traffic and information about the devices that are connected to them to the root FortiGate. See Configuring the root FortiGate and downstream FortiGates on page 2208 for more information about adding FortiGate devices in the Security Fabric. FortiGate documentation: https://docs.fortinet.com/product/fortigate
FortiAnalyzer*	FortiAnalyzer gives you increased visibility into your network, centralized monitoring, and awareness of threats, events, and network activity by collecting and correlating logs from all Security Fabric devices. This gives you a deeper and more comprehensive view across the entire Security Fabric. See Configuring FortiAnalyzer on page 2215 for more information about adding FortiAnalyzer devices in the Security Fabric.

Device	Description
	FortiAnalyzer documentation: https://docs.fortinet.com/product/fortianalyzer
Cloud Logging*	There are two options for cloud logging: FortiAnalyzer Cloud and FortiGate Cloud. Either can be used to enable the Security Fabric root device; however, if using FortiGate Cloud, all downstream devices must belong to the same FortiCloud account. See Configuring FortiGate Cloud on page 2217 for more information about configuring a Security Fabric with FortiGate Cloud. FortiGate Cloud documentation: https://docs.fortinet.com/product/fortigate-cloud

* FortiAnalyzer or Cloud Logging is a required component for the Security Fabric. Either FortiAnalyzer, FortiAnalyzer Cloud, or FortiGate Cloud can be used to met this requirement.

The following devices are recommended:

Device	Description
FortiNDR	FortiNDR (formerly FortiAI) uses artificial neural networks (ANN) that can deliver sub-second malware detection and a verdict. Add FortiNDR to your Security Fabric to automatically quarantine attacks. See Configuring FortiNDR on page 2255 for more information about adding FortiNDR devices in the Security Fabric. FortiNDR documentation: https://docs.fortinet.com/product/fortindr
FortiAP	Add FortiAP devices to extend the Security Fabric to your wireless devices. Devices connected to a FortiAP appear in the Physical and Logical Topology pages in the Security Fabric menu. See Configuring FortiAP and FortiSwitch on page 2252 for more information about adding FortiAP devices in the Security Fabric. FortiAP documentation: https://docs.fortinet.com/product/fortiap
FortiClient	FortiClient adds endpoint control to devices that are located in the Security Fabric, allowing only traffic from compliant devices to flow through the FortiGate. FortiClient compliance profiles are applied by the first FortiGate that a device's traffic flows through. Device registration and on-net status information for a device that is running FortiClient appears only on the FortiGate that applies the FortiClient profile to that device. FortiClient documentation: https://docs.fortinet.com/product/forticlient
FortiDeceptor	FortiDeceptor automatically lays out a layer of decoys and lures, which helps conceal sensitive and critical assets behind a fabricated deception surface to confuse and redirect attackers while revealing their presence on your network. See Configuring FortiDeceptor on page 2259 for more information about adding FortiDeceptor devices in the Security Fabric. FortiDeceptor documentation: https://docs.fortinet.com/product/fortideceptor
FortiClient EMS	FortiClient EMS is used in the Security Fabric to provide visibility across your network, securely share information, and assign security profiles to endpoints. See Configuring FortiClient EMS on page 2234 for more information about adding FortiClient EMS devices in the Security Fabric. FortiClient EMS documentation: https://docs.fortinet.com/product/forticlient

Device	Description
FortiMail	FortiMail antispam processing helps offload from other devices in the Security Fabric that would typically carry out this process. See Configuring FortiMail on page 2253 for more information about adding FortiMail devices in the Security Fabric. FortiMail documentation: https://docs.fortinet.com/product/fortimail
FortiManager	Add FortiManager to simplify the network management of devices in the Security Fabric by centralizing management access in a single device. This allows you to easily control the deployment of security policies, FortiGuard content security updates, firmware revisions, and individual configurations for devices in the Security Fabric. See Configuring FortiManager on page 2225 for more information about adding FortiManager devices in the Security Fabric. FortiManager documentation: https://docs.fortinet.com/product/fortimanager
FortiSandbox	Add FortiSandbox to your Security Fabric to improve security with sandbox inspection. Sandbox integration allows FortiGate devices in the Security Fabric to automatically receive signature updates from FortiSandbox and add the originating URL of any malicious file to a blocked URL list. See Configuring Sandboxing on page 2228 for more information about adding FortiSandbox devices in the Security Fabric. FortiSandbox documentation: https://docs.fortinet.com/product/fortisandbox
FortiSwitch	A FortiSwitch can be added to the Security Fabric when it is managed by a FortiGate that is in the Security Fabric with the FortiLink protocol, and connected to an interface with *Security Fabric Connection* enabled. FortiSwitch ports to become logical extensions of the FortiGate. Devices connected to the FortiSwitch appear in the Physical and Logical Topology pages in the Security Fabric menu, and security features, such as FortiClient compliance profiles, are applied to them. See Configuring FortiAP and FortiSwitch on page 2252 for more information about adding FortiSwitch devices in the Security Fabric. FortiSwitch documentation: https://docs.fortinet.com/product/fortiswitch
FortiWeb	Add FortiWeb to defend the application attack surface from attacks that target application exploits. You can also configure FortiWeb to apply web application firewall features, virus scanning, and web filtering to HTTP traffic to help offload from other devices in the Security Fabric that would typically carry out these processes. See Configuring FortiWeb on page 2262 for more information about adding FortiWeb devices in the Security Fabric. FortiWeb documentation: https://docs.fortinet.com/product/fortiweb

The following devices are optional:

Device	Description
FortiADC	FortiADC devices optimize the availability, user experience, and scalability of enterprise application delivery. They enable fast, secure, and intelligent acceleration and distribution of even the most demanding enterprise applications. FortiADC documentation: https://docs.fortinet.com/product/fortiadc

Device	Description
FortiDDoS	FortiDDoS is a Network Behavior Anomaly (NBA) prevention system that detects and blocks attacks that intend to disrupt network service by overutilizing server resources. FortiDDoS documentation: https://docs.fortinet.com/product/fortiddos
FortiWLC	FortiWLC delivers seamless mobility and superior reliability with optimized client distribution and channel utilization. Both single and multi channel deployment options are supported, maximizing efficiency to make the most of available wireless spectrum. FortiWLC documentation: https://docs.fortinet.com/product/wireless-controller
Other Fortinet products	Other Fortinet products can be added to the Security Fabric, including FortiAuthenticator, FortiToken, FortiCache, and FortiSIEM. Documentation: https://docs.fortinet.com/
Third-party products	Third-party products that belong to the Fortinet Fabric-Ready Partner Program can be added to the Security Fabric.

Configuring the root FortiGate and downstream FortiGates

The following procedures include configuration steps for a typical Security Fabric implementation, where the edge FortiGate is the root FortiGate with other FortiGates that are downstream from the root FortiGate.

For information about the recommended number of downstream FortiGates, see the *FortiOS Best Practices*.

Prerequisite

- The FortiGates must be operating in NAT mode.

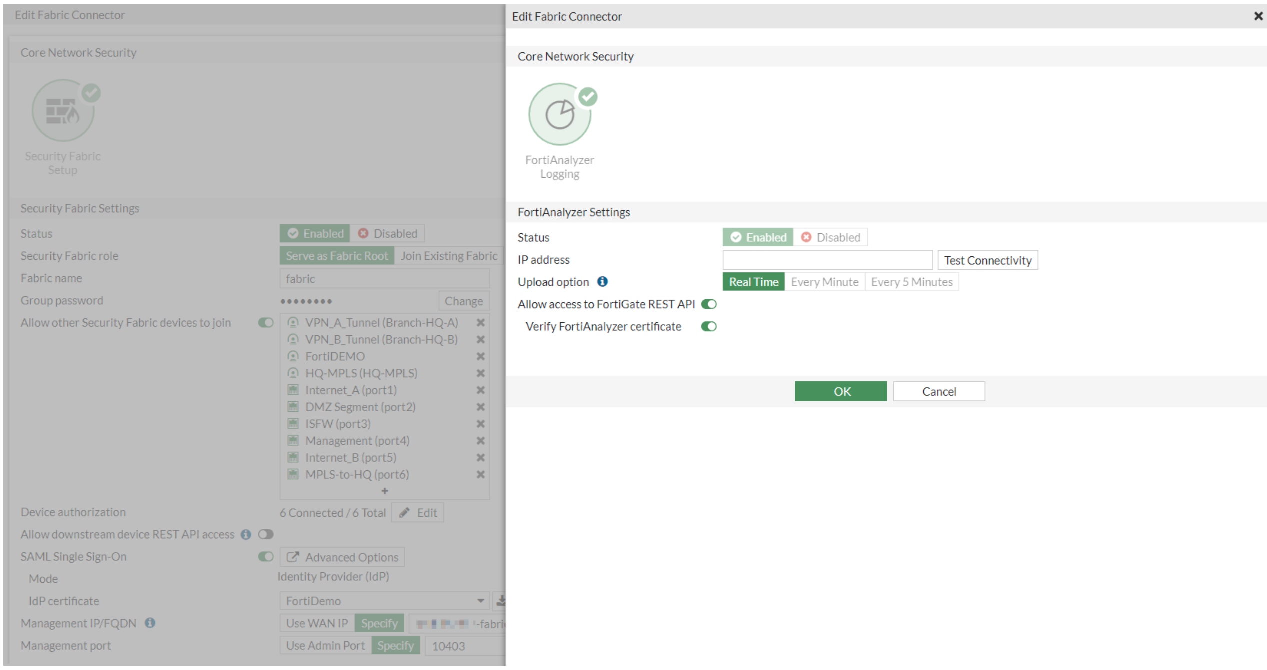
Configuring the root FortiGate

The edge FortiGate is typically configured as the root FortiGate, as this allows you to view the full topology of the Security Fabric from the top down.

The following steps describe how to add the FortiGate to serve as the root device, and how to configure FortiAnalyzer logging.

To configure the root FortiGate:

1. On the root FortiGate, go to *Security Fabric* > *Fabric Connectors* and double-click the *Security Fabric Setup* card.
2. For *Status*, click *Enable*.
3. Set the *Security Fabric role* to *Serve as Fabric Root*. FortiAnalyzer logging is automatically enabled and the settings can be configured in the slide-out pane.



When neither *FortiAnalyzer Logging* nor *Cloud Logging* are enabled, if the FortiGate detects that a FortiAnalyzer Cloud entitlement is available on this FortiGate, the slide-out pane will display *Cloud Logging* configurations. Otherwise, if *Cloud Logging* is enabled, the slide-out pane will display the *Cloud Logging* page. If *Cloud Logging* is disabled but FortiAnalyzer is enabled, then it will display the *FortiAnalyzer Logging* page.

In the *Cloud Logging* card, there are two options available, *FortiGate Cloud* and *FortiAnalyzer Cloud*. If there are multiple services enrolled on the FortiGate, the preference is: *Cloud Logging (FortiAnalyzer Cloud)*, *FortiAnalyzer Logging*, then *Cloud Logging (FortiGate Cloud)*.

4. Enter the FortiAnalyzer IP and select the *Upload option*.
5. In the *FortiAnalyzer Logging* section, in the *IP address* field, enter the IP address of the FortiAnalyzer.
6. If required, enable *Allow access to FortiGate REST API* and, optionally, *Verify FortiAnalyzer certificate*.
   The REST API accesses the FortiGate topology and shares data and results. The FortiGate will verify the FortiAnalyzer by retrieving its serial number and checking it against the FortiAnalyzer certificate. When verified, the FortiAnalyzer serial number is stored in the FortiGate configuration. When authorizing the FortiGate on the FortiAnalyzer, the FortiGate admin credentials do not need to be entered.
7. Click *Test Connectivity*.
   If you select *Test Connectivity* and this is the first time that you are connecting the FortiGate to the FortiAnalyzer, you will receive a warning message because the FortiGate has not yet been authorized on the FortiAnalyzer. You can configure this authorization when you configure the FortiAnalyzer. See Configuring FortiAnalyzer on page 2215.
8. Click *OK*. The FortiAnalyzer serial number is verified.
9. Enter a *Fabric name*.
10. Ensure *Allow other Security Fabric devices to join* is enabled.
11. Select the interfaces that will be listening for device join requests. Enabling an interface here has the same effect as going to *Network > Interfaces*, editing an interface, and enabling *Security Fabric Connection* under *Administrative Access*.

12. Click *OK*.

Using the root FortiGate with disk to store historic user and device information

This backend implementation allows the root FortiGate in a Security Fabric to store historic user and device information in a database on its disk. This will allow administrators to visualize users and devices over a period of time.

The daemon, user_info_history, stores this data on the disk. The information source for the historical data will be the user_info daemon, which would be recorded on the disk when user_info notifies user_info_history that a user has logged out or the device is no longer connected.

Adding downstream devices

Downstream device serial numbers can be pre-authorized from the root FortiGate, or allowed to join by request. New authorization requests include the device serial number, IP address, and HA members. HA members can include up to four serial numbers and is used to ensure that, in the event of a fail over, the secondary FortiGate is still authorized.

A downstream device's certificate can also be used to authorize the device by uploading the certificate to the root FortiGate.

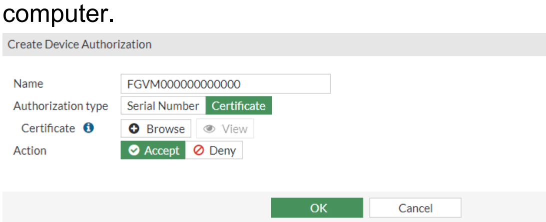
Pre-authorizing the downstream FortiGate

When a downstream Fortinet device's serial number or certificate is added to the trusted list on the root FortiGate, the device can join the Security Fabric as soon as it connects. After the new device is authorized, connected FortiAP and FortiSwitch devices are automatically included in the topology, where they can be authorized with one click.

The interface that connects to the downstream FortiGate must have *Security Fabric Connection* enabled.

To pre-authorize a FortiGate:

1. On the root FortiGate, go to *Security Fabric* > *Fabric Connectors* and double-click the *Security Fabric Setup* card.
2. In the *Device authorization* field click *Edit*. The *Device Authorization* window opens.
3. Click *Create New* to add a new device for pre-authorization.
4. Enter the device name in the *Name* field.
5. Select the *Authorization type*, either *Serial Number* or *Certificate*.
6. If *Certificate* is selected, click *Browse* to upload the downstream device's certificate from the management computer.



7. Select the *Action*, either *Accept* or *Deny*.
8. Click *OK* and add more devices as required.
9. Click *OK*.

To configure a downstream FortiGate to connect to an upstream FortiGate:

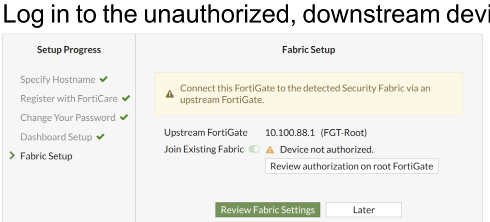
1. Configure the downstream FortiGate:
   a. On the downstream FortiGate, go to *Security Fabric > Fabric Connectors* and double-click the *Security Fabric Setup* card.
   b. Set *Status* to *Enable*.
   c. Set *Security Fabric role* to *Join Existing Fabric*.
   d. Enter the IP address of the root FortiGate in the *Upstream FortiGate IP* field.
   e. Click *OK*.
2. On the root FortiGate, go to *Security Fabric > Physical Topology* and verify that the downstream FortiGate that you added appears in the Security Fabric topology.

Authorizing a downstream FortiGate

When you log in to an unauthorized downstream FortiGate, the log in prompt includes the option to authorize the device on the root FortiGate.

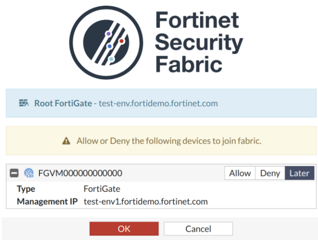
To authorize a downstream FortiGate:

1. Log in to the unauthorized, downstream device.

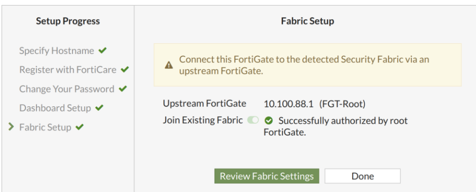


2. In the *Fabric Setup* step, click *Review authorization on root FortiGate*.
   A pop-up window opens to a log in screen for the root FortiGate.

3. Enter the log in credentials for the root FortiGate, then click *Login*.
   A list of pending authorizations is shown.



4. Select *Allow* and then click *OK* to authorize the downstream FortiGate. You can also select *Deny* to reject the authorization, or *Later* to postpone the decision to the next time that you log in.
   When authorization is allowed, the pop-up window closes, and the log in prompt shows that the downstream FortiGate has been authorized.

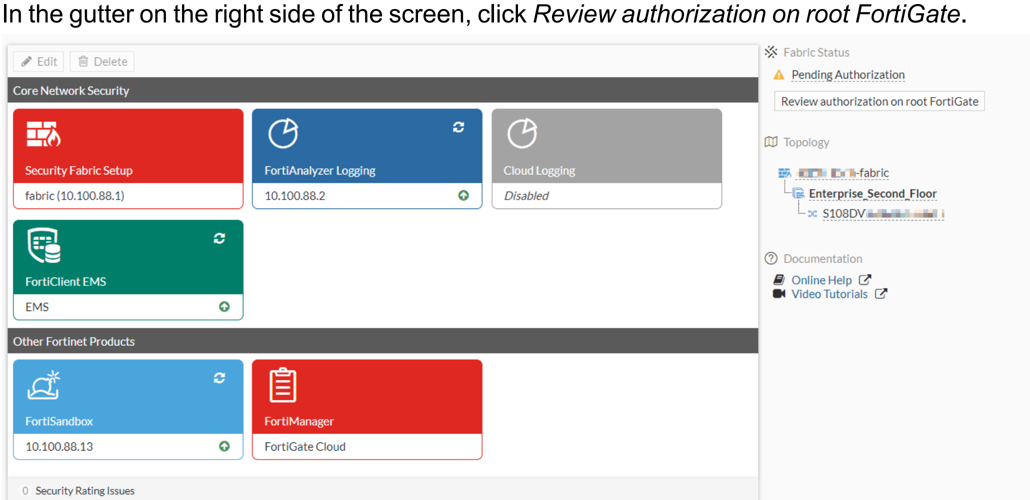


5. Click *Done* to log in to the downstream FortiGate.

Triggering authorization from the Fabric Connectors page

To authorize a downstream device from the Fabric Connectors page:

1. Go to *Security Fabric > Fabric Connectors*.
2. In the gutter on the right side of the screen, click *Review authorization on root FortiGate*.



   The root FortiGate pop-up window shows the state of the device authorization.

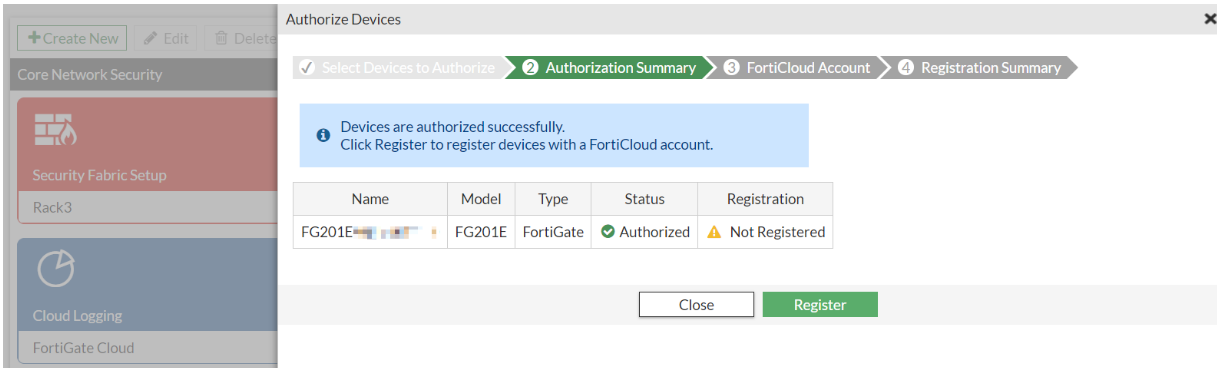
Authorizing the downstream FortiGate from the root

In this example, a downstream FortiGate is unauthorized and it is not registered to a FortiCloud account.

To authorize the downstream FortiGate from the root:

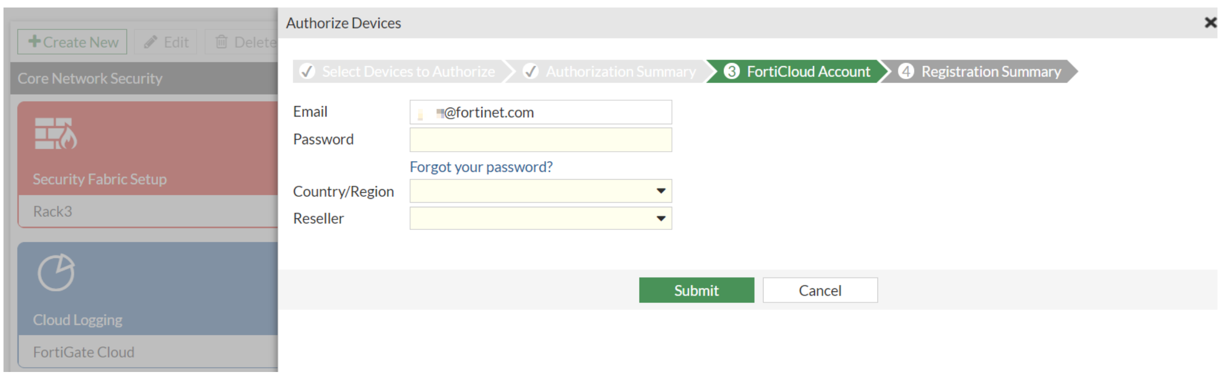
1. Log in to the root FortiGate and go to *Security Fabric > Fabric Connectors*. Devices requiring authorization are highlighted in the *Topology* tree (right-side gutter).

2. Click on a highlighted device and select *Authorize*. The *Authorize Devices* pane opens.
3. Click *Authorize*. The *Authorization Summary* pane opens.
4. The FortiGate is now authorized, so click *Register* to register the device to a FortiCloud account.



The *FortiCloud Account* pane opens.

5. Enter the required information (password, country/region, reseller). On the *Fabric Connectors* page, the same account name is implied for registration.



6. Click *Submit*. The *Registration Summary* pane opens.
7. Click *Close*.

You can use IPAM to automatically assign subnets to downstream FortiGates to prevent duplicate IP addresses from overlapping within the same Security Fabric. See Configure IPAM locally on the FortiGate on page 143.

CLI commands

Use the following commands to view, accept, and deny authorization requests, to view upstream and downstream devices, and to list or test Fabric devices:

Command	Description
`diagnose sys csf authorization pending-list`	View pending authorization requests on the root FortiGate.

Command	Description
diagnose sys csf authorization accept <serial number>	Authorize a device to join the Security Fabric.
diagnose sys csf authorization deny <serial number>	Deny a device from joining the Security Fabric.
diagnose sys csf downstream	Show connected downstream devices.
diagnose sys csf upstream	Show connected upstream devices.
diagnose sys csf fabric-device list	List all known Fabric devices.

Desynchronizing settings

By default, the settings for FortiAnalyzer logging, central management, sandbox inspection, and FortiClient EMS are synchronized between all FortiGates in the Security Fabric.

To disable automatic synchronization:

```
config system csf
    set configuration-sync local
end
```

Deauthorizing a device

A device can be deauthorized to remove it from the Security Fabric.

To deauthorize a device:

1. On the root FortiGate, go to *Security Fabric* > *Fabric Connectors*.
2. In the topology tree, click the device and select *Deauthorize*.
   After a device is deauthorized, the serial number is saved in a trusted list that can be viewed in the CLI using the `show system csf` command. For example, this result shows a deauthorized FortiSwitch:

```
show system csf
    config system csf
        set status enable
        set group-name "Office-Security-Fabric"
        set group-password ************
        config trusted-list
            edit "FGT6HD391800000"
        next
            edit "S248DF3X1700000"
                set action deny
            next
        end
    end
```

Configuring FortiAnalyzer

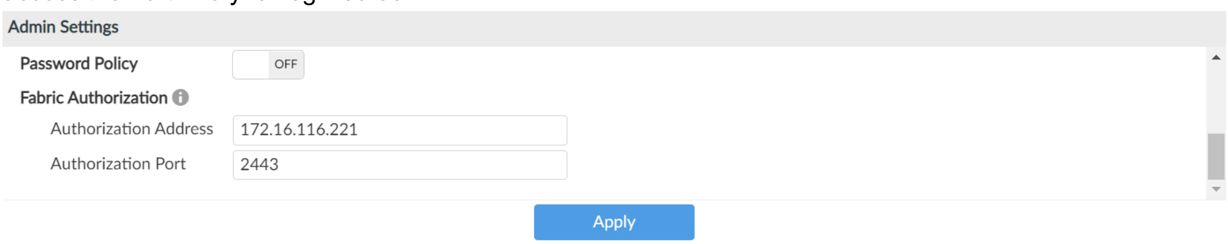
FortiAnalyzer or Cloud Logging is a required component for the Security Fabric. Either FortiAnalyzer, FortiAnalyzer Cloud, or FortiGate Cloud can be used to met this requirement.

FortiAnalyzer allows the Security Fabric to show historical data for the Security Fabric topology and logs for the entire Security Fabric. For more information about using FortiAnalyzer, see the FortiAnalyzer Administration Guide.

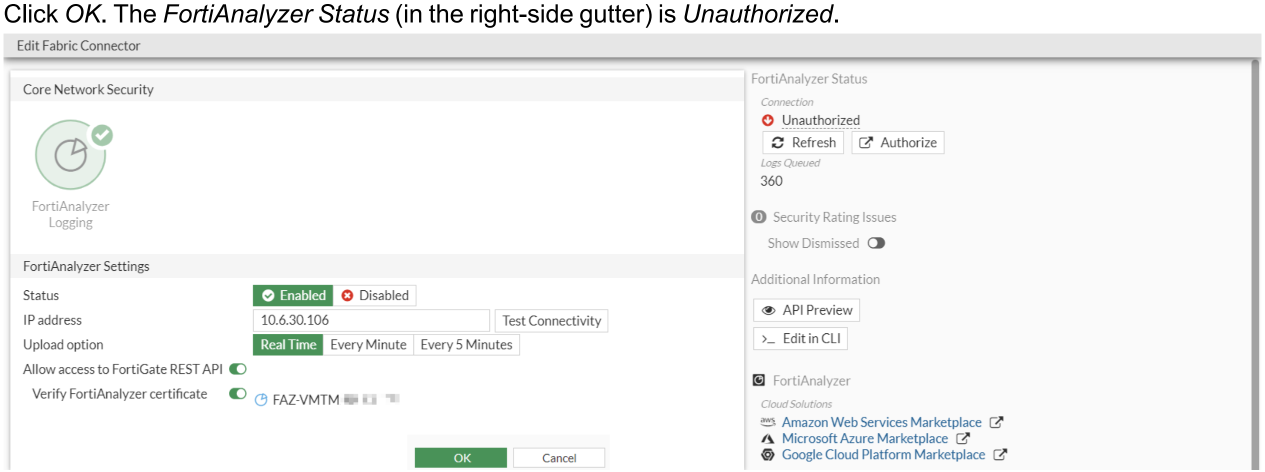
FortiAnalyzer 7.0.1 is used for this configuration example.

To authorize a FortiAnalyzer in the Security Fabric:

1. In FortiAnalyzer, configure the authorization address and port:
   a. Go to *System Settings > Admin > Admin Settings*.
   b. In the *Fabric Authorization* section, enter an *Authorization Address* and *Authorization Port*. This is used to access the FortiAnalyzer login screen.

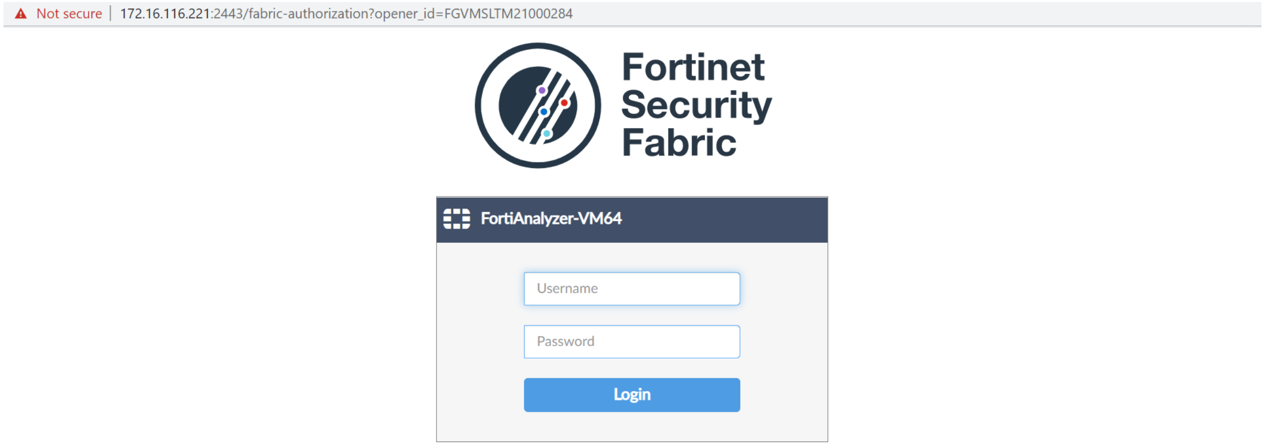


   c. Click *Apply*.
2. In FortiOS, go to *Security Fabric > Fabric Connectors* and double-click the *FortiAnalyzer Logging* card.
3. Enter the FortiAnalyzer IP.
4. Click *OK*. The *FortiAnalyzer Status* (in the right-side gutter) is *Unauthorized*.



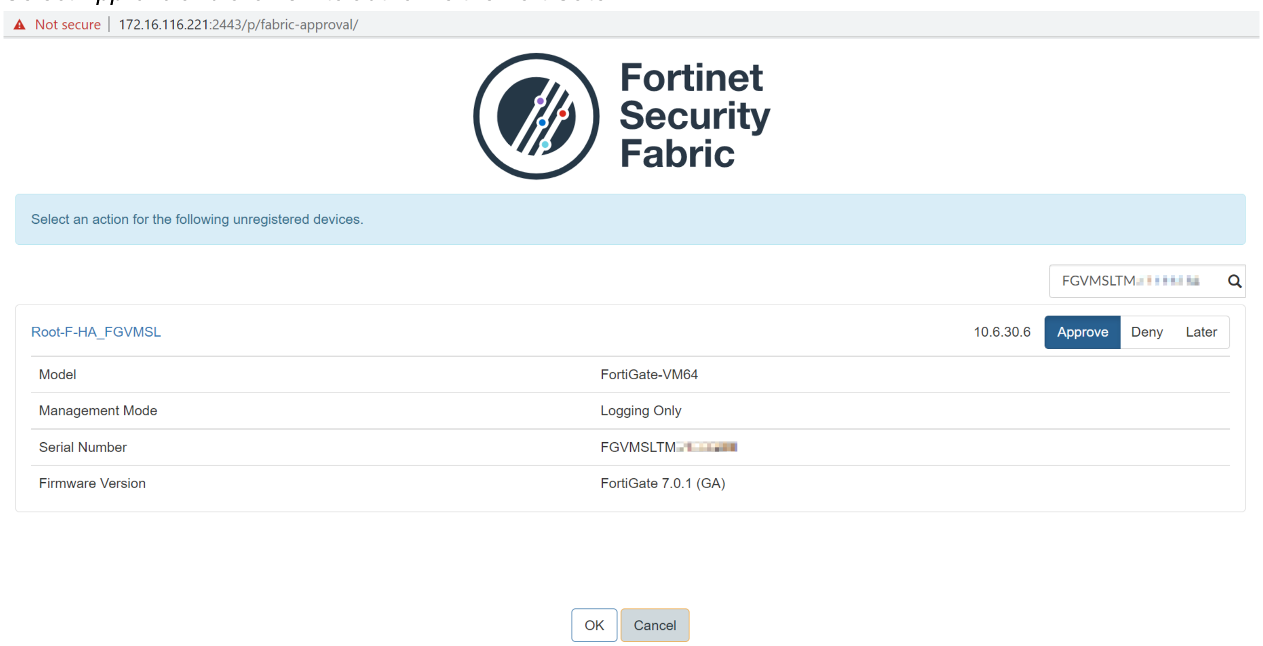
5. Click *Authorize*. You are redirected to a login screen.

6. Enter the username and password, then click *Login*.

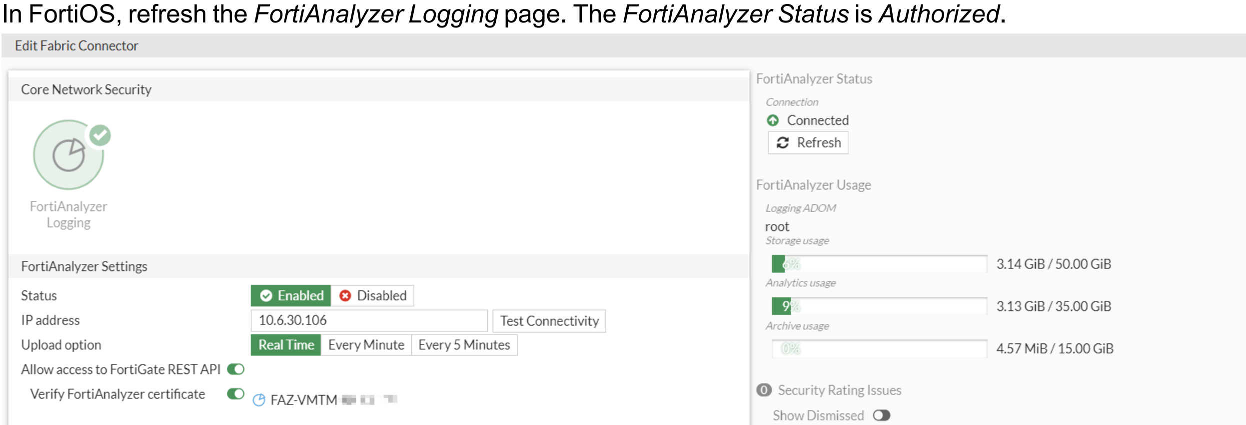


The authorization dialog opens.

7. Select *Approve* and click *OK* to authorize the FortiGate.



8. In FortiOS, refresh the *FortiAnalyzer Logging* page. The *FortiAnalyzer Status* is *Authorized*.



Sending traffic logs to FortiAnalyzer Cloud

FortiGates with a FortiCloud Premium subscription (AFAC) for Cloud-based Central Logging & Analytics, can send traffic logs to FortiAnalyzer Cloud in addition to UTM logs and event logs. After the Premium subscription is registered through FortiCare, FortiGuard will verify the purchase and authorize the AFAC contract. Once the contract is verified, FortiGuard will deliver the contract to FortiGate.

FortiGates with a Standard FortiAnalyzer Cloud subscription (FAZC) can only send UTM and event logs. FortiGates with a Premium subscription will send the UTM and event logs even if the Standard subscription has expired.

For information about cloud logging, see Configuring FortiAnalyzer Cloud service on page 2221

FortiAnalyzer Cloud does not support DLP/IPS archives at this time.

To verify the status a FortiCloud subscription with the CLI:

```
# diagnose test update info
```

The `FAZC` and `AFAC` fields display the subscription expiration date. The `Support contract` field displays the FortiCare account information. The `User ID` field displays the ID for FortiAnalyzer-Cloud instance.

```
...
FAZC,Tue Sep 24 16:00:00 2030
AFAC,Mon Nov 29 16:00:00 2021
...
Support contract: pending_registration=255 got_contract_info=1
account_id=[****@fortinet.com] company=[Fortinet] industry=[Technology]
User ID: 979090
```

Configuring FortiGate Cloud

FortiGate Cloud is a hosted security management and log retention service for FortiGate devices. It provides centralized reporting, traffic analysis, configuration management, and log retention without the need for additional hardware or software.

FortiGate Cloud offers a wide range of features:

- **Simplified central management**
  FortiGate Cloud provides a central GUI to manage individual or aggregated FortiGate and FortiWiFi devices. Adding a device to the FortiGate Cloud management subscription is straightforward. FortiGate Cloud has detailed traffic and application visibility across the whole network.
- **Hosted log retention with large default storage allocated**
  Log retention is an integral part of any security and compliance program, but administering a separate storage system is onerous. FortiGate Cloud takes care of this automatically and stores the valuable log information in the cloud. Different types of logs can be stored, including Traffic, System Events, Web, Applications, and Security Events.
- **Monitoring and alerting in real time**
  Network availability is critical to a good end-user experience. FortiGate Cloud enables you to monitor your FortiGate network in real time with different alerting mechanisms to pinpoint potential issues. Alerting mechanisms can be delivered via email.

- **Customized or pre-configured reporting and analysis tools**
  Reporting and analysis are your eyes and ears into your network's health and security. Pre-configured reports are available, as well as custom reports that can be tailored to your specific reporting and compliance requirements. The reports can be emailed as PDFs, and can cover different time periods.
- **Maintain important configuration information uniformly**
  The correct configuration of the devices within your network is essential for maintaining optimum performance and security posture. In addition, maintaining the correct firmware (operating system) level allows you to take advantage of the latest features.
- **Service security**
  All communication (including log information) between the devices and the cloud is encrypted. Redundant data centers are always used to give the service high availability. Operational security measures have been put in place to make sure your data is secure — only you can view or retrieve it.

For more information about FortiGate Cloud, see the FortiGate Cloud documentation.

Registration and activation

Before you can activate a FortiGate Cloud account, you must first register your device.

FortiGate Cloud accounts can be registered manually through the FortiGate Cloud website, https://www.forticloud.com, or you can easily register and activate your account directly from your FortiGate.

To activate your FortiGate Cloud account:

1. On your device, go to *Dashboard > Status*.
2. In the *FortiGate Cloud* widget, click the *Not Activated > Activate* button in the *Status* field.
3. A pane will open asking you to register your FortiGate Cloud account. Click *Create Account*, enter your information, view and accept the terms and conditions, and then click *OK*.
4. A second dialogue window open , asking you to enter your information to confirm your account. This sends a confirmation email to your registered email. The dashboard widget then updates to show that confirmation is required.
5. Open your email, and follow the confirmation link it contains.
   A FortiGate Cloud page will open, stating that your account has been confirmed. The *Activation Pending* message on the dashboard will change to state the type of account you have, and will provide a link to the FortiGate Cloud portal.

Enabling logging to FortiGate Cloud

To enable logging to FortiGate Cloud:

1. Go to *Security Fabric > Fabric Connectors > Cloud Logging* or *Log & Report > Log Settings*.
2. Enable *Cloud Logging*.
3. Select an upload option: *Realtime*, *Every Minute*, or *Every 5 Minutes* (default).
4. Click *Apply*.

Logging into the FortiGate Cloud portal

Once logging has been configured and you have registered your account, you can log into the FortiGate Cloud portal and begin viewing your logging results. There are two methods to reach the FortiGate Cloud portal:

- If you have direct network access to the FortiGate:
  - **a.** Go to *Dashboard > Status*.
  - **b.** In the *FortiGate Cloud* widget, in the *Status* field, click *Activated > Launch Portal*, or, in the *Licenses* widget, click *FortiCare Support > Launch Portal*.
- If you do not have access to the FortiGate’s interface, visit the FortiGate Cloud website (https://www.forticloud.com) and log in remotely, using your email and password. It will ask you to confirm the FortiGate Cloud account you are connecting to and then you will be granted access.

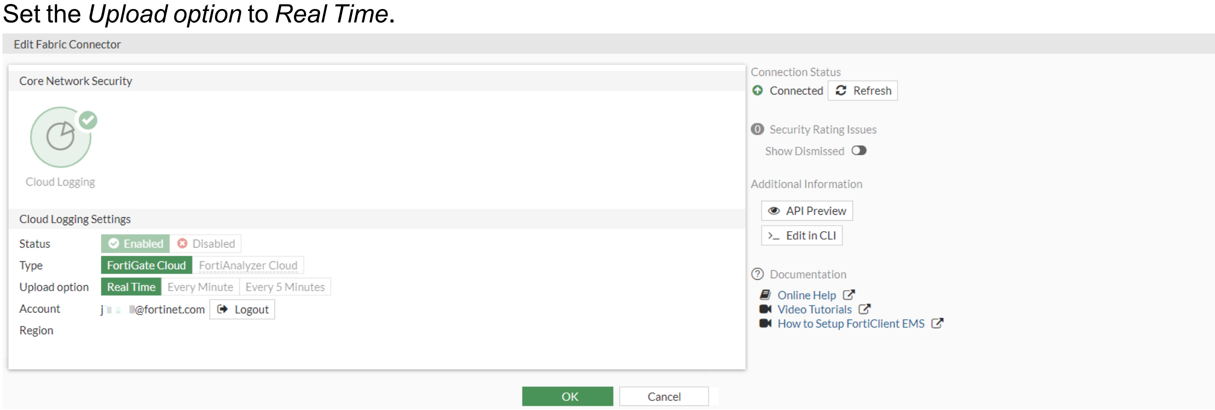
Configuring a Security Fabric with FortiGate Cloud logging

A Security Fabric can be created on the root device using FortiGate Cloud for cloud logging. When the FortiCloud account enforcement is enabled (by default), members joining the Fabric must be registered to the same FortiCloud account. Devices that are not activated with FortiCloud are also allowed.

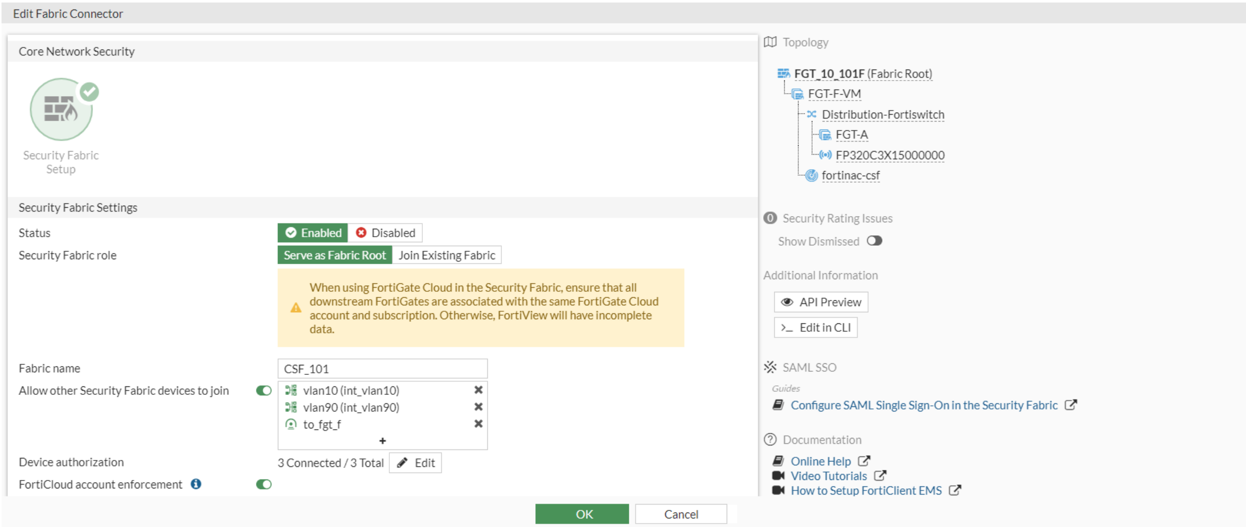
For example, the root FortiGate (FGT_10_101F) is configured with FortiGate Cloud logging. In the Security Fabric settings, the *FortiCloud account enforcement* option is enabled by default. The downstream FortiGate, FGT-F-VM, with the same FortiCloud account ID is able to join the Fabric.

To configure a Security Fabric with FortiCloud logging in the GUI:

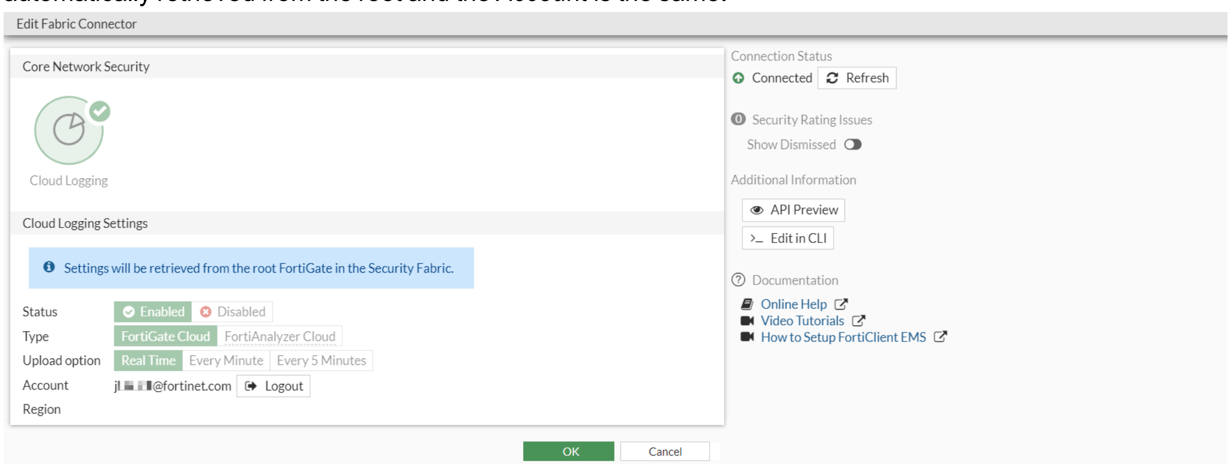
1. On the root FortiGate, configure FortiCloud logging:
   - **a.** Go to *Security Fabric > Fabric Connectors* and double-click the *Cloud Logging* card.
   - **b.** Set the *Type* to *FortiGate Cloud*.
   - **c.** Set the *Upload option* to *Real Time*.



   - **d.** Click *OK*.
2. Configure the Security Fabric settings (see Configuring the root FortiGate and downstream FortiGates on page 2208). The *FortiCloud account enforcement* setting is enabled by default.

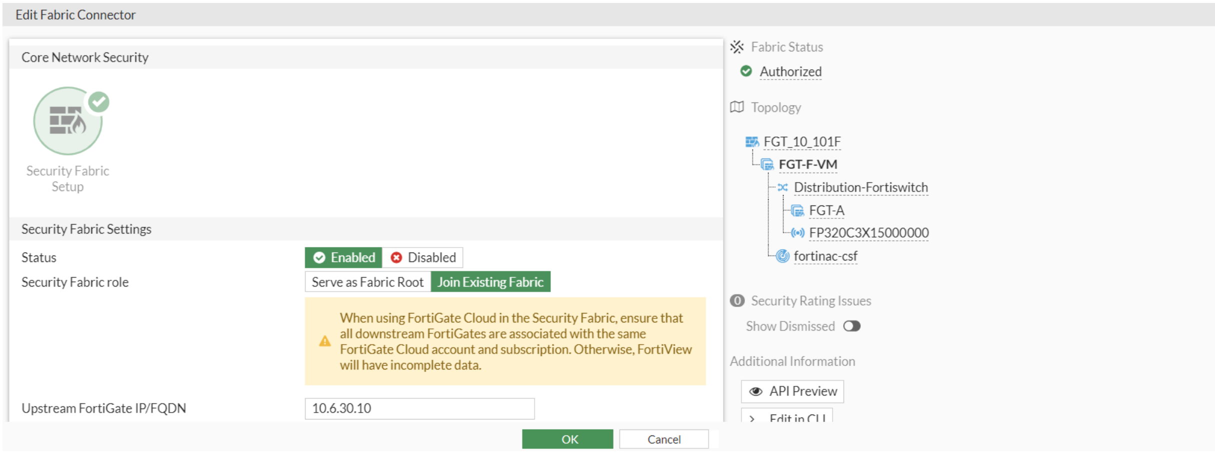


3. On the FGT-F-VM, check the FortiCloud logging settings:
   a. Go to *Security Fabric > Fabric Connectors* and double-click the *Cloud Logging* card. The settings are automatically retrieved from the root and the *Account* is the same.



4. Configure the FGT-F-VM to join the Security Fabric:
   a. Go to *Security Fabric > Fabric Connectors* and double-click the *Security Fabric Setup* card.
   b. Set the *Security Fabric* role to *Join Existing Fabric*.

c. Click *OK*. The FortiGate is authorized and successfully joins the Security Fabric.



To configure a Security Fabric with FortiCloud logging in the CLI:

```
config log fortiguard setting
    set status enable
    set upload-option realtime
end
```

The FortiCloud account enforcement setting is enabled by default in the Security Fabric settings:

```
show system csf
    config system csf
        set status enable
        set group-name "CSF_101"
        set forticloud-account-enforcement enable
    end
```

Cloud sandboxing

FortiGate Cloud can be used for automated sample tracking, or sandboxing, for files from a FortiGate. This allows suspicious files to be sent to be inspected without risking network security. If the file exhibits risky behavior, or is found to contain a virus, a new virus signature is created and added to the FortiGuard antivirus signature database.

See Configuring Sandboxing on page 2228 for instructions to configure FortiGate Cloud Sandbox. Sandboxing results are shown on the *Sandbox* tab in the FortiGate Cloud portal.

Configuring FortiAnalyzer Cloud service

The FortiAnalyzer Cloud service can be used for event logging.

Traffic logs are not currently supported by FortiAnalyzer Cloud without a FortiCloud Premium subscription (AFAC). For information, see Configuring FortiAnalyzer on page 2215.

When FortiAnalyzer Cloud is licensed and enabled (see Deploying FortiAnalyzer Cloud for more information), all event logs are sent to FortiAnalyzer Cloud by default. All traffic logs, security logs, and archive files are not sent to FortiAnalyzer Cloud.

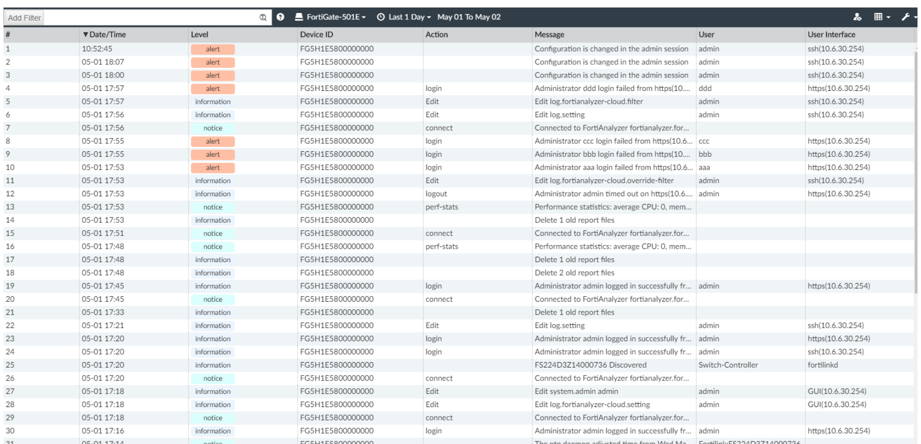
FortiAnalyzer Cloud differs from FortiAnalyzer in the following ways:

- You cannot enable FortiAnalyzer Cloud in `vdom override-setting` when global FortiAnalyzer Cloud is disabled.
- You must use the CLI to retrieve and display logs sent to FortiAnalyzer Cloud. The FortiOS GUI is not supported.
- You cannot enable FortiAnalyzer Cloud and FortiGate Cloud at the same time.

In the *Security Fabric > Fabric Connectors > Cloud Logging* card settings, *FortiAnalyzer Cloud* is grayed out when you do not have a FortiAnalyzer Cloud entitlement. When you have a FortiAnalyzer Cloud entitlement, *FortiAnalyzer Cloud* is available and you can authenticate by the certificate.

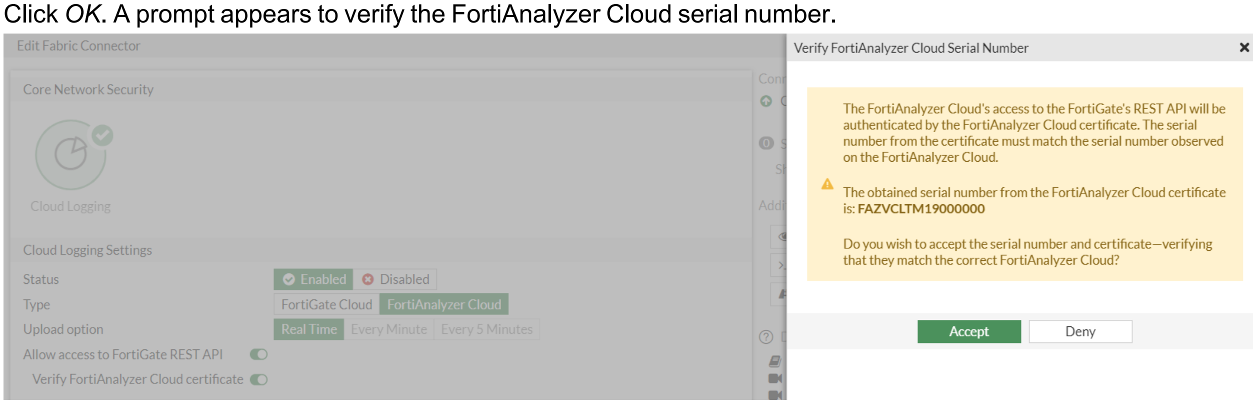
You can also view the FortiAnalyzer Cloud settings in the *Log & Report > Log Settings* page.

In FortiAnalyzer Cloud, you can view logs from FortiOS in the *Event > All Types* page.



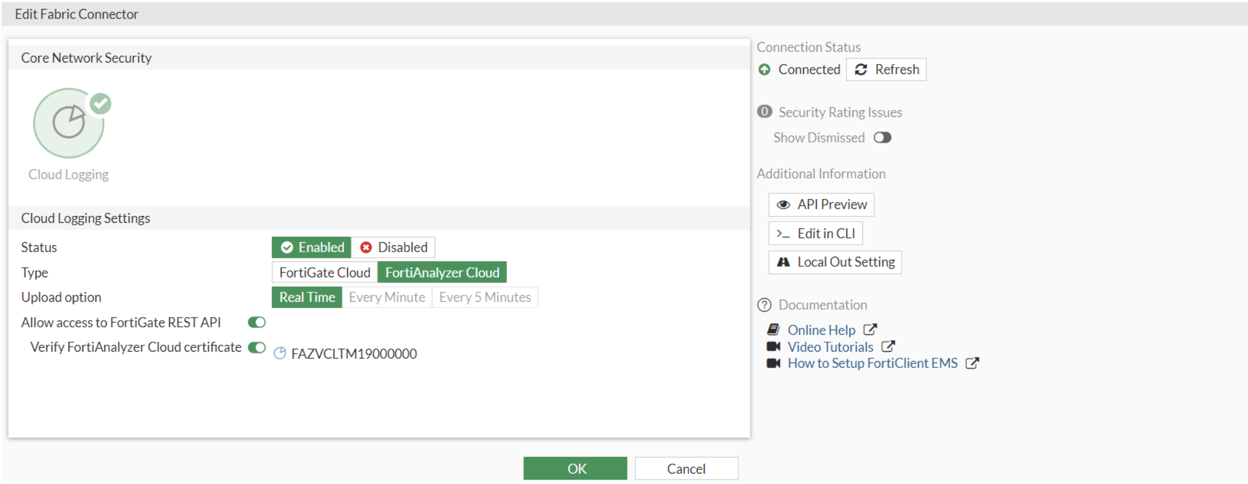
To configure FortiAnalyzer Cloud logging in the GUI:

1. Go to *Security Fabric > Fabric Connectors* and double-click the *Cloud Logging* card.
2. Set the *Type* to *FortiAnalyzer Cloud*.
3. Click *OK*. A prompt appears to verify the FortiAnalyzer Cloud serial number.



4. Click *Accept*.

5. The verified FortiAnalyzer Cloud certificate appears in the settings.



To enable FortiAnalyzer Cloud logging in the CLI:

1. Configure the FortiAnalyzer Cloud settings:

```
config log fortianalyzer-cloud setting
    set status enable
    set ips-archive disable
    set certificate-verification enable
    set serial "FAZVCLTM19000000"
    set access-config enable
    set enc-algorithm high
    set ssl-min-proto-version default
    set conn-timeout 10
    set monitor-keepalive-period 5
    set monitor-failure-retry-period 5
    set upload-option realtime
end
```

2. Configure the FortiAnalyzer Cloud filters:

```
config log fortianalyzer-cloud filter
    set severity information
    set forward-traffic disable
    set local-traffic disable
    set multicast-traffic disable
    set sniffer-traffic disable
    set anomaly disable
    set voip disable
    set dlp-archive disable
end
```

To disable FortiAnalyzer Cloud logging for a specific VDOM in the CLI:

1. Enable override FortiAnalyzer in the general log settings:

```
config log setting
    set faz-override enable
end
```

2. Disable the override FortiAnalyzer Cloud setting:

```
config log fortianalyzer-cloud override-setting
    set status disable
end
```

To set FortiAnalyzer Cloud logging to filter for a specific VDOM in the CLI:

1. Enable override FortiAnalyzer in the general log settings:

```
config log setting
    set faz-override enable
end
```

2. Enable the override FortiAnalyzer Cloud setting:

```
config log fortianalyzer-cloud override-setting
    set status enable
end
```

3. Configure the override filters for FortiAnalyzer Cloud:

```
config log fortianalyzer-cloud override-filter
    set severity information
    set forward-traffic disable
    set local-traffic disable
    set multicast-traffic disable
    set sniffer-traffic disable
    set anomaly disable
    set voip disable
    set dlp-archive disable
end
```

To display FortiAnalyzer Cloud logs in the CLI:

```
# execute log filter device fortianalyzer-cloud
# execute log filter category event
# execute log display
```

Sample log

```
date=2019-05-01 time=17:57:45 idseq=60796052214644736 bid=100926 dvid=1027 itime="2019-05-01
17:57:48" euid=3 epid=3 dsteuid=0 dstepid=3 logver=602000890 logid=0100032002 type="event"
subtype="system" level="alert" srcip=10.6.30.254 dstip=10.6.30.9 action="login"
msg="Administrator ddd login failed from https(10.6.30.254) because of invalid user name"
logdesc="Admin login failed" sn="0" user="ddd" ui="https(10.6.30.254)" status="failed"
reason="name_invalid" method="https" eventtime=1556758666274548325 devid="FG5H1E5818900000"
vd="root" dtime="2019-05-01 17:57:45" itime_t=1556758668 devname="FortiGate-501E"
```

```
date=2019-05-01 time=17:57:21 idseq=60796052214644736 bid=100926 dvid=1027 itime="2019-05-01
17:57:23" euid=3 epid=3 dsteuid=0 dstepid=3 logver=602000890 logid=0100044546 type="event"
subtype="system" level="information" action="Edit" msg="Edit log.fortianalyzer-cloud.filter
" logdesc="Attribute configured" user="admin" ui="ssh(10.6.30.254)" cfgtid=164757536
cfgpath="log.fortianalyzer-cloud.filter" cfgattr="severity[information->critical]"
eventtime=1556758642413367644 devid="FG5H1E5818900000" vd="root" dtime="2019-05-01 17:57:21"
itime_t=1556758643 devname="FortiGate-501E"
```

Configuring FortiManager

When a FortiManager device is added to the Security Fabric, it automatically synchronizes with any connected downstream devices.

To add a FortiManager to the Security Fabric, configure it on the root FortiGate. The root FortiGate then pushes this configuration to downstream FortiGate devices. The FortiManager provides remote management of FortiGate devices over TCP port 541. The FortiManager must have internet access for it to join the Security Fabric.

Once configured, the FortiGate can receive antivirus and IPS updates, and allows remote management through FortiManager or the FortiGate Cloud service. The FortiGate management option must be enabled so that the FortiGate can accept management updates to its firmware and FortiGuard services.

Adding a FortiManager device to the Security Fabric requires the following steps in FortiOS:

- Specify the FortiManager IP address or domain name.
- Approve the FortiManager serial number that is returned by the provided IP address or domain name.

You can complete the steps in FortiOS by using the GUI or CLI.

After you complete the steps in FortiOS, go to FortiManager to complete the process by authorizing the FortiGate.

To add a FortiManager to the Security Fabric using the CLI:

1. Provide FortiManager connection information:

```
config system central-management
    set type fortimanager
    set fmg {<IP_address> | <Domain name>}
end
```

2. Approve the returned FortiManager serial number:

   When configuring the FortiManager connection from the CLI, no prompt is available to approve the returned FortiManager serial number. Therefore you must provide the following command:

```
execute central-mgmt <fmg-serial-no> <PSK>
```

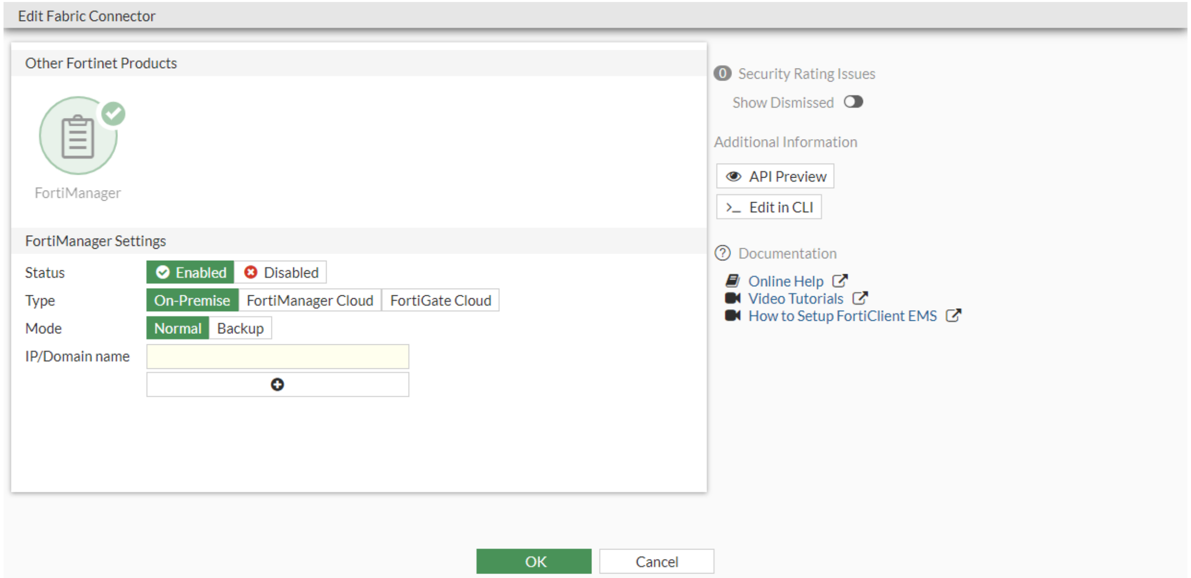
   If you have not previously configured a model device in FortiManager and leveraged a pre-shared key for registration, you can enter any character for the PSK field in the `execute central-mgmt` command.

3. Go to FortiManager and authorize the FortiGate. See Authorizing the FortiGate in FortiManager on page 2226.

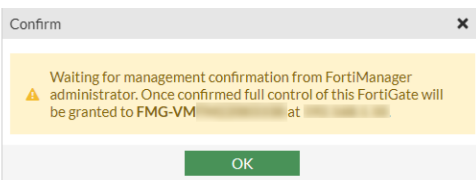
To add a FortiManager to the Security Fabric using the GUI:

1. On the root FortiGate, go to *Security Fabric* > *Fabric Connectors* and double-click the *FortiManager* card.
   The FortiManager card is used to configure the FortiManager connection information.
2. For *Status*, click *Enable*.

3. For *Type*, click *On-Premise*.



4. Enter the *IP/Domain Name* of the FortiManager.
5. Click *OK*.
   The *Confirm* pane appears.



6. Review the serial number, and click *OK*.
7. Go to FortiManager and authorize the FortiGate. See Authorizing the FortiGate in FortiManager on page 2226.

Authorizing the FortiGate in FortiManager

After completing the GUI or CLI steps in FortiOS, go to FortiManager to authorize the FortiGate, which completes the process.

To authorize the FortiGate in FortiManager:

1. On FortiManager, go to *Device Manager* and find the FortiGate in the *Unauthorized Devices* list.
   The unauthorized device list is located in the root ADOM, regardless of the firmware version of the root ADOM or FortiOS.
2. Select the FortiGate device or devices, and click *Authorize* in the toolbar.
3. In the *Authorize Device* pop-up, adjust the device names as needed, select the appropriate ADOM (if applicable), and click *OK*.

For more information about using FortiManager, see the FortiManager Administration Guide.

Configuring FortiManager Cloud service

This cloud-based SaaS management service is available through FortiManager. This service is included in FortiCloud accounts with a FortiManager Cloud account level subscription (ALCI).

Configuring a per-device license

Once the FortiGate has acquired a contract named *FortiManager Cloud*, FortiCloud creates a cloud-based FortiManager instance under the user account. You can launch the portal for the cloud-based FortiManager from FortiCloud, and its URL starts with the User ID.

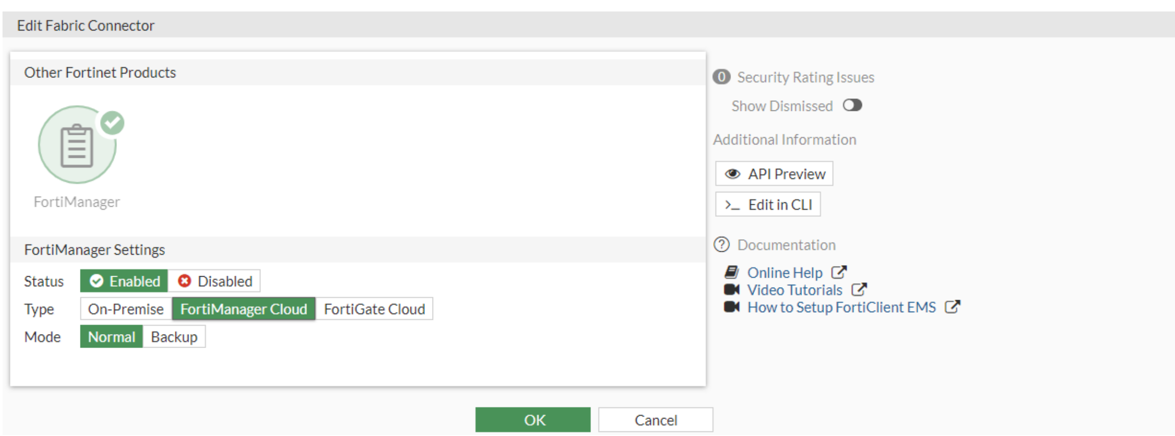
You can use a FortiGate with a contract for *FortiManager Cloud* to configure central management by using the FQDN of *fortimanager.forticloud.com*. A FortiGate-FortiManager tunnel is established between FortiGate and the FortiManager instance.

After the tunnel is established, you can execute FortiManager functions from the cloud-based FortiManager portal.

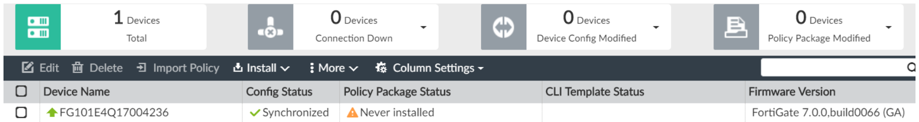
To configure FortiManager Cloud central management:

1. Enable FortiManager Cloud.
   a. Go to *Security Fabric > Fabric Connectors* and double-click the *FortiManager* card.
   b. For *Status*, click *Enable*.
   c. For *Type*, click *FortiManager Cloud*.
   d. Click *OK*.

   The *FortiManager Cloud* button can only be selected if you have a FortiManager Cloud product entitlement.

2. In the FortiManager Cloud instance, go to *Device Manager* and authorize the FortiGate. See Authorizing devices for more information.
   When using the FortiGate to enable FortiManager Cloud, the FortiGate appears as an unauthorized device. After authorizing the FortiGate, it becomes a managed device.



In FortiOS, the *Security Fabric > Fabric Connectors* page now displays green arrow in the *FortiManager* card because FortiManager Cloud is registered.

Diagnostics

To verify the contract information:

```
# diagnose test update info contract
...
System contracts:
    ...
Account contracts:
    FMGC,Thu Dec  2 16:00:00 2021
...
```

To verify the FortiManager Cloud instance has launched and the FortiGate is registered:

```
# diagnose fdsm central-mgmt-status
Connection status: Up
Registration status: Registered
```

Configuring Sandboxing

The Security Fabric supports the following FortiSandbox deployments.

Type	Description	Requirements
FortiGate Cloud Sandbox	Files are sent to Fortinet's Cloud Sandbox cluster for processing.	• The FortiGate must have a valid AV license. • The FortiCloud account provides access to a portal to view submissions. This is not required for the Security Fabric.
FortiSandbox Cloud	Files are sent to a dedicated FortiCloud hosted instance of FortiSandbox for processing.	• FortiCloud premium license • FortiSandbox Cloud entitlement • The FortiGate and FortiCloud license are registered to the same account.
FortiSandbox appliance	Files are sent to a physical appliance or VM, typically residing on premise, for processing.	• None

To apply sandboxing in a Security Fabric, connect one of the FortiSandbox deployments, then configure an antivirus profile to submit files for dynamic analysis. The submission results supplement the AV signatures on the FortiGate. FortiSandbox inspection can also be used in web filter profiles.

In a Security Fabric environment, sandbox settings are configured on the root FortiGate. Once configured, the root FortiGate pushes the settings to other FortiGates in the Security Fabric.

FortiGate Cloud Sandbox

FortiGate Cloud Sandbox allows users to take advantage of FortiSandbox features without having to purchase, operate, and maintain a physical appliance. It also allows you to control the region where your traffic is sent to for analysis. This allows you to meet your country's compliance needs regarding data storage locations.

Users are not required to have a FortiCloud account to use FortiGate Sandbox Cloud.

The submission to the cloud with a valid FortiGuard Antivirus (AVDB) license is rate limited per FortiGate model. Refer to the Service Description for details. For those without any AVDB license, the submission is limited to only 100 per day.

To configure FortiGate Cloud Sandbox, you must first activate the connection from the CLI. Note that FortiGate Cloud Sandbox is decoupled from FortiGate Cloud logging, so you do not need to have a FortiCloud account or have cloud logging enabled.

To activate the FortiGate Cloud Sandbox connection:

```
# execute forticloud-sandbox region
0  Europe
1  Global
2  Japan
3  US
Please select cloud sandbox region[0-3]:3
```

After a region is selected, the following configuration is added:

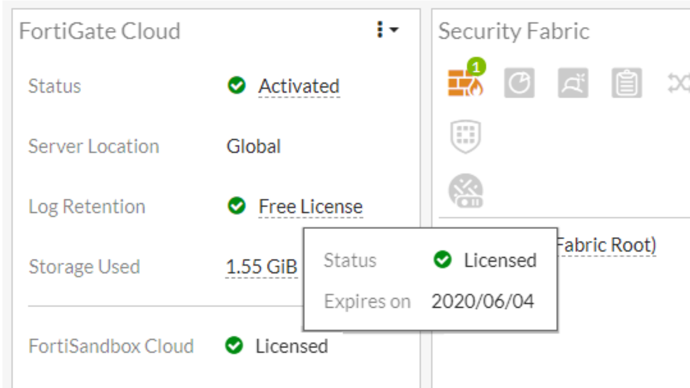
```
config system fortiguard
    set sandbox-region {0 | 1 | 2 | 3}
end
```

Alternatively, using the `execute forticloud-sandbox update` command also works.

To obtain or renew a FortiGuard antivirus license:

1. See the How to Purchase or Renew FortiGuard Services video for FortiGuard antivirus license purchase instructions.

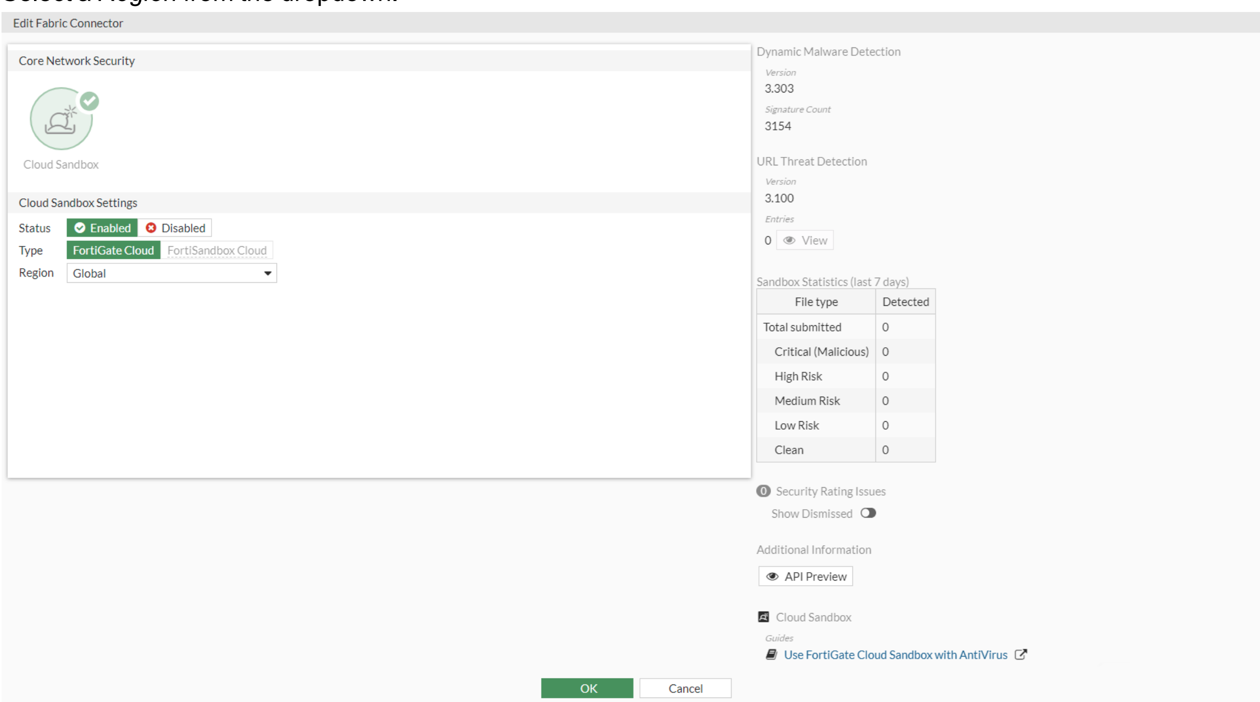
2. Once a FortiGuard license is purchased and activated, users are provided with a paid FortiSandbox Cloud license.
   a. Go to *Dashboard* > *Status* to view the FortiSandbox Cloud license indicator.



   b. Alternatively, go to *System* > *FortiGuard* to view the FortiSandbox Cloud license indicator.

To enable FortiGate Cloud Sandbox in the GUI:

1. Go to *Security Fabric* > *Fabric Connectors* and double-click the *Cloud Sandbox* card.
2. Set *Status* to *Enable*.
3. For Type, select *FortiGate Cloud*.
4. Select a *Region* from the dropdown.



5. Click *OK*.

FortiSandbox Cloud

FortiSandbox Cloud offers more features and better detection capability. Connecting to FortiSandbox Cloud will automatically use the cloud user ID of the FortiGate to connect to the dedicated FortiSandbox Cloud instance. The

FortiGate automatically detects if there is a valid entitlement.

The following items are required to initialize FortiSandbox Cloud:

- A FortiCloud premium account.
- A valid FortiSandbox Cloud contract on the FortiGate. To view contract information in the CLI, enter `diagnose test update info`. The `User ID` at the end of the output shows FortiCloud which FortiSandbox Cloud account the FortiGate is connected to.
- A provisioned FortiSandbox Cloud. See Deploying FortiSandbox Cloud for information.

To configure FortiSandbox Cloud in the GUI:

1. Go to *Security Fabric > Fabric Connectors* and double-click the *Cloud Sandbox* card.
2. Set *Status* to *Enable*.
3. For *Type*, select *FortiSandbox Cloud*.

If the *FortiSandbox Cloud* option is grayed out or not visible, enter the following in the CLI:

```
config system global
    set gui-fortigate-cloud-sandbox enable
end
```

4. Click *OK*.

To configure FortiSandbox Cloud in the CLI:

```
config system fortisandbox
    set status enable
    set forticloud enable
    set server <string>
end
```

If the FortiGate does not detect the proper entitlement, a warning is displayed and the CLI configuration will not save.

If the FortiSandbox Cloud is running version 4.0.0 and later, the FortiGate will automatically connect to fortisandboxcloud.com, and then discover the specific region and server to connect to based on which region the customer selected to deploy their FortiSandbox Cloud instance. The FortiGate must have a FortiCloud premium account license and a FortiSandbox Cloud VM license for this functionality.

FortiSandbox appliance

FortiSandbox appliance is the on-premise option for a full featured FortiSandbox. Connecting to a FortiSandbox appliance requires that Cloud Sandbox is disabled.

To switch from Cloud Sandbox to FortiSandbox in the Security Fabric:

1. Go to *Security Fabric > Fabric Connectors* and double-click the *Cloud Sandbox* card.
2. Set *Status* to *Disabled*.
3. Click *OK*.

To enable FortiSandbox appliance in the GUI:

1. Go to *Security Fabric > Fabric Connectors* and double-click the *FortiSandbox* card.
2. Set *Status* to *Enable*.
3. In the *Server* field, enter the FortiSandbox device's IP address.
4. Optionally, enter a *Notifier email*.
5. Click *OK*.

To enable FortiSandbox appliance in the CLI:

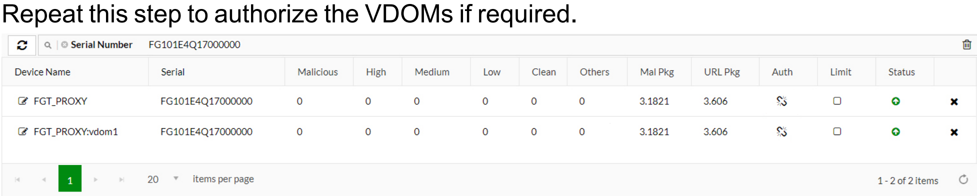
```
config system fortisandbox
    set status enable
    set forticloud disable
    set server <address>
end
```

Authorizing the FortiGate from FortiSandbox Cloud and a FortiSandbox appliance

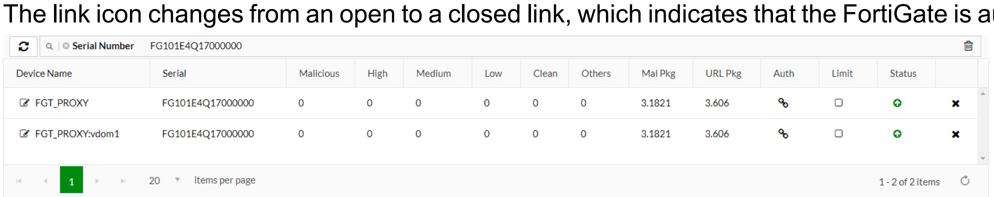
Once the FortiGate makes a connection to the FortiSandbox Cloud or appliance, the FortiGate must be authorized.

To authorize a FortiGate from FortiSandbox:

1. In the FortiSandbox GUI, go to *Scan Input > Device* in 3.2 or *Security Fabric > Device* in 4.0.
2. Search using the FortiGate serial number to locate the FortiGate. In the *Auth* column, click the link icon to authorize the FortiGate.
3. Repeat this step to authorize the VDOMs if required.



The link icon changes from an open to a closed link, which indicates that the FortiGate is authorized.



4. In the FortiGate GUI, go to *Security Fabric > Fabric Connectors* and double-click the *FortiSandbox* card.
5. Click *Test connectivity*. The FortiGate is now authorized and the status displays as *Connected*.

Antivirus profiles

An antivirus profile must be configured to send files to the sandbox. Once submitted, sandbox inspection is performed on the file to detect malicious activities. The FortiGate can use the dynamic malware detection database from the sandbox to supplement the AV signature database. See Using FortiSandbox post-transfer scanning with antivirus on page 1111 for more information.

FortiSandbox inline scanning is supported on FortiSandbox appliances in proxy inspection mode. When inline scanning is enabled, the client's file is held while it is sent to FortiSandbox for inspection. Once a verdict is returned, the appropriate action is performed on the held file. If there is an error or timeout on the FortiSandbox, the FortiGate's configuration determines what to do with the held file. See Using FortiSandbox inline scanning with antivirus on page 1113 for more information.

Inline scanning requires a FortiSandbox appliance running version 4.2 or later. This feature is not supported on FortiSandbox Cloud or FortiGate Cloud Sandbox.

Web filter profiles

Sandbox inspection can be used in web filter profiles. The FortiGate uses URL threat detection database from the sandbox to block malicious URLs. See Block malicious URLs discovered by FortiSandbox on page 1146 for more information.

Top FortiSandbox Files FortiView monitor

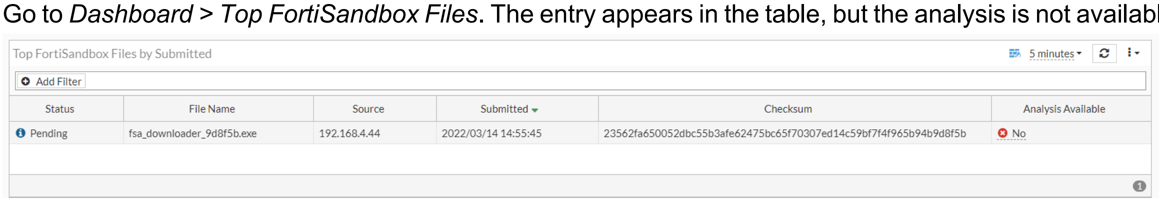
In the *Top FortiSandbox Files* FortiView monitor, users can select a submitted file and drill down to view its static and dynamic file analysis. The full FortiSandbox report can be downloaded in PDF format. This feature works with FortiGate Cloud Sandbox, FortiSandbox Cloud, and FortiSandbox appliance. FortiSandbox must be running version 3.2.1 and later.

Prerequisites:

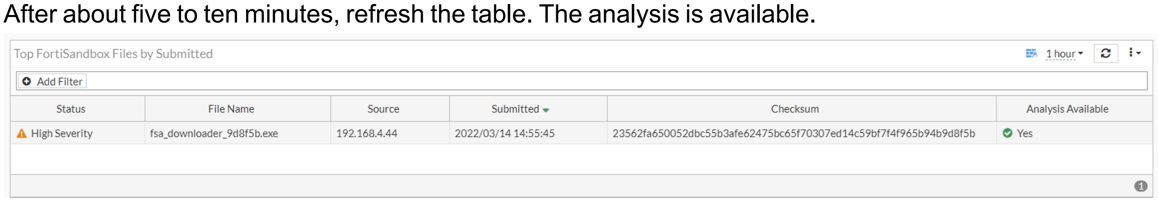
1. Add FortiSandbox to the Security Fabric.
2. Configure an AV profile with *Send files to FortiSandbox for inspection* enabled (see Using FortiSandbox post-transfer scanning with antivirus on page 1111).
3. Configure a firewall policy with the AV profile that allows traffic to the internet.
4. Add the *Top FortiSandbox Files* FortiView monitor (see Adding FortiView monitors on page 108).
5. On a client PC, attempt to download a suspicious file.

To view the FortiSandbox analysis and download the PDF:

1. Go to *Dashboard > Top FortiSandbox Files*. The entry appears in the table, but the analysis is not available yet.

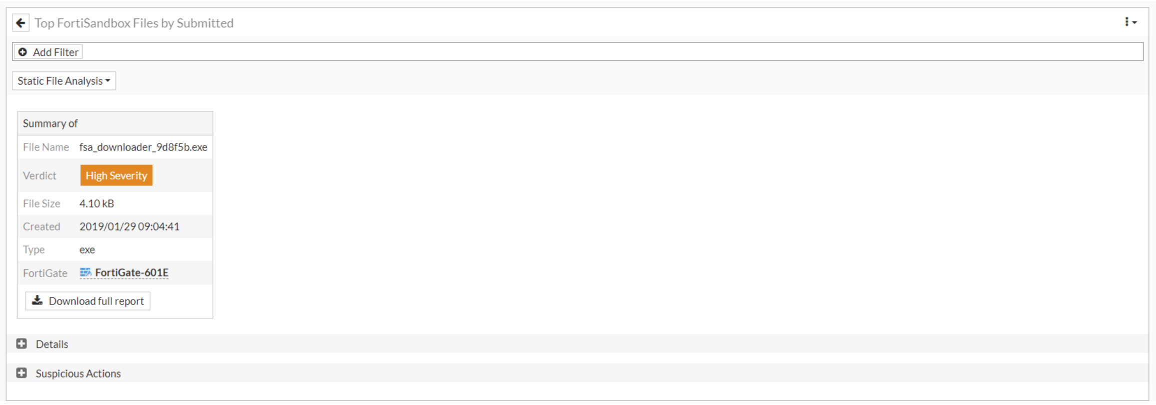


2. After about five to ten minutes, refresh the table. The analysis is available.

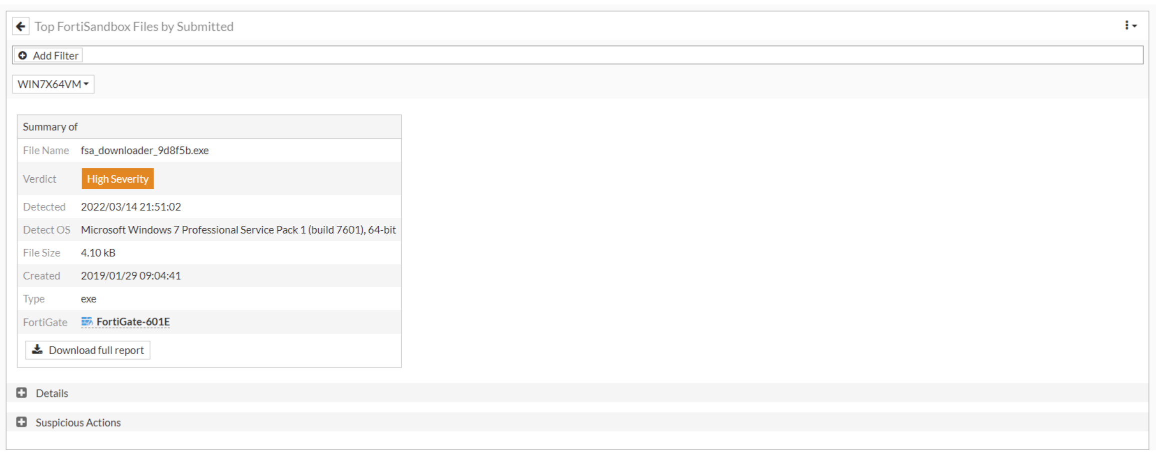


3. Select the entry, then right-click and select *Drill Down to Details*.

4. In the dropdown, select *Static File Analysis* to view the static file analysis.



5. In the dropdown, select the client device to view the dynamic file analysis.



6. Click *Download full report* to download the detailed PDF report. The reports contains FortiSandbox job information, detailed file information, static analysis results, and dynamic analysis results.

Configuring FortiClient EMS

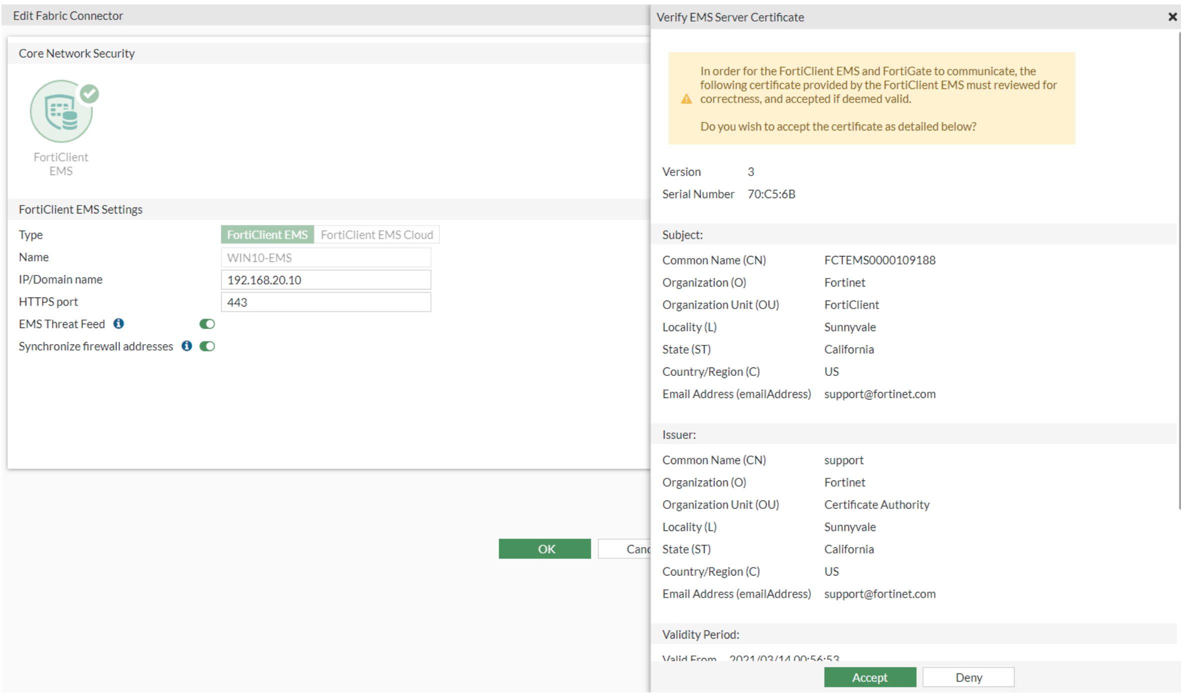
The FortiGate Security Fabric root device can link to FortiClient Endpoint Management System (EMS) and FortiClient EMS Cloud (a cloud-based EMS solution) for endpoint connectors and automation. Up to three EMS servers can be added to the Security Fabric, including a FortiClient EMS Cloud server. EMS settings are synchronized between all fabric members.

To enable cloud-based EMS services, the FortiGate must be registered to FortiCloud with an appropriate user account. The following examples presume that the EMS certificate has already been configured.

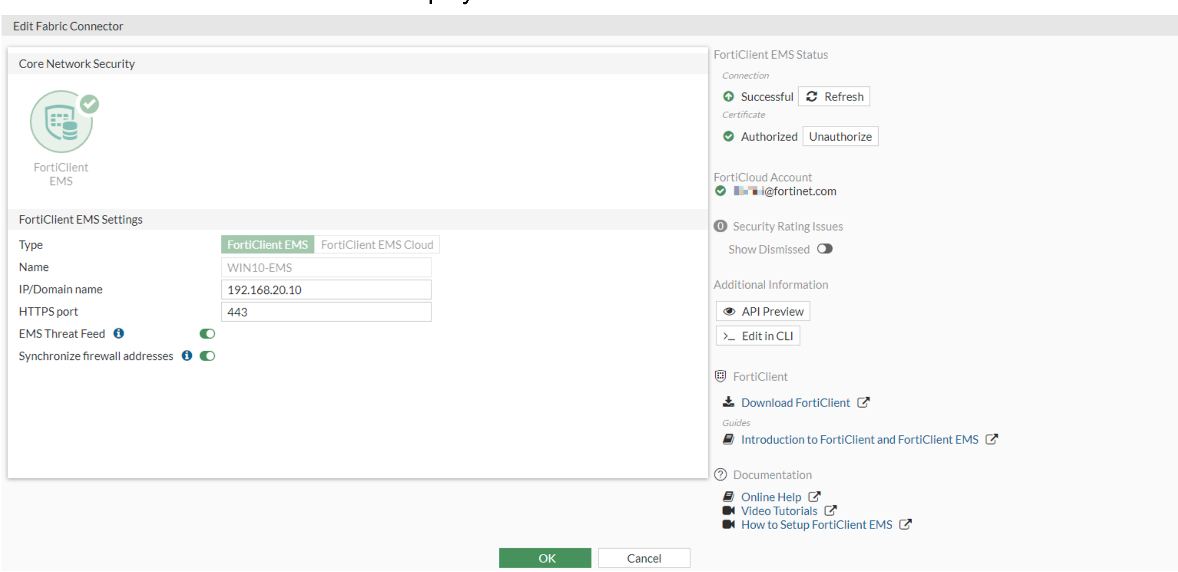
To add an on-premise FortiClient EMS server to the Security Fabric in the GUI:

1. On the root FortiGate, go to *System* > *Feature Visibility* and enable *Endpoint Control*.
2. Go to *Security Fabric* > *Fabric Connectors* and double-click the *FortiClient EMS* card.
3. For *Type*, click *FortiClient EMS*.
4. Optionally, enable *EMS Threat Feed*. See Malware threat feed from EMS on page 1101 for more information about using this setting in an AV profile.

5. Enter a name and IP address or FQDN. When connecting to a multitenancy-enabled EMS, Fabric connectors must use an FQDN to connect to EMS, where the FQDN hostname matches a site name in EMS (including "Default"). The following are examples of FQDNs to provide when configuring the connector to connect to the default site and to a site named SiteA, respectively: default.ems.yourcompany.com, sitea.ems.yourcompany.com. See Multitenancy.
6. Click *OK*.
   A window appears to verify the EMS server certificate:



7. Click *Accept*.
   The *FortiClient EMS Status* section displays a *Successful* connection and an *Authorized* certificate:



8. If the device is not authorized, log in to the FortiClient EMS to authorize the FortiGate under *Administration > Fabric Devices*.

To add a FortiClient EMS Cloud server to the Security Fabric in the GUI:

FortiClient EMS Cloud can only be configured when the FortiGate is registered to FortiCloud and the EMS Cloud entitlement is verified.

If the FortiCloud account does not pass the FortiClient EMS Cloud entitlement check, the option is not selectable in the FortiClient EMS connector settings.

1. Go to *Security Fabric > Fabric Connectors* and double-click the *FortiClient EMS* card.
2. Set *Type* to *FortiClient EMS Cloud*.
3. Enter a name.
4. Click *OK*.
   A window appears to verify the EMS server certificate.
5. Click *Accept*.
   The *FortiClient EMS Status* section displays a *Successful* connection and an *Authorized* certificate.

To test connectivity with the EMS server:

1. Go to *Security Fabric > Fabric Connectors* and double-click the *FortiClient EMS* or *FortiClient EMS Cloud* card.
2. In the *FortiClient EMS Status* section under *Connection*, click *Refresh*.

To add an on-premise FortiClient EMS server to the Security Fabric in the CLI:

```
config endpoint-control fctems
    edit <ems_name>
        set server <ip_address>
        set certificate <string>
        set https-port <integer>
        set source-ip <ip_address>
    next
end
```

The `https-port` is the EMS HTTPS access port number, and the `source-ip` is the REST API call source IP address.

To add a FortiClient EMS Cloud server to the Security Fabric in the CLI:

```
config endpoint-control fctems
    edit <name>
        set fortinetone-cloud-authentication enable
        set certificate <string>
    next
end
```

To verify the EMS Cloud entitlement in the CLI:

```
# diagnose test update info
```

To verify an EMS certificate in the CLI:

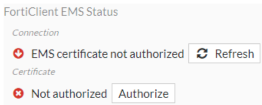
```
# execute fctems verify ems137

        Subject:     C = CA, ST = bc, L = burnaby, O = devqa, OU = top3, CN =
sys169.qa.fortinet.cm, emailAddress = xxxx@xxxxxxxx.xxx
```

```
        Issuer:      CN = 155-sub1.fortinet.com
        Valid from:  2017-12-05 00:37:57  GMT
        Valid to:    2027-12-02 18:08:13  GMT
        Fingerprint: D3:7A:1B:84:CC:B7:5C:F0:A5:73:3D:BB:ED:21:F2:E0
        Root CA:     No
        Version:     3
        Serial Num:
                01:86:a2
        Extensions:
                Name:     X509v3 Basic Constraints
                Critical: yes
                Content:
                CA:FALSE

                Name:     X509v3 Subject Key Identifier
                Critical: no
                Content:
                35:B0:E2:62:AF:9A:7A:E6:A6:8E:AD:CB:A4:CF:4D:7A:DE:27:39:A4

                Name:     X509v3 Authority Key Identifier
                Critical: no
                Content:
                keyid:66:54:0F:78:78:91:F2:E4:08:BB:80:2C:F6:BC:01:8E:3F:47:43:B1

DirName:/C=CA/ST=bc/L=burnaby/O=devqa/OU=top3/CN=fac155.fortinet.com/emailAddress=xyguo@fort
inet.com
serial:01:86:A4

                Name:     X509v3 Subject Alternative Name
                Critical: no
                Content:
                DNS:sys169.qa.fortinet.cm

                Name:     X509v3 Key Usage
                Critical: no
                Content:
                Digital Signature, Non Repudiation, Key Encipherment, Data Encipherment, Key
Agreement, Certificate Sign, CRL Sign, Encipher Only, Decipher Only

                Name:     X509v3 Extended Key Usage
                Critical: no
                Content:
                TLS Web Server Authentication, TLS Web Client Authentication

EMS configuration needs user to confirm server certificate.
Do you wish to add the above certificate to trusted remote certificates? (y/n)y
```

Troubleshooting

Certificate not trusted

When configuring a new connection to an EMS server, the certificate might not be trusted.



When you click *Authorize*, a warning displays: *The server certificate cannot be authenticated with installed CA certificates. Please install its CA certificates on this FortiGate*.

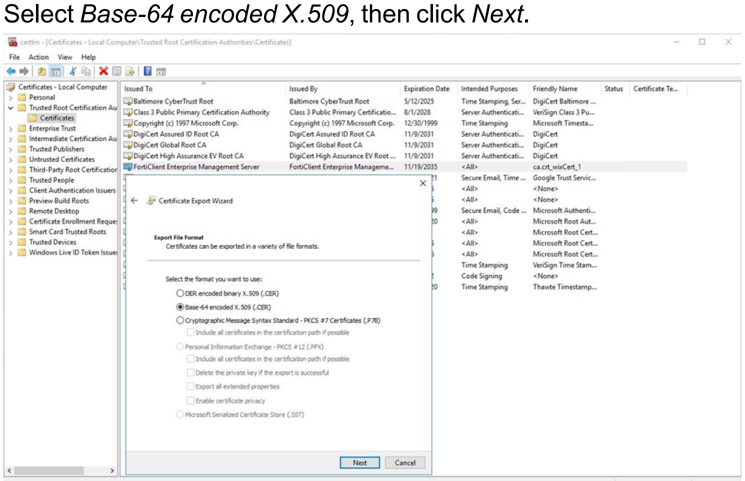
In the CLI, an error message displays when you try to verify the certificate:

```
# execute fctems verify Win2K16-EMS
certificate not configured/verified: 2
Could not verify server certificate based on current certificate authorities.
Error 1--92-60-0 in get SN call: EMS Certificate is not signed by a known CA.
```

The default FortiClient EMS certificate that is used for the SDN connection is signed by the CA certificate that is saved on the Windows server when FortiClient EMS is first installed. You can manually export and install it on the FortiGate.

To manually export and install the certificate on to the FortiGate:

1. Export the EMS certificate on the server that EMS is installed on:
   a. On the Windows server that EMS is installed on, go to *Settings > Manage computer certificates*.
   b. In the certificate management module, go to *Trusted Root Certification Authorities > Certificates*.
   c. Right click on the certificate issued by FortiClient Enterprise Management Server and select *All Tasks > Export*.
   d. The *Certificate Export Wizard* opens. Click *Next*.
   e. Select *Base-64 encoded X.509*, then click *Next*.



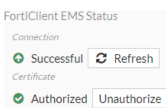
   f. Enter a file name for the certificate and click *Browse* to select the folder where it will be located, then click *Next*.
   g. Review the settings, then click *Finish*. The certificate is downloaded to the specified folder.
2. On the FortiGate, import the certificate:
   a. Go to *System > Certificate*. By default, the *Certificate* option is not visible, see Feature visibility on page 2163 for information.
   b. Click *Import > CA Certificate*.
   c. Set *Type* to *File*, and click *Upload* to import the certificate from the management computer.
   d. Click *OK*. The imported certificate is shown in the *Remote CA Certificate* section of the certificate table.
3. Try to authorize the certificate on the FortiGate:
   a. Go to *Security Fabric > Fabric Connectors* and edit the FortiClient EMS connector. The connection status should now say that the certificate is not authorized.

b. Click *Authorize*. The following warning is shown:

The warning can also be seen in the CLI:

```
# execute fctems verify Win2K16-EMS
failure in certificate configuration/verification: -4
Could not verify EMS. Error 1--94-0-401 in get SN call: Authentication denied.
```

4. Authorize the FortiGate on EMS:
   a. Log in to the EMS server console and go to *Administration > Fabric Devices*.
   b. Select the serial number of the FortiGate device, then click *Authorize*.
5. Try to authorize the certificate on the FortiGate again:
   a. On the FortiGate, go to *Security Fabric > Fabric Connectors* and edit the FortiClient EMS connector.
   b. Click *Authorize*.
   c. When presented with the EMS server certificate, click *Accept* to accept the certificate.
   Your connection should now be successful and authorized.



   d. Click *OK*.

Using EMS silent approval in the Security Fabric

FortiClient EMS with Fabric authorization and silent approval capabilities can approve the root FortiGate in a Security Fabric once, and then silently approve remaining downstream FortiGates in the Fabric. Similarly in an HA scenario, an approval only needs to be made once to the HA primary unit. The remaining cluster members are approved silently.

To use EMS silent approval:

1. Configure the EMS entry on the root FortiGate or HA primary:

```
config endpoint-control fctems
    edit "ems139"
        set fortinetone-cloud-authentication disable
        set server "172.16.200.139"
        set https-port 443
        set source-ip 0.0.0.0
        set pull-sysinfo enable
        set pull-vulnerabilities enable
        set pull-avatars enable
        set pull-tags enable
        set pull-malware-hash enable
        unset capabilities
        set call-timeout 30
        set websocket-override disable
    next
end
```

When the entry is created, the capabilities are unset by default.

2. Authenticate the FortiGate with EMS:

```
# execute fctems verify ems_139
...
```

The FortiGate will enable the Fabric authorization and silent approval based on the EMS supported capabilities.

```
config endpoint-control fctems
    edit "ems139"
        set server "172.18.62.12"
        set capabilities fabric-auth silent-approval websocket
    next
end
```

3. Configure a downstream device in the Security Fabric (see Configuring the root FortiGate and downstream FortiGates on page 2208 for more details). The downstream device will be silently approved.
4. Configure a secondary device in an HA system (see HA active-passive cluster setup on page 2035 and HA active-active cluster setup on page 2037 for more details). The secondary device will be silently approved.

Allowing deep inspection certificates to be synchronized to EMS and distributed to FortiClient

On FortiClient EMS versions that support `push CA certs` capability, the FortiGate will push CA certificates used in SSL deep inspection (see Deep inspection on page 1296 for more details) to the EMS server. On the EMS server, the CA certificates can be selected in the managed endpoint profiles so they can be installed on managed endpoints. FortiClient EMS 7.0.1 and later is required to use this feature.

To configure deep inspection certificate synchronization to EMS:

1. Configure the EMS Fabric connector:

```
config endpoint-control fctems
    edit "ems138"
        set fortinetone-cloud-authentication disable
        set server "172.16.200.138"
        set https-port 443
        set source-ip 0.0.0.0
        set pull-sysinfo enable
        set pull-vulnerabilities enable
        set pull-avatars enable
        set pull-tags enable
        set pull-malware-hash enable
        set capabilities fabric-auth silent-approval websocket websocket-malware push-
ca-certs
        set call-timeout 30
        set websocket-override disable
        set preserve-ssl-session disable
    next
end
```

2. Apply the certificate to an SSL/SSH profile for deep inspection:

```
config firewall ssl-ssh-profile
    edit "deep-inspection"
```

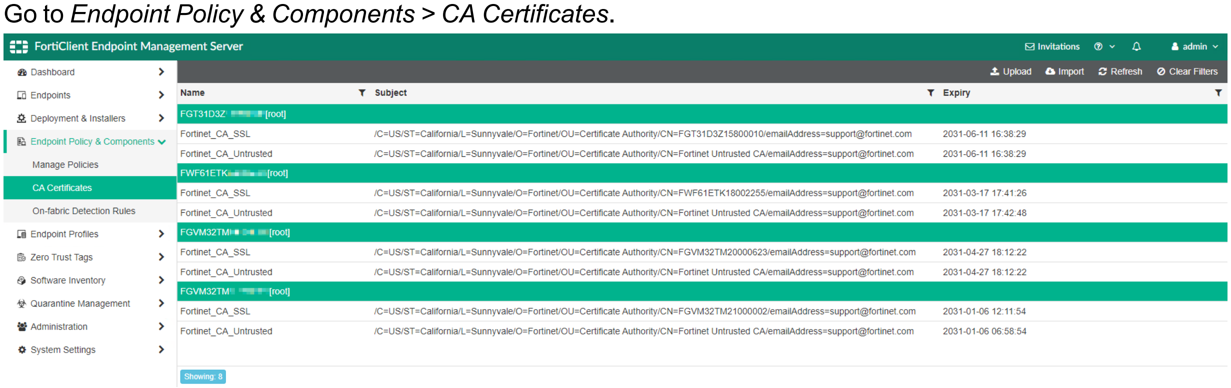
```
            set comment "Read-only deep inspection profile."
            config https
                set ports 443
                set status deep-inspection
            end
            ...
            set server-cert-mode re-sign
            set caname "Fortinet_CA_SSL"
            set untrusted-caname "Fortinet_CA_Untrusted"
        next
    end
```

The default deep inspection profile, CA certificate, and untrusted CA certificates are used in this example.

3. Configure the firewall policy:

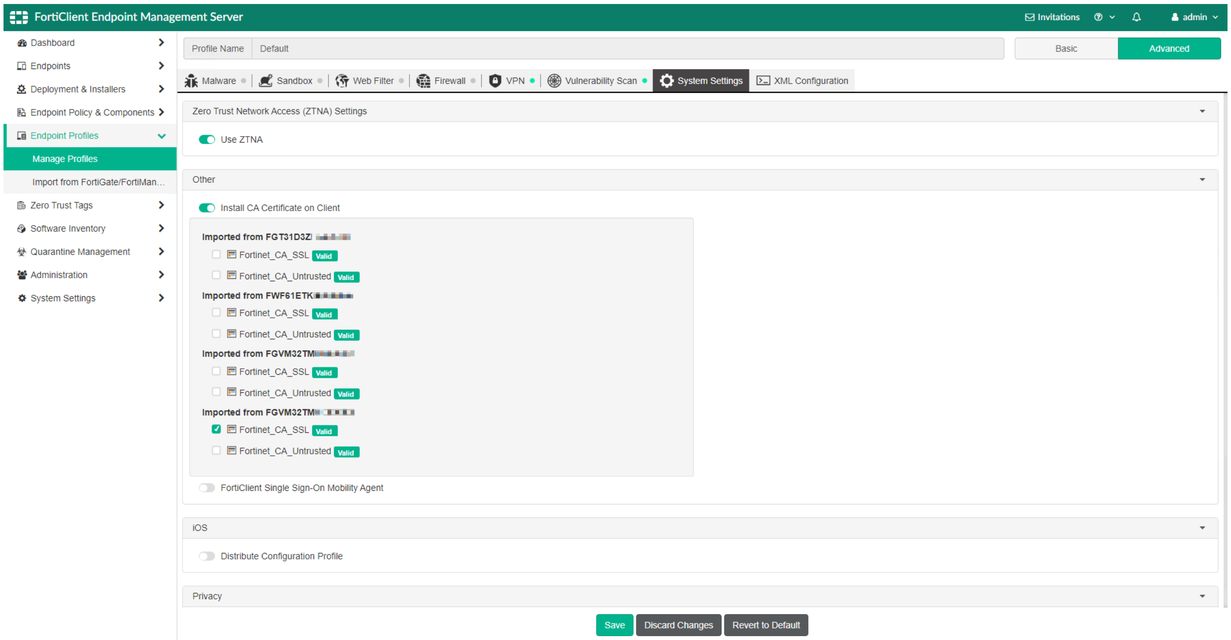
```
config firewall policy
    edit 1
        set name "deep-inspection"
        set srcintf "port14"
        set dstintf "port13"
        set action accept
        set srcaddr "all"
        set dstaddr "all"
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set inspection-mode proxy
        set ssl-ssh-profile "deep-inspection"
        set av-profile "default"
        set nat enable
    next
end
```

4. In EMS, verify that the CA certificate was pushed to EMS:
    a. Go to *Endpoint Policy & Components > CA Certificates*.



    b. Verify the certificate table to see that the EMS server received the CA certification from the different FortiGates.

5. Select the CA certificate in the endpoint profile:
    a. Go to *Endpoint Profiles > Manage Profiles* and edit a profile. The default profile is used in this example.
    b. Click *Advanced* in the top right corner and click the *System Settings* tab.
    c. In the *Other* section, enable *Install CA Certificate on Client* and select the *Fortinet_CA_SSL* certificate for the desired endpoint.

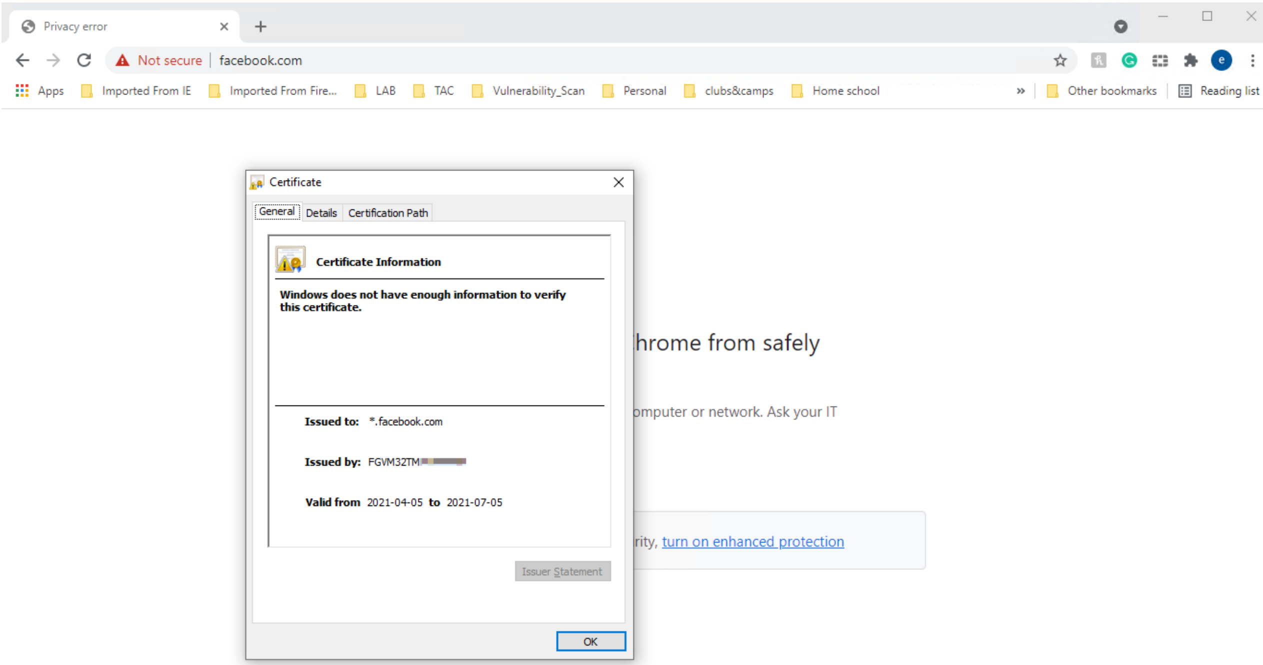


d. Click *Save*.
Once the FortiClient endpoint is registered, it receives the CA certificate. When the FortiClient endpoint tries to access the internet through the FortiGate with the firewall policy that has deep inspection, no warning message is displayed. The server certificate is trusted with the installed CA certificate to complete the certificate chain.

Verification

Before configuring deep inspection certificate synchronization, a warning message is displayed when a FortiClient endpoint accesses the internet through the FortiGate with the firewall policy that has deep inspection. The FortiClient certificate store does not have the FortiGate's CA that is used in the deep inspection SSL/SSH profile.

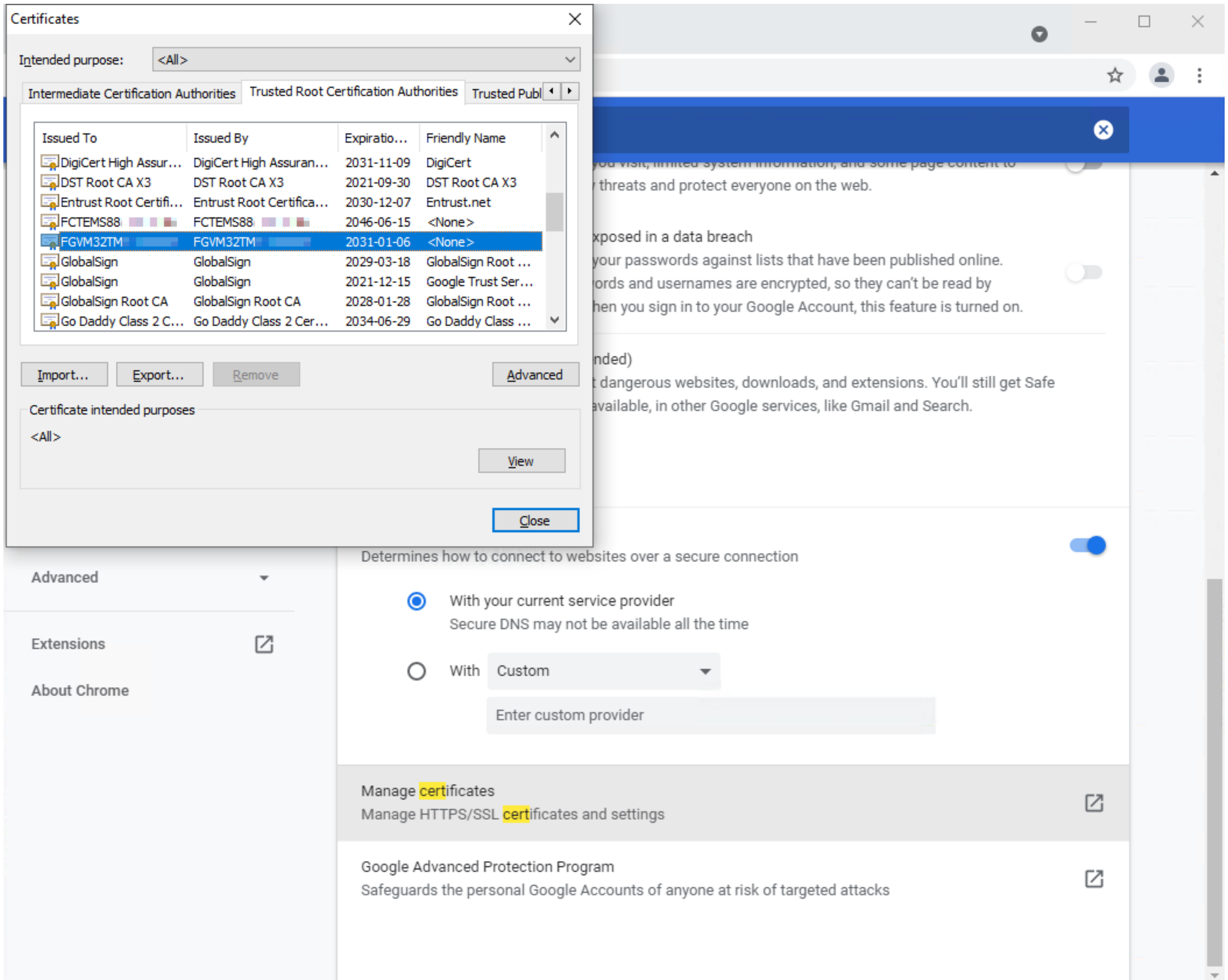
For example, accessing https://www.facebook.com in Chrome shows a warning. In the address bar, clicking *Not secure > Certificate* opens the *Certificate* dialog, which indicates that *Windows does not have enough information to verify the certificate.*



After the EMS profile is pushed to FortiClient endpoint, the expected FortiGate's certificate is shown in its certificate store.

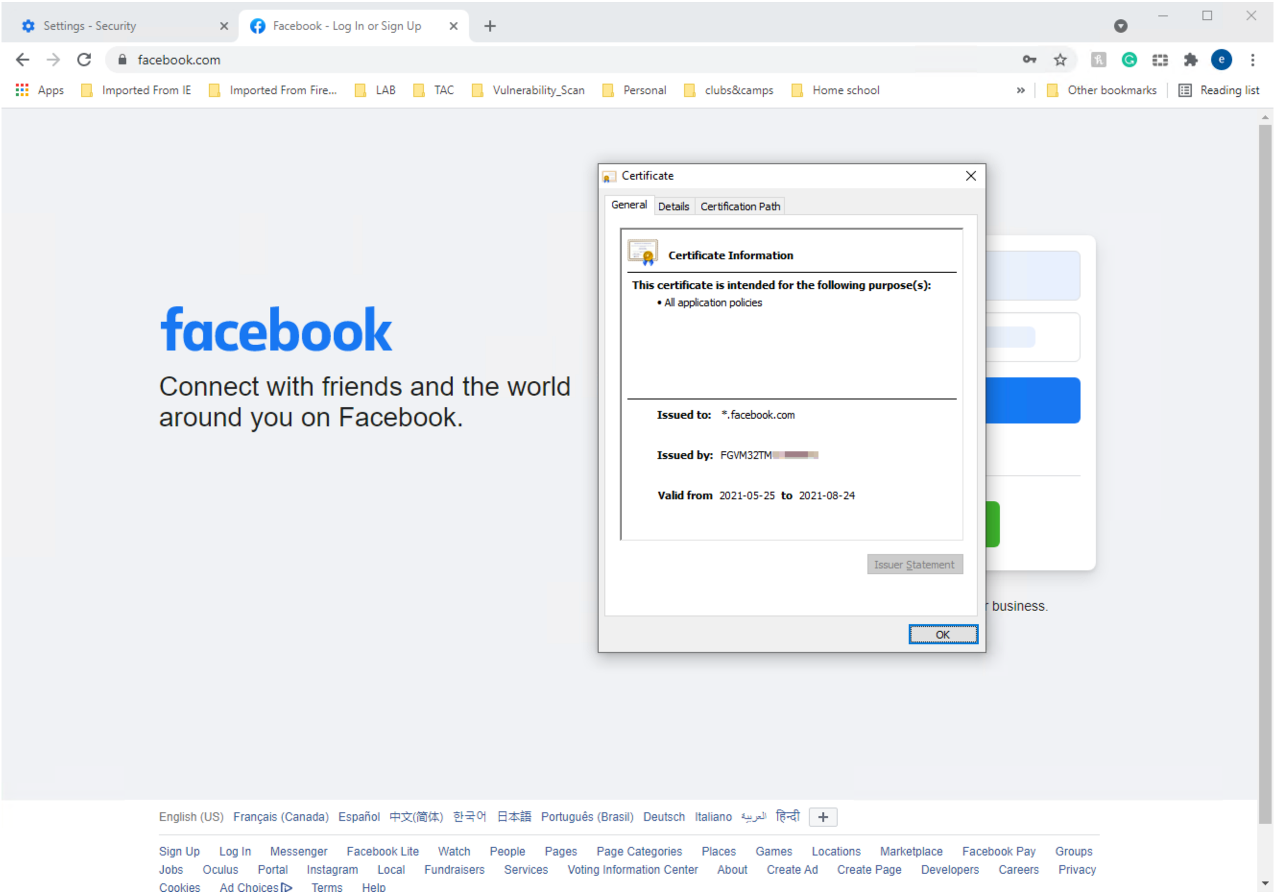
To verify the deep inspection certificate synchronization:

1. In Chrome, go to *Settings > Privacy and security* and open *Manage certificates*.
2. Click the *Trusted Root Certification Authorities* tab. The FortiGate's certificate appears in the list.



3. On the FortiClient endpoint using Chrome, go to https://www.facebook.com. The website is displayed.

4. In the address bar, click the padlock, then click *Certificate*. The dialog displays the valid certificate information.



Diagnostics

Use the `diagnose endpoint fctems json deep-inspect-cert-sync` command in FortiOS to verify the certificate information. In the following example, there are multiple VDOMs with FortiGates in HA mode.

To verify the primary FortiGate:

```
FGT_EC_Primary (global) # diagnose endpoint fctems json deep-inspect-cert-sync
JSON:
"""
{
  "fortigates":[
    "FG2K5E39169*****",
    "FG2K5E39169*****"
  ],
  "vdoms":[
    {
      "vdom":"root",
      "certs":[
        {
          "name":"Fortinet_CA_SSL",
          "cert":"-----BEGIN CERTIFICATE-----\\nMIID5jCCAs6g...Sfu+Q8zE8Crmt6L1X\/bv+q\\n---
--END CERTIFICATE-----\\n"
```

```
        },
        {
          "name":"Fortinet_CA_Untrusted",
          "cert":"-----BEGIN CERTIFICATE-----\\nMIID8DCCAtig...3zBbfzP+nVUpC\\nZDPRZA==\\n--
---END CERTIFICATE-----"
        }
      ]
    },
    {
      "vdom":"vdom1",
      "certs":[
        {
          "name":"Fortinet_CA_SSL",
          "cert":"-----BEGIN CERTIFICATE-----\\nMIID5jCCAs6g...Sfu+Q8zE8Crmt6L1X\/bv+q\\n---
--END CERTIFICATE-----\\n"
        },
        {
          "name":"Fortinet_CA_Untrusted",
          "cert":"-----BEGIN CERTIFICATE-----\\nMIID8DCCAtig...3zBbfzP+nVUpC\\nZDPRZA==\\n--
---END CERTIFICATE-----"
        }
      ]
    }
  ]
}
"""
```

To verify the secondary FortiGate:

```
FGT_EC_Secondary(global) # diagnose endpoint fctems json deep-inspect-cert-sync
JSON:
"""
{
  "fortigates":[
    "FG2K5E39169*****",
    "FG2K5E39169*****"
  ],
  "vdoms":[
    {
      "vdom":"root",
      "certs":[
        {
          "name":"Fortinet_CA_SSL",
          "cert":"-----BEGIN CERTIFICATE-----\\nMIID5jCCAs6g...Sfu+Q8zE8Crmt6L1X\/bv+q\\n---
--END CERTIFICATE-----\\n"
        },
        {
          "name":"Fortinet_CA_Untrusted",
          "cert":"-----BEGIN CERTIFICATE-----\\nMIID8DCCAtig...3zBbfzP+nVUpC\\nZDPRZA==\\n--
---END CERTIFICATE-----"
        }
      ]
    },
    {
      "vdom":"vdom1",
      "certs":[
```

```
        {
          "name":"Fortinet_CA_SSL",
          "cert":"-----BEGIN CERTIFICATE-----\\nMIID5jCCAs6g...Sfu+Q8zE8Crmt6L1X\/bv+q\\n---
--END CERTIFICATE-----\\n"
        },
        {
          "name":"Fortinet_CA_Untrusted",
          "cert":"-----BEGIN CERTIFICATE-----\\nMIID8DCCAtig...3zBbfzP+nVUpC\\nZDPRZA==\\n--
---END CERTIFICATE-----"
        }
      ]
    }
  ]
}
"""
```

Synchronizing FortiClient ZTNA tags

ZTNA tags (formerly FortiClient EMS tags in FortiOS 6.4 and earlier) are tags synchronized from FortiClient EMS as dynamic address objects on the FortiGate. FortiClient EMS uses zero-trust tagging rules to automatically tag managed endpoints based on various attributes detected by the FortiClient. When the FortiGate establishes a connection with the FortiClient EMS server via the EMS Fabric connector, it pulls zero-trust tags containing device IP and MAC addresses and converts them to read-only dynamic address objects. It also establishes a persistent WebSocket connection to monitor for changes in zero-trust tags, which keeps the device information current. These ZTNA tags can then be used in ZTNA rules, firewall rules, and NAC policies to perform security posture checks. ZTNA tags are displayed in the *Device Inventory* widget, *FortiClient* widget, and *Asset Identity Center* page.

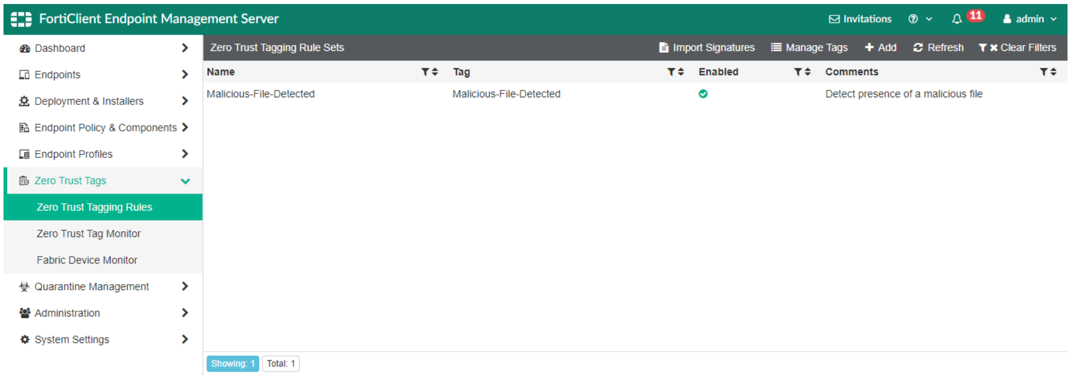
When using WebSocket, EMS pushes notifications to the corresponding FortiGate when there are updates to tags or other monitored attributes. The FortiGate then fetches the updated information using the REST API over TCP/8013. When WebSocket is not used (due to an override or unsupported EMS version), updates are triggered on demand from the FortiGate side over the REST API.

If the WebSocket capability is detected, the capabilities setting will automatically display the WebSocket option. You can use the `diagnose test application fcnacd 2` command to view the status of the WebSocket connection.

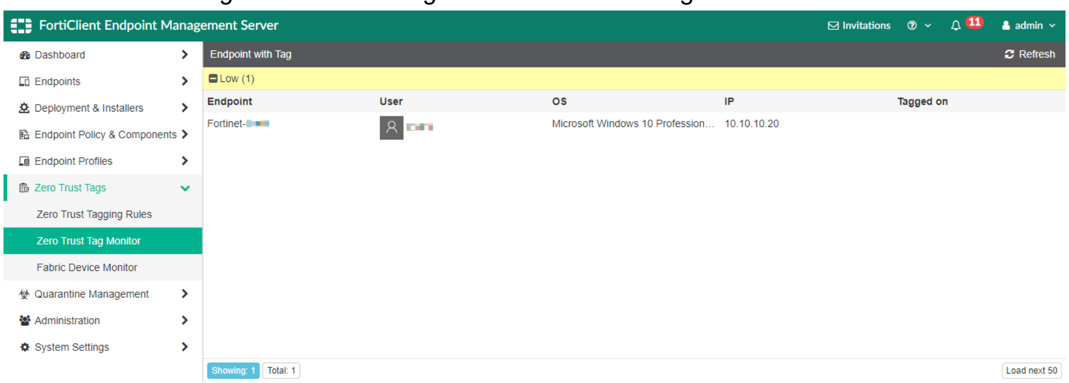
In the following example, the FortiGate connects to and retrieves ZTNA tags from a FortiClient EMS configured with tagging rules. It is assumed that zero-trust tags and rules are already created on the FortiClient EMS. For more information, see the Zero Trust Tags section of the EMS Administration Guide.

To verify zero-trust tags in FortiClient EMS:

1. Go to *Zero Trust Tags > Zero Trust Tagging Rules* to view the tags.

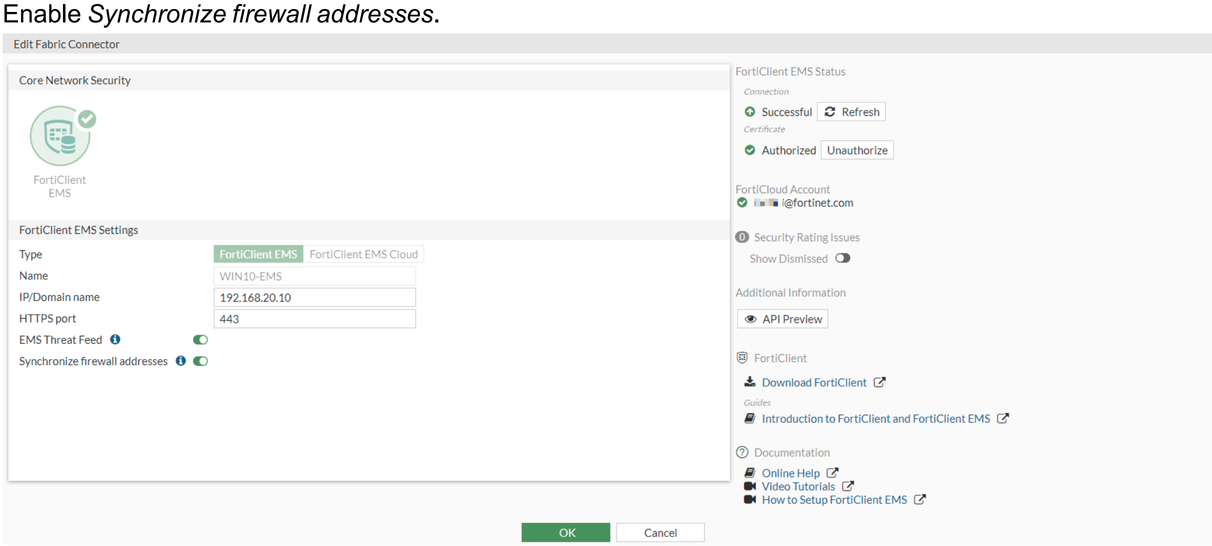


2. Go to *Zero Trust Tags > Zero Trust Tag Monitor* to view the registered users who match the defined tag.

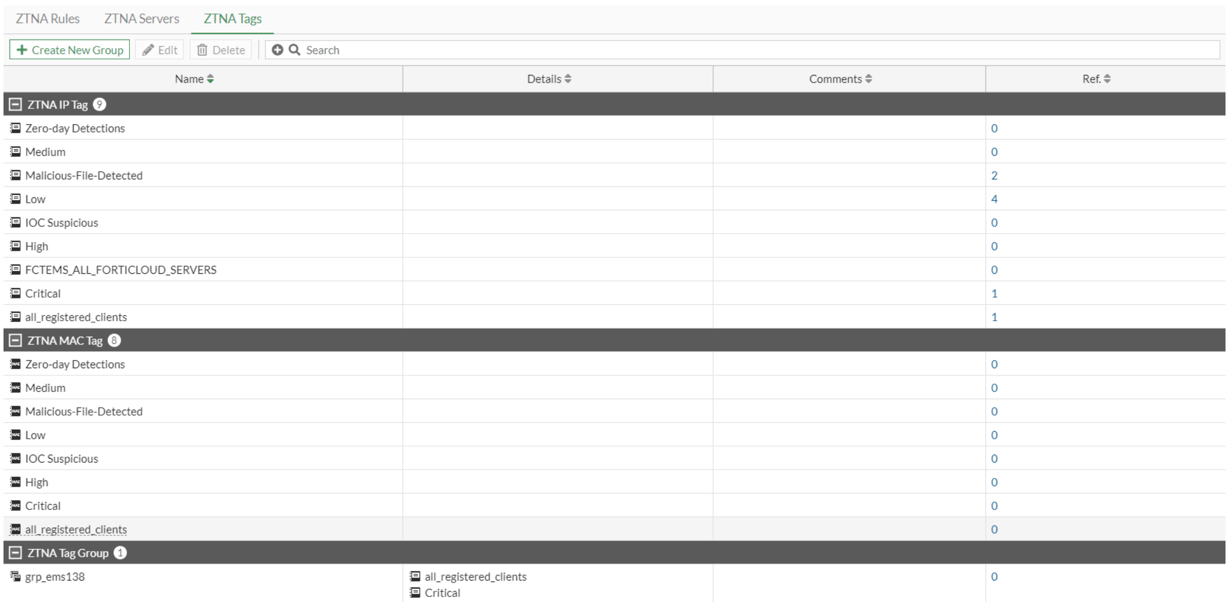


To configure the EMS Fabric connector to synchronize ZTNA tags in the GUI:

1. Configure the EMS Fabric connector:
   a. On the root FortiGate, go to *Security Fabric > Fabric Connectors*.
   b. Click *Create New* and click *FortiClient EMS*.
   c. Enable *Synchronize firewall addresses*.



   d. Configure the other settings as needed and validate the certificate.
   e. Click *OK*.
2. Enable ZTNA:
   a. Go to *System* > *Feature Visibility* and enable *Zero Trust Network Access*.
   b. Click *Apply*.
3. Go to *Policy & Objects* > *ZTNA* and select the *ZTNA Tags* tab. You will see the ZTNA IP and ZTNA MAC tags synchronized from the FortiClient EMS.



To configure the EMS Fabric connector to synchronize ZTNA tags in the CLI:

1. Configure the EMS Fabric connector on the root FortiGate:

```
config endpoint-control fctems
    edit "WIN10-EMS"
        set server "192.168.20.10"
        set https-port 443
        set pull-sysinfo enable
        set pull-vulnerabilities enable
        set pull-avatars enable
        set pull-tags enable
        set pull-malware-hash enable
        set capabilities fabric-auth silent-approval websocket
    next
end
```

2. Verify which IPs the dynamic firewall address resolves to:

```
# diagnose firewall dynamic list
List all dynamic addresses:
FCTEMS0000100000_all_registered_clients: ID(51)
        ADDR(172.17.194.209)
        ADDR(10.10.10.20)
…

FCTEMS0000100000_Low: ID(78)
```

```
        ADDR(172.17.194.209)
        ADDR(10.10.10.20)
…

FCTEMS0000100000_Malicious-File-Detected: ID(190)
        ADDR(172.17.194.209)
        ADDR(10.10.10.20)
…
```

Configuring FortiNAC

A FortiNAC device can be added to the Security Fabric on the root FortiGate. After the device has been added and authorized, you can log in to the FortiNAC from the FortiGate topology views.

Adding a FortiNAC to the Security Fabric requires a FortiNAC with a license issued in the year 2020 or later that includes an additional certificate. The device cannot be added if it has an older license. Use the `licensetool` in the FortiNAC CLI to determine if your license includes the additional certificate.

The FortiNAC tags connector under *Security Fabric* > *Fabric Connectors* has been deprecated. It was replaced with a REST API (in FortiNAC and FortiOS) that is used by FortiNAC to send user logon and logoff information to the FortiGate. The FortiNAC tag dynamic firewall address type is used to store the device IP, FortiNAC firewall tags, and FortiNAC group information sent from FortiNAC by the REST API when user logon and logoff events are registered (see FortiNAC tag dynamic address on page 892 for more information).

For upgrade support, the FSSO FortiNAC user type can still be configured in the CLI.

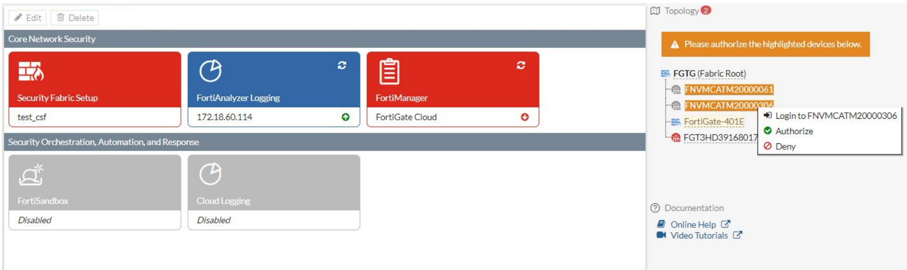
To add a FortiNAC to the Security Fabric:

1. On the FortNAC, configure telemetry and input the IP address of the root FortiGate. See Security Fabric Connection in the *FortiNAC Administration Guide* for more information.
2. On the root FortiGate, authorize the FortiNAC.
3. Verify the connection status in the topology views.

To authorize the FortiNAC on the root FortiGate in the GUI:

1. Go to *Security Fabric* > *Fabric Connectors*.
2. The FortiNAC device will be highlighted in the topology list in the right panel with the status *Waiting for Authorization*.

3. Click on the highlighted FortiNAC and select *Authorize*.



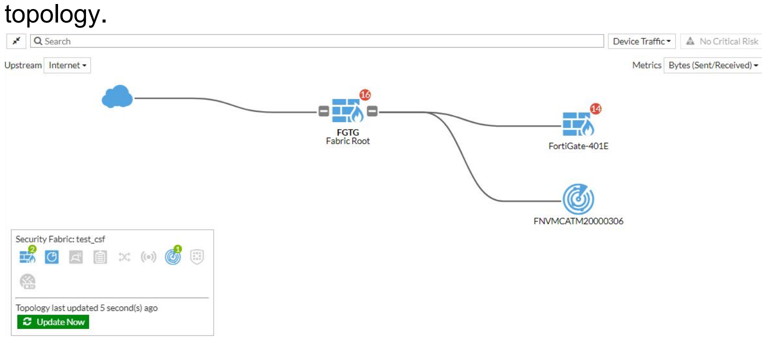
Optionally, you can also deny authorization to the FortiNAC to remove it from the list.

To authorize the FortiNAC on the root FortiGate in the CLI:

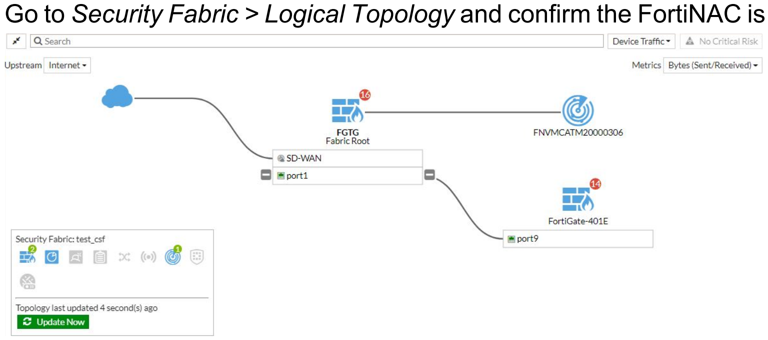
```
config system csf
    config trusted-list
        edit "FNVMCATM20000306"
            set action accept
        next
    end
end
```

To verify the connection status:

1. After the FortiNAC is authorized, go to *Security Fabric > Physical Topology* and confirm that it is included in the topology.



2. Go to *Security Fabric > Logical Topology* and confirm the FortiNAC is also displayed there.



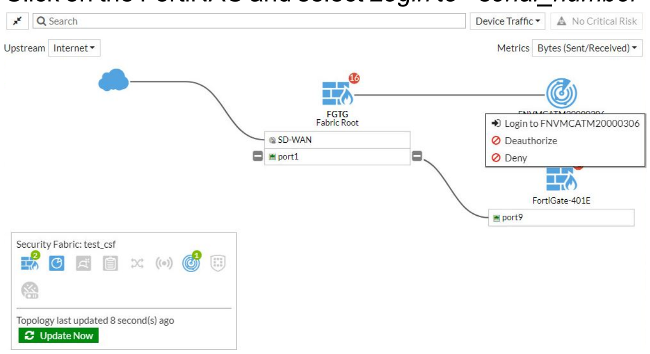
3. Run the following command in the CLI to view information about the FortiNAC device's status:

```
# diagnose sys csf downstream-devices fortinac
{
  "path":"FG5H1E5818900126:FNVMCATM20000306",
  "mgmt_ip_str":"10.1.100.197",
```

```
    "mgmt_port":0,
    "admin_port":8443,
    "serial":"FNVMCATM20000306",
    "host_name":"adnac",
    "device_type":"fortinac",
    "upstream_intf":"port2",
    "upstream_serial":"FG5H1E5818900126",
    "is_discovered":true,
    "ip_str":"10.1.100.197",
    "downstream_intf":"eth0",
    "authorizer":"FG5H1E5818900126",
    "idx":1
  }
```

To log in to the FortiNAC from the FortiGate:

1. On the FortiGate, go to *Security Fabric > Physical Topology* or *Security Fabric > Logical Topology*.
2. Click on the FortiNAC and select *Login to <serial_number>*.



   A new tab will open to the FortiNAC log in page.
3. Enter the username and password to log in to the FortiNAC.

Configuring FortiAP and FortiSwitch

FortiAP and FortiSwitch devices can be authorized in the Security Fabric with one click. After connecting a FortiAP or FortiSwitch device to an authorized FortiGate, it will automatically be listed in the topology tree.

If the default `auto-auth-extension-device` settings on the FortiAP or FortiSwitch have been modified, manual authorization in the Security Fabric may not be required.

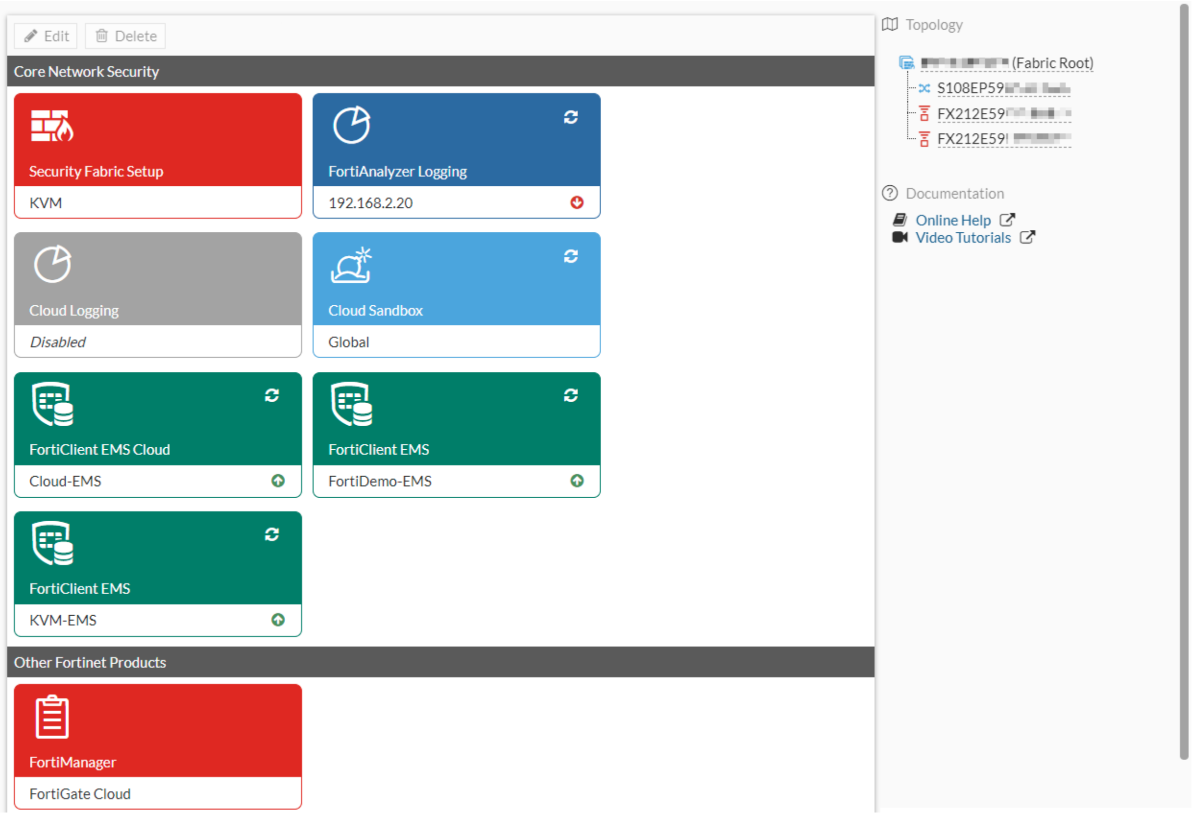
For more information about configuring FortiAPs, see Configuring the FortiGate interface to manage FortiAP units and Discovering, authorizing, and deauthorizing FortiAP units.

For more information about configuring FortiSwitches, see Using the FortiGate GUI.

To authorize FortiAP and FortiSwitch devices:

1. Connect the FortiAP or FortiSwitch device to a FortiGate.
2. On the root FortiGate, go to *Security Fabric > Fabric Connectors*. The new device is shown in the *Topology* tree.

3. Click the device and select *Authorize*.



Configuring FortiMail

FortiMail can be authorized into the Security Fabric using either the gutter on the *Fabric Connectors* page, or by pre-authorizing using the FortiMail serial number or certificate.

To join the Security Fabric from FortiMail:

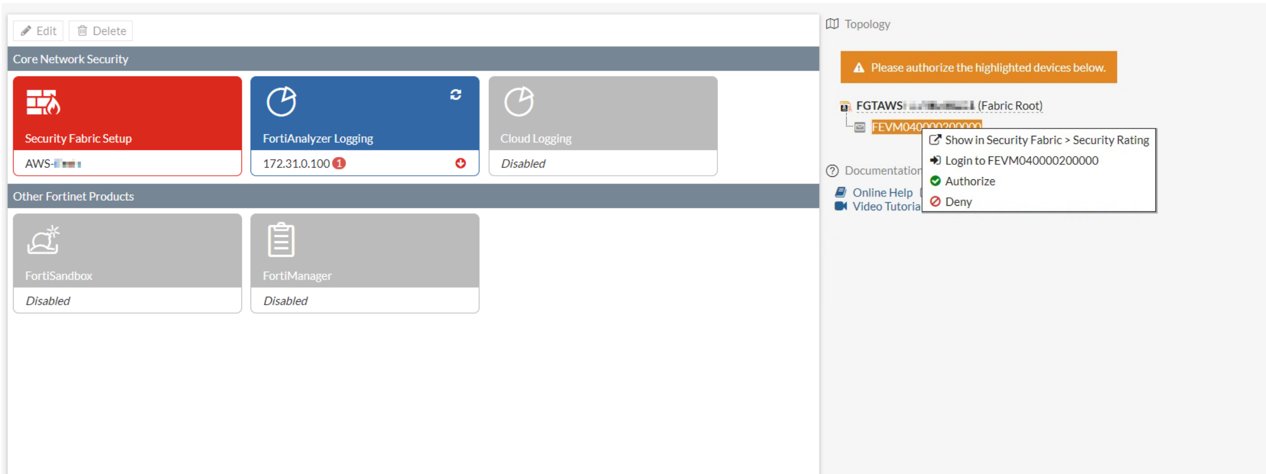
1. Go to *System* > *Customization* and click the *Corporate Security Fabric* tab (or the *Corporate Security Fabric* tab in FortiMail 6.4.2 and earlier).
2. Click the toggle to enable the Fabric.
3. Enter the *Upstream IP Address* (root FortiGate) and the *Management IP* of the FortiMail.
4. Click *Apply*.

Authorizing using FortiOS

If the FortiMail was added to the Security Fabric but not pre-authorized, you can authorize it in FortiOS on the *Fabric Connectors* page.

To authorize FortiMail:

1. Go to *Security Fabric > Fabric Connectors*.
2. In the topology tree, hover over the FortiMail and click *Authorize*.



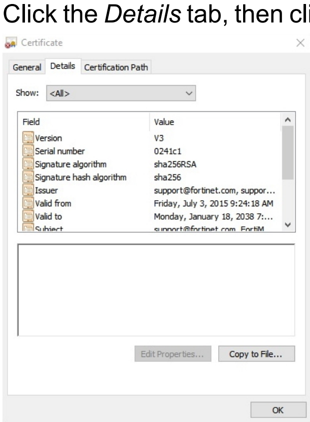
3. Verify the certificate is correct, then click *Accept*.

Pre-authorizing using the FortiMail certificate

FortiMail can be pre-authorized using its serial number or certificate. When you pre-authorize, the FortiMail can join at any time, and you will not need to authorize it FortiOS. In this example, FortiMail is pre-authorized using a certificate.

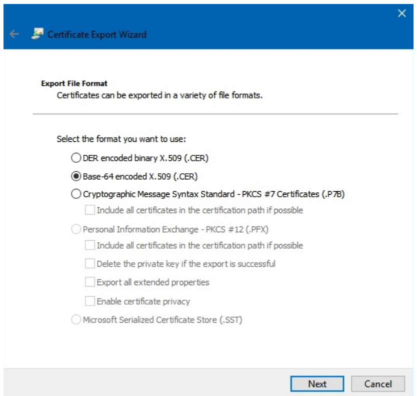
To pre-authorize FortiMail using a third-party or default certificate:

1. Log in to FortiMail.
2. Download the certificate. For example, in Chrome:
   a. In the left side of the address bar, click the icon to view the site information.
   b. Click *Certificate*.
   c. Click the *Details* tab, then click *Copy to File*.



   d. The *Certificate Export Wizard* opens. Click *Next* to continue.

e. For the file format, select *Base-64 encoded X.509 (.CER)*, then click *Next*.



f. Browse to the folder location and enter a file name, then click *Next*.

g. Click *Finish*, then click *OK* to close the dialog box.

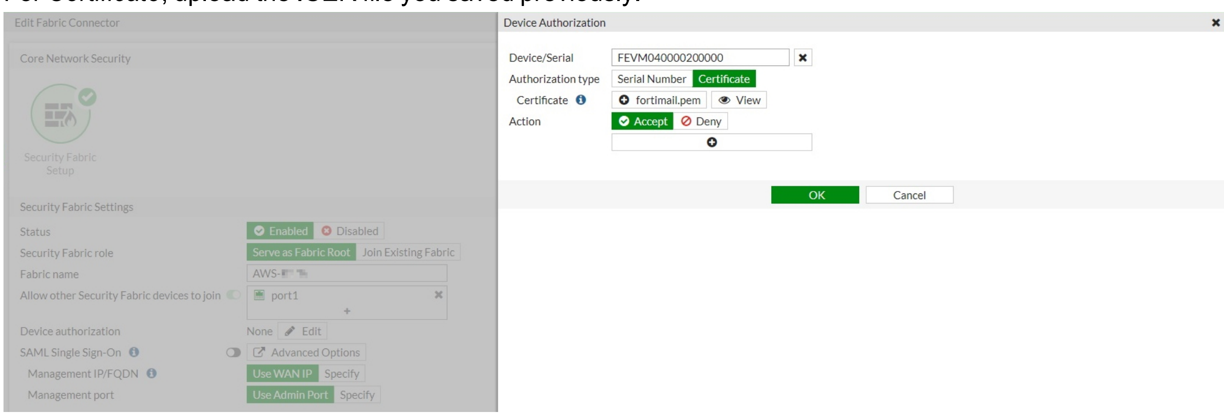
3. In FortiOS, go to *Security Fabric > Fabric Connectors* and double-click the *Security Fabric Setup* card.

4. Beside *Device authorization*, click *Edit* and configure the following:

a. Enter the FortiMail serial number.

b. For *Authorization type*, select *Serial Number*.

c. For *Certificate*, upload the .CER file you saved previously.



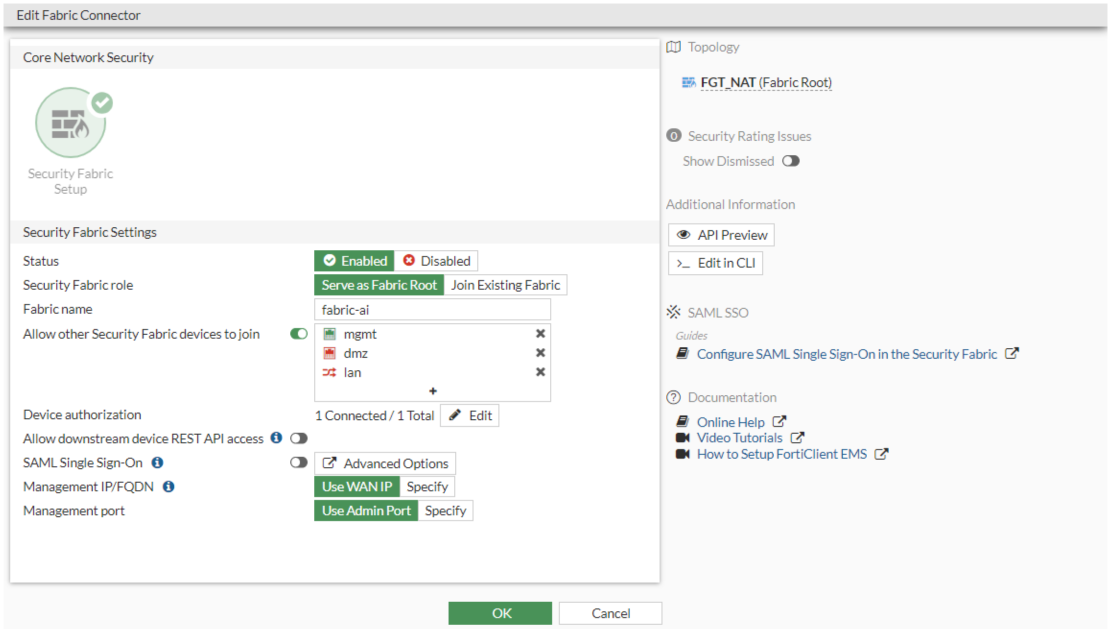
d. Click *OK*.

Configuring FortiNDR

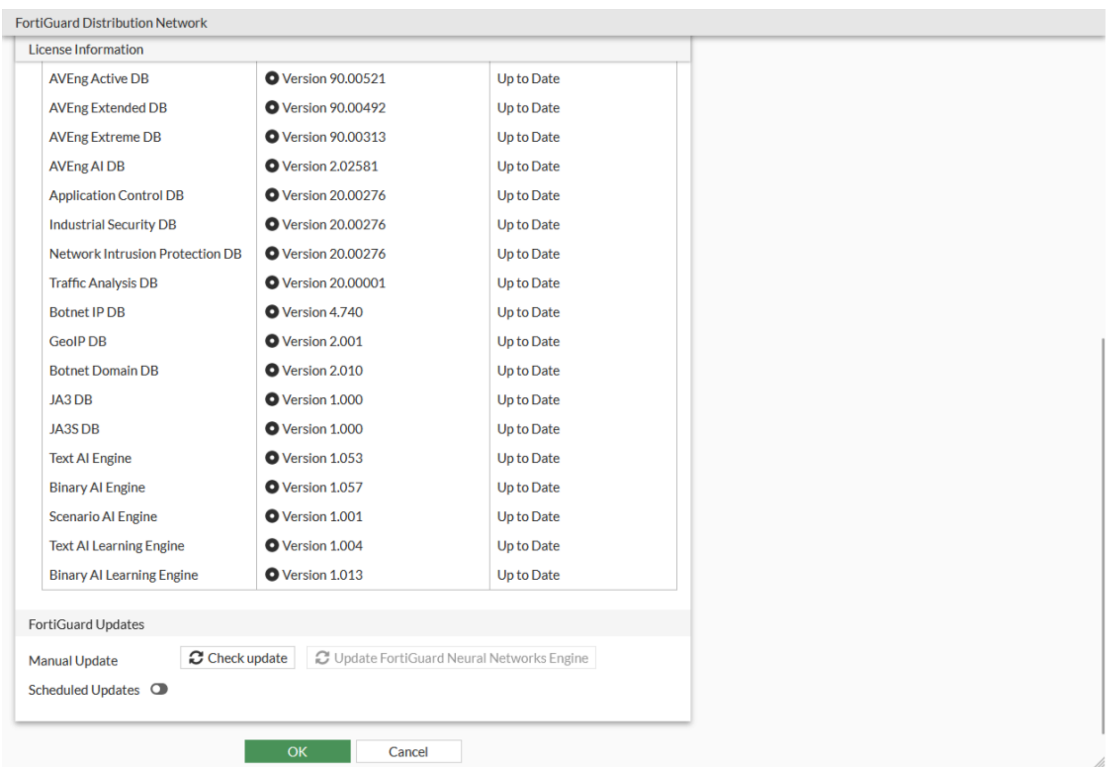
FortiNDR (formerly FortiAI) can be added to the Security Fabric so it appears in the topology views and the dashboard widgets.

To add FortiNDR to the Security Fabric in the GUI:

1. Enable the Security Fabric and configure the interface to allow other Security Fabric devices to join (see Configuring the root FortiGate and downstream FortiGates on page 2208).

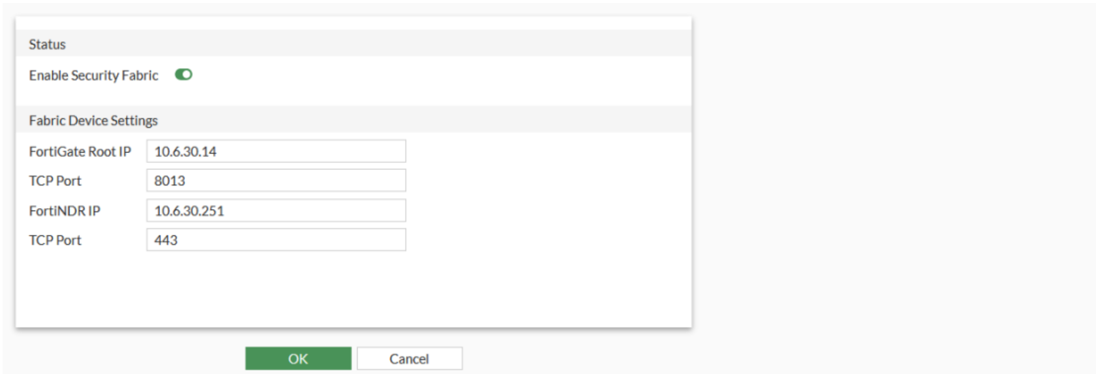


2. Install the FortiNDR appliance and activate the product with a valid license (see Registering products in the Asset Management Guide). A license file is provided after the product is registered.
3. In FortiNDR, go to *System* > *FortiGuard* and verify that the pre-trained models (engines) are up to date. Refer to the FortiGuard website for the latest FortiNDR ANN versions.



4. Configure and authorize the FortiGate in the FortiNDR GUI to join the Security Fabric:
   a. Go to *Security Fabric* > *Fabric Connectors* and double-click the *Security Fabric* card.
   b. Click the toggle to *Enable Security Fabric*.

c. Enter the IP addresses for the root FortiGate and the FortiNDR.

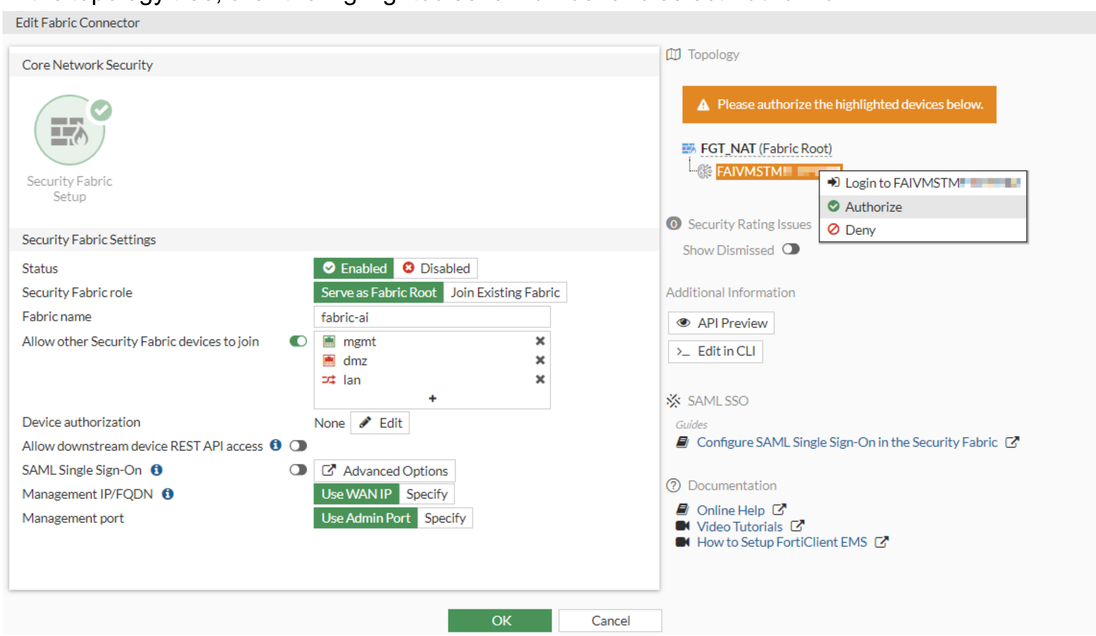


d. Click *OK*. The FortiNDR is now authorized.

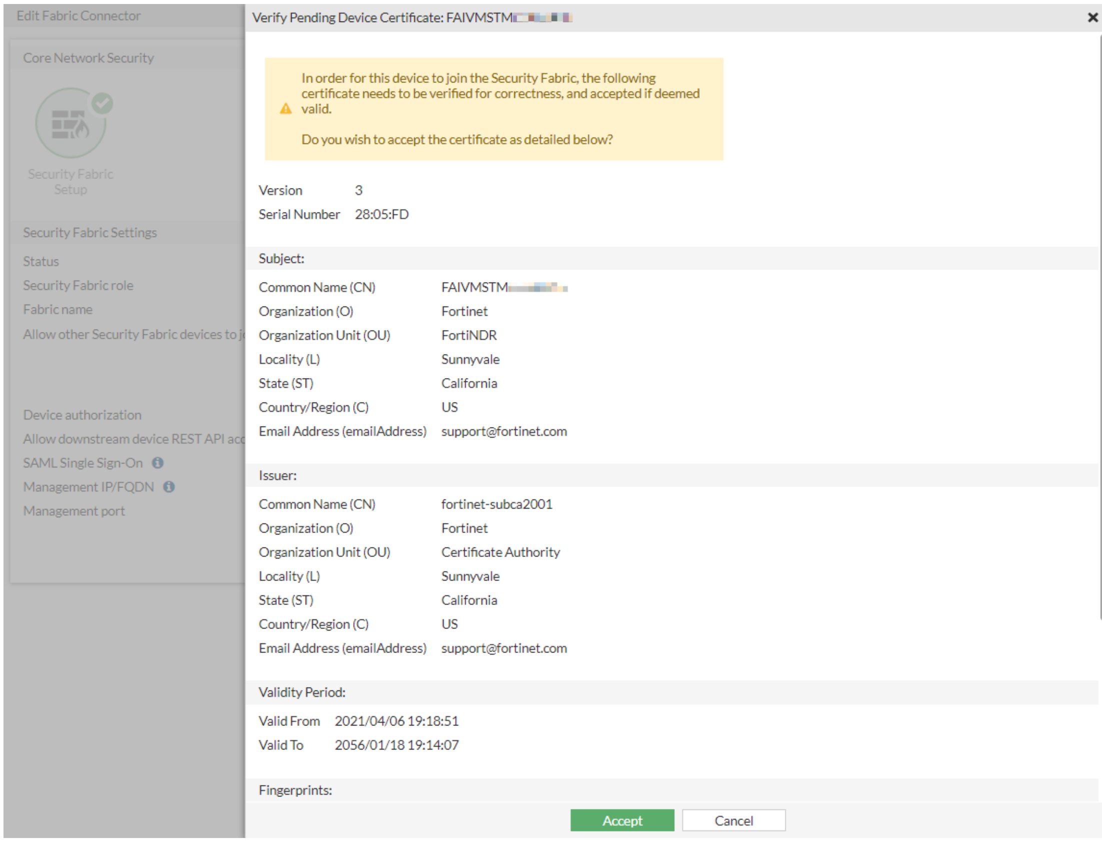
5. Authorize the FortiNDR in FortiOS:

a. Go to *Security Fabric > Fabric Connectors* and double-click the *Security Fabric Setup* card.

b. In the topology tree, click the highlighted serial number and select *Authorize*.



c. Click *Accept* to verify the device certificate.



The *Security Fabric* widget on the dashboard also updates when the FortiNDR is authorized.

6. Go to *Security Fabric* > *Physical Topology* or *Security Fabric* > *Logical Topology* to view more information.

To add FortiNDR to the Security Fabric in the CLI:

1. Configure the interface to allow other Security Fabric devices to join:

```
config system interface
    edit "port1"
        ...
        set allowaccess ping https ssh http fgfm fabric
        ...
    next
end
```

2. Enable the Security Fabric:

```
config system csf
    set status enable
    set group-name "fabric-ai"
end
```

3. In FortiNDR, configure the device to join the Security Fabric:

```
config system csf
    set status enable
    set upstream-ip 10.6.30.14
```

```
    set managment-ip 10.6.30.251
end
```

4. Authorize the FortiNDR in FortiOS:

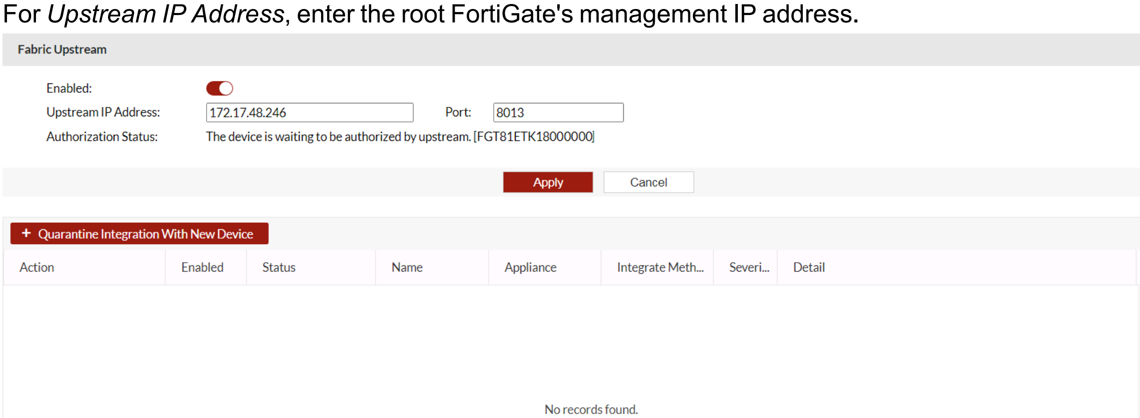
```
config system csf
    set status enable
    set group-name "fabric-ai"
    config trusted-list
        edit "FAIVMSTM21000000"
            set authorization-type certificate
            set certificate "********************"
        next
    end
end
```

Configuring FortiDeceptor

FortiDeceptor can be added to the Security Fabric so it appears in the topology views and the dashboard widgets.

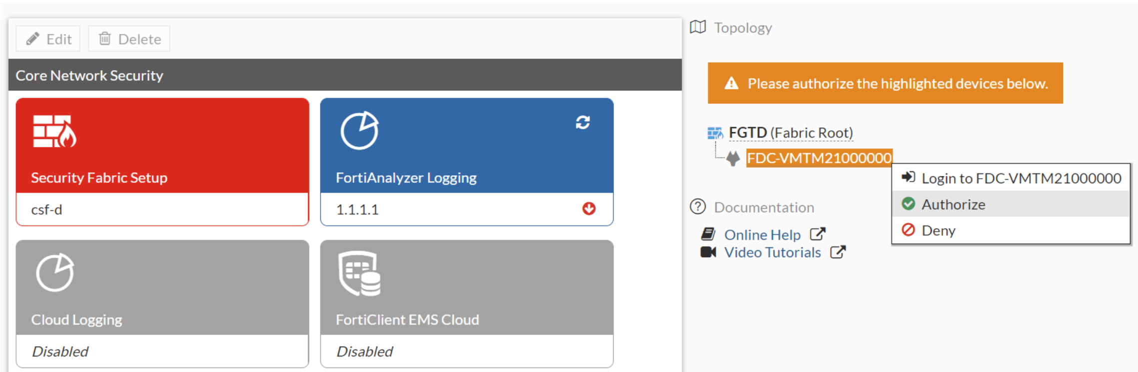
To add FortiDeceptor to the Security Fabric in the GUI:

1. Enable the Security Fabric (see Configuring the root FortiGate and downstream FortiGates on page 2208 for more details) with the following settings:
   a. Configure the interface to allow other Security Fabric devices to join.
   b. Enable *Allow downstream device REST API access* so the FortiDeceptor can communicate with the FortiGate, and select an *Administrator profile*. The minimum permission required for the selected *Administrator profile* is *Read/Write* for *User & Device* (`set authgrp read-write`).
2. In FortiDeceptor, integrate the device:
   a. Go to *Fabric > Integration Devices*.
   b. Click *Quarantine Integration With New Device*.
   c. Click the toggle to enable the device.
   d. For *Upstream IP Address*, enter the root FortiGate's management IP address.

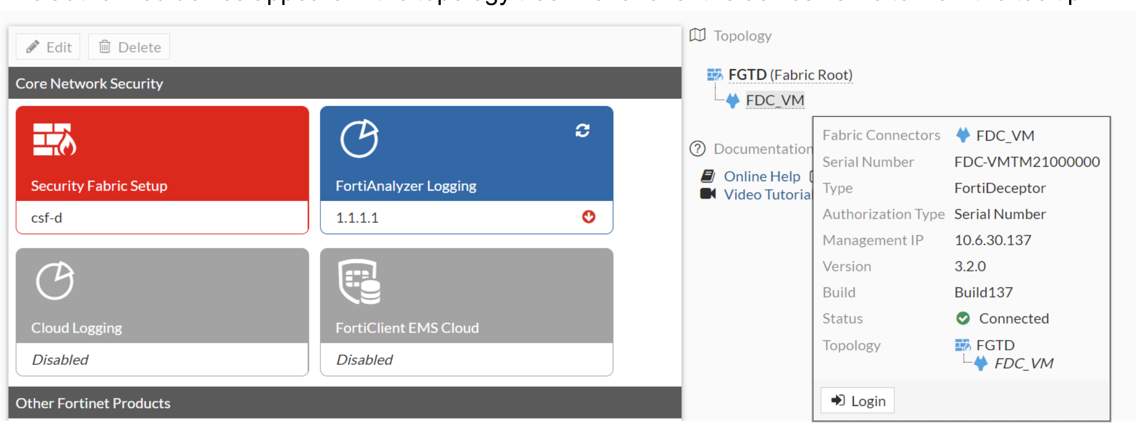


   e. Click *Apply*.

3. Authorize the FortiDeceptor in FortiOS:
   a. Go to *Security Fabric > Fabric Connectors*.
   b. In the topology tree, click the highlighted FortiDeceptor serial number and select *Authorize*.

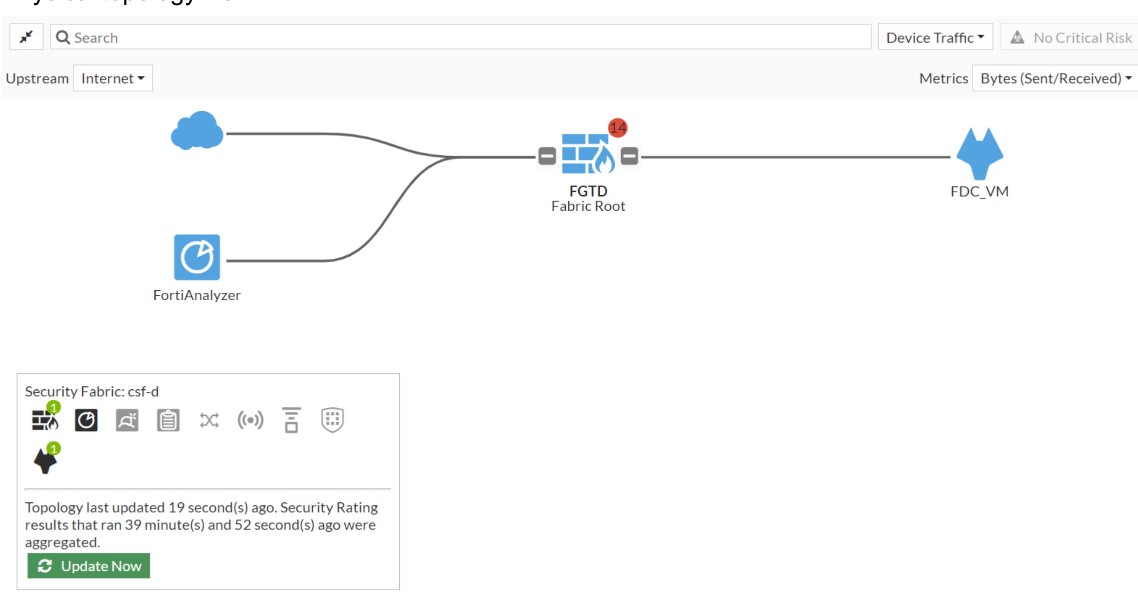


The authorized device appears in the topology tree. Hover over the device name to view the tooltip.

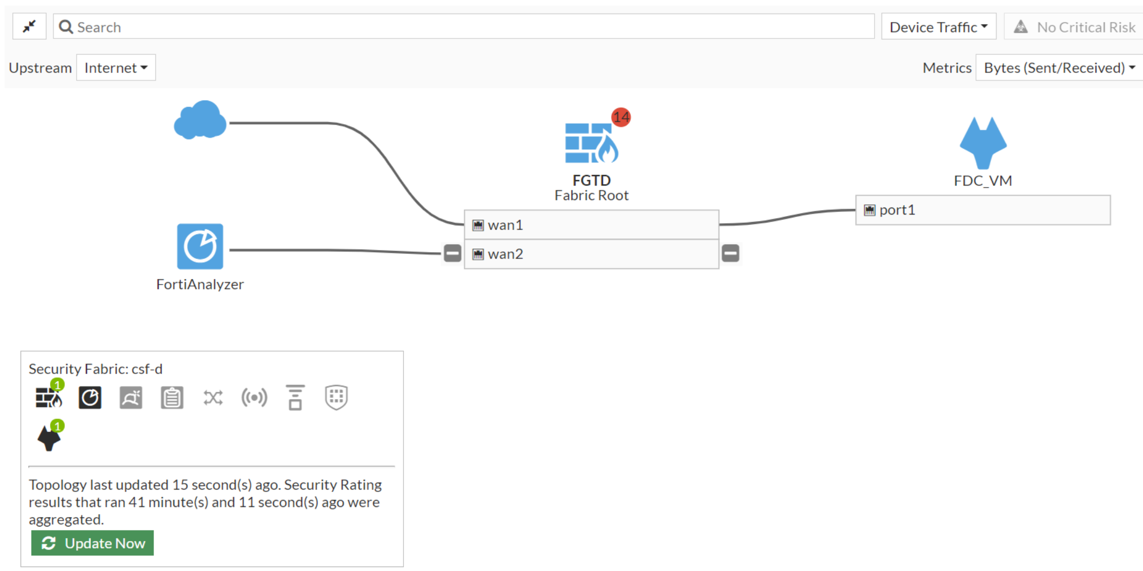


The *Security Fabric* widget on the dashboard also updates when the FortiDeceptor is authorized.

4. Go to *Security Fabric > Physical Topology* or *Security Fabric > Logical Topology* to view more information.
   Physical topology view:



Logical topology view:



To add FortiDeceptor to the Security Fabric in the CLI:

1. Configure the interface to allow other Security Fabric devices to join:

```
config system interface
    edit "wan1"
        ...
        set allowaccess ping https ssh snmp http fabric
        ...
    next
end
```

2. Enable the Security Fabric:

```
config system csf
    set status enable
    set group-name "csf-d"
    set downstream-access enable
    set downstream-accprofile "super_admin"
end
```

3. In FortiDeceptor, integrate the device:
   a. Go to *Fabric > Integration Devices*.
   b. Click *Quarantine Integration With New Device*.
   c. Click the toggle to enable the device.
   d. For *Upstream IP Address*, enter the root FortiGate's management IP address.
   e. Click *Apply*.
4. Authorize the FortiDeceptor in FortiOS:

```
config system csf
    set status enable
    set group-name "csf-d"
    config trusted-list
        edit "FDC-VMTM21000000"
            set serial "FDC-VMTM21000000"
        next
```

```
    end
end
```

Configuring FortiWeb

A FortiWeb can be configured to join a Security Fabric through the root or downstream FortiGate. There are two methods to add a FortiWeb to the Security Fabric:

- Trigger the authorization on the FortiWeb side and authorize from the FortiOS side.
- Pre-authorize the FortiWeb from the FortiOS side.

Once the FortiWeb joins the Fabric, the following features are available:

- View the FortiWeb on topology pages.
- Create a dashboard Fabric Device widget to view FortiWeb data.
- Configure single sign-on using SAML.

Triggering the authorization in FortiWeb

In this example, a FortiWeb triggers the authorization process, and then the device is approved in FortiOS. This is example assumes the Security Fabric has already been configured.

To trigger the Security Fabric authorization in FortiWeb:

1. Edit the FortiGate *Fabric Connector* settings in FortiWeb (see Fabric Connector: Single Sign On with FortiGate). The *Connection Status* is currently *Authorize pending*.
2. In FortiOS, go to *Security Fabric > Fabric Connectors*.
3. In the topology tree, hover over the FortiWeb and click *Authorize*.
4. Verify that the certificate is correct, then click *Accept*.
5. In FortiWeb, verify that the FortiGate *Connection Status* is now *Authorized*.

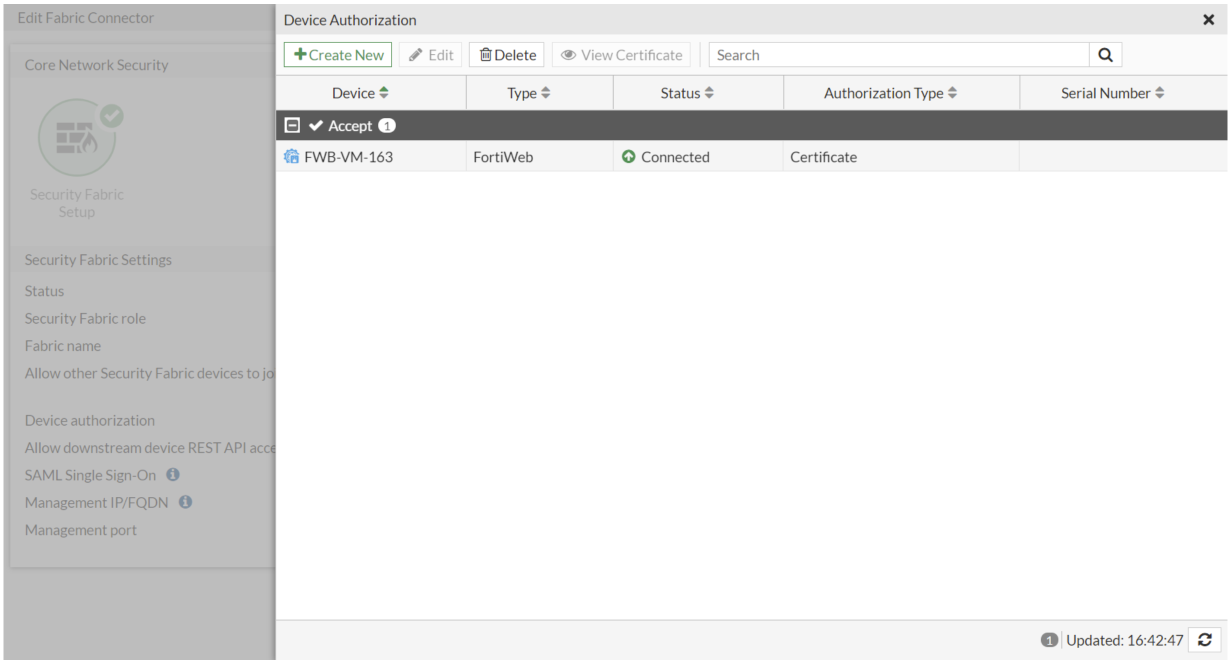
Pre-authorizing the FortiWeb in FortiOS

In this example, a FortiWeb is pre-authorized on the root FortiGate using certificate authorization. This is example assumes the Security Fabric has already been configured.

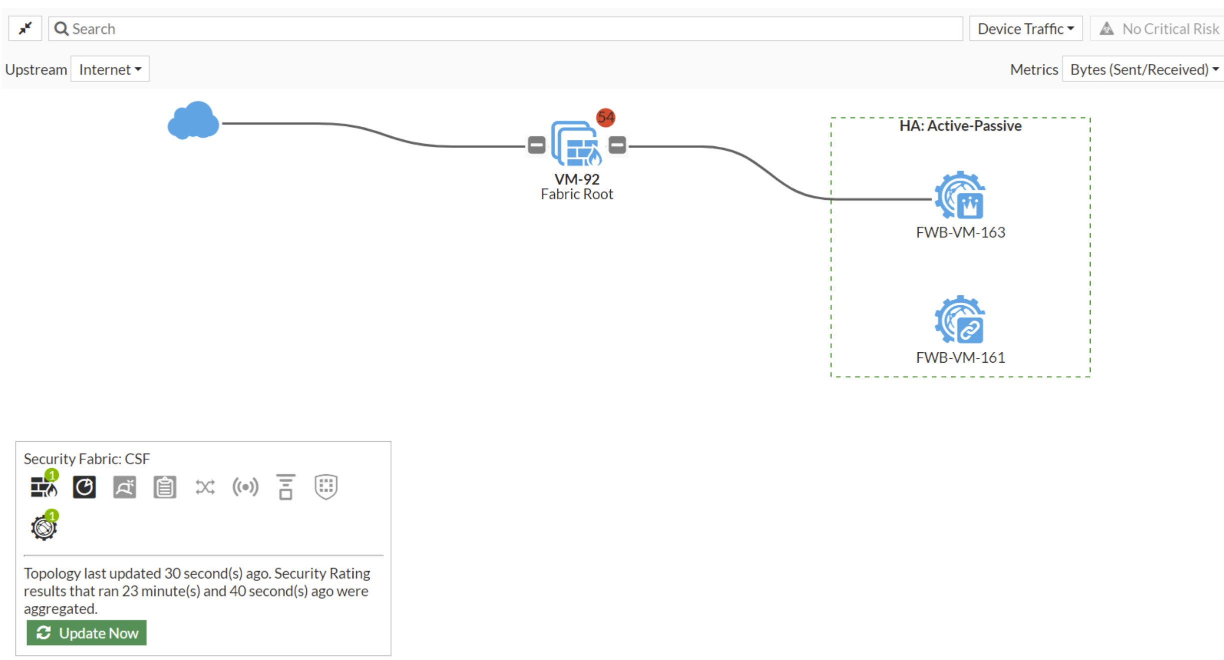
To authorize a FortiWeb to join the Security Fabric in FortiOS:

1. Go to *Security Fabric > Fabric Connectors* and double-click the *Security Fabric Setup* card.
2. Beside *Device authorization*, click *Edit*. The *Device authorization* pane opens.
3. Add the FortiWeb:
   a. Click *Create New* and enter a device name.
   b. For *Authorization type*, select *Certificate*.
   c. Click *Browse* to upload the certificate.
   d. For *Action*, select *Accept*.

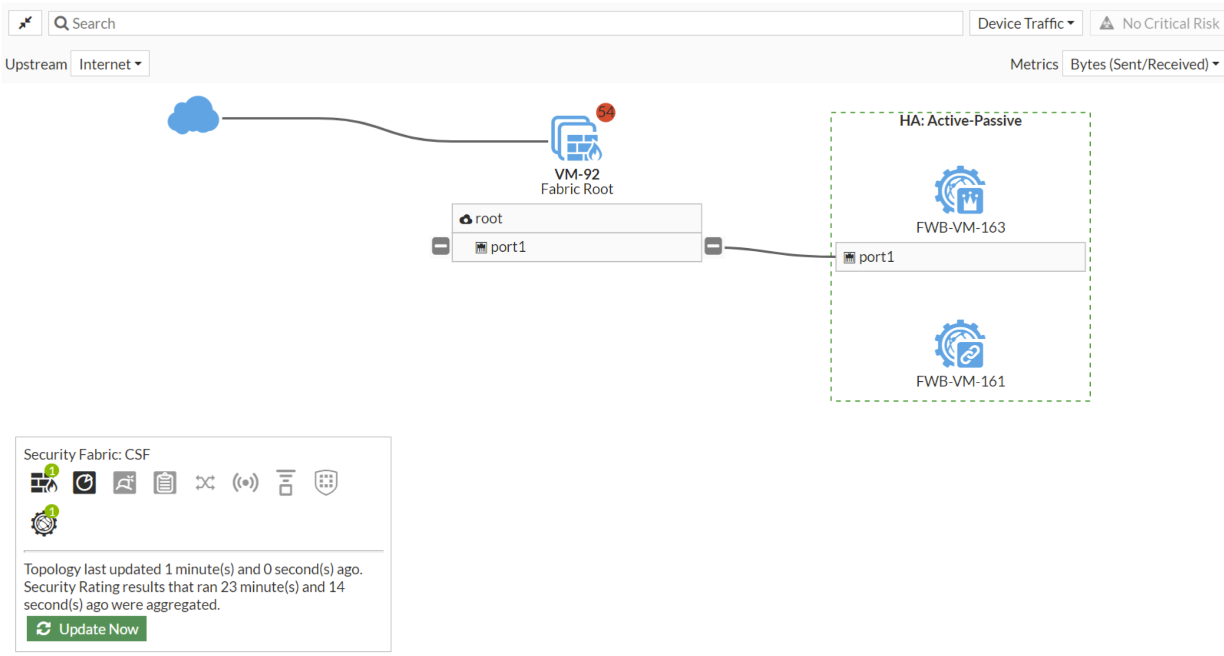
e. Click *OK*. The FortiWeb appears in the table.



4. Go to *Security Fabric > Physical Topology* or *Security Fabric > Logical Topology* to view more information.
Physical topology view:



Logical topology view:

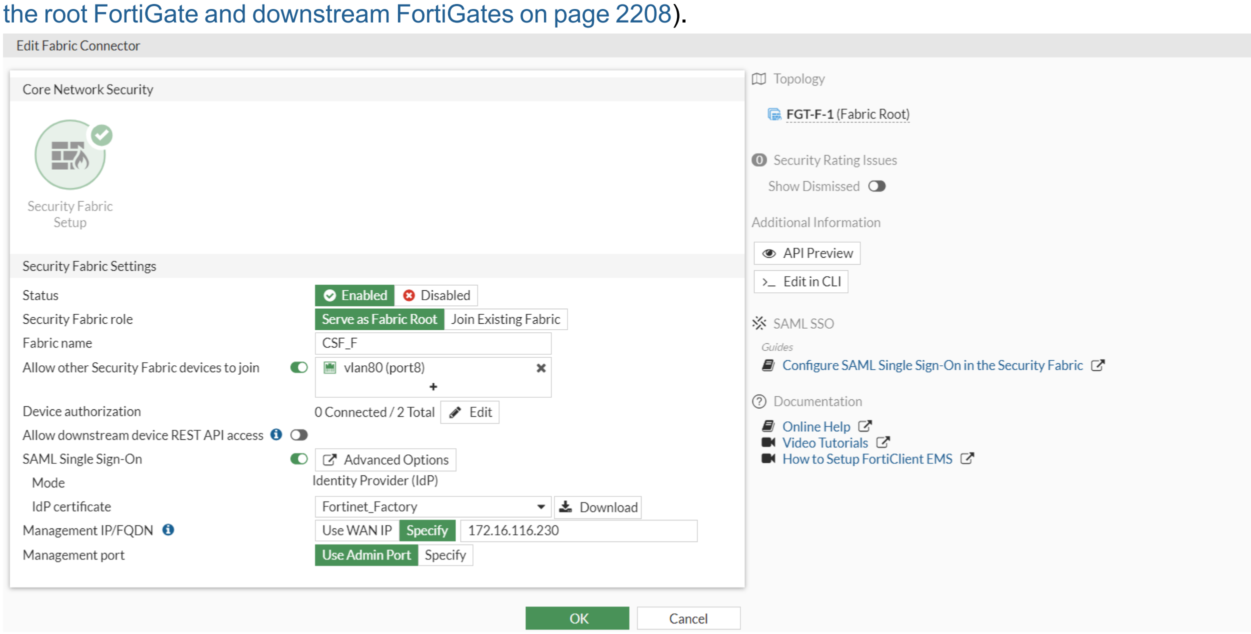


Configuring FortiTester

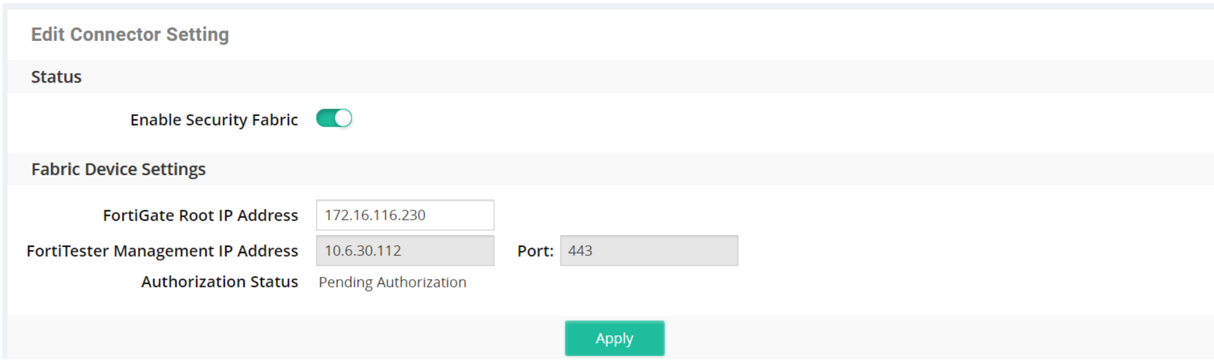
FortiTester can be added to the Security Fabric and authorized from the Security Fabric topology views. Once added, the FortiTester will appear in the *Security Fabric* widget on the dashboard. A FortiTester can be added to the dashboard as a Fabric device widget.

To add FortiTester to the Security Fabric in the GUI:

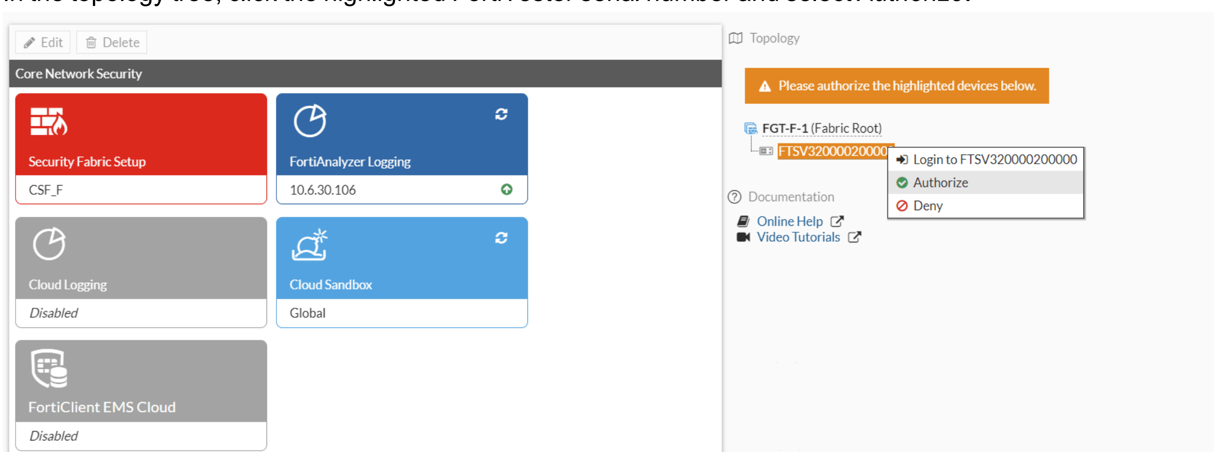
1. Enable the Security Fabric and configure the interface to allow other Security Fabric devices to join (see Configuring the root FortiGate and downstream FortiGates on page 2208).



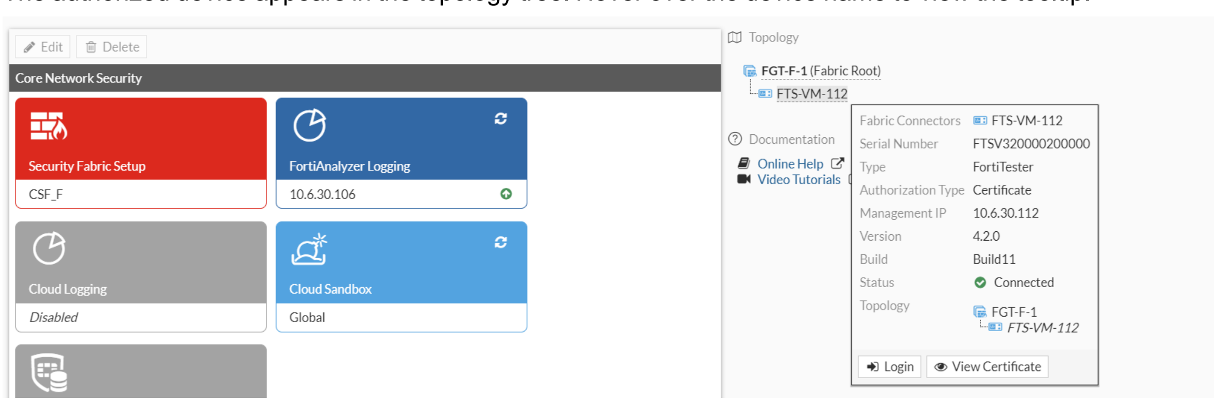
2. In FortiTester, enable the Security Fabric:
   a. Go to *System Settings > Security Fabric > Settings*.
   b. Click the toggle to enable the device (*Enable Security Fabric*).
   c. Enter the *FortiGate Root IP Address*.



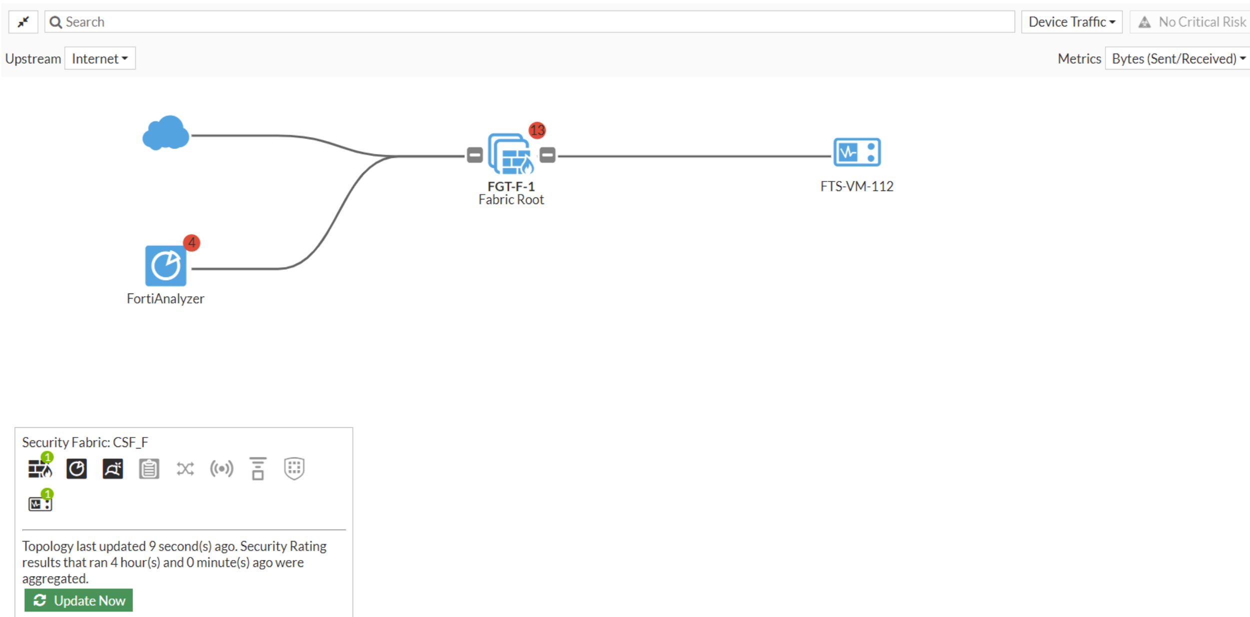
   d. Click *Apply*.
3. Authorize the FortiTester in FortiOS:
   a. Go to *Security Fabric > Fabric Connectors*.
   b. In the topology tree, click the highlighted FortiTester serial number and select *Authorize*.



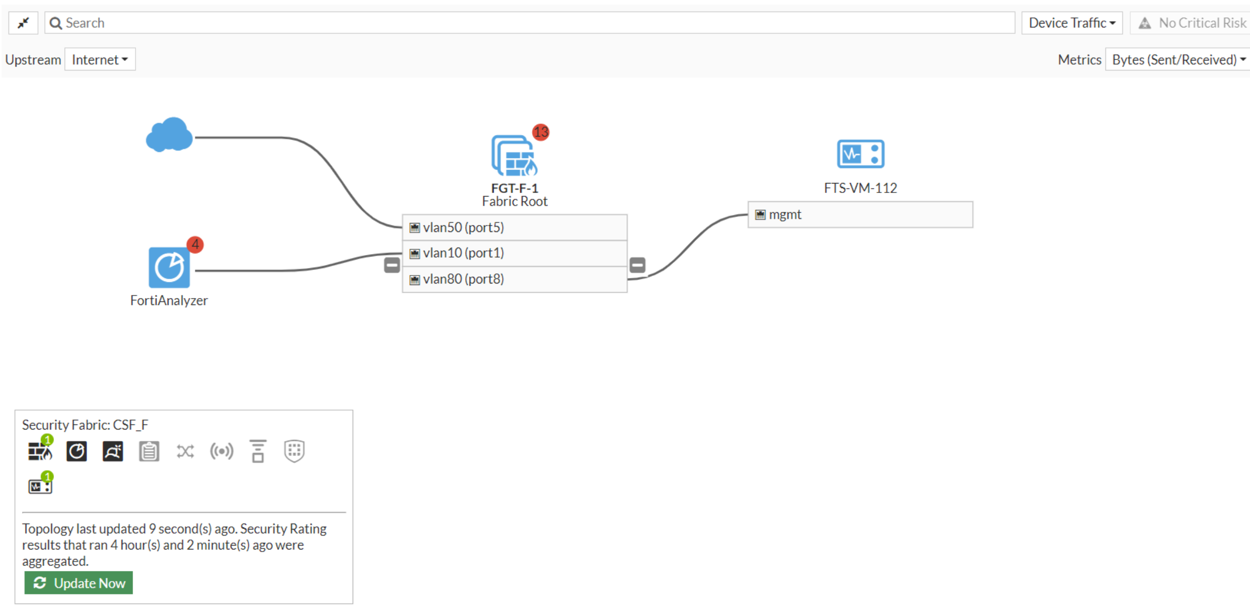
   The authorized device appears in the topology tree. Hover over the device name to view the tooltip.



   The *Security Fabric* widget on the dashboard also updates when the FortiTester is authorized.
4. Go to *Security Fabric > Physical Topology* or *Security Fabric > Logical Topology* to view more information.
   Physical topology view:



Logical topology view:



To add FortiTester to the Security Fabric in the CLI:

1. Configure the interface to allow other Security Fabric devices to join:

```
config system interface
    edit "port8"
        ...
        set allowaccess ping https ssh http fgfm fabric
        ...
    next
end
```

2. Enable the Security Fabric:

```
config system csf
    set status enable
    set group-name "CSF_F"
end
```

3. In FortiTester, enable the Security Fabric:

```
config system csf
    set ip 172.16.116.230
    set port 8013
    set status enable
end
```

4. Authorize the FortiTester in FortiOS:

```
config system csf
    set status enable
    set group-name "CSF_F"
    config trusted-list
        edit "FTSV320000200000"
            set authorization-type certificate
            set certificate "*******************"
        next
    end
end
```

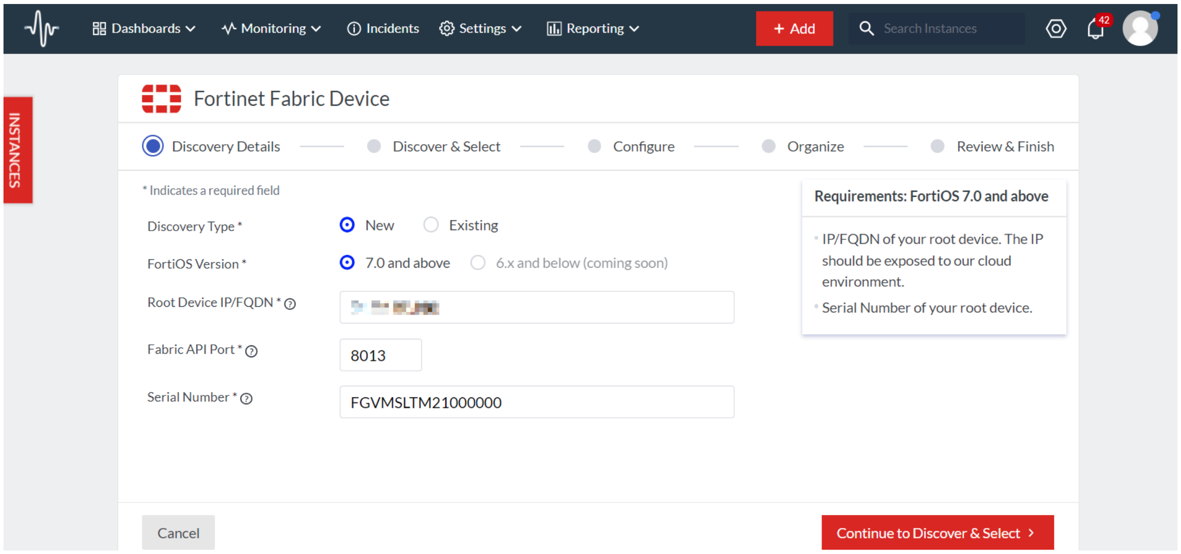
Configuring FortiMonitor

FortiMonitor can be added to the Security Fabric. When a FortiMonitor joins the Security Fabric and is authorized, it appears in the Fabric topology pages.

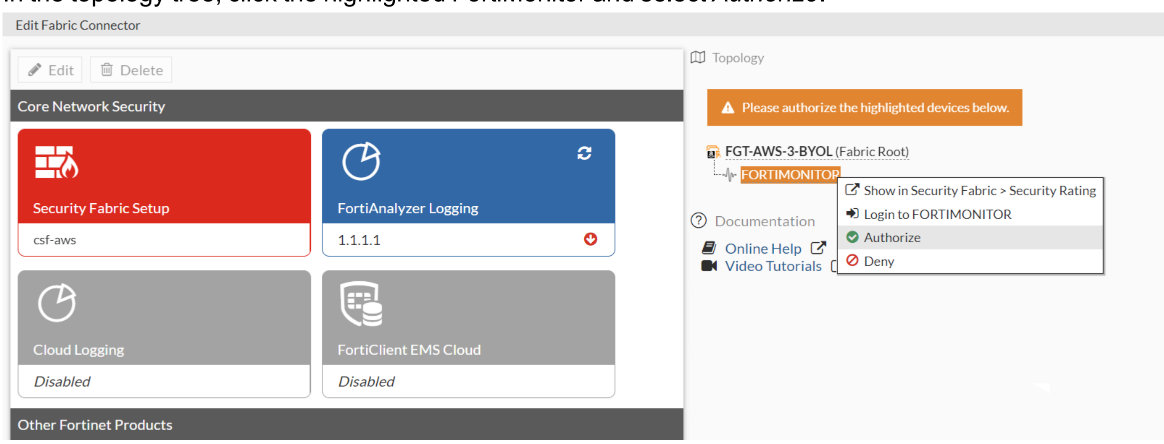
To add FortiMonitor to the Security Fabric:

1. Enable the Security Fabric (see Configuring the root FortiGate and downstream FortiGates on page 2208) with the following settings:
   a. Configure the interface to allow other Security Fabric devices to join.
   b. Enable *Allow downstream device REST API access* and select an *Administrator profile*.
2. In FortiMonitor, start configuring the device to join the Security Fabric (see Enable Security Fabric monitoring for detailed instructions):

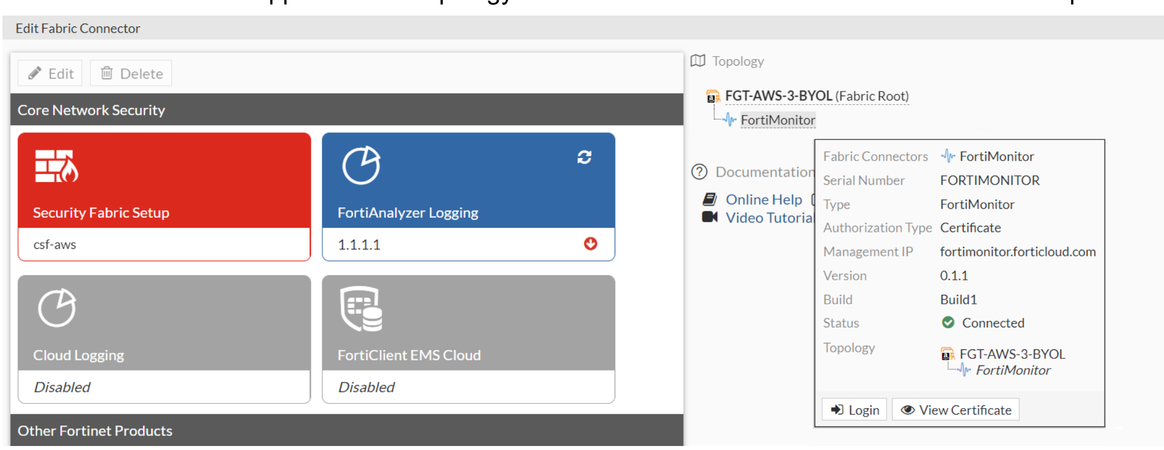
a. Complete the *Discovery Details* page.



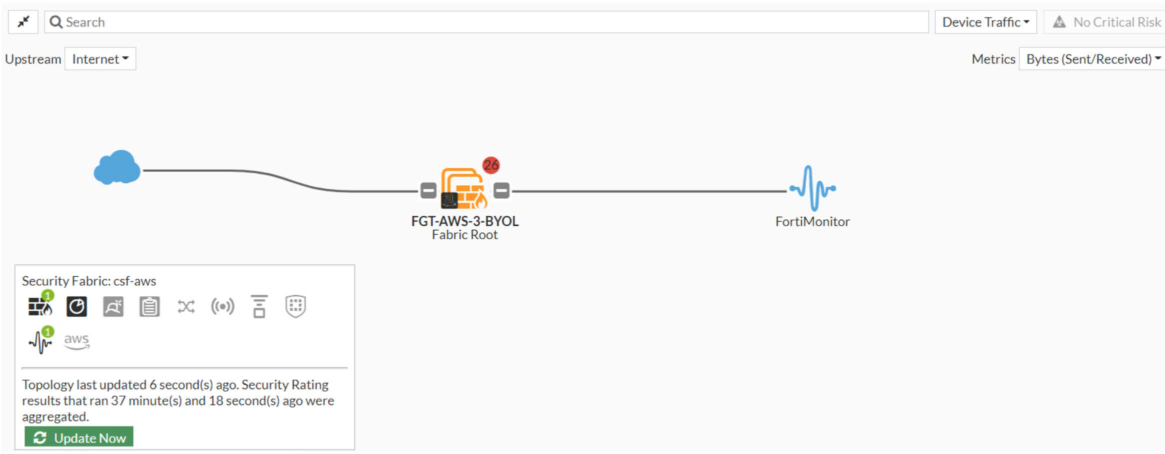
3. Authorize the FortiMonitor in FortiOS:
   a. Go to *Security Fabric > Fabric Connectors*.
   b. In the topology tree, click the highlighted FortiMonitor and select *Authorize*.



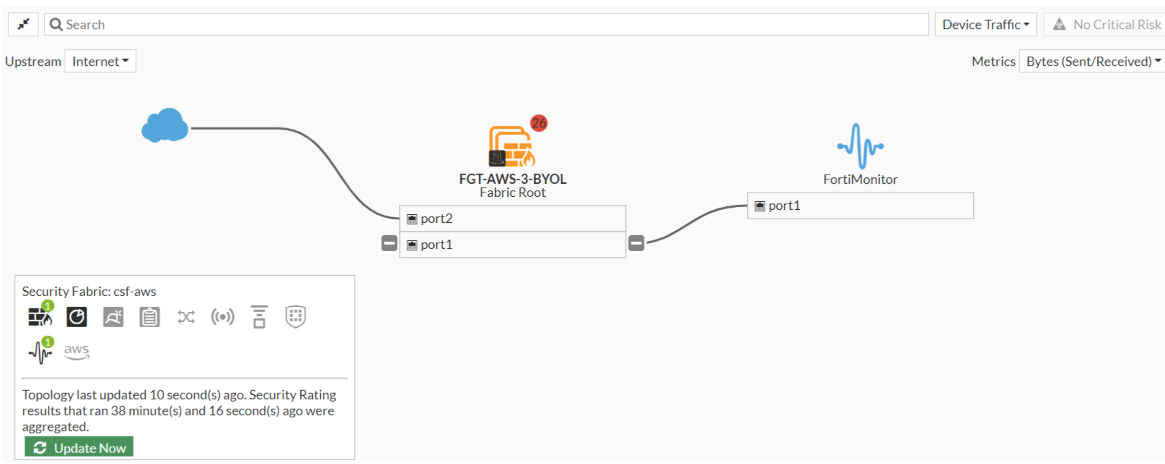
The authorized device appears in the topology tree. Hover over the device name to view the tooltip.



4. Go to *Security Fabric > Physical Topology* or *Security Fabric > Logical Topology* to view more information.
   Physical topology view:



Logical topology view:



5. In FortiMonitor, complete the device configuration (see Enable Security Fabric monitoring for detailed instructions).

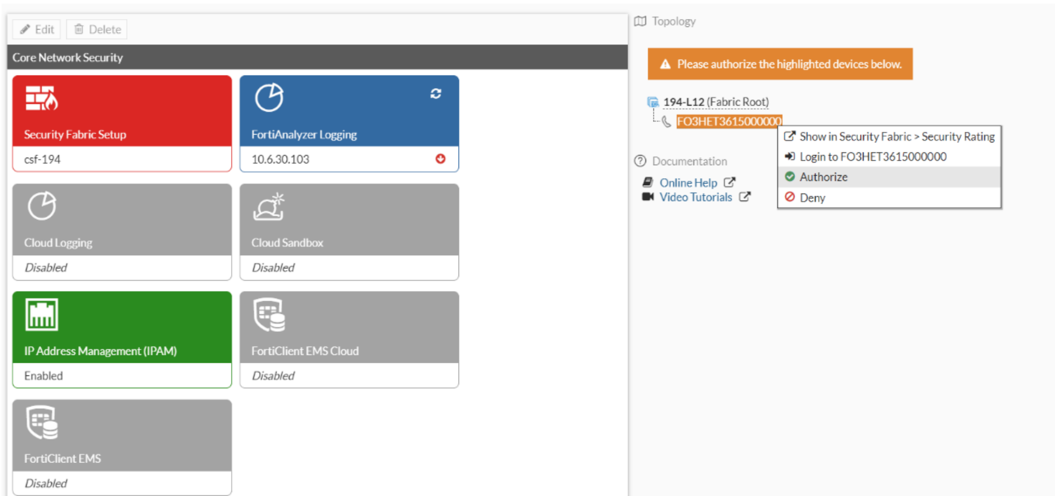
Configuring FortiVoice

A FortiVoice can be added to the Security Fabric on the root FortiGate. Once the FortiVoice is added and authorized, you can log in to the device from the Security Fabric topology pages or the topology tree. A FortiVoice can be authorized in FortiOS, or can be pre-authroized with its serial number or certificate. A FortiVoice can be added to the dashboard as a Fabric device widget.

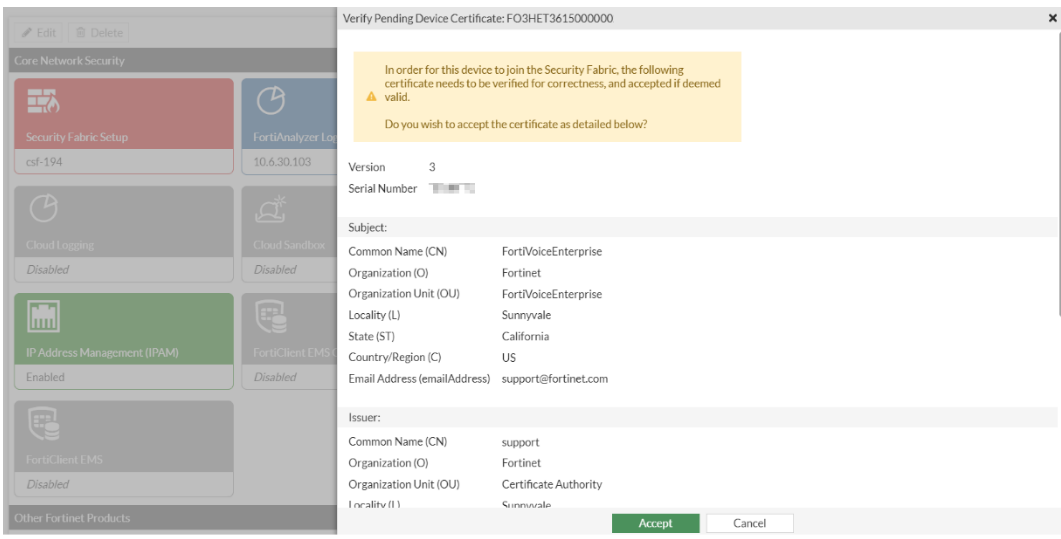
Authorizing using the FortiOS GUI

To authorize a FortiVoice to join the Security Fabric:

1. On the FortiVoice, enable the Security Fabric. See Enabling Security Fabric in the FortiVoice Phone System Administration Guide.
2. On the root FortiGate, go to *Security Fabric > Fabric Connectors*. The FortiVoice is highlighted in the topology list in the right panel with the status *Waiting for Authorization*.
3. Click the highlighted FortiVoice and select *Authorize*.



4. Verify that the certificate is correct, then click *Accept*.

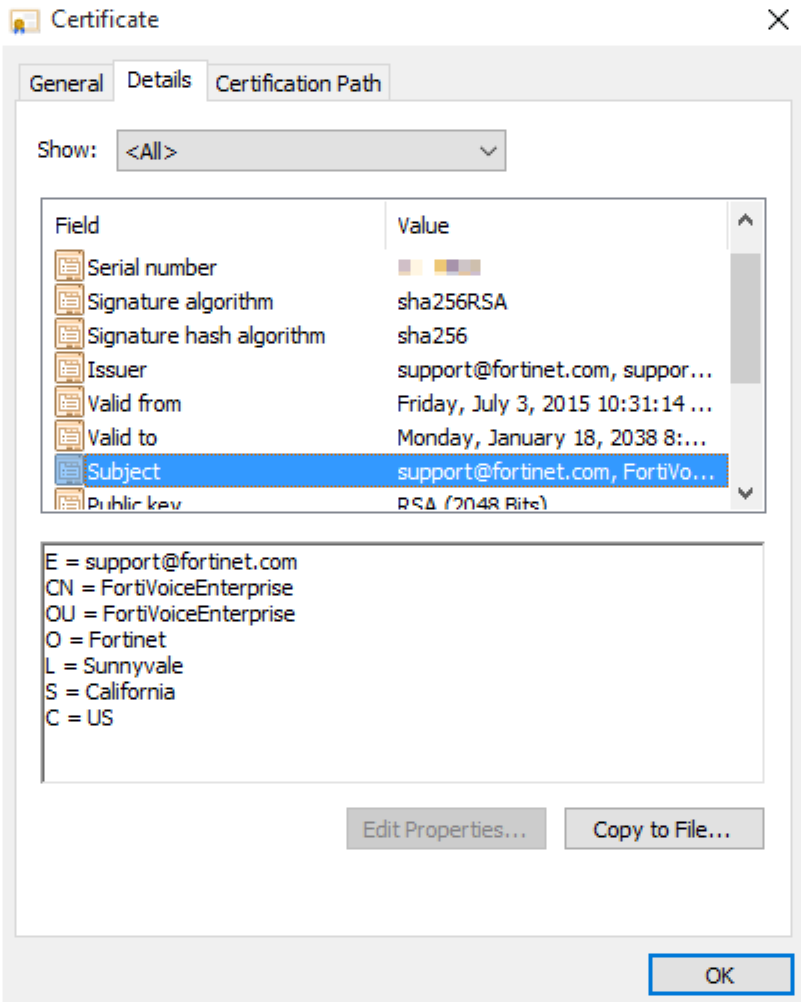


Pre-authorizing using the FortiVoice certificate

A FortiVoice can be pre-authorized using its serial number or certificate. When pre-authorizing, the FortiVoice can join at any time, and it will not need to be authorized in FortiOS. In the following example, the FortiVoice is pre-authorized using a certificate.

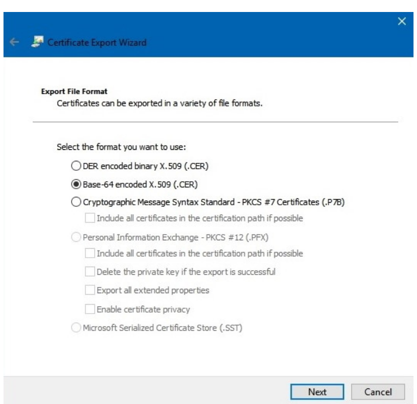
To pre-authorize a FortiVoice using a third-party or default certificate in the GUI:

1. Log in to the FortiVoice.
2. Download the certificate. For example, in Chrome:
   a. In the left side of the address bar, click the icon to view the site information.
   b. Click *Certificate*.
   c. In the *Certificate* window, click the *Details* tab, then click *Copy to File*.



d. The *Certificate Export Wizard* opens. Click *Next*.

e. Set the format to *Base-64 encoded X.509 (.CER)*, then click *Next*.



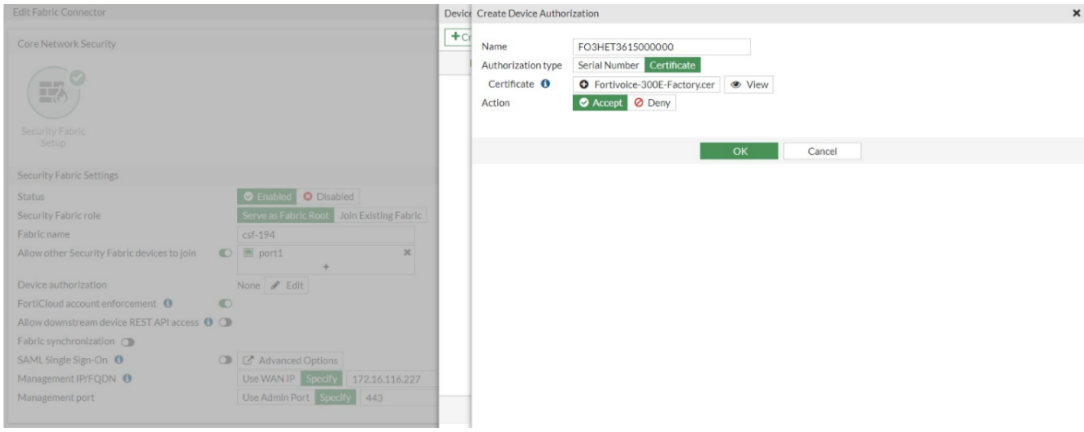
f. Browse to the folder location, enter a file name, then click *Next*.

g. Click *Finish*, then click *OK* to close the wizard.

3. In FortiOS, go to *Security Fabric > Fabric Connectors* and double-click the *Security Fabric Setup* card.

4. Beside *Device authorization*, click *Edit*.

5. Click *Create New* and enter the following:
   a. In the *Name* field, enter the FortiVoice serial number.
   b. Set the *Authorization type* to *Certificate*.
   c. Upload the .CER file.



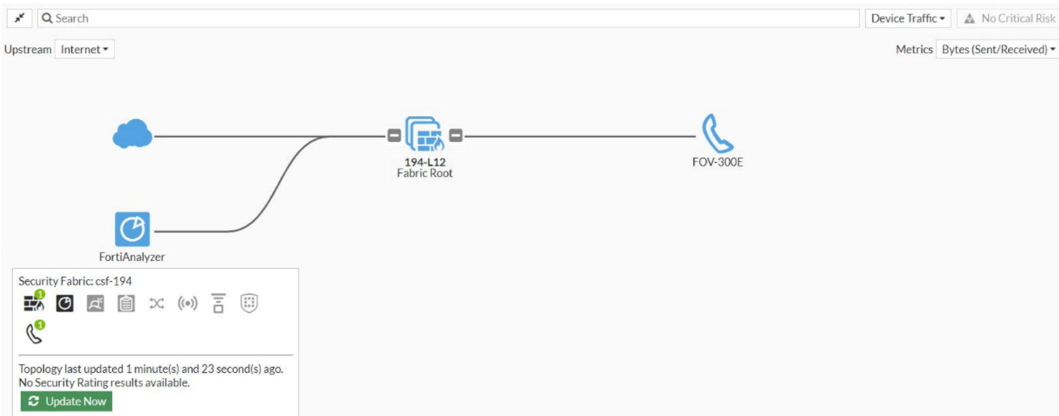
   d. Click *OK*, then close the *Device authorization* pane.

To pre-authorize a FortiVoice using a third-party or default certificate in the CLI:

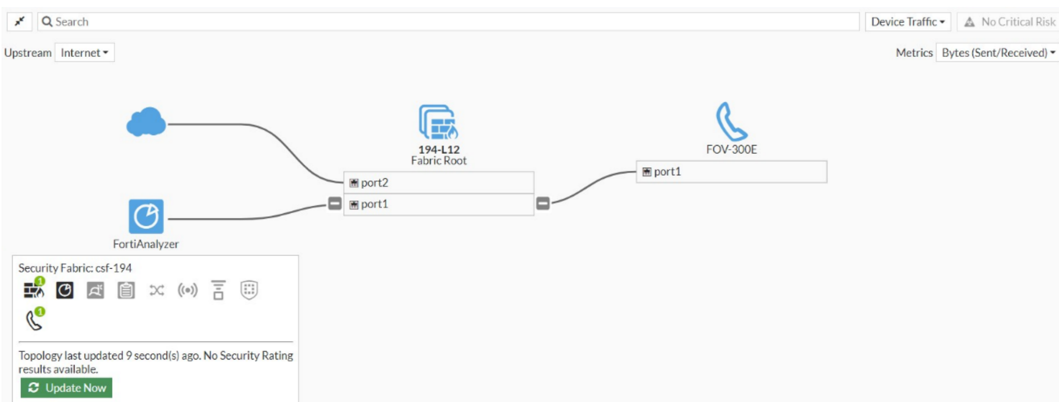
```
config system csf
    config trusted-list
        edit "FOV-300E"
            set action accept
            set authorization-type certificate
            set certificate "-----BEGIN CERTIFICATE-----
...
<encrypted_certificate_data>
...
-----END CERTIFICATE-----"
        next
    end
end
```

To verify the connection status:

1. After the FortiVoice is authorized, go to *Security Fabric > Physical Topology* and confirm that it is included in the topology.



2. Go to *Security Fabric* > *Logical Topology* and confirm the FortiVoice is also displayed there.



Logging in to the FortiVoice using the Security Fabric

To log in using a topology page:

1. Go to *Security Fabric* > *Physical Topology* or *Security Fabric* > *Logical Topology*.
2. Click on the FortiVoice and select *Login to <serial_number>*.

To log in using the Fabric Connectors page:

1. Go to *Security Fabric* > *Fabric Connectors*.
2. In the topology tree, click the FortiVoice and select *Login to <serial_number>*.

Using the Security Fabric

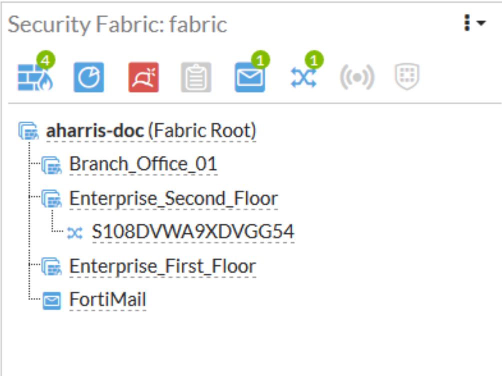
Dashboard widgets

Security Fabric widgets can be added to FortiGate dashboards, including:

- Security Fabric status on page 2274
- Fabric Device on page 2274
- FortiGate Cloud on page 2275

Security Fabric status

The Security Fabric status widget shows a summary of the devices in the Security Fabric.



Hover the cursor over the top icons to view pop-ups showing the statuses of the devices in the fabric.

The device tree shows devices that are connected, or could be connected, to you Security Fabric, according to the following color scheme:

- Blue: connected to the network
- Gray: not configured or not detected
- Red: no longer connected or not authorized

Hover over a device in the tree to view details about the device, such as it's serial number, operation mode, IP address, CPU and memory usage, and others, depending on the device type.

Unauthorized FortiAP and FortiSwitch devices are highlighted in the list, and can be authorized by clicking on the device name.

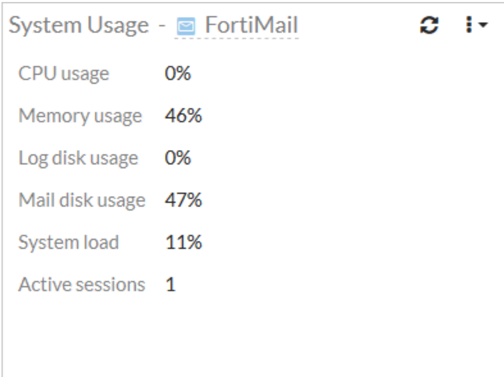
Fabric Device

A Fabric Device widget shows statistics and system information about the selected fabric device. Widgets can be added for various Fabric devices including FortiMail, FortiNDR (formerly FortiAI), and FortiDeceptor.

For a FortiMail device, the widget can show:

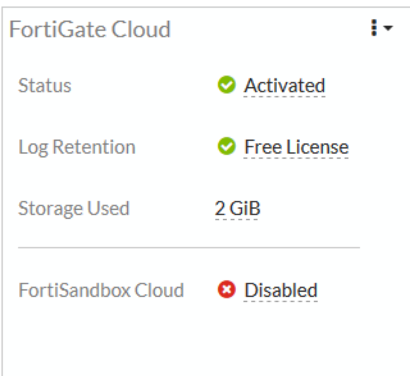
- Mail Statistics: a chart of the total messages and total spam messages over time.
- Statistics Summary: a pie chart summarizes mail statistics.
- System Information: The FortiMail System Information widget
- System Usage: System usage information, such as CPU, memory, and disk usage, as well as the number of active

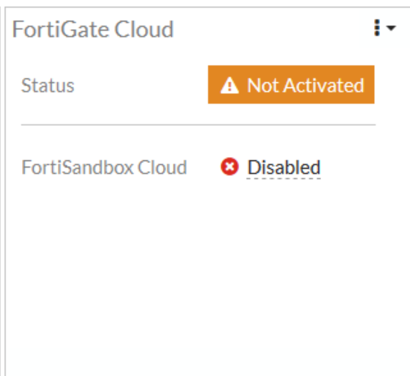
sessions.



FortiGate Cloud

The FortiGate Cloud widget shows the FortiGate Cloud status and information. If your account is not activated, you can activate it from the widget.





To activate your FortiGate Cloud account:

1. Click on the *Not Activated* button and select *Activate*. The *Activate FortiGate Cloud* pane opens.
2. If you already have an account:
   a. Fill in your email address, password, country or region, and reseller.
   b. Click *OK*.
3. If you are creating an account:
   a. In the *FortiCloud* field select *Create Account*.
   b. Fill in all of the required information.
   c. Click *OK*.

Topology

The full Security Fabric topology can be viewed on the root FortiGate. Downstream FortiGate devices' topology views do not include upstream devices.

The *Physical Topology* page shows the physical structure of your network, including all connected devices and the connections between them. The *Logical Topology* page shows information about the interfaces that connect devices to the Security Fabric.

In both topology pages, you can use filtering and sorting options to control the information that is shown. Hover the cursor over a device icon, port number, or endpoint to open a tooltip that shows information about that specific device, port, or endpoint. Right-click on a device to log into, configure, or deauthorize it. Right-click on an endpoint to perform various tasks, such as drilling down for more details in FortiView, quarantining the host, and banning the IP address.

The small number that might be shown in the top right corner of a device icon is the number of security ratings recommendations or warnings for that device. The circle color indicates the severity of the highest security rating check that failed. Clicking it opens the *Security Rating* page. See Security rating on page 2335 for more information.

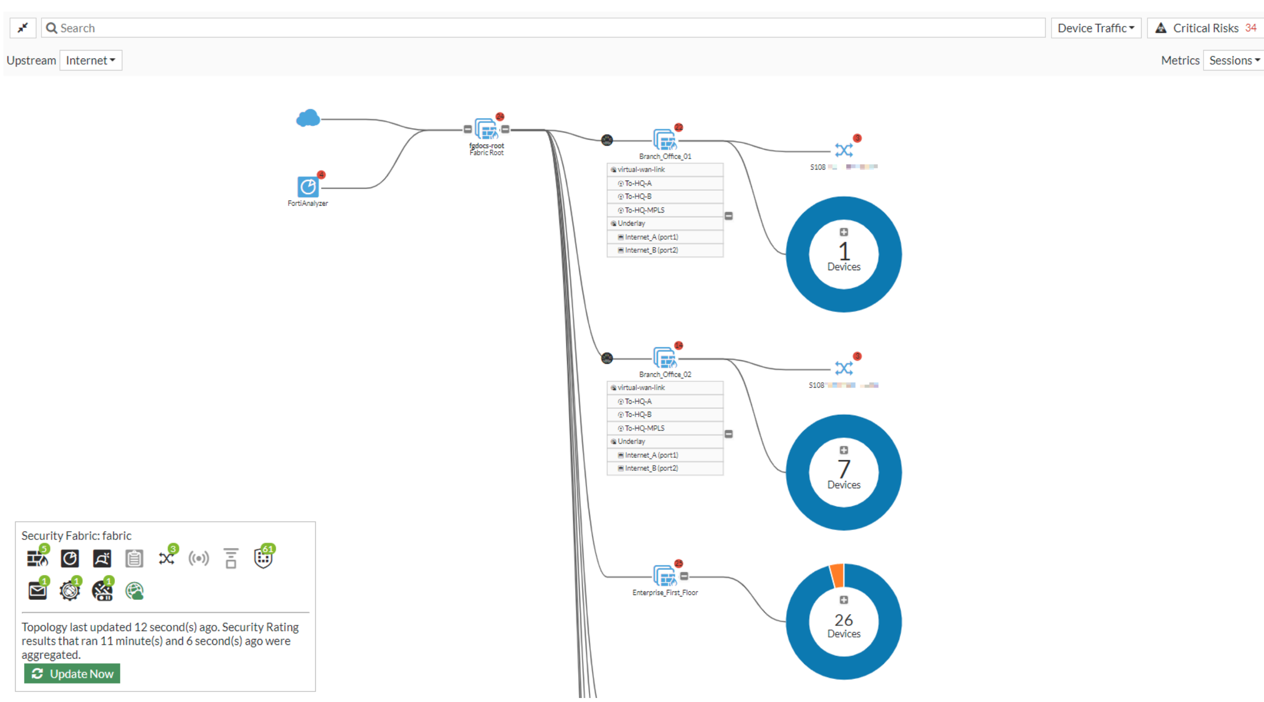
Views

From the dropdown list beside the search bar, select one of the following views:

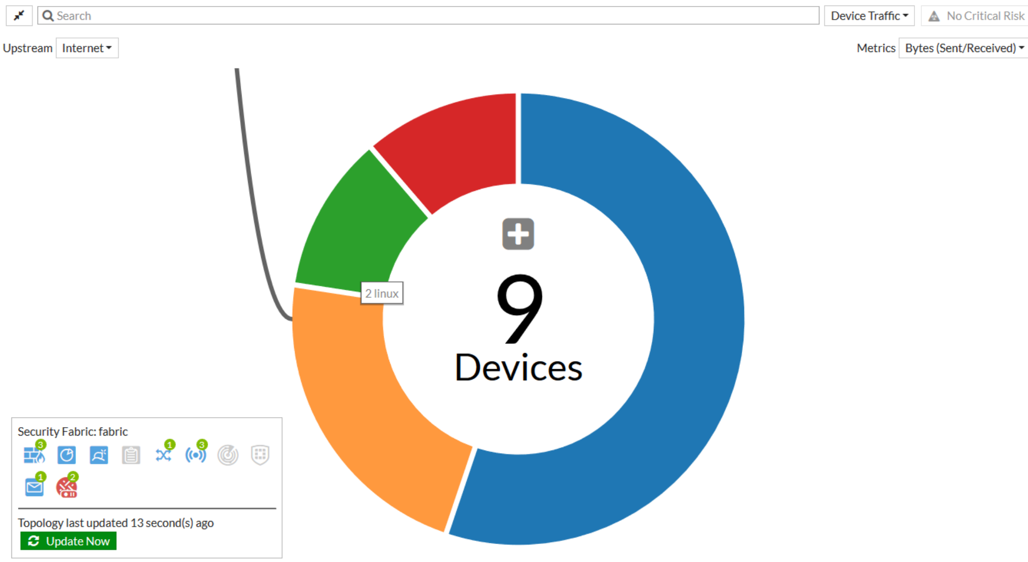
- *Device Traffic*: organize devices by traffic.
- *Device Count*: organize devices by the number of devices connected to it.
- *Device Operating System*: organize devices by operating system.
- *Device Hardware Vendor*: organize devices by hardware vendor.
- *Risk*: only include devices that have endpoints with medium, high, or critical risk values of the specified type: *All*, *Compromised Host*, *Vulnerability*, or *Threat Score*.
- *No Devices*: do not show endpoints.

Endpoint groups

The *Device Traffic* and *Device Count* views display endpoint groups as donut charts, with the total number of endpoints in the group in the center of the chart. Each sector of the donut chart represents a different endpoint operating system.



To zoom in on a donut chart, click any chart sector. Each sector represents a different endpoint OS. Hovering over each sector allows you to see the OS that the sector represents and the number of endpoints that have that OS installed.



In this example, the endpoint group contains a total of nine endpoints, with the following OSes installed:

Donut sector color	OS	Number of endpoints
Orange	Linux	2
Green	FortiMail	1
Red	FortiManager	1
Blue	Other	5

To view the endpoint group in a bubble pack display, click the + button in the center of the donut chart. You can view each individual endpoint in the bubble pack view.

FortiAP and FortiSwitch devices

Newly discovered FortiAP and FortiSwitch devices are initially shown in the topologies with gray icons to indicate that they have not been authorized. To authorize a device, click on the device icon or name and select *Authorize*. Once authorized, the device icon will turn blue.

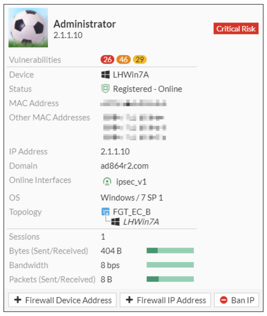
Right-click on an authorized FortiAP device to *Deauthorize* or *Restart* the device. Right-click on a FortiSwitch device to *Deauthorize*, *Restart*, or *Upgrade* the device, or to *Connect to the CLI*.

FortiAP and FortiSwitch links are enhanced to show link aggregation groups for the inter-switch link (ISL-LAG). To differentiate them from physical links, ISL-LAG links are shown with a thicker line. The endpoint circles can also be used as a reference to identify ISL-LAG groups that have more than two links.

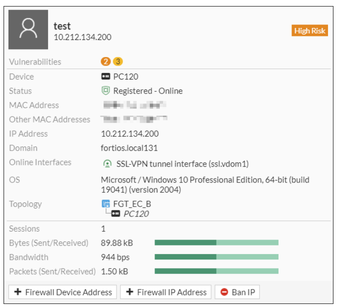
Managed clients connected over a VPN

When managed clients are connected over a VPN, EMS collects user information about these registered clients, such as the VPN connection information. The FortiGate can synchronize this user information from EMS and display it in the logical topology view to provide a detailed picture of clients and their associated VPN interfaces.

Client using an IPsec VPN interface:



Client using an SSL VPN interface:



Critical risks

Click the *Critical Risks* button to see a list of endpoints that are deemed critical risks, organized by threat severity. These are the red endpoints in the current topology view.

For each endpoint, the user's photo, name, IP address, email address, and phone number are shown. The number of vulnerabilities of each severity is shown, and if the IoC verdict is that the endpoint is compromised.

If applicable, the endpoint's host can be quarantined (click *Quarantine Host*) or their IP address can be banned (click *Ban IP*).

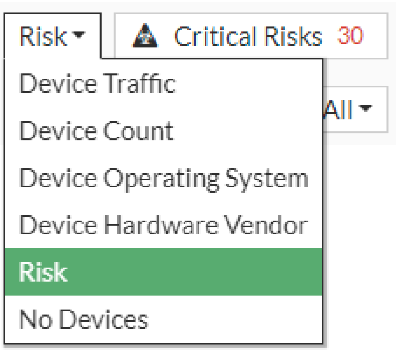
The dropdown menu also provides options to drill down to more information on compromised hosts or endpoint vulnerabilities.

Consolidated risk view

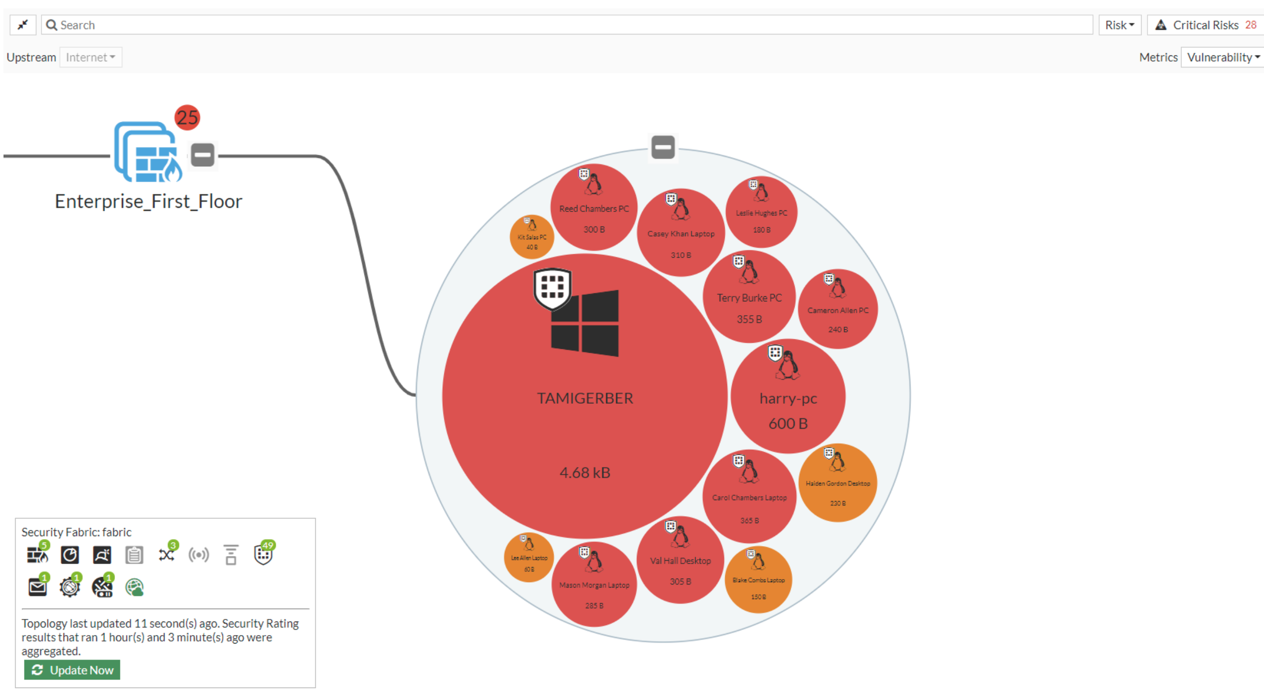
The consolidated *Risk* view mode displays different risks within the Security Fabric topology. You can use the *Risk* view mode to filter threats by *Compromised Hosts*, *Vulnerability*, and *Threat Score*.

To access the consolidated risk view mode:

1. On one of the topology pages, in the view option dropdown list beside the search bar, select *Risk*.



2. Select one of the following options from the *Risk Type* dropdown menu:
   a. *All*
   b. *Compromised Hosts*
   c. *Vulnerability*
   d. *Threat Score*
3. When devices fit into the risk metric, they will appear in the endpoint groups. Click the + in the endpoint group to display the devices in a bubble chart.

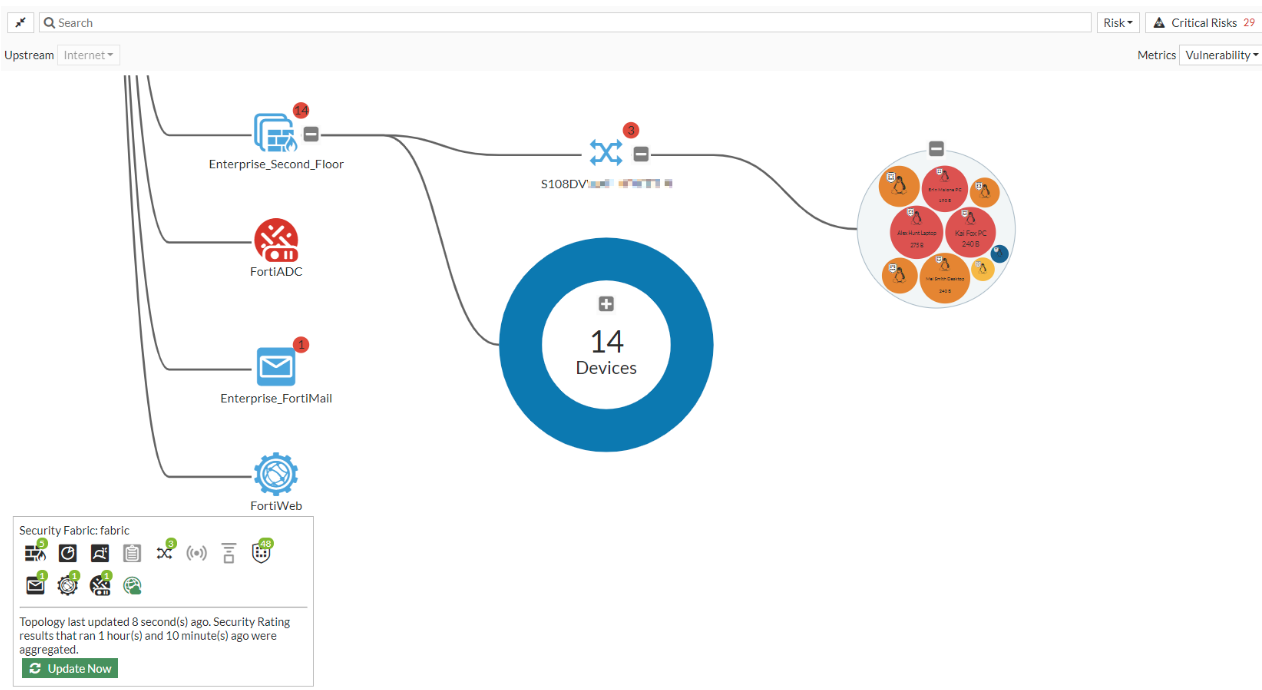


Viewing and controlling network risks in topology view

On the physical and logical topology pages, you can view and control compromised hosts. Compromised hosts behind a FortiSwitch or FortiAP can be quarantined.

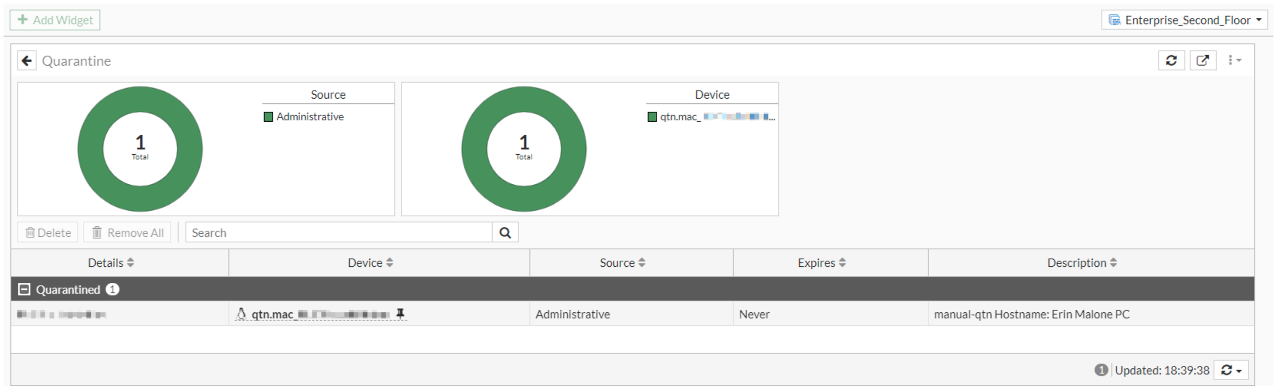
To view a compromised endpoint host:

1. Test that FortiGate detects a compromised endpoint host by opening a browser on the endpoint host and entering a malicious website URL. The browser displays a *Web Page Blocked!* warning and does not allow access to the website.
2. On the root FortiGate, go to *Security Fabric > Physical Topology* or *Security Fabric > Logical Topology*. Expand the endpoint group connected to a FortiSwitch or FortiAP.
   The endpoint host connected to the switch is highlighted in red. Mouse over the endpoint host to view a tooltip that shows the IoC verdict. The endpoint host is compromised.



To quarantine a compromised endpoint host:

1. On the *Physical Topology* or *Logical Topology* page, right-click the endpoint host and select *Quarantine Host*.
   A dialog displays the FortiGate, host MAC address, and description of the host to be quarantined. Quarantine entries for each MAC address will be created on the FortiGate that the FortiSwitch or FortiAP is connected to.
2. Click *OK*.
3. Go to *Dashboard > User & Devices* and click the *Quarantine* widget to expand it.
4. In the top-right corner, use the dropdown to select the FortiGate in which this host was quarantined. In this example, it is the Enterprise_Second_Floor FortiGate.



5. On the endpoint host, open a browser and visit a website such as https://www.fortinet.com/. If the website cannot be accessed, this confirms that the endpoint host is quarantined.

To show the quarantined device from the CLI:

1. Log in to the downstream device where the host was quarantined (Enterprise_Second_Floor).
2. Enter the following show command:

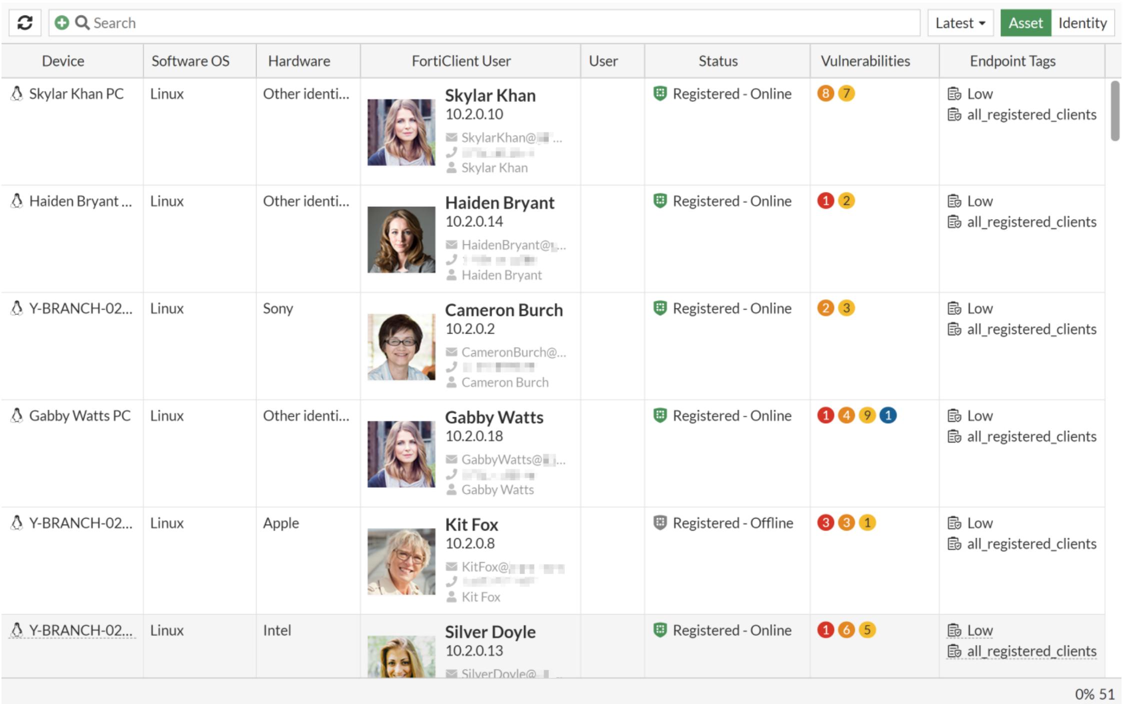
```
Enterprise_Second_Floor # show user quarantine
config user quarantine
    set firewall-groups "QuarantinedDevices"
    config targets
        edit "Erin Malone PC"
            set description "Manually quarantined"
            config macs
                edit **:**:**:**:**:**
                    set description "manual-qtn Hostname: Erin Malone PC"
                next
            end
        next
    end
end
```

Asset Identity Center page

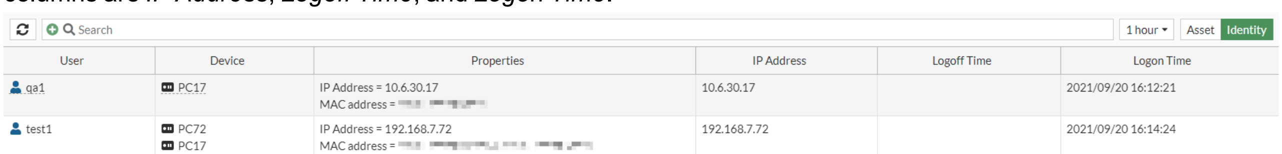
The *Asset Identity Center* page unifies information from detected addresses, devices, and users into a single page, while building a data structure to store the user and device information in the backend. *Asset* view groups information by *Device*, while *Identity* view groups information by *User*. Hover over a device or a user in the GUI to perform different actions relevant to the object, such as adding a firewall device address, adding an IP address, banning the IP, quarantining the host, and more.

To view the Asset Identity Center page:

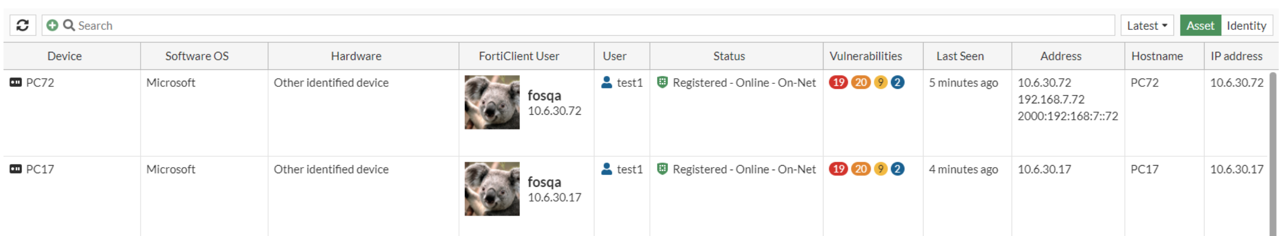
1. Go to *Security Fabric > Asset Identity Center*.
2. Click *Asset* to view information by device. The default columns are *Device*, *Software OS*, *Hardware*, *FortiClient User*, *User*, *Status*, *Vulnerabilities*, *Endpoint Tags*, and *Last Seen*. The optional columns are *Address*, *Firewall Address*, *Hostname*, *IP Address*, and *Server*.

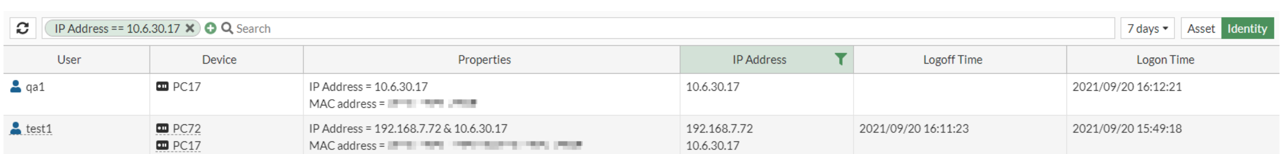


3. Click *Identity* to view information by user. The default columns are *User*, *Device*, and *Properties*. The optional columns are *IP Address*, *Logoff Time*, and *Logon Time*.

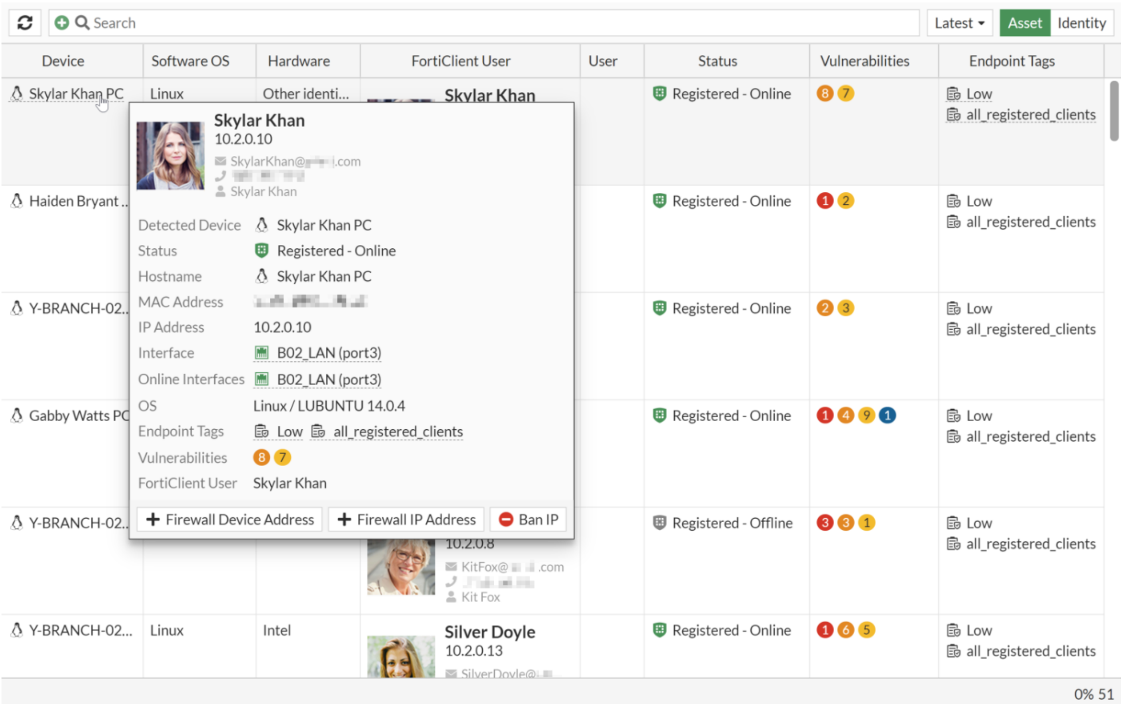


Each view has a dropdown option to view the information within different time frames (*Latest*, *1 hour*, *24 hours*, and *7 days*). Vulnerability information is displayed when applicable. The page displays user and device relationships, such as which users are logged in to multiple devices or if multiple users are logged in to single devices.





4. Hover over a device in the list to view the tooltip and possible actions. In this example, the available actions are add firewall device address, add firewall IP address, and ban the IP.



Diagnostics for the unified user device store

The following options are available for `diagnose user-device-store unified <option>`:

Option	Description
`device-memory-query`	Get device records and associated user records from memory.
`device-query`	Get device records and associated user records from memory and disk.
`user-memory-query`	Get user records and associated device records from memory.
`user-query`	Get user records and associated device records from memory and disk.
`re-query`	Retrieve query by `<query-id> <iteration-start> <iteration-count>` (takes 0-3 arguments).
`list`	List unified queries.
`clear`	Delete all unified queries.
`dump`	Dump unified query stats by `<query-id>` (takes 0-1 arguments).
`delete`	Delete unified query by `<query-id>` (takes 0-1 arguments).
`stats`	Get statistics for unified queries.
`debug`	Enable/disable debug logs for unified queries.

OT asset visibility and network topology

When the *Operational Technology (OT)* feature is enabled, tabs are added in the *Asset Identity Center* page to view the OT asset list and OT network topology using Purdue Levels. This feature is available regardless of whether a Security

Fabric is enabled.

To enable the OT features in the GUI:

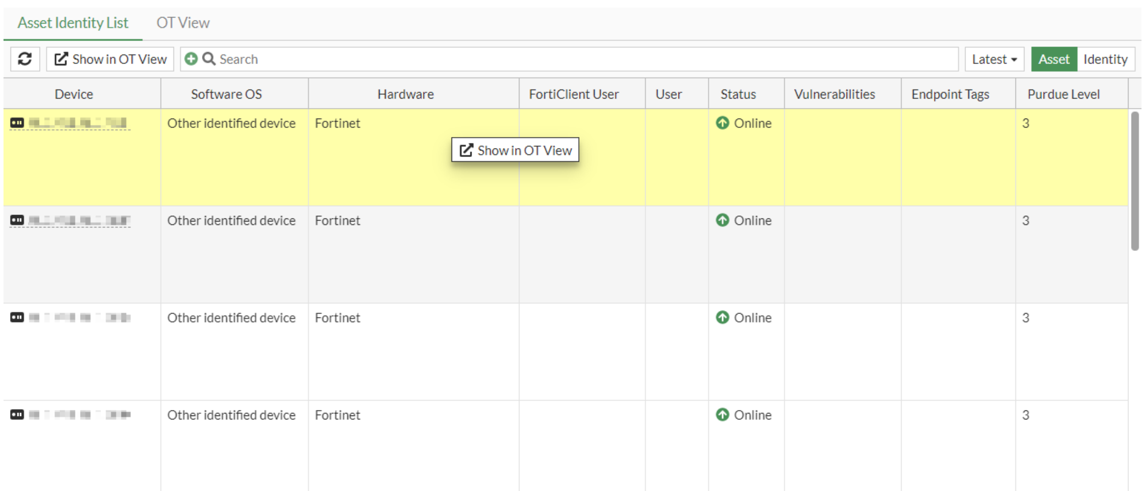
1. Go to *System > Feature Visibility*.
2. In the *Additional Features* section, enable *Operational Technology (OT)*.
3. Click *Apply*.

To enable the OT features in the CLI:

```
config system settings
    set gui-ot enable
end
```

Once enabled, the *Security Fabric > Asset Identity Center* page displays an *Asset Identity List* tab and an *OT View* tab.

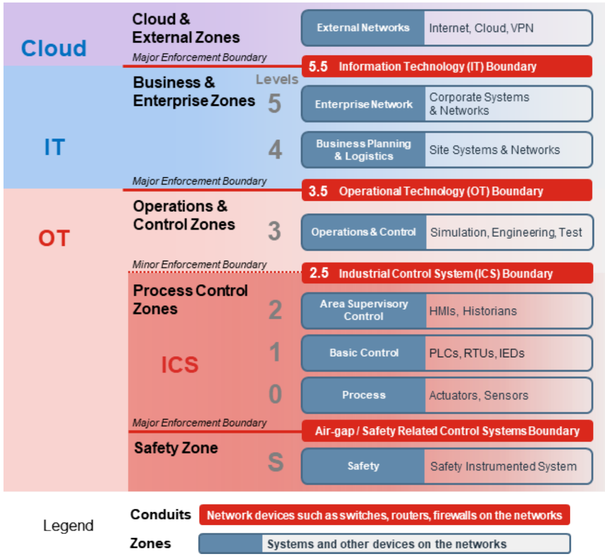
- The *Asset Identity List* tab includes a configurable *Purdue Level* column and a *Show in OT View* option for selected devices in the table.



- The *OT View* tab shows a topology of detected components and connections mapped to Purdue Levels. The default view is locked, but devices can be dragged and dropped to other Purdue Levels if the view is unlocked.

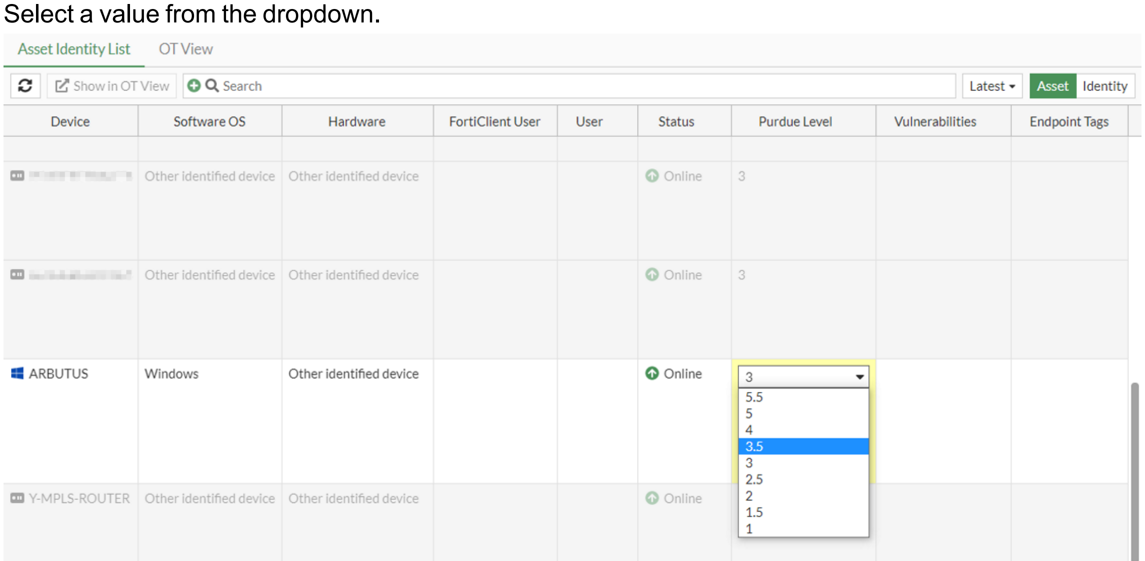
FortiGates and managed FortiSwitches are statically assigned Purdue Level 2 and cannot be changed. Other detected devices are assigned Purdue Level 3 by default and can be changed (except to level S, 0, or external).

The following diagram lists the Purdue Levels based on OT network topologies:



To change the Purdue Level in the Asset Identity List tab:

1. Go to *Security Fabric > Asset Identity Center* and select the *Asset Identity List* tab.
2. Add the *Purdue Level* column to the table:
   a. Hover over the table header and click the gear icon (*Configure Table*).
   b. Select *Purdue Level*.
   c. Click *Apply*.
3. Select a device and hover over the *Purdue Level* value.
4. Click the pencil icon to edit the level.
5. Select a value from the dropdown.

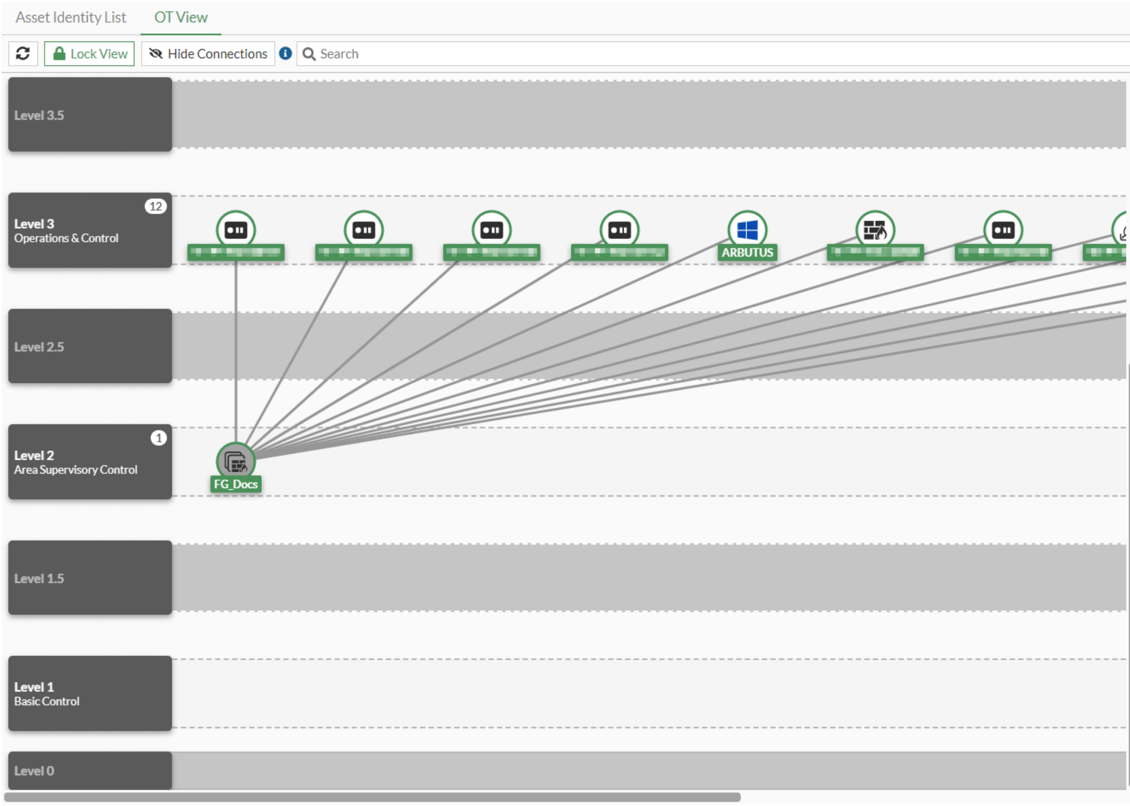


6. Click *Apply*.

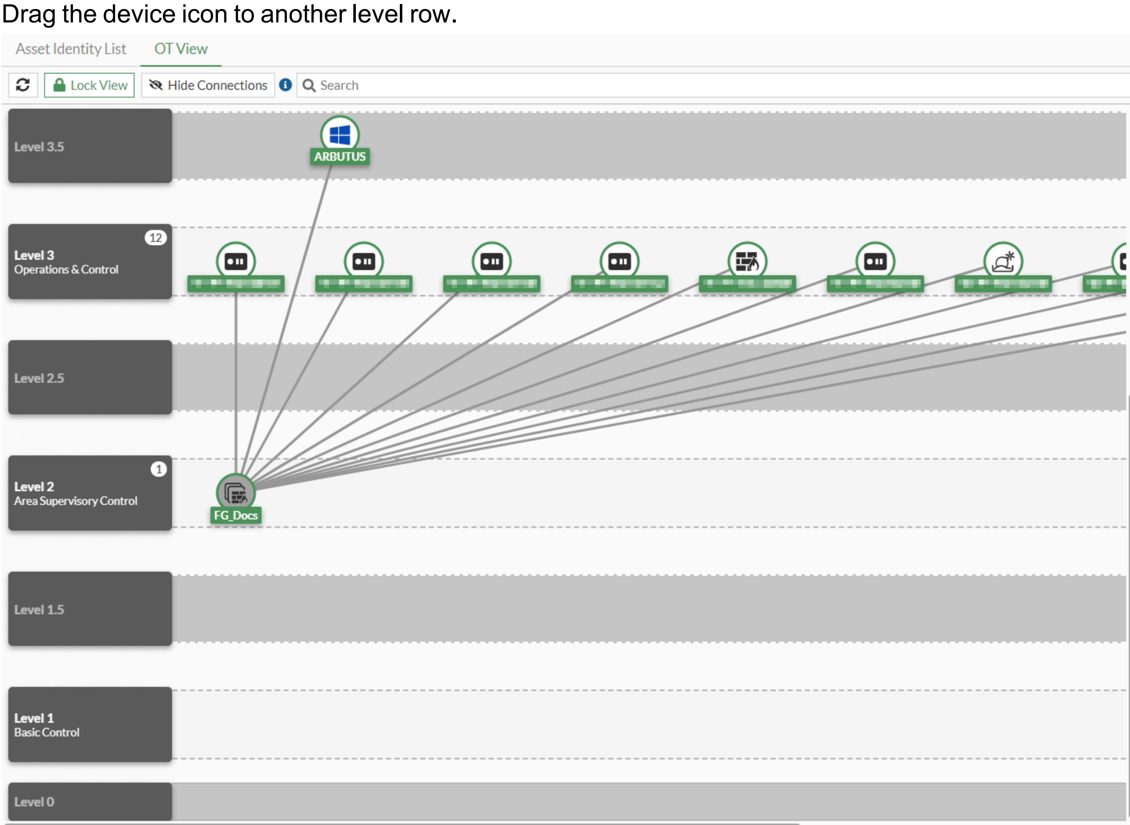
To change the Purdue Level in the OT View tab:

1. Go to *Security Fabric > Asset Identity Center* and select the *OT View* tab.
2. Click *Unlock View*.

3. Select a device.



4. Drag the device icon to another level row.



5. Optionally, click *Lock View* to revert to the locked view.

To change the Purdue Level in the CLI:

```
# diagnose user-device-store device memory ot-purdue-set <mac> <ip> <level>
```

`mac`	Enter the MAC address of the device.
`ip`	Enter the IPv4 address of the device.
`level`	Enter the Purdue Level: 1, 1.5, 2, 2.5, 3, 3.5, 4, 5, 5.5.

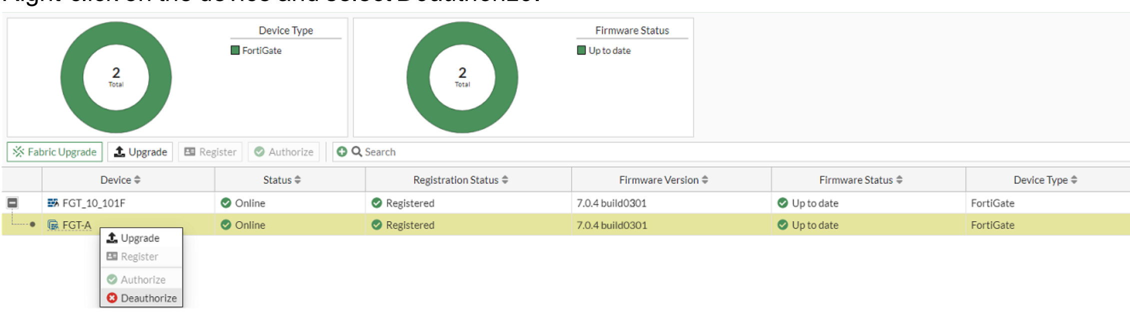
WebSocket for Security Fabric events

With the WebSocket for Security Fabric events, subscribers to the WebSocket (such as the *Fabric Management* page) are updated upon new Fabric events and alert users to reload the page.

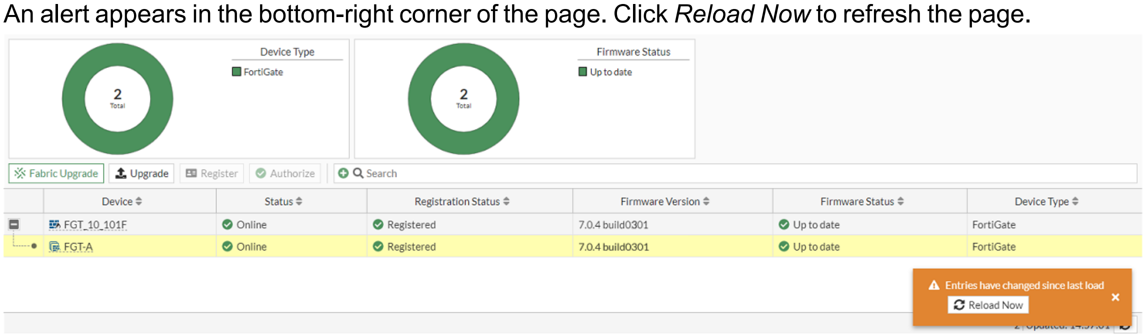
Example

To deauthorize a downstream FortiGate:

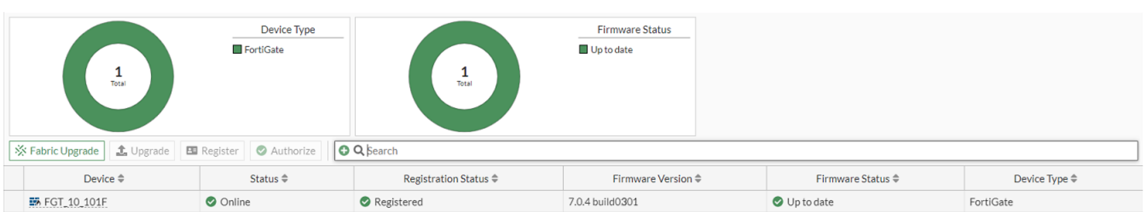
1. Go to *System* > *Fabric Management* and select a downstream FortiGate in the table.
2. Right-click on the device and select *Deauthorize*.



3. An alert appears in the bottom-right corner of the page. Click *Reload Now* to refresh the page.



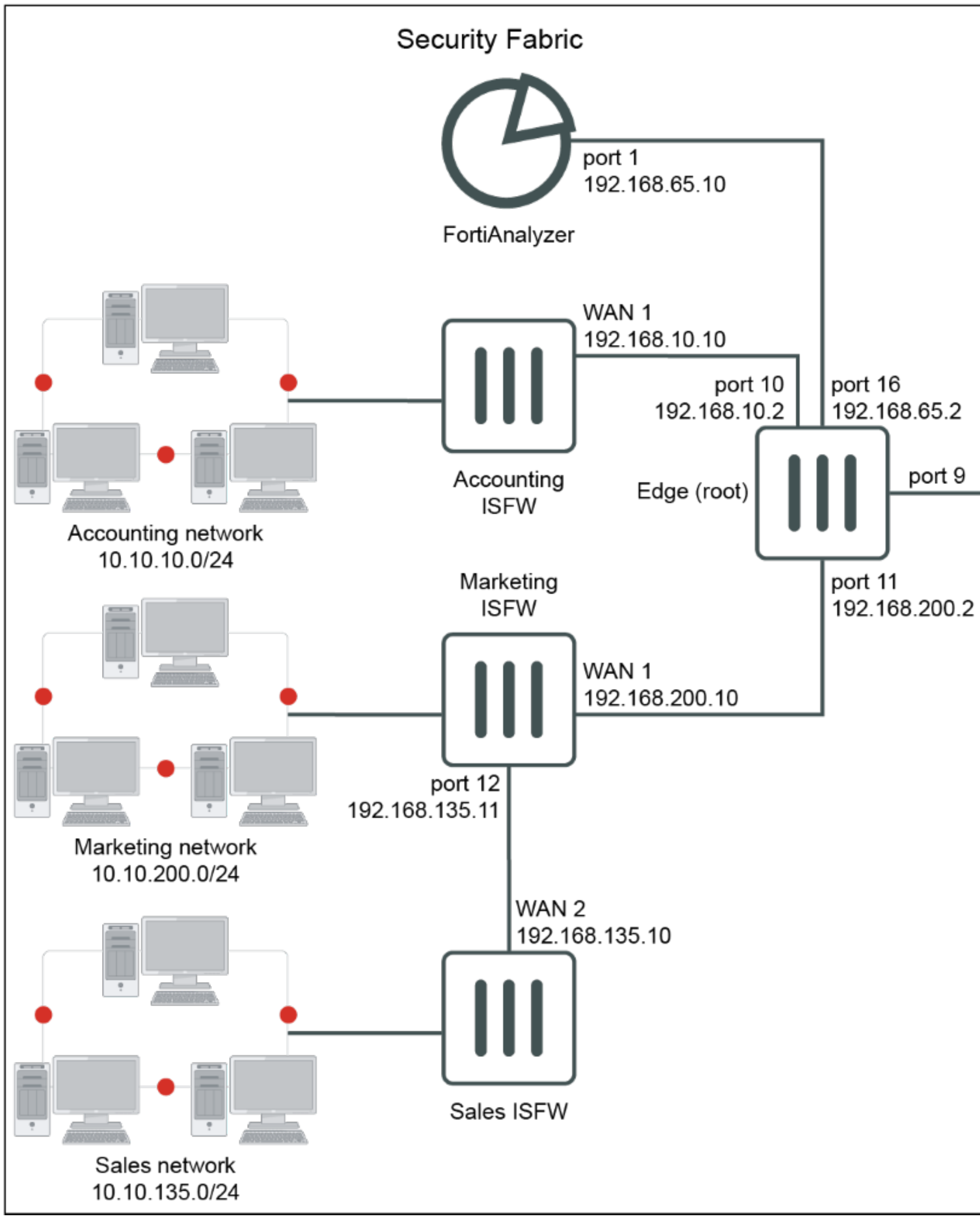
The deauthorized FortiGate is removed from the table.



Deploying the Security Fabric

This topic provides an example of deploying Security Fabric with three downstream FortiGates connecting to one root FortiGate. To deploy Security Fabric, you need a FortiAnalyzer running firmware version 6.2 or later.

The following shows a sample network topology with three downstream FortiGates (Accounting, Marketing, and Sales) connected to the root FortiGate (Edge).



To configure the root FortiGate (Edge):

1. Configure interfaces:
   a. In the root FortiGate (Edge), go to *Network > Interfaces*.
   b. Edit *port16*:
      - Set *Role* to *DMZ*.
      - For the interface connected to FortiAnalyzer, set the *IP/Network Mask* to *192.168.65.2/255.255.255.0*
   c. Edit *port10*:
      - Set *Role* to *LAN*.
      - For the interface connected to the downstream FortiGate (Accounting), set the *IP/Network Mask* to *192.168.10.2/255.255.255.0*
   d. Edit *port11*:
      - Set *Role* to *LAN*.
      - For the interface connected to the downstream FortiGate (Marketing), set the *IP/Network Mask* to *192.168.200.2/255.255.255.0*
2. Configure Security Fabric:
   a. In the root FortiGate (Edge), go to *Security Fabric > Fabric Connectors* and double-click the *Security Fabric Setup* card.
   b. For *Status*, click *Enable*.
   c. Set the *Security Fabric role* to *Serve as Fabric Root*. The FortiAnalyzer settings can be configured.
   d. Enter the FortiAnalyzer IP (*192.168.65.10*) and select and *Upload option* (the default is *Real Time*).
   e. Click *Test Connectivity*.
      A warning message indicates that the FortiGate is not authorized on the FortiAnalyzer. The authorization is configured in a later step on the FortiAnalyzer.
   f. Click *OK*. The FortiAnalyzer serial number is verified.
   g. Enter a *Fabric name*, such as *Office-Security-Fabric*.
   h. Ensure *Allow other Security Fabric devices to join* is enabled and add *port10* and *port11*.
   i. Click *OK*.
3. Create a policy to allow the downstream FortiGate (Accounting) to access the FortiAnalyzer:
   a. In the root FortiGate (Edge), go to *Policy & Objects > Addresses*.
   b. Click *Create New*.
      - Set *Name* to *FAZ-addr*.
      - Set *Type* to *Subnet*.
      - Set *Subnet/IP Range* to *192.168.65.10/32*.
      - Set *Interface* to *any*.
   c. Click *OK*.
   d. Click *Create New*.
      - Set *Name* to *Accounting*.
      - Set *Type* to *Subnet*.
      - Set *Subnet/IP Range* to *192.168.10.10/32*.
      - Set *Interface* to *any*.
   e. Click *OK*.
   f. In the root FortiGate (Edge), go to *Policy & Objects > Firewall Policy* and click *Create New*.
      - Set *Name* to *Accounting-to-FAZ*.
      - Set *srcintf* to *port10*.
      - Set *dstintf* to *port16*.

- Set *srcaddr* to *Accounting-addr*.
- Set *dstaddr* to *FAZ-addr*.
- Set *Action* to *Accept*.
- Set *Schedule* to *Always*.
- Set *Service* to *All*.
- Enable *NAT*.
- Set *IP Pool Configuration* to *Use Outgoing Interface Address*.

g. Click *OK*.

4. Create a policy to allow the two downstream FortiGates (Marketing and Sales) to access the FortiAnalyzer:
   a. In the root FortiGate (Edge), go to *Policy & Objects > Addresses* and click *Create New*.
      - Set *Name* to *Marketing-addr*.
      - Set *Type* to *Subnet*.
      - Set *Subnet/IP Range* to *192.168.200.10/32*.
      - Set *Interface* to *any*.
   b. Click *OK*.
   c. In the root FortiGate (Edge), go to *Policy & Objects > Firewall Policy* and click *Create New*.
      - Set *Name* to *Marketing-to-FAZ*.
      - Set *srcintf* to *port11*.
      - Set *dstintf* to *port16*.
      - Set *srcaddr* to *Marketing-addr*.
      - Set *dstaddr* to *FAZ-addr*.
      - Set *Action* to *Accept*.
      - Set *Schedule* to *Always*.
      - Set *Service* to *All*.
      - Enable *NAT*.
      - Set *IP Pool Configuration* to *Use Outgoing Interface Address*.
   d. Click *OK*.

To configure the downstream FortiGate (Accounting):

1. Configure interface:
   a. In the downstream FortiGate (Accounting), go to *Network > Interfaces*.
   b. Edit interface *wan1*:
      - Set *Role* to *WAN*.
      - For the interface connected to root, set the *IP/Network Mask* to *192.168.10.10/255.255.255.0*
2. Configure the default static route to connect to the root FortiGate (Edge):
   a. In the downstream FortiGate (Accounting), go to *Network > Static Routes* and click *Create New* or *Create New > IPv4 Static Route*.
      - Set *Destination* to *0.0.0.0/0.0.0.0*.
      - Set *Interface* to *wan1*.
      - Set *Gateway Address* to *192.168.10.2*.
   b. Click *OK*.
3. Configure Security Fabric:
   a. In the downstream FortiGate (Accounting), go to *Security Fabric > Fabric Connectors* and double-click the *Security Fabric Setup* card.

   b. For *Status*, click *Enable*.
      FortiAnalyzer automatically enables logging. Settings for the FortiAnalyzer are retrieved from the root FortiGate (Edge) when FortiGate (Accounting) connects to the root FortiGate (Edge).
   c. Set the *Security Fabric role* to *Join Existing Fabric*.
   d. *Upstream FortiGate IP* is filled in automatically with the default static route *Gateway Address* of *192.168.10.2* set in the previous step.
   e. Disable *Allow other FortiGates to join*, because there is no downstream FortiGate connecting to it.
   f. Click *OK*.

To configure the downstream FortiGate (Marketing):

1. Configure interface:
   a. In the downstream FortiGate (Marketing), go to *Network > Interfaces*.
   b. Edit *port12*:
      - Set *Role* to *LAN*.
      - For the interface connected to the downstream FortiGate (Sales), set the *IP/Network Mask* to *192.168.135.11/255.255.255.0*.
   c. Edit *wan1*:
      - Set *Role* to *WAN*.
      - For the interface connected to the root FortiGate (Edge), set the *IP/Network Mask* to *192.168.200.10/255.255.255.0*.
2. Configure the default static route to connect to the root FortiGate (Edge):
   a. In the downstream FortiGate (Marketing), go to *Network > Static Routes* and click *Create New* or *Create New > IPv4 Static Route*.
      - Set *Destination* to *0.0.0.0/0.0.0.0*.
      - Set *Interface* to *wan1*.
      - Set *Gateway Address* to *192.168.200.2*.
   b. Click *OK*.
3. Configure Security Fabric:
   a. In the downstream FortiGate (Marketing), go to *Security Fabric > Fabric Connectors* and double-click the *Security Fabric Setup* card.
   b. For *Status*, click *Enable*.
      FortiAnalyzer automatically enables logging. Settings for the FortiAnalyzer are retrieved from the root FortiGate (Edge) when FortiGate (Marketing) connects to the root FortiGate (Edge).
   c. Set the *Security Fabric role* to *Join Existing Fabric*.
   d. *Upstream FortiGate IP* is filled in automatically with the default static route *Gateway Address* of *192.168.200.2* set in the previous step.
   e. Enable *Allow other FortiGates to join* and add *port12*.
   f. Click *OK*.
4. Create a policy to allow another downstream FortiGate (Sales) going through FortiGate (Marketing) to access the FortiAnalyzer:
   a. In the downstream FortiGate (Marketing), go to *Policy & Objects > Addresses* and click *Create New*.
      - Set *Name* to *FAZ-addr*.
      - Set *Type* to *Subnet*.
      - Set *Subnet/IP Range* to *192.168.65.10/32*.
      - Set *Interface* to *any*.
   b. Click *OK*.

c. Click *Create New*.
   - Set *Name* to *Sales-addr*.
   - Set *Type* to *Subnet*.
   - Set *Subnet/IP Range* to *192.168.135.10/32*.
   - Set *Interface* to *any*.
d. Click *OK*.
e. In the downstream FortiGate (Marketing), go to *Policy & Objects* > *Firewall Policy* and click *Create New*.
   - Set *Name* to *Sales-to-FAZ*.
   - Set *srcintf* to *port12*.
   - Set *dstintf* to *wan1*.
   - Set *srcaddr* to *Sales-addr*.
   - Set *dstaddr* to *FAZ-addr*.
   - Set *Action* to *Accept*.
   - Set *Schedule* to *Always*.
   - Set *Service* to *All*.
   - Enable *NAT*.
   - Set *IP Pool Configuration* to *Use Outgoing Interface Address*.
f. Click *OK*.

To configure the downstream FortiGate (Accounting):

1. Configure interface:
   a. In the downstream FortiGate (Accounting), go to *Network* > *Interfaces*.
   b. Edit interface *wan1*:
      - Set *Role* to *WAN*.
      - For the interface connected to root, set the *IP/Network Mask* to *192.168.10.10/255.255.255.0*
2. Configure the default static route to connect to the root FortiGate (Edge):
   a. In the downstream FortiGate (Accounting), go to *Network* > *Static Routes* and click *Create New* or *Create New > IPv4 Static Route*.
      - Set *Destination* to *0.0.0.0/0.0.0.0*.
      - Set *Interface* to *wan1*.
      - Set *Gateway Address* to *192.168.10.2*.
   b. Click *OK*.
3. Configure Security Fabric:
   a. In the downstream FortiGate (Accounting), go to *Security Fabric* > *Fabric Connectors* and double-click the *Security Fabric Setup* card.
   b. For *Status*, click *Enable*.
      FortiAnalyzer automatically enables logging. Settings for the FortiAnalyzer are retrieved from the root FortiGate (Edge) when FortiGate (Accounting) connects to the root FortiGate (Edge).
   c. Set the *Security Fabric role* to *Join Existing Fabric*.
   d. *Upstream FortiGate IP* is filled in automatically with the default static route *Gateway Address* of *192.168.10.2* set in the previous step.
   e. Disable *Allow other FortiGates to join*, because there is no downstream FortiGate connecting to it.
   f. Click *OK*.

To configure the downstream FortiGate (Sales):

1. Configure interface:
   a. In the downstream FortiGate (Sales), go to *Network* > *Interfaces*.
   b. Edit *wan2*:
      - Set *Role* to *WAN*.
      - For the interface connected to the upstream FortiGate (Marketing), set the *IP/Network Mask* to *192.168.135.10/255.255.255.0*.
2. Configure the default static route to connect to the upstream FortiGate (Marketing):
   a. In the downstream FortiGate (Sales), go to *Network* > *Static Routes* and click *Create New* or *Create New* > *IPv4 Static Route*.
      - Set *Destination* to *0.0.0.0/0.0.0.0*.
      - Set *Interface* to *wan2*.
      - Set *Gateway Address* to *192.168.135.11*.
   b. Click *OK*.
3. Configure Security Fabric:
   a. In the downstream FortiGate (Sales), go to *Security Fabric* > *Fabric Connectors* and double-click the *Security Fabric Setup* card.
   b. For *Status*, click *Enable*.
      FortiAnalyzer automatically enables logging. Settings for the FortiAnalyzer are retrieved from the root FortiGate (Edge) when FortiGate (Sales) connects to the root FortiGate (Edge).
   c. Set the *Security Fabric role* to *Join Existing Fabric*.
   d. *Upstream FortiGate IP* is filled in automatically with the default static route *Gateway Address* of *192.168.135.11* set in the previous step.
   e. Disable *Allow other FortiGates to join*, because there is no downstream FortiGate connecting to it.
   f. Click *OK*.

To authorize downstream FortiGates (Accounting, Marketing, and Sales) on the root FortiGate (Edge):

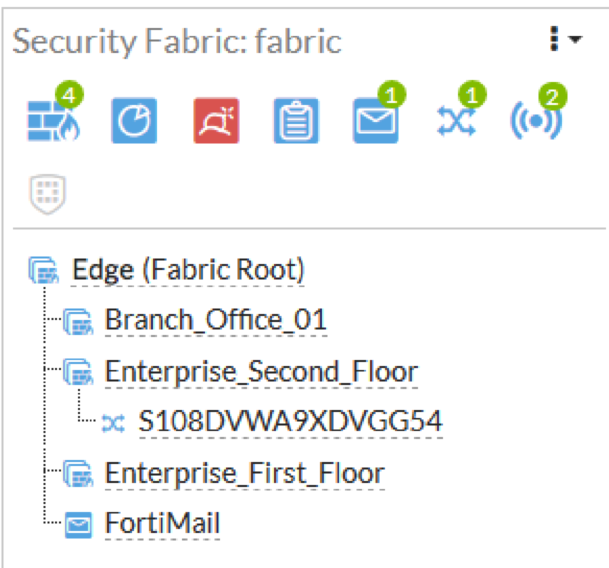
1. In the root FortiGate (Edge), go to *Security Fabric* > *Fabric Connectors* and double-click the *Security Fabric Setup* card.
   The *Topology* tree highlights two connected FortiGates with their serial numbers and asks you to authorize the highlighted devices.
2. Select the highlighted FortiGates and select *Authorize*.
   After they are authorized, the two downstream FortiGates (Accounting and Marketing) appear in the *Topology* tree in the *Security Fabric* > *Fabric Connectors* > *Security Fabric Setup* page. This means that the two downstream FortiGates (Accounting and Marketing) have successfully joined the Security Fabric.
3. The *Topology* tree now highlights the FortiGate with the serial number that is connected to the downstream FortiGate (Marketing) and asks you to authorize the highlighted device.
4. Select the highlighted FortiGates and select *Authorize*.
   After it is authorized, the downstream FortiGate (Sales) appears in the *Topology* tree in the *Security Fabric* > *Fabric Connectors* > *Security Fabric Setup* page. This means that the downstream FortiGates (Sales) has successfully joined the Security Fabric.

To use FortiAnalyzer to authorize all the Security Fabric FortiGates:

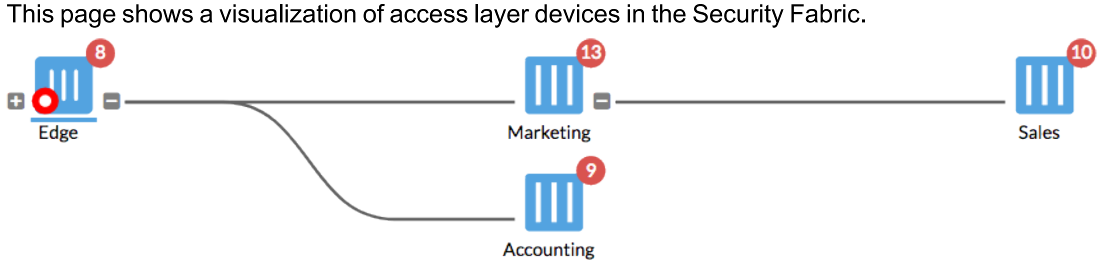
1. Authorize all the Security Fabric FortiGates on the FortiAnalyzer side:
   a. On the FortiAnalyzer, go to *System Settings > Network > All Interfaces*.
   b. Edit *port1* and set *IP Address/Netmask* to *192.168.65.10/255.255.255.0*.
   c. Go to *Device Manager > Unauthorized*. All of the FortiGates are listed as unauthorized.
      i. Select all the FortiGates and select *Authorize*. The FortiGates are now listed as authorized.
         After a moment, a warning icon appears beside the root FortiGate (Edge) because the FortiAnalyzer needs administrative access to the root FortiGate (Edge) in the Security Fabric.
      ii. Click the warning icon and enter the admin username and password of the root FortiGate (Edge).
2. Check FortiAnalyzer status on all the Security Fabric FortiGates:
   a. On each FortiGate, go to *Security Fabric > Fabric Connectors* and double-click the *FortiAnalyzer Logging* card.
   b. Check that *Storage usage* information is shown.

To check Security Fabric deployment result:

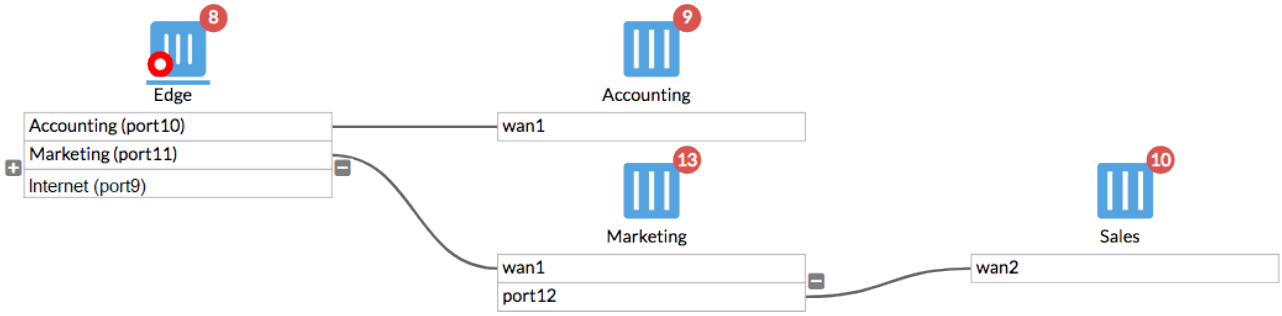
1. On FortiGate (Edge), go to *Dashboard > Status*.
   The *Security Fabric* widget displays all the FortiGates in the Security Fabric.



2. On FortiGate (Edge), go to *Security Fabric > Physical Topology*.
   This page shows a visualization of access layer devices in the Security Fabric.



3. On FortiGate (Edge), go to *Security Fabric > Physical Topology*.
   This dashboard shows information about the interfaces of each device in the Security Fabric.



To run diagnose commands:

1. Run the `diagnose sys csf authorization pending-list` command in the root FortiGate to show the downstream FortiGate pending for root FortiGate authorization:

```
Edge # diagnose sys csf authorization pending-list
Serial                  IP Address      HA-Members     Path
------------------------------------------------------------------------------------
FG201ETK18902514        0.0.0.0                        FG3H1E5818900718:FG201ETK18902514
```

2. Run the `diagnose sys csf downstream` command in the root or middle FortiGate to show the downstream FortiGates after they join Security Fabric:

```
Edge # diagnose sys csf downstream
1:     FG201ETK18902514 (192.168.200.10) Management-IP: 0.0.0.0 Management-port:0
parent: FG3H1E5818900718
        path:FG3H1E5818900718:FG201ETK18902514
        data received: Y downstream intf:wan1 upstream intf:port11 admin-port:443
        authorizer:FG3H1E5818900718
2:     FGT81ETK18002246 (192.168.10.10) Management-IP: 0.0.0.0 Management-port:0 parent:
FG3H1E5818900718
        path:FG3H1E5818900718:FGT81ETK18002246
        data received: Y downstream intf:wan1 upstream intf:port10 admin-port:443
        authorizer:FG3H1E5818900718
3:     FG101ETK18002187 (192.168.135.10) Management-IP: 0.0.0.0 Management-port:0
parent: FG201ETK18902514
        path:FG3H1E5818900718:FG201ETK18902514:FG101ETK18002187
        data received: Y downstream intf:wan2 upstream intf:port12 admin-port:443
        authorizer:FG3H1E5818900718
```

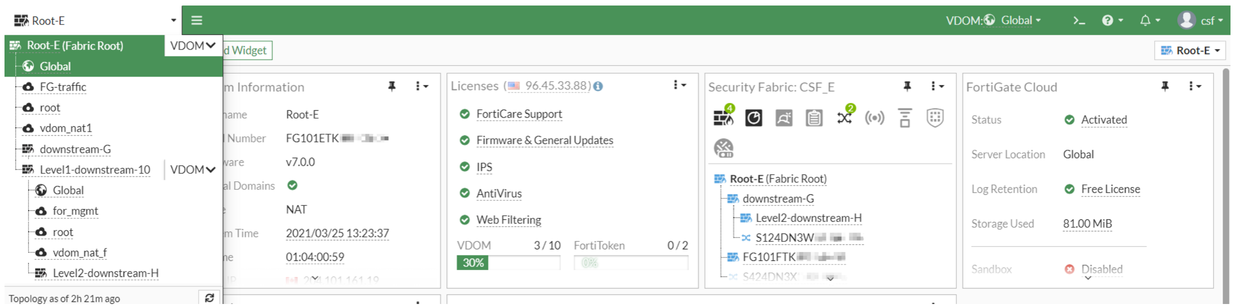
3. Run the `diagnose sys csf upstream` command in any downstream FortiGate to show the upstream FortiGate after downstream FortiGate joins Security Fabric:

```
Marketing # diagnose sys csf upstream
Upstream Information:
Serial Number:FG3H1E5818900718
IP:192.168.200.2
Connecting interface:wan1
Connection status:Authorized
```

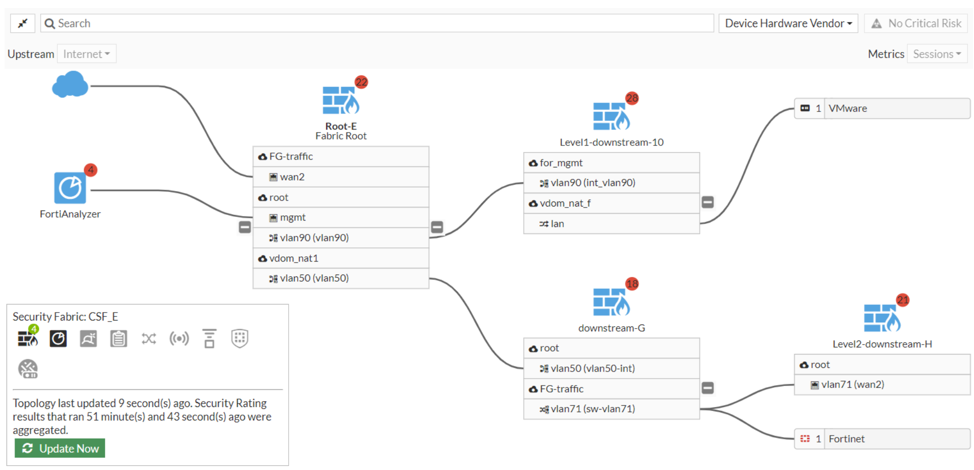
Deploying the Security Fabric in a multi-VDOM environment

A Security Fabric can be enabled in multi-VDOM environments. This allows access to all of the Security Fabric features, including automation, security rating, and topologies, across the VDOM deployment.

- Users can navigate to downstream FortiGate devices and VDOMs directly from the root FortiGate using the Fabric selection menu.



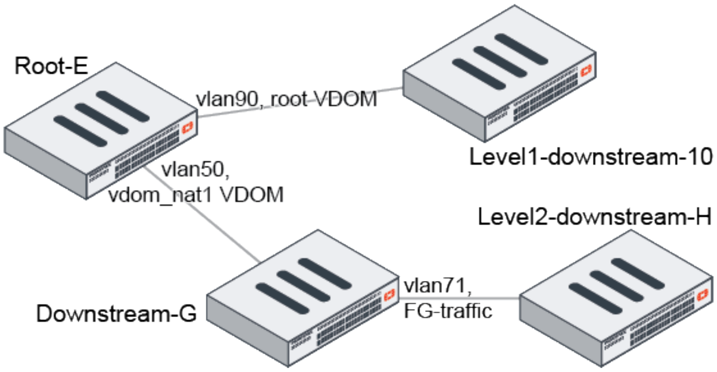
- The logical topology shows all of the configured VDOMs.



- Security rating reports include results for all of the configured VDOMs as well the entire Fabric.

Downstream FortiGate devices must connect to the upstream FortiGate from its management VDOM.

Topology



In this topology, there is a root FortiGate with three FortiGates connected through two different VDOMs. The root FortiGate is able to manage all devices running in multi-VDOM mode.

This example assumes multi-VDOM mode is already configured on each FortiGate, and that FortiAnalyzer logging is configured on the root FortiGate (see Configuring FortiAnalyzer on page 2215 and Configuring the root FortiGate and downstream FortiGates on page 2208 for more details).

To enable multi-VDOM mode:

```
config system global
    set vdom-mode multi-vdom
end
```

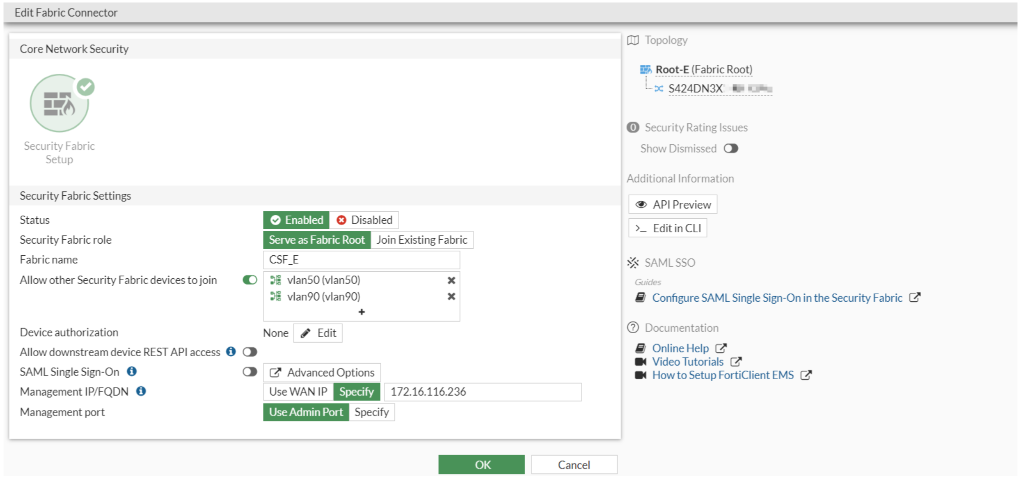
Device configurations

Root FortiGate (Root-E)

The Security Fabric is enabled, and configured so that downstream interfaces from all VDOMs can allow other Security Fabric devices to join.

To configure Root-E in the GUI:

1. Go to *Security Fabric > Fabric Connectors* and double-click the *Security Fabric Setup* card.
2. Ensure that the *Status* is *Enabled* and the *Security Fabric role* is set to *Serve as Fabric Root*.
3. Enable *Allow other Security Fabric devices to join* and click the + to add the interfaces (vlan50 and vlan90) from the vdom_nat1 and root VDOMs.



4. Configure the other settings as needed.
5. Click *OK*.

To configure Root-E in the CLI:

1. Enable the Security Fabric:

```
config system csf
    set status enable
    set group-name "CSF_E"
end
```

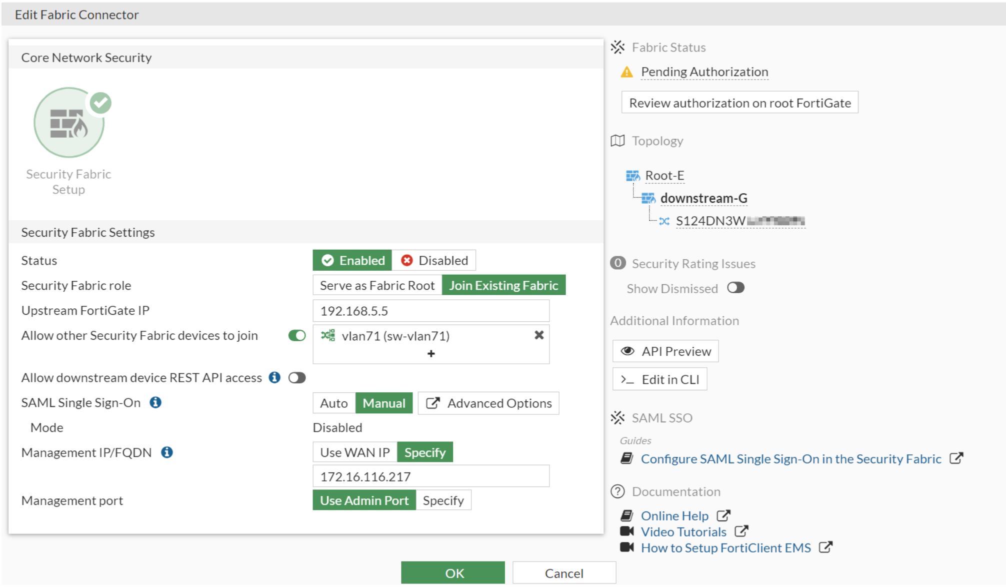
2. Configure the interfaces:

```
config system interface
    edit "vlan50"
        set vdom "vdom_nat1"
        ...
        set allowaccess ping https ssh http fgfm fabric
        ...
    next
    edit "vlan90"
        set vdom "root"
        ...
        set allowaccess ping https ssh http fgfm fabric
        ...
    next
end
```

Downstream FortiGate 1 (Downstream-G)

To configure Downstream-G in the GUI:

1. Go to *Security Fabric > Fabric Connectors* and double-click the *Security Fabric Setup* card.
2. For *Status*, select *Enabled* and set the role to *Join Existing Fabric*.
3. Enter the *Upstream FortiGate IP*, which is the IP of the root FortiGate vdom_nat1 interface (192.168.5.5). Downstream-G must use the interface from the management VDOM to connect to the upstream FortiGate IP.
4. Enable *Allow other Security Fabric devices to join* and click the + to add the downstream interface (sw-vlan71) from the FG-traffic VDOM.



5. Configure the other settings as needed.
6. Click *OK*.

To configure Downstream-G in the CLI:

1. Enable the Security Fabric:

```
config system csf
    set status enable
    set upstream-ip 192.168.5.5
end
```

2. Configure the interfaces:

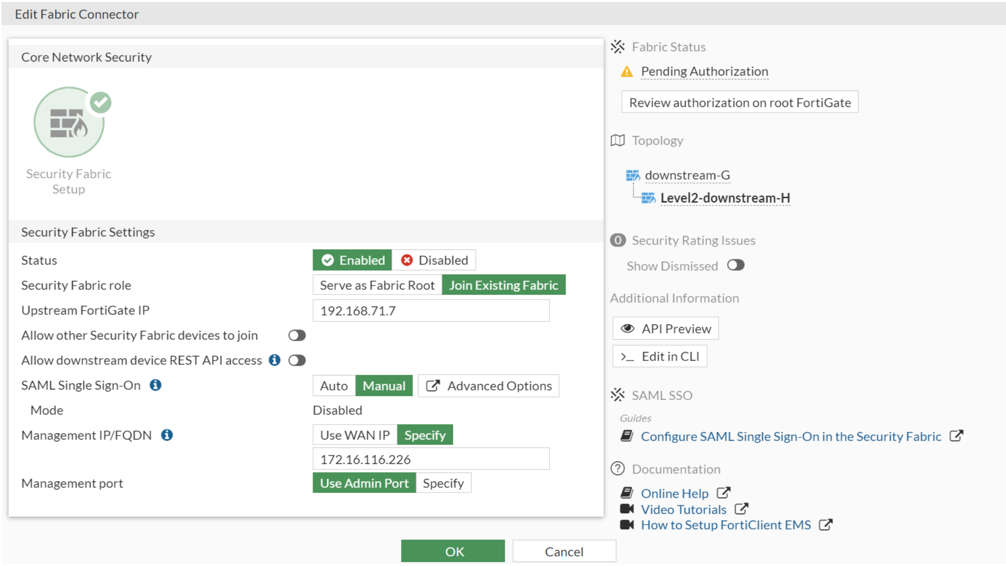
```
config system interface
    edit "sw-vlan71"
        set vdom "FG-traffic"
        ...
        set allowaccess ping https ssh http fgfm fabric
        ...
    next
end
```

Downstream FortiGate 2 (Level2-downstream-H)

To configure Level2-downstream-H in the GUI:

1. Go to *Security Fabric > Fabric Connectors* and double-click the *Security Fabric Setup* card.
2. For *Status*, select *Enabled* and set the role to *Join Existing Fabric*.

3. Enter the *Upstream FortiGate IP*, which is the IP of the root VDOM on Downstream-G (192.168.71.7).



4. Configure the other settings as needed.
5. Click *OK*.

To configure Level2-downstream-H in the CLI:

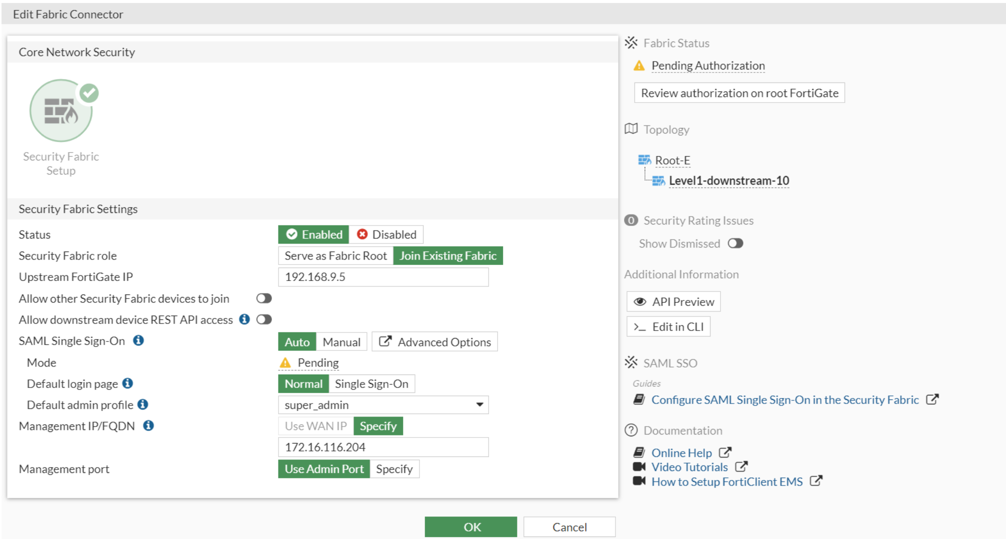
```
config system csf
    set status enable
    set upstream-ip 192.168.71.7
end
```

Downstream FortiGate 3 (Level1-downstream-10)

To configure Level1-downstream-10 in the GUI:

1. Go to *Security Fabric* > *Fabric Connectors* and double-click the *Security Fabric Setup* card.
2. For *Status*, select *Enabled* and set the role to *Join Existing Fabric*.

3. Enter the *Upstream FortiGate IP*, which is the IP of the root VDOM on Root-E (192.168.9.5).



4. Configure the other settings as needed.
5. Click *OK*.

To configure Level1-downstream-10 in the CLI:

```
config system csf
    set status enable
    set upstream-ip 192.168.9.5
end
```

Device authorization and verification

To authorize the downstream devices on the root FortiGate:

1. On Root-E, go to *Security Fabric > Fabric Connectors*.
2. In the topology tree, click the highlighted serial number and select *Authorize* for each downstream FortiGate. Once all the devices are authorized, the physical topology page shows the root and downstream FortiGates. The logical topology page shows the root and downstream FortiGates connected to interfaces in their corresponding VDOMs.

Synchronizing objects across the Security Fabric

When the Security Fabric is enabled, various objects such as addresses, services, and schedules are synced from the upstream FortiGate to all downstream devices by default. FortiOS has the following settings for object synchronization across the Security Fabric:

- Set object synchronization (`fabric-object-unification`) to `default` or `local` on a downstream device.
- Set a per object option to toggle whether the specific Fabric object will be synchronized or not. After upgrading from 6.4.3, this option is disabled for supported Fabric objects. The synchronized Fabric objects are kept as locally created objects on downstream FortiGates.
- Define the number of task workers to handle synchronizations.

The firewall object synchronization wizard helps identify objects that are not synchronized and resolves any conflicts. A warning message appears in the topology tree if there is a conflict.

Summary of CLI commands

To configure object synchronization:

```
config system csf
    set fabric-object-unification {default | local}
    set configuration-sync {default | local}
    set fabric-workers <integer>
end
```

Parameter	Description
`fabric-object-unification`	*default*: Global CMDB objects will be synchronized in the Security Fabric. *local*: Global CMDB objects will not be synchronized to and from this device. This command is available on the root FortiGate. If set to local, the device does not synchronize objects from the root, but will send the synchronized objects downstream.
`configuration-sync`	*default*: Synchronize configuration for FortiAnalyzer, FortiSandbox, and Central Management to root node. *local*: Do not synchronize configuration with root node. If downstream FortiGates are set to local, the synchronized objects from the root to downstream are not applied locally. However, the downstream FortiGate will send the configuration to lower FortiGates.
`fabric-workers`	Define how many task worker process are created to handle synchronizations (1-4, default = 2). The worker processes dies if there is no task to perform after 60 seconds.

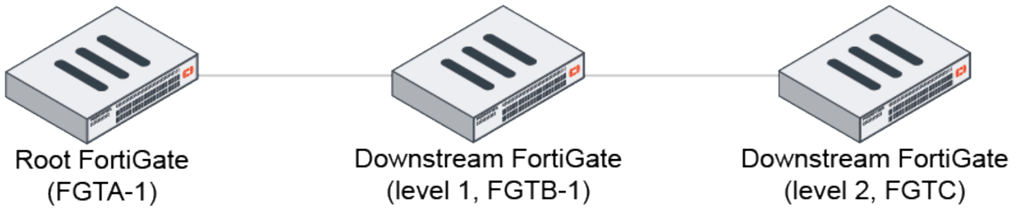
The per object setting can be configured on the root FortiGate as follows:

```
config firewall <object>
    edit <name>
        set fabric-object {enable | disable}
    next
end
```

Where:

- `<object>` is one of the following: `address`, `address6`, `addrgrp`, `addrgrp6`, `service category`, `service custom`, `service group`, `schedule group`, `schedule onetime`, or `schedule recurring`.
- Enabling `fabric-object` sets the object as a Security Fabric-wide global object that is synchronized to downstream FortiGates.
- Disabling `fabric-object` sets the object as local to this Security Fabric member.
- If a device in the Fabric is in multi-VDOM mode, the GUI will not display the Fabric synchronization option. Even if this is enabled in the CLI, the object will not be synchronized to any downstream devices.

Sample topology



In this Security Fabric, the root FortiGate (FGTA-1) has `fabric-object-unification` set to `default` so the Fabric objects can be synchronized to the downstream FortiGate. The level 1 downstream FortiGate (FGTB-1) has `configuration-sync` set to `local`, so it will not apply the synchronized objects locally. The level 2 downstream FortiGate (FGTC) has `configuration-sync` set to `default`, so it will apply the synchronized objects locally.

In this example, firewall addresses and address groups are used. Other supported Fabric objects have the same behaviors. The following use cases illustrate common synchronization scenarios:

- If no conflicts exist, firewall addresses and address groups can be synchronized to downstream FortiGates (see example below).
- If a conflict exists between the root and downstream FortiGates, it can be resolved with the conflict resolution wizard. After the conflict is resolved, the firewall addresses and address groups can be synchronized to downstream FortiGates (see example below).
- If `set fabric-object` (*Fabric synchronization* option in the GUI) is disabled for firewall addresses and address groups on the root FortiGate, they will not be synchronized to downstream FortiGates (see example below).

To configure the FortiGates used in this example:

```
FGTA-1 # config system csf
    set status enable
    set group-name "fabric"
    set fabric-object-unification default
    ...
end
```

```
FGTB-1 # config system csf
    set status enable
    set upstream-ip 10.2.200.1
    set configuration-sync local
    ...
end
```

```
FGTC # config system csf
    set status enable
    set upstream-ip 192.168.7.2
    set configuration-sync default
    ...
end
```

To synchronize a firewall address and address group in the Security Fabric:

1. Configure the firewall address on the root FortiGate:

```
FGTA-1 # config firewall address
    edit "add_subnet_1"
```

```
        set fabric-object enable
        set subnet 22.22.22.0 255.255.255.0
    next
end
```

2. Configure the address group on the root FortiGate:

```
FGTA-1 # config firewall addrgrp
    edit "group_subnet_1"
        set member "add_subnet_1"
        set fabric-object enable
    next
end
```

3. Check the firewall address and address group on the downstream FortiGates:

```
FGTB-1 # show firewall address add_subnet_1
entry is not found in table
```

```
FGTB-1 # show firewall addrgrp group_subnet_1
entry is not found in table
```

The synchronized objects are not applied locally on this FortiGate because `configuration-sync` is set to `local`.

```
FGTC # show firewall address add_subnet_1
config firewall address
    edit "add_subnet_1"
        set uuid 378a8094-34cb-51eb-ce40-097f298fcfdc
        set fabric-object enable
        set subnet 22.22.22.0 255.255.255.0
    next
end
```

```
FGTC # show firewall addrgrp group_subnet_1
config firewall addrgrp
    edit "group_subnet_1"
        set uuid 4d7a8a52-34cb-51eb-fce7-d93f76915319
        set member "add_subnet_1"
        set color 19
        set fabric-object enable
    next
end
```

The objects are synchronized on this FortiGate because `configuration-sync` is set to `default`.

To resolve a firewall address and address group conflict in the Security Fabric:

1. On FGTC, create a firewall address:
   a. Go to *Policy & Objects > Addresses* and click *Create New > Address*.
   b. Configure the following:

Name	sync_add_1
IP/Netmask	33.33.33.0 255.255.255.0

   c. Click *OK*.

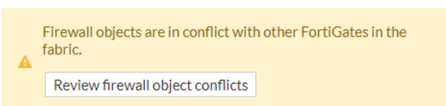
2. On FGTA-1 (Fabric root), create the firewall address with same name but a different subnet:
   a. Go to *Policy & Objects* > *Addresses* and click *Create New* > *Address*.
   b. Configure the following:

Name	sync_add_1
IP/Netmask	11.11.11.0 255.255.255.0
Fabric synchronization	Enable

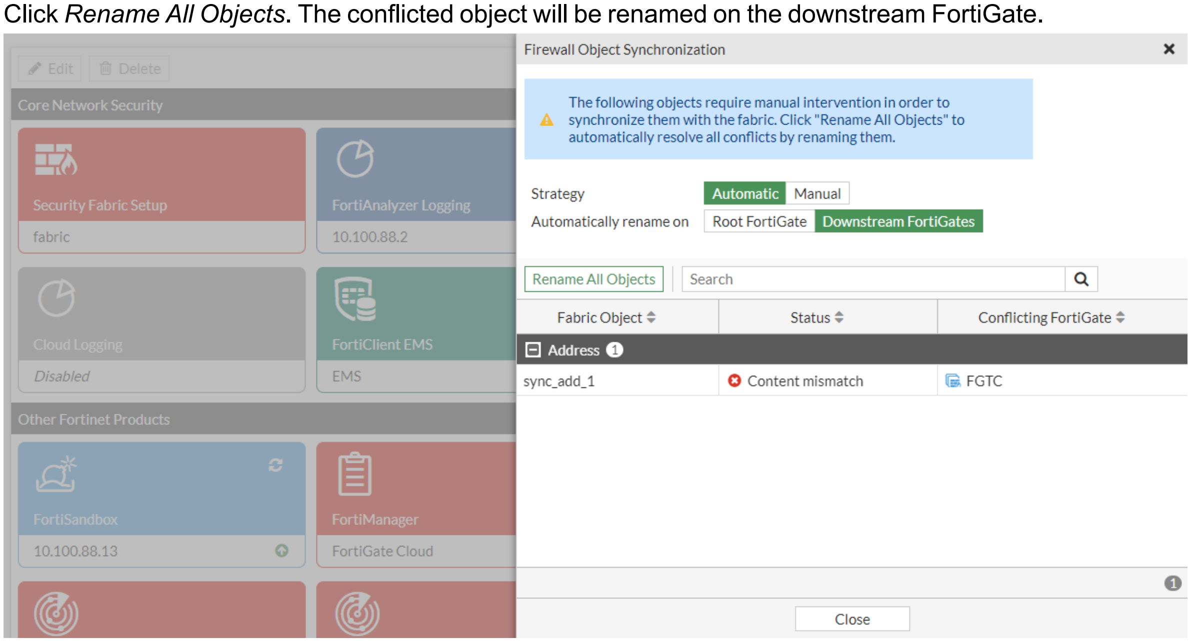
   c. Click *OK*.
3. Add the address to a different address group than what is configured on FGTC:
   a. Go to *Policy & Objects* > *Addresses* and click *Create New* > *Address Group*.
   b. Configure the following:

Name	sync_group4
Members	sync_add_1
Fabric synchronization	Enable

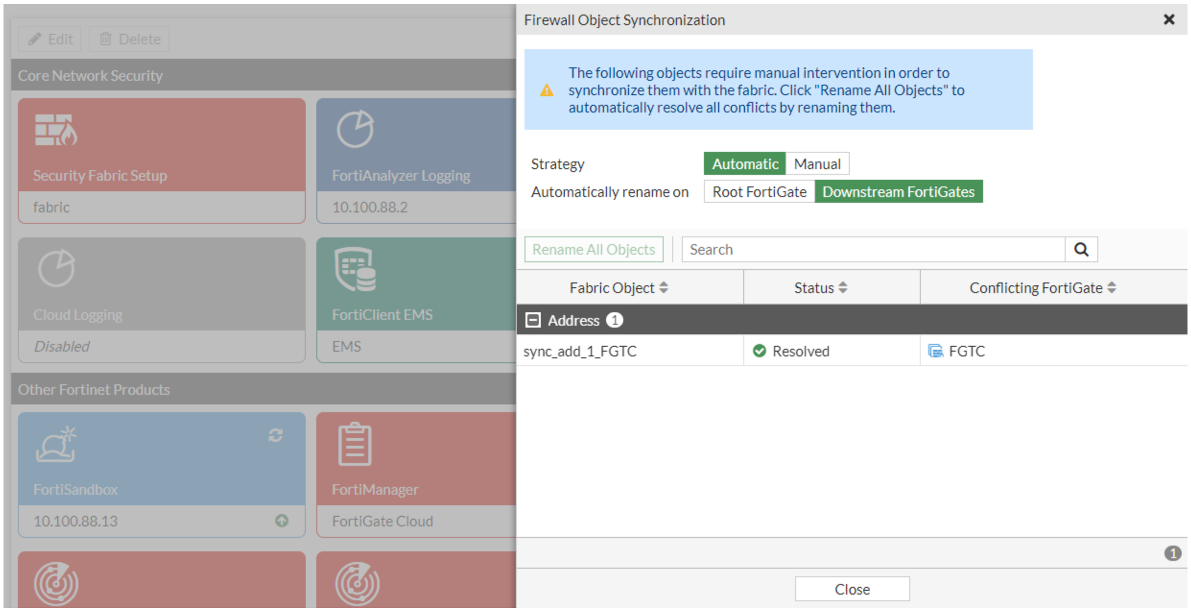
   c. Click *OK*.
4. Go to *Security Fabric* > *Fabric Connectors*. In the topology tree, there is a message that *Firewall objects are in conflict with other FortiGates in the fabric*.



5. Resolve the conflict:
   a. Click *Review firewall object conflicts*. The *Firewall Object Synchronization* pane opens.
   b. Click *Rename All Objects*. The conflicted object will be renamed on the downstream FortiGate.

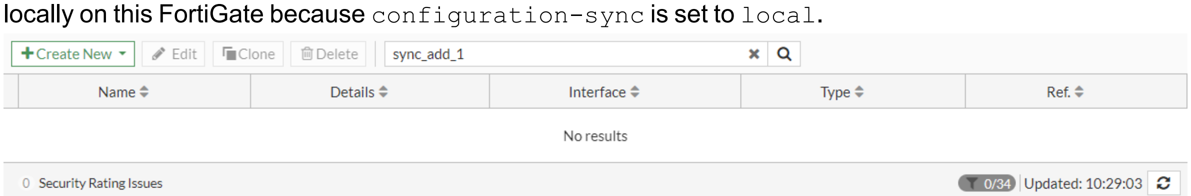


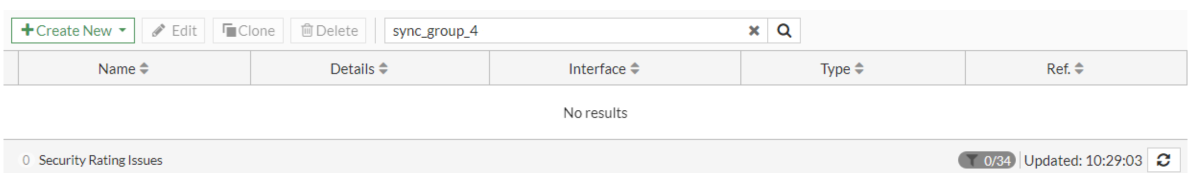
c. The conflict is resolved. Click *Close* to exit the *Firewall Object Synchronization* pane.



d. The topology tree no longer indicates there is a conflict.

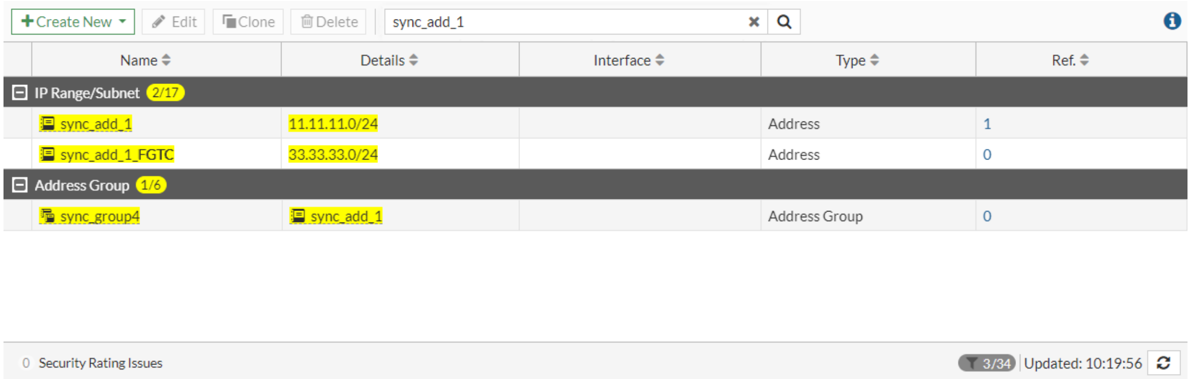
6. Verify the results on the downstream FortiGates:
   a. On FGTB-1, go to *Policy & Objects > Addresses*.
   b. Search for *sync_add_1* and *sync_group4*. No results are found. The synchronized objects are not applied locally on this FortiGate because `configuration-sync` is set to `local`.





   c. On FGTC, go to *Policy & Objects > Addresses*.
   d. Search for *sync_add_1*. The original firewall address *sync_add_1* was renamed to *sync_add_1_FGTC* by resolving the conflict on FGTA-1. The address *sync_add_1* and address group *sync_group4* are synchronized

from FGTA-1.



To disable Fabric synchronization on the root FortiGate in the GUI:

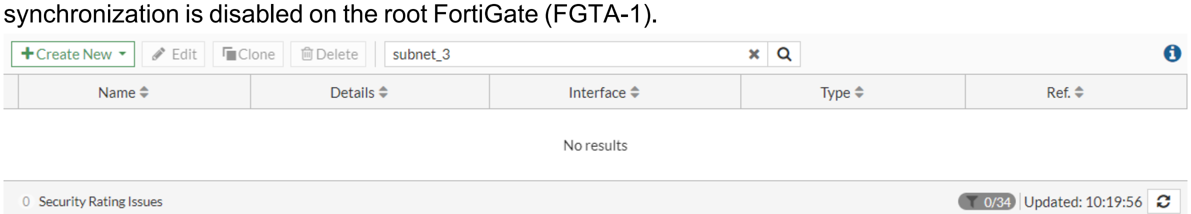
1. On FGTA-1, create a firewall address:
   a. Go to *Policy & Objects > Addresses* and click *Create New > Address*.
   b. Configure the following:

Name	add_subnet_3
IP/Netmask	33.33.33.0 255.255.255.0
Fabric synchronization	Disable

   c. Click *OK*.
2. Create the firewall address group and add the address:
   a. Go to *Policy & Objects > Addresses* and click *Create New > Address Group*.
   b. Configure the following:

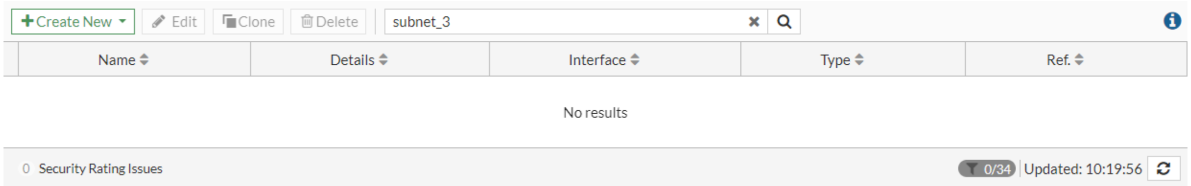
Name	group_subnet_3
Members	add_subnet_3
Fabric synchronization	Disable

   c. Click *OK*.
3. On FGTB-1, go to *Policy & Objects > Addresses* and search for *subnet_3*. No results are found because Fabric synchronization is disabled on the root FortiGate (FGTA-1).



4. On FGTC, go to *Policy & Objects > Addresses* and search for *subnet_3*. No results are found because Fabric

synchronization is disabled on the root FortiGate (FGTA-1).



To disable Fabric synchronization on the root FortiGate in the CLI:

1. Configure the firewall address on the root FortiGate:

```
FGTA-1 # config firewall address
    edit "add_subnet_3"
        set subnet 33.33.33.0 255.255.255.0
        set fabric-object disable
    next
end
```

2. Configure the address group on the root FortiGate:

```
FGTA-1 # config firewall addrgrp
    edit "group_subnet_3"
        set member "add_subnet_3"
        set fabric-object disable
    next
end
```

3. Check the firewall address and address group on the downstream FortiGates:

```
FGTB-1 # show firewall address add_subnet_3
entry is not found in table
```

```
FGTB-1 # show firewall addrgrp group_subnet_3
entry is not found in table
```

```
FGTC # show firewall address add_subnet_3
entry is not found in table
```

```
FGTC # show firewall addrgrp group_subnet_3
entry is not found in table
```

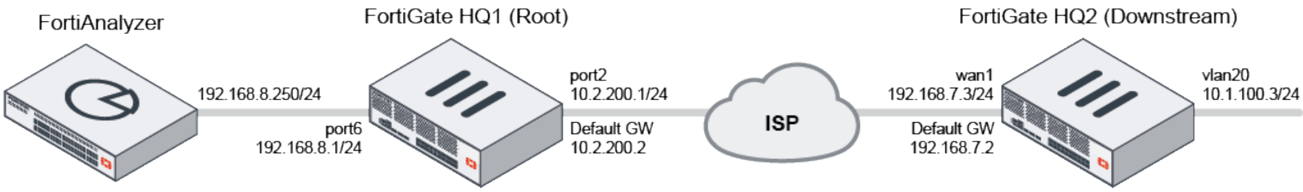
The objects are not synchronized from the root FortiGate (FGTA-1) because the `fabric-object` setting is disabled.

Security Fabric over IPsec VPN

This is an example of configuring Security Fabric over IPsec VPN.

Sample topology

This sample topology shows a downstream FortiGate (HQ2) connected to the root FortiGate (HQ1) over IPsec VPN to join Security Fabric.



Sample configuration

To configure the root FortiGate (HQ1):

1. Configure interface:
   - **a.** In the root FortiGate (HQ1), go to *Network > Interfaces*.
   - **b.** Edit *port2*:
     - Set *Role* to *WAN*.
     - For the interface connected to the Internet, set the *IP/Network Mask* to *10.2.200.1/255.255.255.0*
   - **c.** Edit *port6*:
     - Set *Role* to *DMZ*.
     - For the interface connected to FortiAnalyzer, set the *IP/Network Mask* to *192.168.8.250/255.255.255.0*
2. Configure the static route to connect to the Internet:
   - **a.** Go to *Network > Static Routes* and click *Create New* or *Create New > IPv4 Static Route*.
     - Set *Destination* to *0.0.0.0/0.0.0.0*.
     - Set *Interface* to *port2*.
     - Set *Gateway Address* to *10.2.200.2*.
   - **b.** Click *OK*.
3. Configure IPsec VPN:
   - **a.** Go to *VPN > IPsec Wizard*.
     - Set *Name* to *To-HQ2*.
     - Set *Template Type* to *Custom*.
     - Click *Next*.
     - Set *Authentication* to *Method*.
     - Set *Pre-shared Key* to *123456*.
   - **b.** Leave all other fields in their default values and click *OK*.
4. Configure the IPsec VPN interface IP address which will be used to form Security Fabric:
   - **a.** Go to *Network > Interfaces*.
   - **b.** Edit *To-HQ2*:
     - Set *Role* to *LAN*.
     - Set the *IP/Network Mask* to *10.10.10.1/255.255.255.255*.
     - Set *Remote IP/Network Mask* to *10.10.10.3/255.255.255.0*.

5. Configure IPsec VPN local and remote subnet:
   a. Go to *Policy & Objects > Addresses*.
   b. Click *Create New*
      - Set *Name* to *To-HQ2_remote_subnet_2*.
      - Set *Type* to *Subnet*.
      - Set *IP/Network Mask* to *10.10.10.3/32*.
   c. Click *OK*.
   d. Click *Create New*
      - Set *Name* to *To-HQ2_local_subnet_1*.
      - Set *Type* to *Subnet*.
      - Set *IP/Network Mask* to *192.168.8.0/24*.
   e. Click *OK*.
   f. Click *Create New*
      - Set *Name* to *To-HQ2_remote_subnet_1*.
      - Set *Type* to *Subnet*.
      - Set *IP/Network Mask* to *10.1.100.0/24*.
   g. Click *OK*.
6. Configure IPsec VPN static routes:
   a. Go to *Network > Static Routes*
   b. Click *Create New* or *Create New > IPv4 Static Route*.
      - For *Named Address*, select *Type* and select *To-HQ2_remote_subnet_1*.
      - Set *Interface* to *To-HQ2*.

      Click *OK*.
   c. Click *Create New* or *Create New > IPv4 Static Route*.
      - For *Named Address*, select *Type* and select *To-HQ2_remote_subnet_1*.
      - Set *Interface* to *Blackhole*.
      - Set *Administrative Distance* to *254*.
   d. Click *OK*.
7. Configure IPsec VPN policies:
   a. Go to *Policy & Objects > Firewall Policy*
   b. Click *Create New*.
      - Set *Name* to *vpn_To-HQ2_local*.
      - Set *Incoming Interface* to *port6*.
      - Set *Outgoing Interface* to *To-HQ2*.
      - Set *Source* to *To-HQ2_local_subnet_1*.
      - Set *Destination* to *To-HQ2_remote_subnet_1*.
      - Set *Schedule* to *Always*.
      - Set *Service* to *All*.
      - Disable *NAT*.
   c. Click *OK*.
   d. Click *Create New*.
      - Set *Name* to *vpn_To-HQ2_remote*.
      - Set *Incoming Interface* to *To-HQ2*.
      - Set *Outgoing Interface* to *port6*.
      - Set *Source* to *To-HQ2_remote_subnet_1, To-HQ2_remote_subnet_2*.

- Set *Destination* to *To-HQ2_local_subnet_1*.
- Set *Schedule* to *Always*.
- Set *Service* to *All*.
- Enable *NAT*.
- Set *IP Pool Configuration* to *Use Outgoing Interface Address*.

e. Click *OK*.

8. Configure Security Fabric:
   a. Go to *Security Fabric > Fabric Connectors* and double-click the *Security Fabric Setup* card.
   b. For *Status*, click *Enable*.
      After *FortiGate Telemetry* is enabled, FortiAnalyzer automatically enables *Logging* and *Upload* is set to *Real Time*.
   c. Set the *Security Fabric role* to *Serve as Fabric Root*. The FortiAnalyzer settings can be configured.
   d. Enter the FortiAnalyzer IP (*192.168.8.250*).
   e. Click *OK*. The FortiAnalyzer serial number is verified.
   f. Enter a *Fabric name*, such as *Office-Security-Fabric*.
   g. Ensure *Allow other Security Fabric devices to join* is enabled and add VPN interface *To-HQ2*.
   h. Click *OK*.

To configure the downstream FortiGate (HQ2):

1. Configure interface:
   a. Go to *Network > Interfaces*.
   b. Edit interface *wan1*:
      - Set *Role* to *WAN*.
      - For the interface connected to the Internet, set the *IP/Network Mask* to *192.168.7.3/255.255.255.0*.
   c. Edit interface *vlan20*:
      - Set *Role* to *LAN*.
      - For the interface connected to local endpoint clients, set the *IP/Network Mask* to *10.1.100.3/255.255.255.0*.
2. Configure the static route to connect to the Internet:
   a. Go to *Network > Static Routes* and click *Create New* or *Create New > IPv4 Static Route*.
      - Set *Destination* to *0.0.0.0/0.0.0.0*.
      - Set *Interface* to *wan1*.
      - Set *Gateway Address* to *192.168.7.2*.
   b. Click *OK*.
3. Configure IPsec VPN:
   a. Go to *VPN > IPsec Wizard*.
      - Set *VPN Name* to *To-HQ1*.
      - Set *Template Type* to *Custom*.
      - Click *Next*.
      - In the *Network IP Address*, enter *10.2.200.1*.
      - Set *Interface* to *wan1*.
      - Set *Authentication* to *Method*.
      - Set *Pre-shared Key* to *123456*.
   b. Leave all other fields in their default values and click *OK*.

4. Configure the IPsec VPN interface IP address which will be used to form Security Fabric:
   a. Go to *Network > Interfaces*.
   b. Edit *To-HQ1*:
      - Set *Role* to *WAN*.
      - Set the *IP/Network Mask* to *10.10.10.3/255.255.255.255*.
      - Set *Remote IP/Network Mask* to *10.10.10.1/255.255.0.0*.
5. Configure IPsec VPN local and remote subnet:
   a. Go to *Policy & Objects > Addresses*.
   b. Click *Create New*
      - Set *Name* to *To-HQ1_local_subnet_1*.
      - Set *Type* to *Subnet*.
      - Set *IP/Network Mask* to *10.1.100.0/24*.
   c. Click *OK*.
   d. Click *Create New*
      - Set *Name* to *To-HQ1_remote_subnet_1*.
      - Set *Type* to *Subnet*.
      - Set *IP/Network Mask* to *192.168.8.0/24*.
   e. Click *OK*.
6. Configure IPsec VPN static routes:
   a. Go to *Network > Static Routes* and click *Create New* or *Create New > IPv4 Static Route*.
      - For *Named Address*, select *Type* and select *To-HQ1_remote_subnet_1*.
      - Set *Interface* to *To-HQ1*.
   b. Click *OK*.
   c. Click *Create New* or *Create New > IPv4 Static Route*.
      - For *Named Address*, select *Type* and select *To-HQ1_remote_subnet_1*.
      - Set *Interface* to *Blackhole*.
      - Set *Administrative Distance* to *254*.
   d. Click *OK*.
7. Configure IPsec VPN policies:
   a. Go to *Policy & Objects > Firewall Policy* and click *Create New*.
      - Set *Name* to *vpn_To-HQ1_local*.
      - Set *Incoming Interface* to *vlan20*.
      - Set *Outgoing Interface* to *To-HQ1*.
      - Set *Source* to *To-HQ1_local_subnet_1*.
      - Set *Destination* to *To-HQ1_remote_subnet_1*.
      - Set *Schedule* to *Always*.
      - Set *Service* to *All*.
      - Disable *NAT*.
   b. Click *OK*.
   c. Click *Create New*.
      - Set *Name* to *vpn_To-HQ1_remote*.
      - Set *Incoming Interface* to *To-HQ1*.
      - Set *Outgoing Interface* to *vlan20*.
      - Set *Source* to *To-HQ1_remote_subnet_1*.
      - Set *Destination* to *-HQ1_local_subnet_1*.

- Set *Schedule* to *Always*.
- Set *Service* to *All*.
- Disable *NAT*.

d. Click *OK*.

8. Configure Security Fabric:

a. Go to *Security Fabric > Fabric Connectors* and double-click the *Security Fabric Setup* card.

b. For *Status*, click *Enable*.
FortiAnalyzer automatically enables logging. FortiAnalyzer settings will be retrieved when the downstream FortiGate connects to the root FortiGate.

c. Set the *Security Fabric role* to *Join Existing Fabric*.

d. Set the *Upstream FortiGate IP* to *10.10.10.1*.

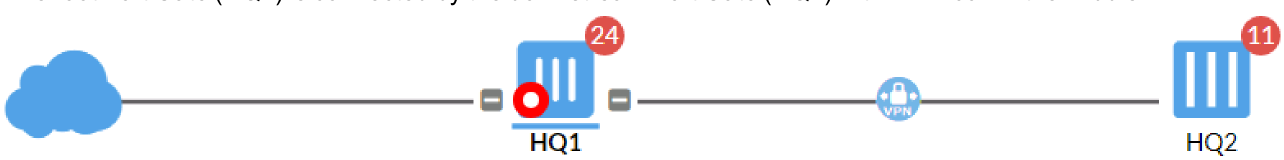
e. Click *OK*.

To authorize the downstream FortiGate (HQ2) on the root FortiGate (HQ1):

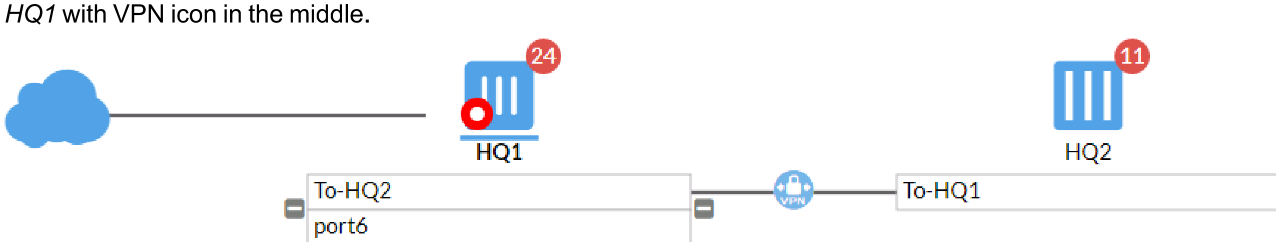
1. In the root FortiGate (HQ1), go to *Security Fabric > Fabric Connectors* and double-click the *Security Fabric Setup* card.
   The *Topology* tree highlights the connected FortiGate (HQ2) with the serial number and asks you to authorize the highlighted device.
2. Select the highlighted FortiGates and select *Authorize*.
   After authorization, the downstream FortiGate (HQ2) appears in the *Topology* tree in the *Security Fabric > Fabric Connectors > Security Fabric Setup* page. This means the downstream FortiGate (HQ2) has successfully joined the Security Fabric.

To check Security Fabric over IPsec VPN:

1. On the root FortiGate (HQ1), go to *Security Fabric > Physical Topology*.
   The root FortiGate (HQ1) is connected by the downstream FortiGate (HQ2) with VPN icon in the middle.



2. On the root FortiGate (HQ1), go to *Security Fabric > Logical Topology*.
   The root FortiGate (HQ1) VPN interface *To-HQ2* is connected by downstream FortiGate (HQ2) VPN interface *To-HQ1* with VPN icon in the middle.



To run diagnose commands:

1. Run the `diagnose sys csf authorization pending-list` command in the root FortiGate (HQ1) to show the downstream FortiGate pending for root FortiGate authorization:

```
HQ1 # diagnose sys csf authorization pending-list
Serial                  IP Address      HA-Members
Path
------------------------------------------------------------------------------------
FG101ETK18002187        0.0.0.0
FG3H1E5818900718:FG101ETK18002187
```

2. Run the `diagnose sys csf downstream` command in the root FortiGate (HQ1) to show the downstream FortiGate (HQ2) after it joins Security Fabric:

```
HQ1 # diagnose sys csf downstream
 1:     FG101ETK18002187 (10.10.10.3) Management-IP: 0.0.0.0 Management-port:0 parent:
FG3H1E5818900718
        path:FG3H1E5818900718:FG101ETK18002187
        data received: Y downstream intf:To-HQ1 upstream intf:To-HQ2 admin-port:443
        authorizer:FG3H1E5818900718
```

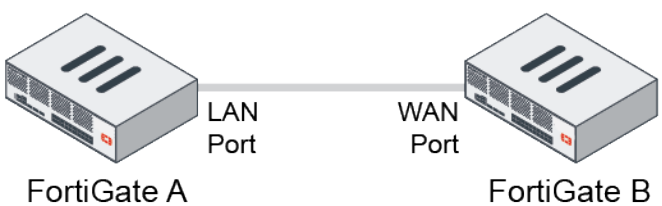
3. Run the `diagnose sys csf upstream` command in the downstream FortiGate (HQ2) to show the root FortiGate (HQ1) after the downstream FortiGate joins Security Fabric:

```
HQ2 # diagnose sys csf upstream
Upstream Information:
Serial Number:FG3H1E5818900718
IP:10.10.10.1
Connecting interface:To-HQ1
Connection status:Authorized
```

Leveraging LLDP to simplify Security Fabric negotiation

LLDP reception is enabled on WAN interfaces, which prompts FortiGates that are joining the Security Fabric if the upstream FortiGate asks.

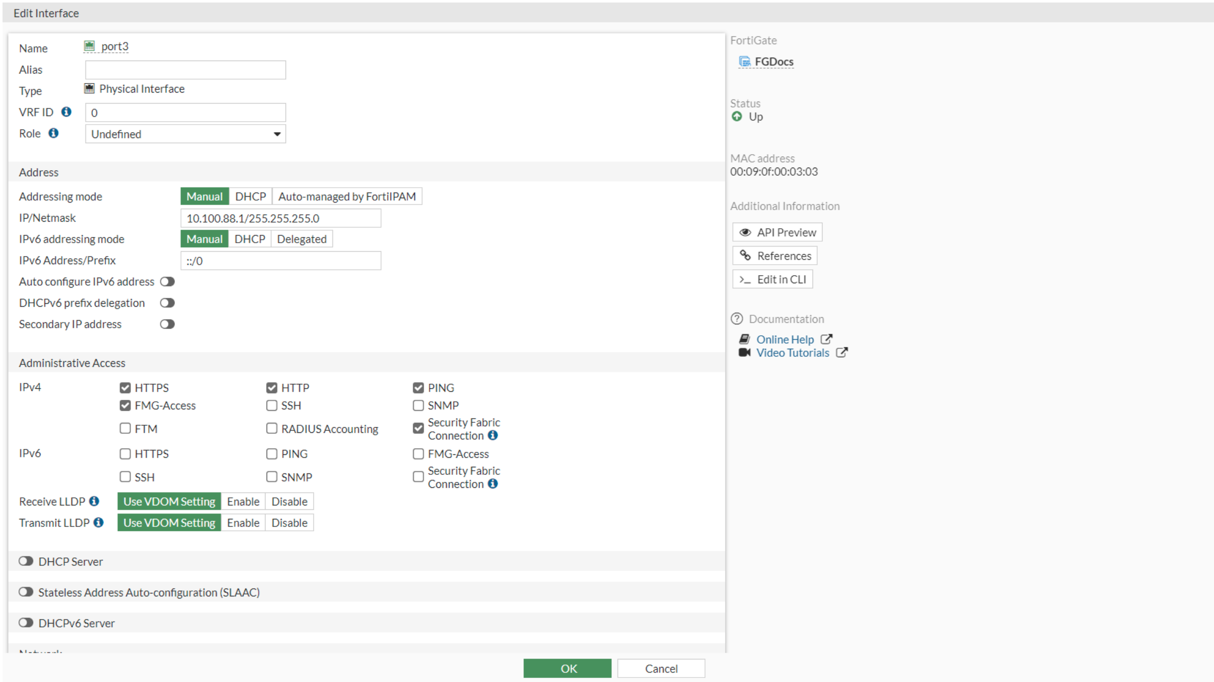
- If the interface role is undefined, LLDP reception and transmission inherit settings from the VDOM.
- If the interface role is WAN, LLDP reception is enabled.
- If the interface role is LAN, LLDP transmission is enabled.



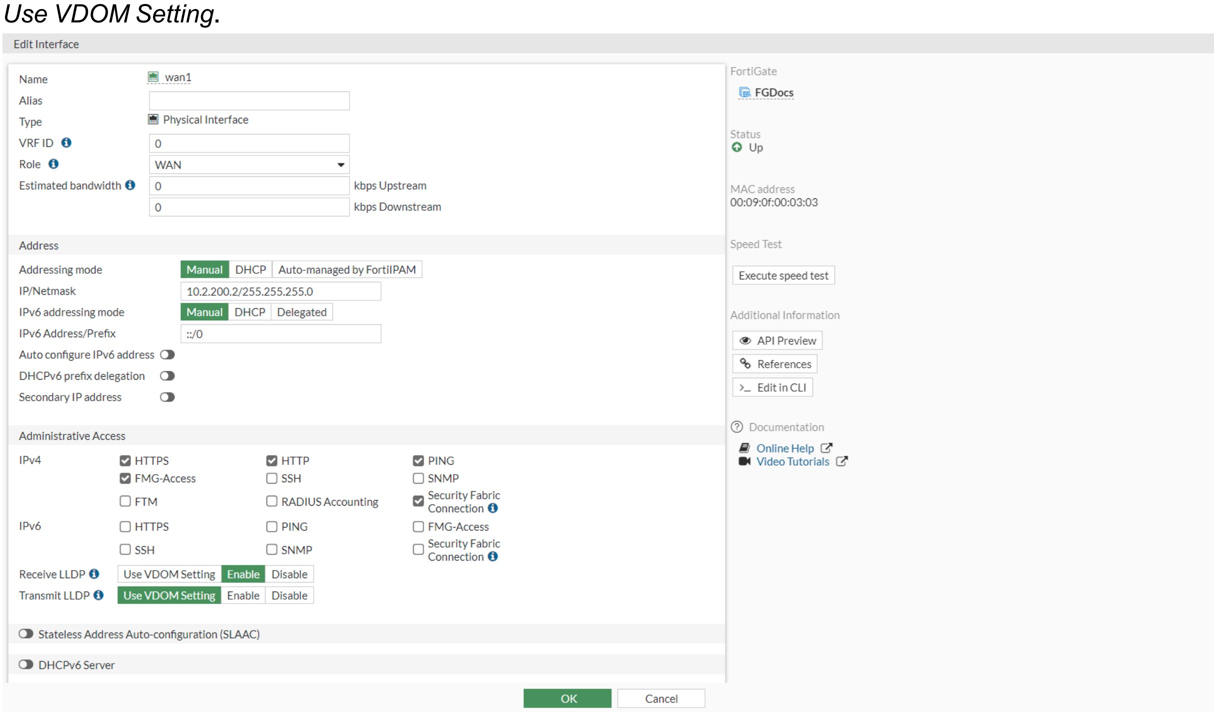
When a FortiGate B's WAN interface detects that FortiGate A's LAN interface is immediately upstream (through the default gateway), and FortiGate A has Security Fabric enabled, FortiGate B will show a notification on the GUI asking to join the Security Fabric.

To configure LLDP reception and join a Security Fabric in the GUI:

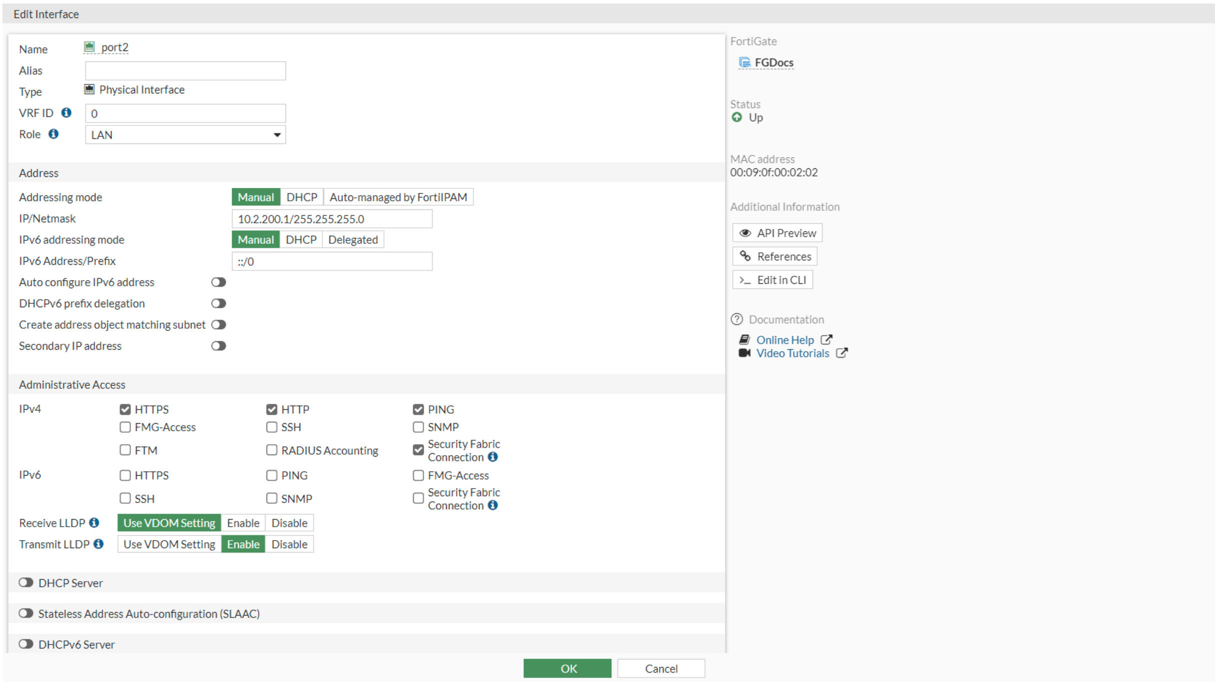
1. On FortiGate A, go to *Network > Interfaces*.
2. Configure an interface:
   - If the interface's role is undefined, under *Administrative Access*, set *Receive LLDP* and *Transmit LLDP* to *Use VDOM Setting*.



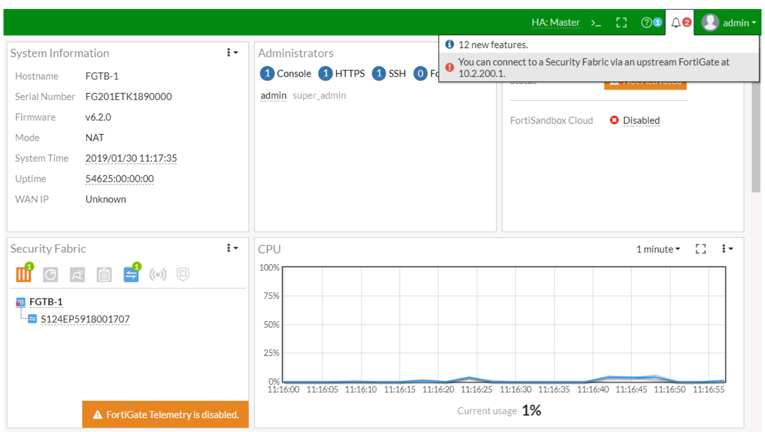
- If the interface's role is WAN, under *Administrative Access*, set *Receive LLDP* to *Enable* and *Transmit LLDP* to *Use VDOM Setting*.



- If the interface's role is LAN, under *Administrative Access*, set *Receive LLDP* to *Use VDOM Setting* and *Transmit LLDP* to *Enable*.



3. Click *OK*. A notification is shown on FortiGate B, *You can connect to a Security Fabric via an upstream FortiGate at 10.2.200.1*.



4. Click the notification. The *Core Network Security* page with the Security Fabric settings opens. All the required settings automatically configured.
5. Click *OK* to apply the settings.

To configure LLDP reception and join a Security Fabric in the CLI:

1. Configure the interface on FortiGate A:
   - Undefined role

```
config system interface
    edit "port3"
        set lldp-reception vdom
        set lldp-transmission vdom
        set role undefined
        ...
```

```
    next
end
```

- WAN role

```
config system interface
    edit "wan1"
        set lldp-reception enable
        set lldp-transmission vdom
        set role wan
        ...
    next
end
```

- LAN role

```
config system interface
    edit "port2"
        set lldp-reception vdom
        set lldp-transmission enable
        set role lan
        ...
    next
end
```

2. Edit the Security Fabric settings on FortiGate B:

```
config system csf
    set status enable
    set upstream-ip 10.2.200.1
end
```

Configuring the Security Fabric with SAML

Security Assertion Markup Language (SAML) is an open standard for exchanging authentication and authorization data between one Identity Provider (IdP) and one or more Service Providers (SP). Both parties exchange messages using the XML protocol as transport. FortiGate firewall devices can be configured as IdPs or SPs.

When the Security Fabric is enabled, you can configure the root FortiGate as the IdP. You can also configure downstream FortiGates to be automatically configured as SPs, with all links required for SAML communication, when added to the Security Fabric. Administrators must still be authorized on each device. Credentials are verified by the root FortiGate, and login credentials are shared between devices. Once authorized, an administrator can move between fabric devices without logging in again.

Optionally, the downstream FortiGate can also be manually configured as an SP, and then linked to the root FortiGate.

The authentication service is provided by the root FortiGate using local system admin accounts for authentication. Any of the administrator account types can be used for SAML log in. After successful authentication, the administrator logs in to the first downstream FortiGate SP, and can then connect to other downstream FortiGates that have the SSO account properly configured, without needing to provide credentials again, as long as admins use the same browser session. In summary, the root FortiGate IdP performs SAML SSO authentication, and individual device administrators define authorization on FortiGate SPs by using security profiles.

Configuring single-sign-on in the Security Fabric

SAML SSO enables a single FortiGate device to act as the identify provider (IdP), while other FortiGate devices act as service providers (SP) and redirect logins to the IdP.

Only the root FortiGate can be the identity provider (IdP). The downstream FortiGates can be configured as service providers (SP).

The process is as follows:

1. Configuring the root FortiGate as the IdP on page 2318
2. Configuring a downstream FortiGate as an SP on page 2319
3. Configuring certificates for SAML SSO on page 2321
4. Verifying the single-sign-on configuration on page 2323

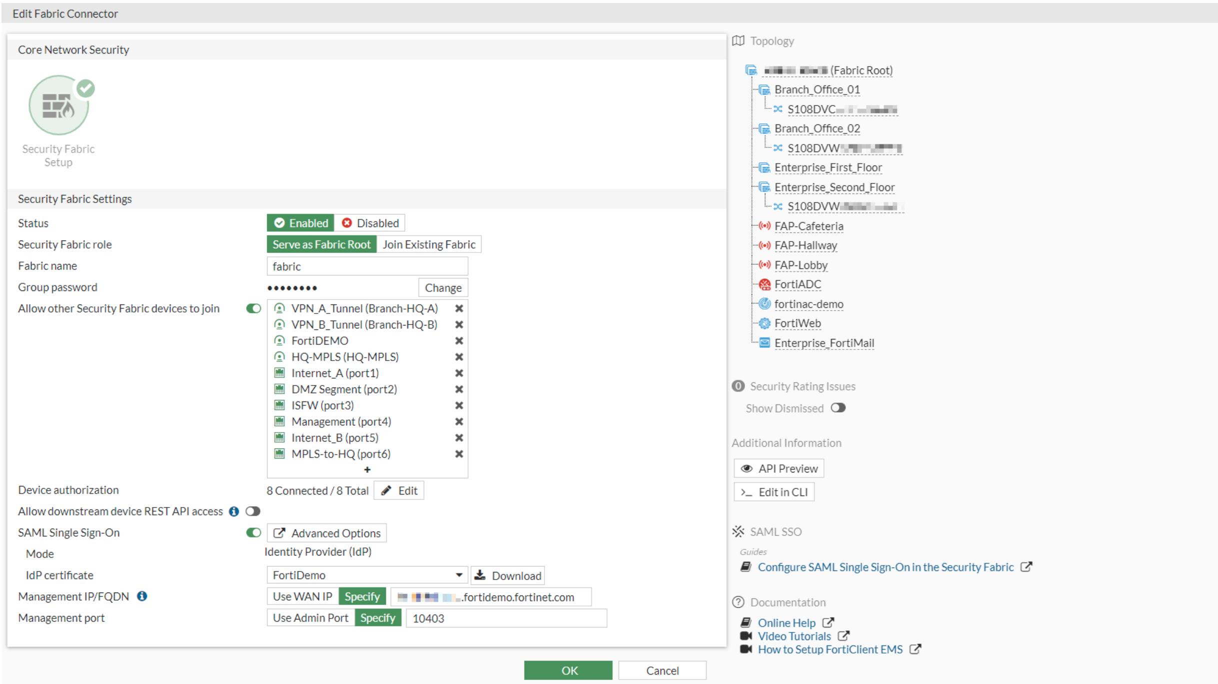
You can also use the CLI. See CLI commands for SAML SSO on page 2324.

Configuring the root FortiGate as the IdP

To configure the root FortiGate as the IdP:

1. Log in to the root FortiGate.
2. Go to *Security Fabric > Fabric Connectors* and double-click the *Security Fabric Setup* card.
3. Enable *SAML Single Sign-On*. The *Mode* field is automatically populated as *Identity Provider (IdP)*.
4. Enter an IP address in the *Management IP/FQDN* box.
5. Enter a management port in the *Management port* box.
   The *Management IP/FQDN* will be used by the SPs to redirect the login request. The *Management IP/FQDN* and *Management port* must be reachable from the user's device.
6. Select the *IdP certificate*.

7. Click *OK*.



Configuring a downstream FortiGate as an SP

There are two ways to configure the downstream FortiGate:

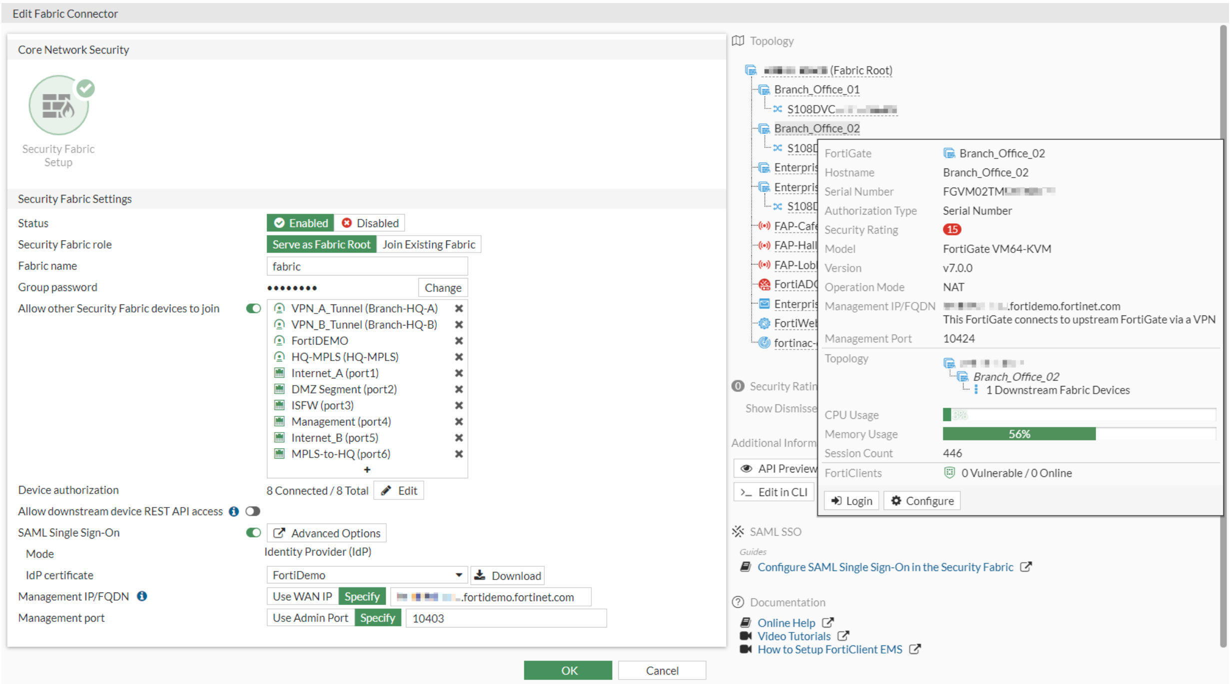
- From the root FortiGate
- From within the downstream device

An SP must be a member of the Security Fabric before you configure it.

To configure the downstream FortiGate from the root FortiGate:

1. Log in to the root FortiGate.
2. Go to *Security Fabric > Fabric Connectors* and double-click the *Security Fabric Setup* card.

3. In the *Topology* tree, hover over a FortiGate and click *Configure*.

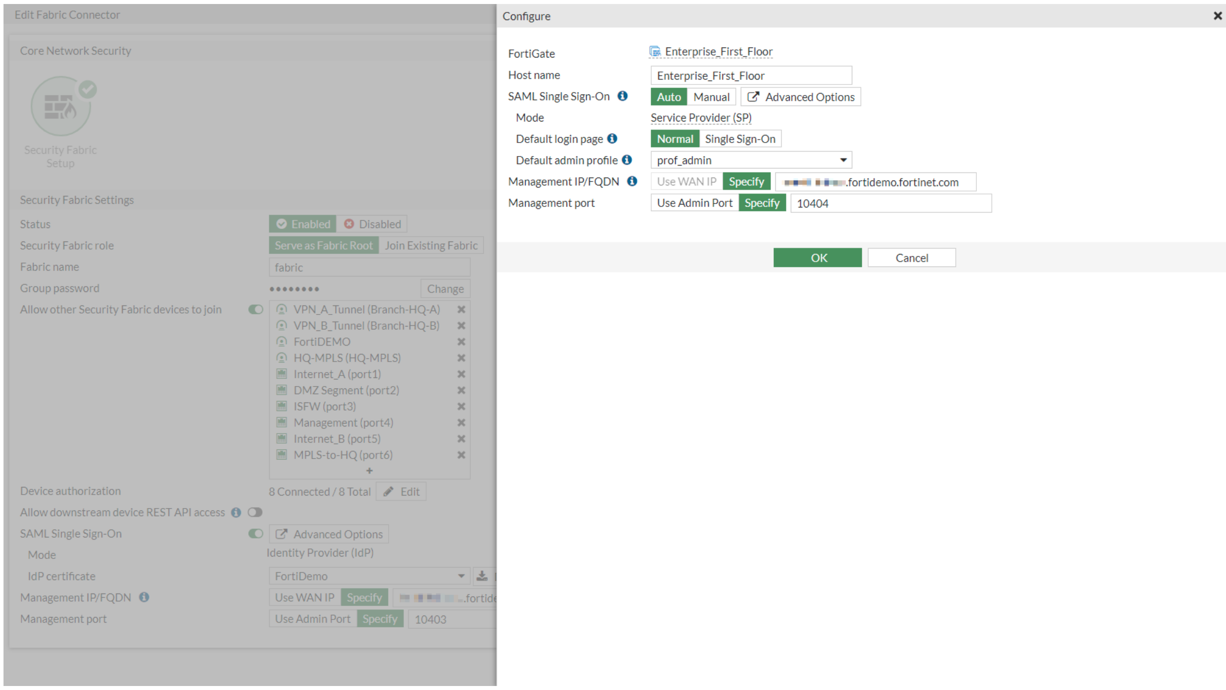


The *Configure* pane opens.

4. Select a *SAML Single Sign-On* option. *Auto* sets the device to SP mode. *Manual* allows you to configure the SSO settings by clicking *Advanced Options*.
5. Select a *Default login page* option.
6. Select one of the following *Default admin profile* types: *prof_admin*, *super_admin*, or *super_admin_readonly*.
7. Enter an IP address in the *Management IP/FQDN* box.
8. Enter a management port in the *Management port* box.

   The *Management IP/FQDN* will be used by the IdP and so other SPs can redirect to each other. The *Management port* must be reachable from the user's device.

9. Click *OK*.



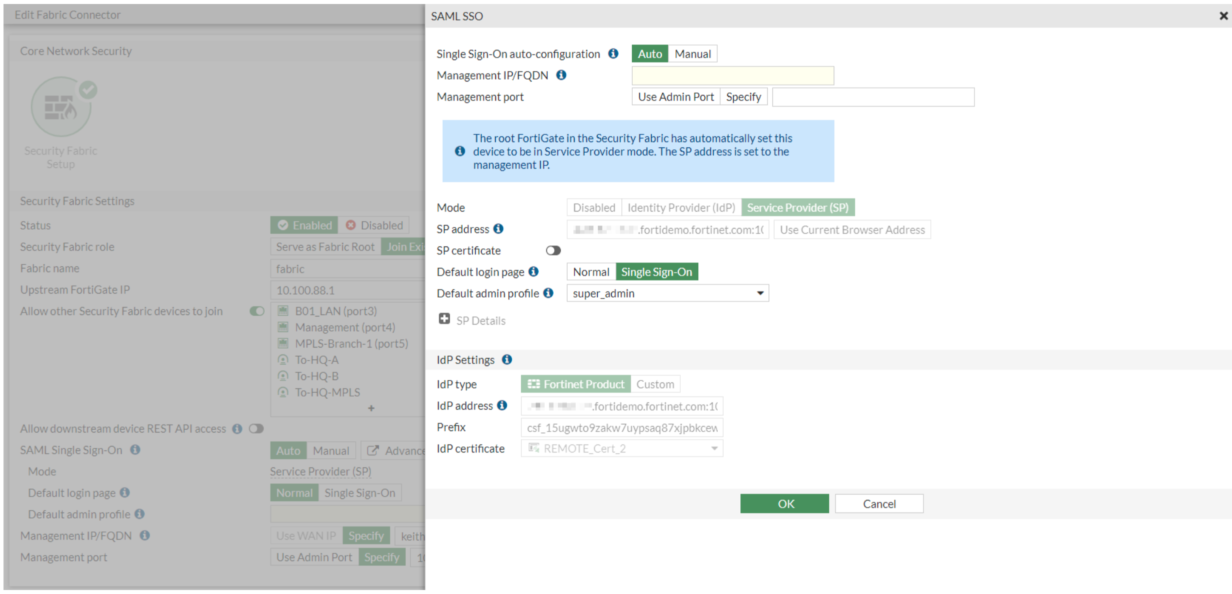
To configure the downstream FortiGate within the device:

1. Log in to the downstream FortiGate.
2. Go to *Security Fabric > Fabric Connectors* and double-click the *Security Fabric Setup* card.
3. Select a *SAML Single Sign-On* option. *Auto* sets the device to SP mode. *Manual* allows you to configure the SSO settings by clicking *Advanced Options*.
4. Select a *Default login page* option.
5. Select one of the following *Default admin profile* types: *prof_admin*, *super_admin*, or *super_admin_readonly*.
6. Enter an IP address in the *Management IP/FQDN* box.
7. Enter a management port in the *Management port* box.
   The *Management IP/FQDN* will be used by the IdP and so other SPs can redirect to each other. The *Management port* must be reachable from the user's device.
8. Click *OK*.

Configuring certificates for SAML SSO

Because communication between the root FortiGate IdP and FortiGate SPs is secured, you must select a local server certificate in the *IdP certificate* option on the root FortiGate. When downstream SPs join the IdP (root FortiGate), the SP automatically obtains the certificate.

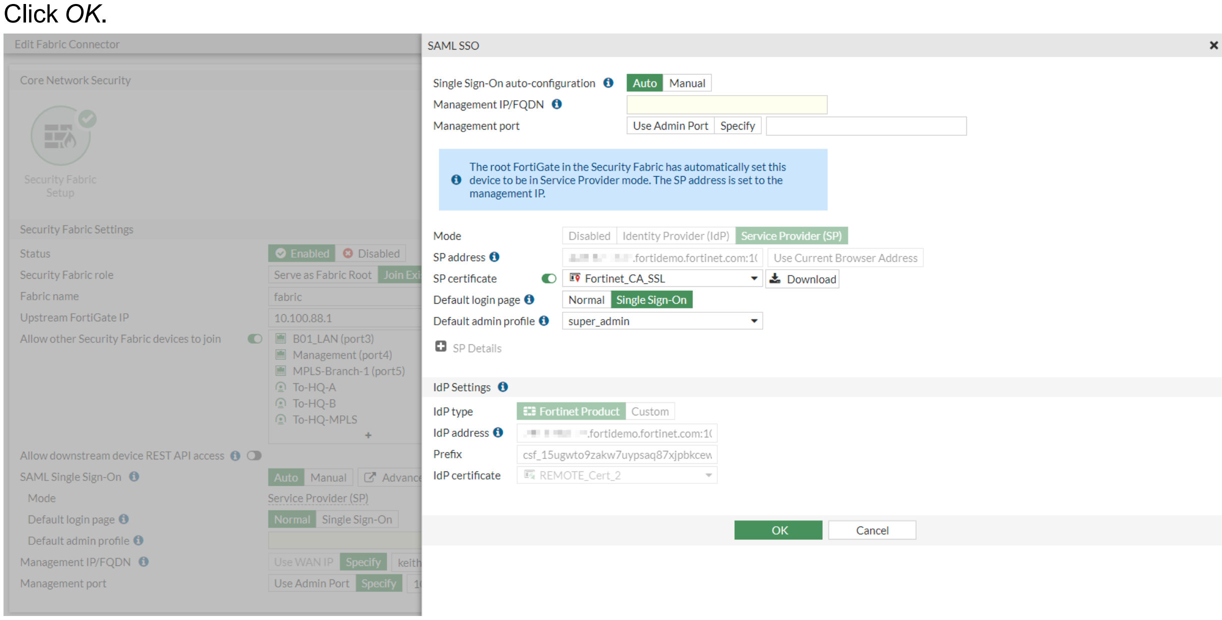
In the following SP example, the *IdP certificate* displays *REMOTE_Cert_2*, which is the root server certificate for the IdP:



It is possible to manually import a certificate from an SP to the IdP so it can be used for authentication.

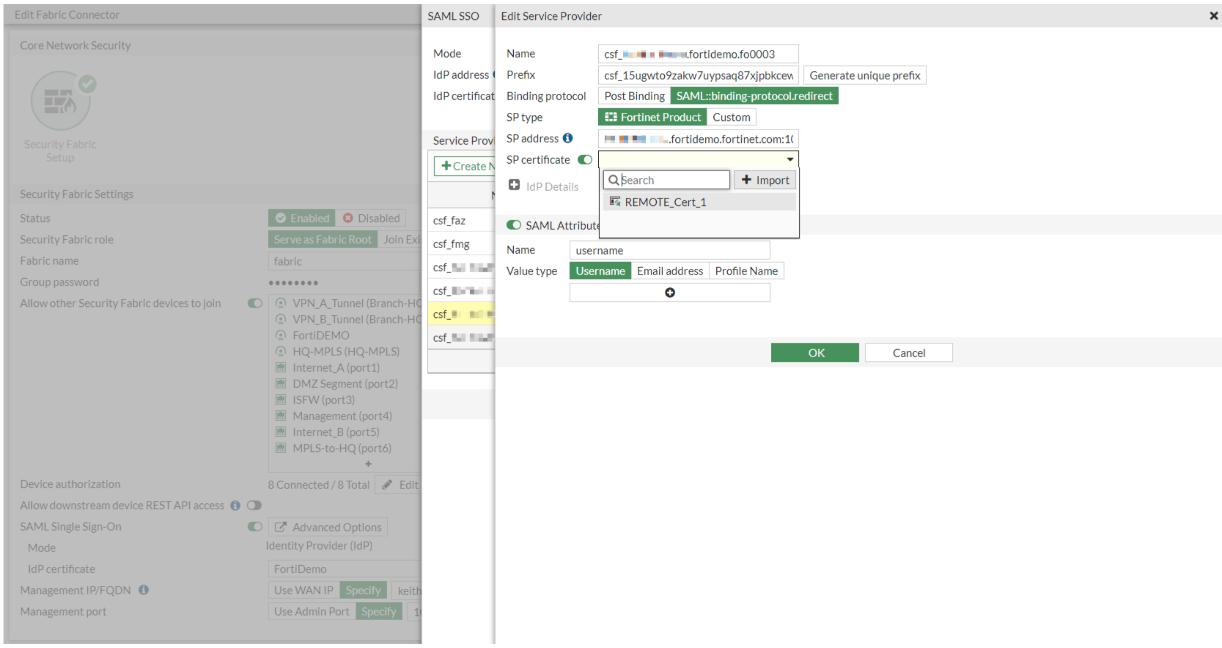
To manually import an SP certificate to an IdP:

1. Add the certificate:
   a. On the SP, go to *Security Fabric > Fabric Connectors* and double-click the *Security Fabric Setup* card.
   b. Click *Advanced Options*. The *SAML SSO* pane opens.
   c. Enable *SP certificate* and select a certificate from the dropdown box.
   d. Click *Download*. The certificate is downloaded on the local file system.
   e. Click *OK*.



2. Import the certificate:
   a. On the IdP, go to *Security Fabric > Fabric Connectors* and double-click the *Security Fabric Setup* card.
   b. Click *Advanced Options*. The *SAML SSO* pane opens.
   c. In the *Service Providers* table, select the SP from step 1 and click *Edit*.

d. Enable *SP certificate* and in the dropdown box, click *Import*.



The *Upload Remote Certificate* window opens.

e. Click *Upload* and select the certificate downloaded in step 1.

f. Click *OK*. The certificate is imported.

g. Click *OK*.

h. In the *IdP certificate* list, select the certificate that you imported.

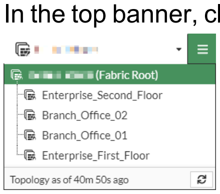
i. Click *OK*.

Verifying the single-sign-on configuration

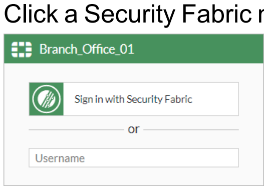
After you have logged in to a Security Fabric member using SSO, you can navigate between any Security Fabric member with SSO configured.

To navigate between Security Fabric members:

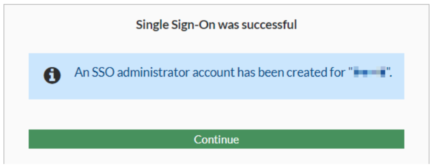
1. Log in to a Security Fabric member that is using SSO.
2. In the top banner, click the name of the device you are logged in to. A list of Security Fabric members displays.



3. Click a Security Fabric member. The login page appears. Click *Sign in with Security Fabric*.

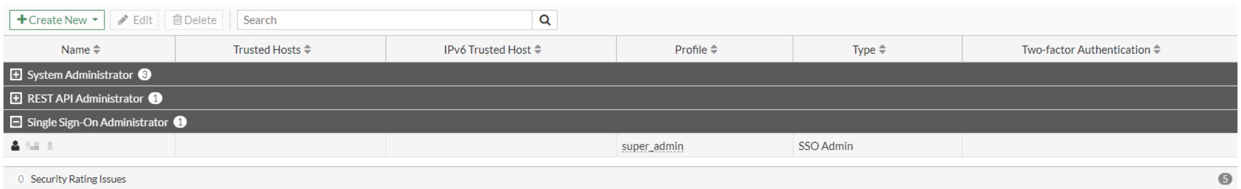


4. A prompt appears that an SSO administrator account has been created. Click *Continue*.



You are now logged in to the Security Fabric member with SSO. The letters "SSO" also display beside the user name in the top banner.

5. Go to *System* > *Administrators* > *Single-Sign-On Administrator* to view the list of SSO admins created.



CLI commands for SAML SSO

To enter a question mark (`?`) or a tab, Ctrl + V must be entered first. Question marks and tabs cannot be typed or copied into the CLI Console or some SSH clients.

To configure the IdP:

```
config system saml
    set status enable
    set role identity-provider
    set cert "Fortinet_Factory"
    set server-address "172.16.106.74"
    config service-providers
        edit "csf_172.16.106.74:12443"
            set prefix "csf_ngczjwqxujfsbhgr9ivhehwu37fml20"
            set sp-entity-id "http://172.16.106.74/metadata/"
            set sp-single-sign-on-url "https://172.16.106.74/saml/?acs"
            set sp-single-logout-url "https://172.16.106.74/saml/?sls"
            set sp-portal-url "https://172.16.106.74/saml/login/"
            config assertion-attributes
                edit "username"
                next
                edit "tdoc@fortinet.com"
                    set type email
                next
            end
        next
    end
end
```

To configure an SP:

```
config system saml
    set status enable
    set cert "Fortinet_Factory"
    set idp-entity-id "http://172.16.106.74/saml-idp/csf_
```

```
ngczjwqxujfsbhgr9ivhehwu37fml20/metadata/"
    set idp-single-sign-on-url "https://172.16.106.74/csf_
ngczjwqxujfsbhgr9ivhehwu37fml20/login/"
    set idp-single-logout-url "https://172.16.106.74/saml-idp/csf_
ngczjwqxujfsbhgr9ivhehwu37fml20/logout/"
    set idp-cert "REMOTE_Cert_1"
    set server-address "172.16.106.74:12443"
end
```

To configure an SSO administrator:

```
config system sso-admin
    edit "SSO-admin-name"
        set accprofile <SSO admin user access profile>
        set vdom <Virtual domain(s) that the administrator can access>
    next
end
```

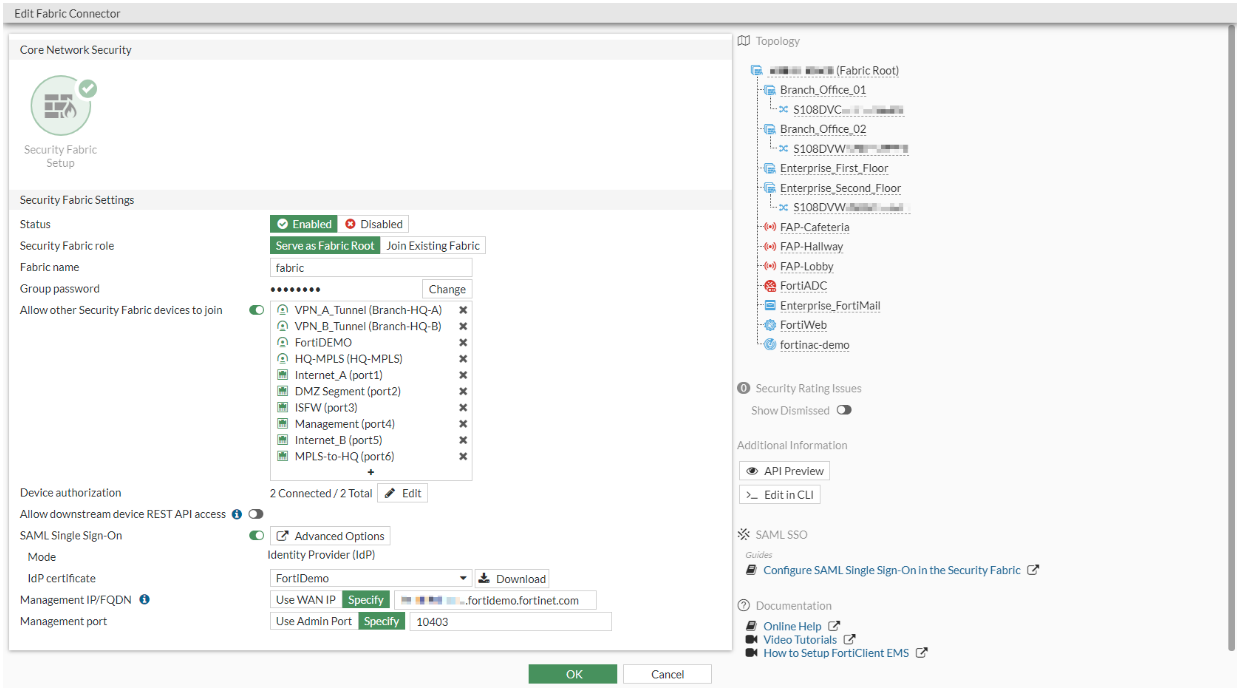
SAML SSO with pre-authorized FortiGates

You can set up SAML SSO authentication in a Security Fabric environment by starting with a root FortiGate that has one or more pre-authorized FortiGates.

After the initial configuration, you can add more downstream FortiGates to the Security Fabric, and they are automatically configured with default values for a service provider.

To set up basic SAML SSO for the Security Fabric:

1. Log in to the root FortiGate of the Security Fabric.
2. Go to *Security Fabric > Fabric Connectors* and double-click the *Security Fabric Setup* card.
3. Join two pre-authorized FortiGates to the root FortiGate (see Configuring the root FortiGate and downstream FortiGates on page 2208).



4. Configure the IdP (see Configuring the root FortiGate as the IdP on page 2318).
5. Configure the SPs (see Configuring a downstream FortiGate as an SP on page 2319).

Navigating between Security Fabric members with SSO

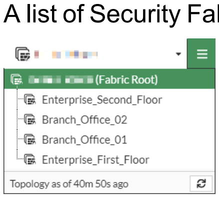
After you have logged in to a Security Fabric member by using SSO, you can navigate between any Security Fabric member with SSO configured. This can be done using the Security Fabric members dropdown menu or by logging in to a FortiGate SP from the root FortiGate IdP.

Security Fabric members dropdown

The Security Fabric members dropdown menu allows you to easily switch between all FortiGate devices that are connected to the Security Fabric. You can also use this menu to customize a FortiGate in the Security Fabric.

To navigate between Security Fabric members:

1. Log in to a Security Fabric member by using SSO.
2. In the top banner, click the name of the device you are logged into with SSO.
   A list of Security Fabric members is displayed.



3. Click the Security Fabric member.
   You are logged in to the Security Fabric member without further authentication.

To customize a FortiGate in the Security Fabric:

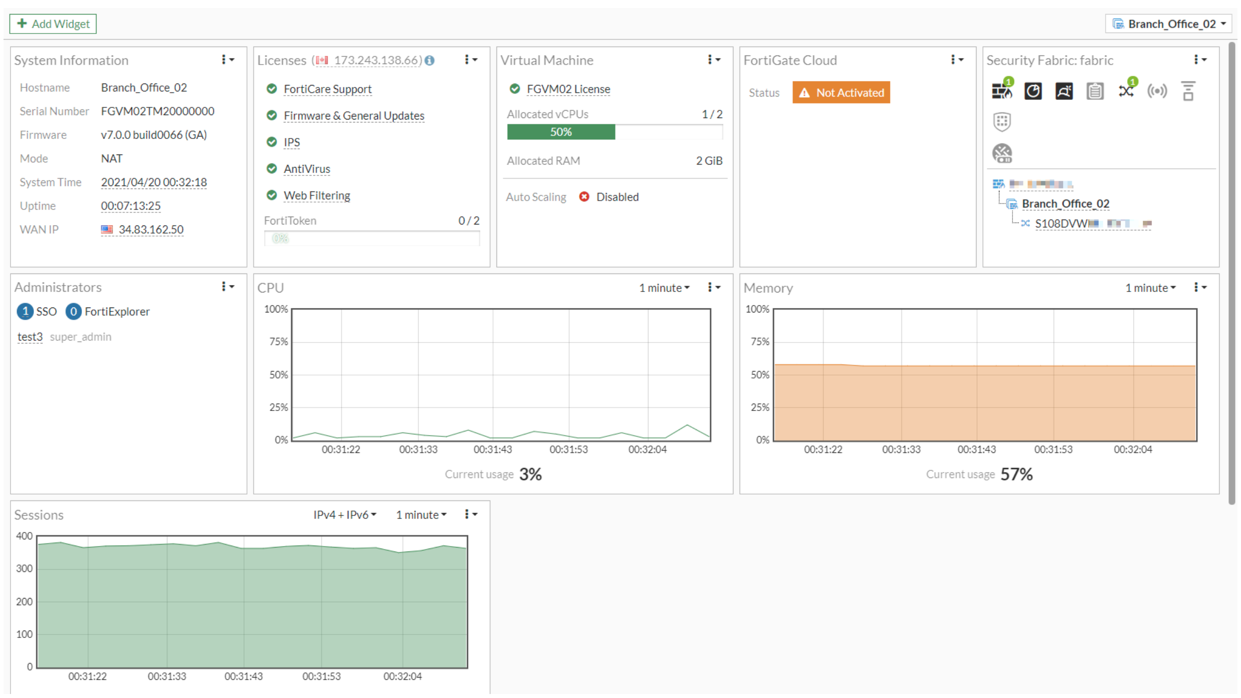
1. In the Security Fabric members dropdown menu, hover the cursor over a FortiGate so the tooltip is shown.
2. Click *Configure*. The *Configure* pane opens.
3. Edit the settings as required.
4. Click *OK*.

Logging in to an SP from the root IdP

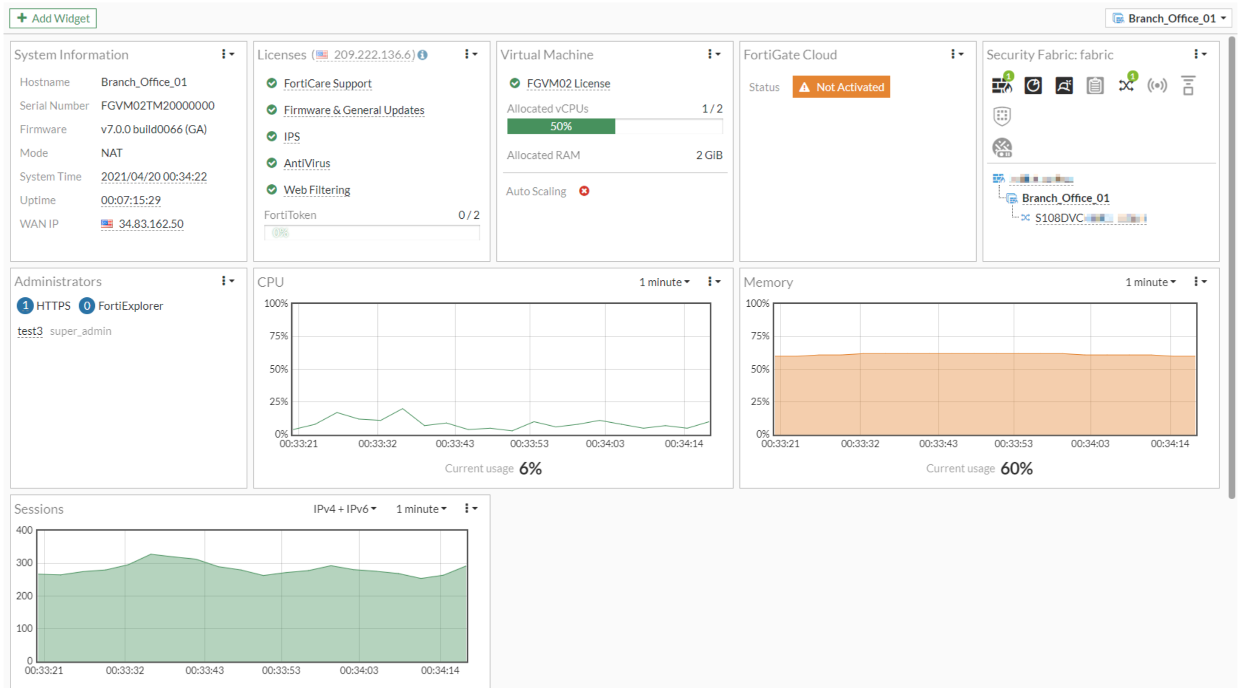
The following example describes how to log in to a root FortiGate IdP, and navigate to other FortiGate SPs in the Security Fabric without further authentication. The local administrator account is named *test3*. The local administrator account must also be available as an SSO administrator account on all downstream FortiGate SPs. Different tabs of the same browser are used to log in to the various FortiGates.

To log in to a FortiGate SP from a root FortiGate IdP:

1. Log in to the root FortiGate IdP by using the local administrator account.
   In this example, the local administrator account is named *test3*.
2. Go to *Security Fabric > Fabric Connectors* and double-click the *Security Fabric Setup* card.
3. In the *Topology* tree, click one of the downstream FortiGate SPs, and select *Login to <name of FortiGate>*. The login screen is displayed.
4. In the login screen, select *Single Sign-On*.
   By using cookies in your local browser for the already-authenticated SSO administrator, FortiGate logs you in to the downstream FortiGate SP as the SSO administrator. In this example, the SSO administrator name is *test3*.



5. While still logged into the root FortiGate IdP in your browser, go to the browser tab for the root FortiGate IdP, and log in to another FortiGate SP that is displayed on the *Security Fabric* widget in the GUI.



SAML SSO login uses *SAML_IDP* session cookies of already authenticated admin users in your local browser cache to send to the root FortiGate IdP for authentication. If your browser cache is manually cleared, or you close your browser, you must authenticate again.

It is possible to log in to one downstream FortiGate SP in a Security Fabric, and then open another tab in your browser to connect to another FortiGate SP that is not a member of the Security Fabric.

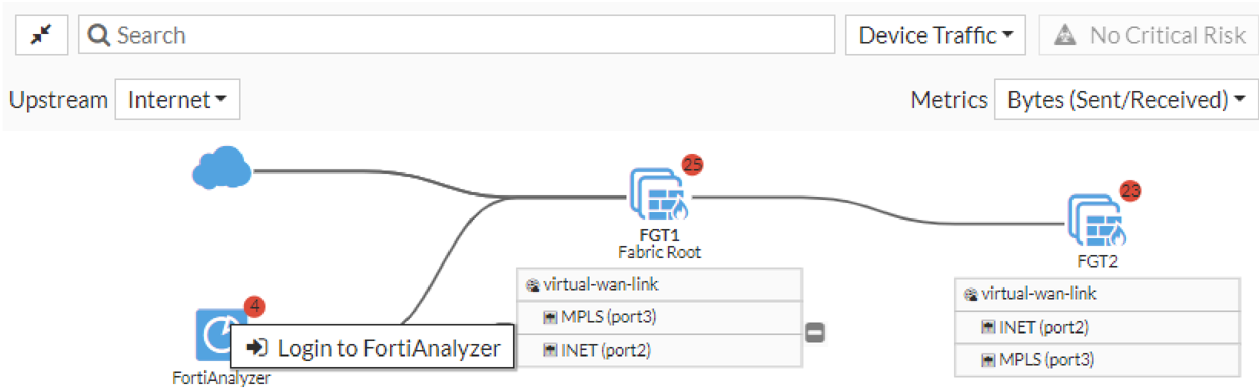
This is useful in cases where the SSO administrator and the local system administrator on the FortiGate SP both have the same login name, but are two different entities.

Integrating FortiAnalyzer management using SAML SSO

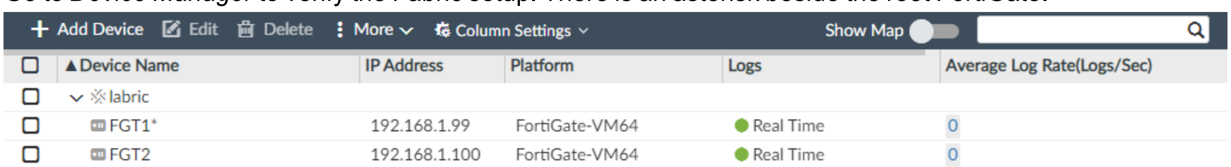
When a FortiGate acting as a Security Fabric root is configured as a SAML SSO identity provider (IdP), the FortiAnalyzer of the Security Fabric can register itself as a service provider (SP). This simplifies the configuration by enabling the setting in FortiAnalyzer to facilitate Fabric SSO access to the FortiAnalyzer once authenticated to the root FortiGate. When signed in using SSO, the FortiAnalyzer includes a Security Fabric navigation dropdown, which allows easy navigation to FortiGates in the Fabric.

To enable FortiAnalyzer as a Fabric SP in the GUI:

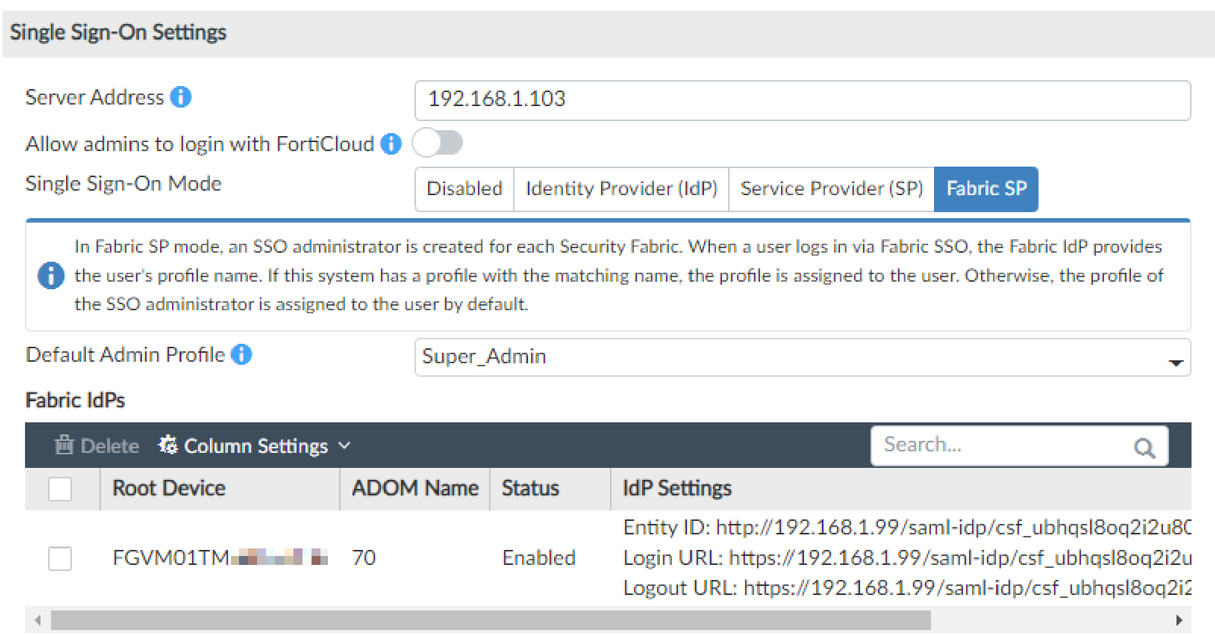
1. On the root FortiGate, go to *Security Fabric > Physical Topology* or *Logical Topology*.
2. In the topology, click the *FortiAnalyzer* icon and select *Login to FortiAnalyzer*.



3. Enter the credentials to log in. A Security Fabric must be configured with the Fabric devices listed under the Fabric name.
   a. Go to *Device Manager* to verify the Fabric setup. There is an asterisk beside the root FortiGate.

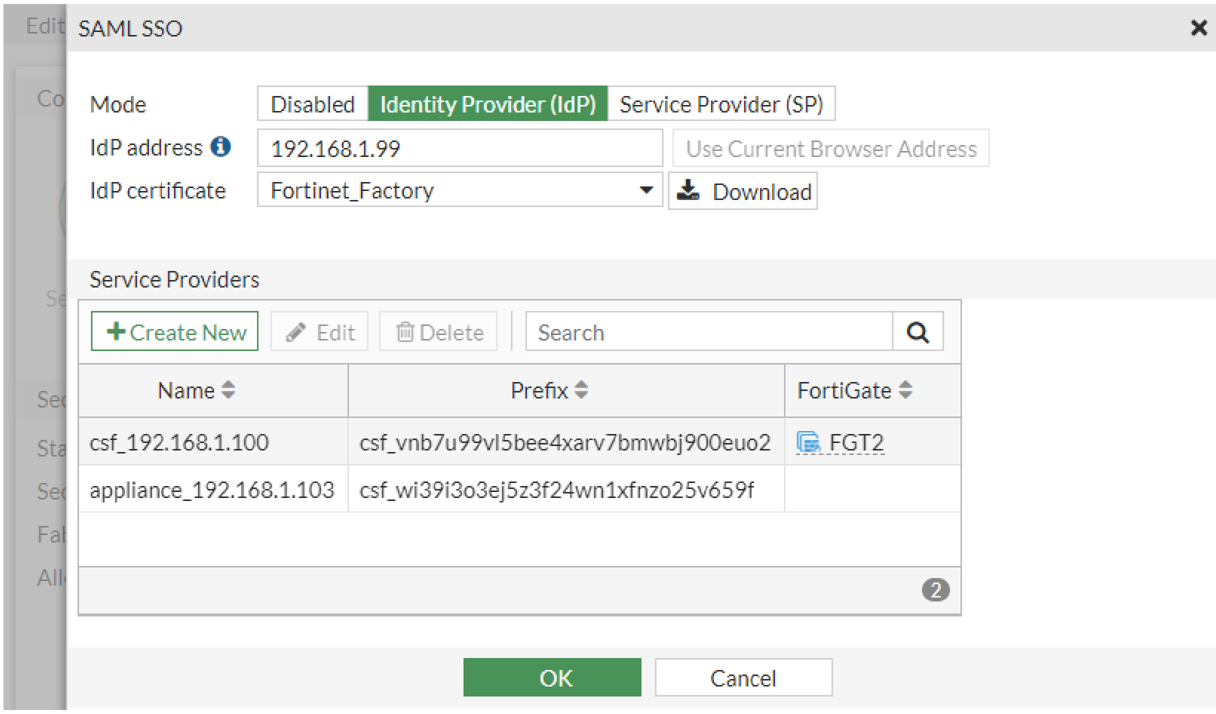


4. Edit the FortiAnalyzer SAML SSO settings:
   a. Go to *System Settings > Admin > SAML SSO*.
   b. For *Single Sign-On Mode*, select *Fabric SP* and enter the address to access the FortiAnalyzer in *Server Address*.

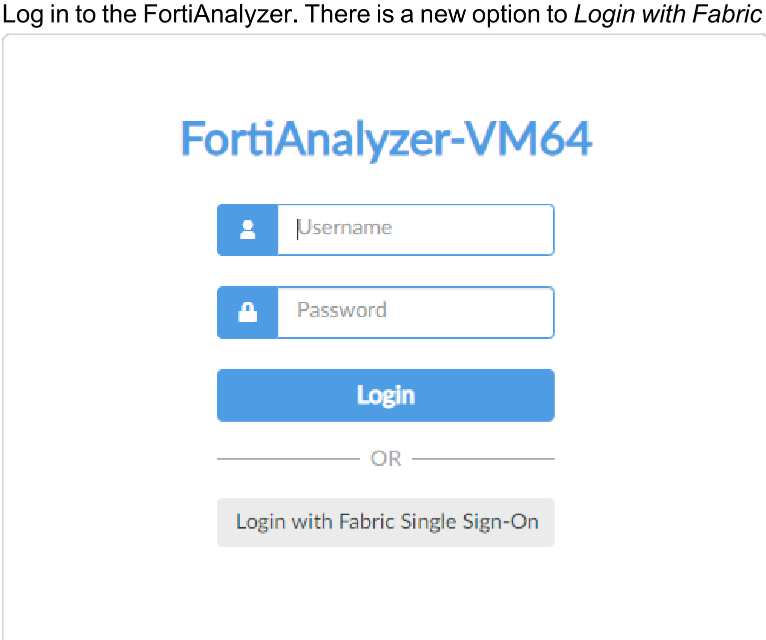


   c. Click *Apply* and log out of the FortiAnalyzer. The FortiAnalyzer will automatically register itself on the FortiGate and is a visible appliance in the list of SPs.

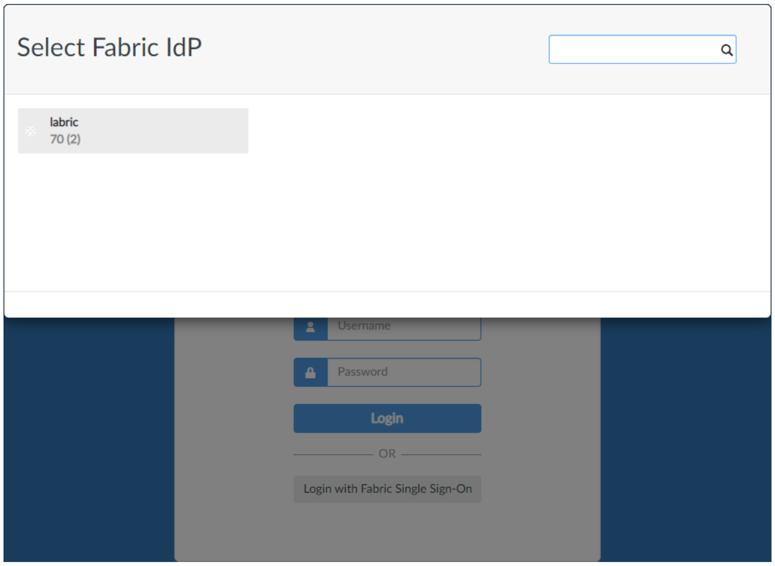
5. Verify that the FortiAnalyzer registration was successful:
   a. In FortiOS, go to *Security Fabric > Fabric Connectors* and double-click the *Security Fabric Setup* card.
   b. In the *SAML Single Sign-On* section click *Advanced Options*. There should be an entry for the FortiAnalyzer in the *Service Providers* table (*appliance_192.168.1.103*).



6. Log in to the FortiAnalyzer. There is a new option to *Login with Fabric Single Sign-On*.



7. Click *Login with Fabric Single Sign-On*. A dialog appears to select a Fabric IdP.



8. Select a FortiGate. The ADOM containing that FortiGate opens.

To enable FortiAnalyzer as a Fabric SP in the CLI:

1. In FortiAnalyzer, enable the device as a Fabric SP:

```
config system saml
    set status enable
    set role FAB-SP
    set server-address "192.168.1.99"
end
```

FortiAnalyzer will register itself on the FortiGate as an appliance.

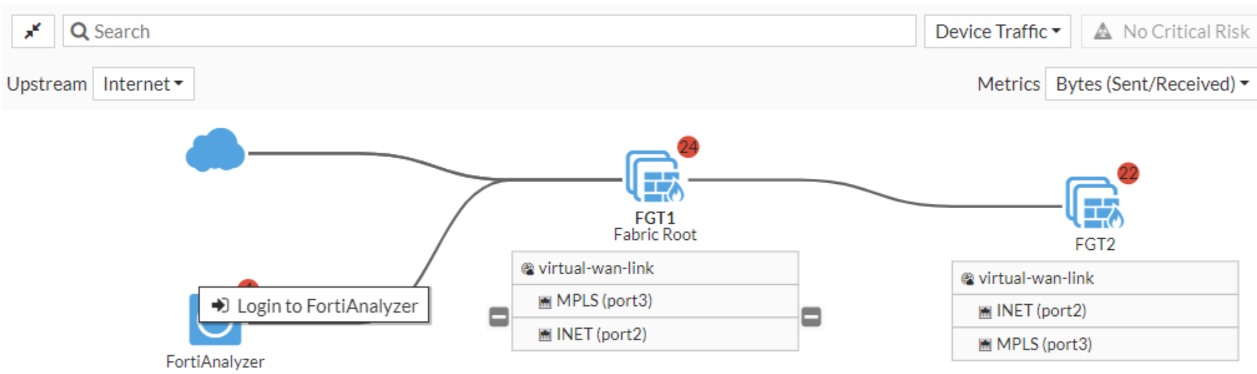
2. Verify the configuration in FortiOS:

```
show system saml
config system saml
    set status enable
    set role identity-provider
    set cert "fortigate.domain.tld"
    set server-address "192.168.1.99"
    config service-providers
        edit "appliance_192.168.1.103"
            set prefix "csf_76sh0bm4e7hf1ty54w42yrrv88tk8uj"
            set sp-entity-id "http://192.168.1.103/metadata/"
            set sp-single-sign-on-url "https://192.168.1.103/saml/?acs"
            set sp-single-logout-url "https://192.168.1.103/saml/?sls"
            set sp-portal-url "https://192.168.1.103/saml/login/"
            config assertion-attributes
                edit "username"
                next
                edit "profilename"
                    set type profile-name
                next
```

```
                end
            next
        end
    end
```

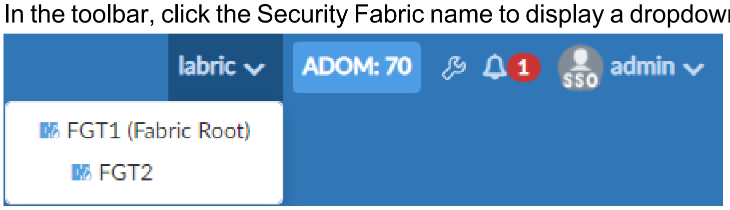
To navigate between devices using SAML SSO in FortiOS:

1. Log in to the root FortiGate.
2. Go to *Security Fabric > Physical Topology* or *Logical Topology*.
3. In the topology, click the *FortiAnalyzer* icon and select *Login to FortiAnalyzer*.



To navigate between devices using SAML SSO in FortiAnalyzer:

1. Log in to the FortiAnalyzer using SSO.
2. Navigate to the ADOM that contains the root FortiGate of the Security Fabric.
3. In the toolbar, click the Security Fabric name to display a dropdown a list of the Fabric FortiGates.



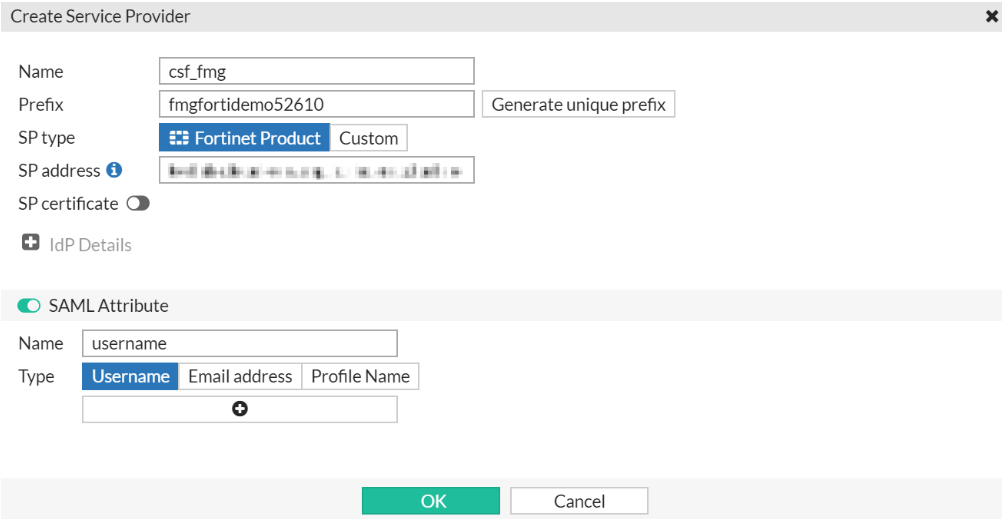
Integrating FortiManager management using SAML SSO

When a FortiGate is configured as the SAML SSO IdP, FortiManager can be added as an SP.

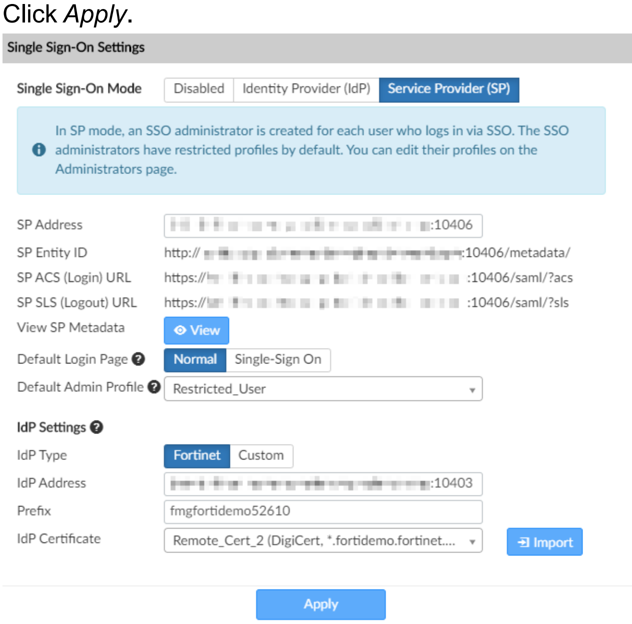
To configure FortiManager as a Fabric SP:

1. On the root FortiGate, go to *Security Fabric > Fabric Connectors*, and edit the *Security Fabric Setup* connector.
2. In the *Security Fabric Settings* section, click *Advanced Options*.
3. In the *Service Providers* section, click *Create New*.
4. Enter a name and a prefix for the SP. FortiOS generates a unique prefix, but you can enter your own.

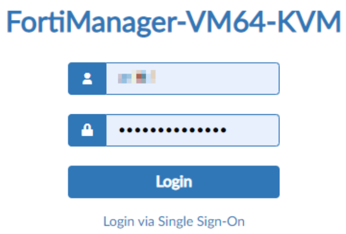
5. In *SP address*, enter the FortiManager address including the port number.



6. Click *OK*.
7. In FortiManager, go to *System Settings > Admin > SAML SSO* and in the *Single Sign-On Mode* section, click *Service Provider (SP)*.
8. Configure the *IdP Settings*:
   a. For *IdP Type*, click *Fortinet*.
   b. For *IdP Address*, enter the root FortiGate address including the port number.
   c. Enter the *Prefix* of the SP.
   d. For *IdP Certificate*, import the same certificate used on the root FortiGate.
   e. Click *Apply*.



9. To verify that the configuration works, log out of FortiManager and log in using the *Login via Single-Sign-On* link.

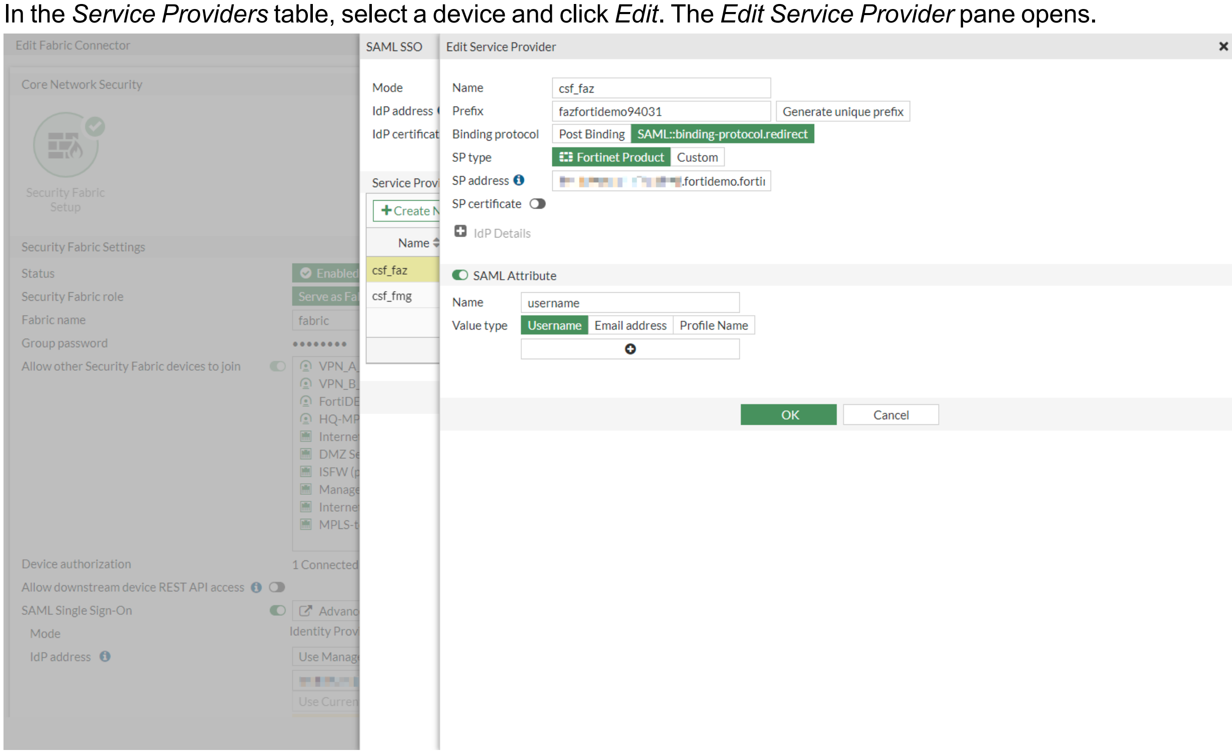


Advanced option - FortiGate SP changes

From a root FortiGate IdP, you can edit each of the FortiGate SPs. For example, you can edit a FortiGate SP to generate a new prefix, or you can add or modify SAML attributes. When you generate a new prefix value, it is propagated to the respective downstream FortiGates.

To edit an SP from the root FortiGate (IdP):

1. Go to *Security Fabric > Fabric Connectors* and double-click the *Security Fabric Setup* card.
2. Click *Advanced Options*. The *SAML SSO* pane opens.
3. In the *Service Providers* table, select a device and click *Edit*. The *Edit Service Provider* pane opens.



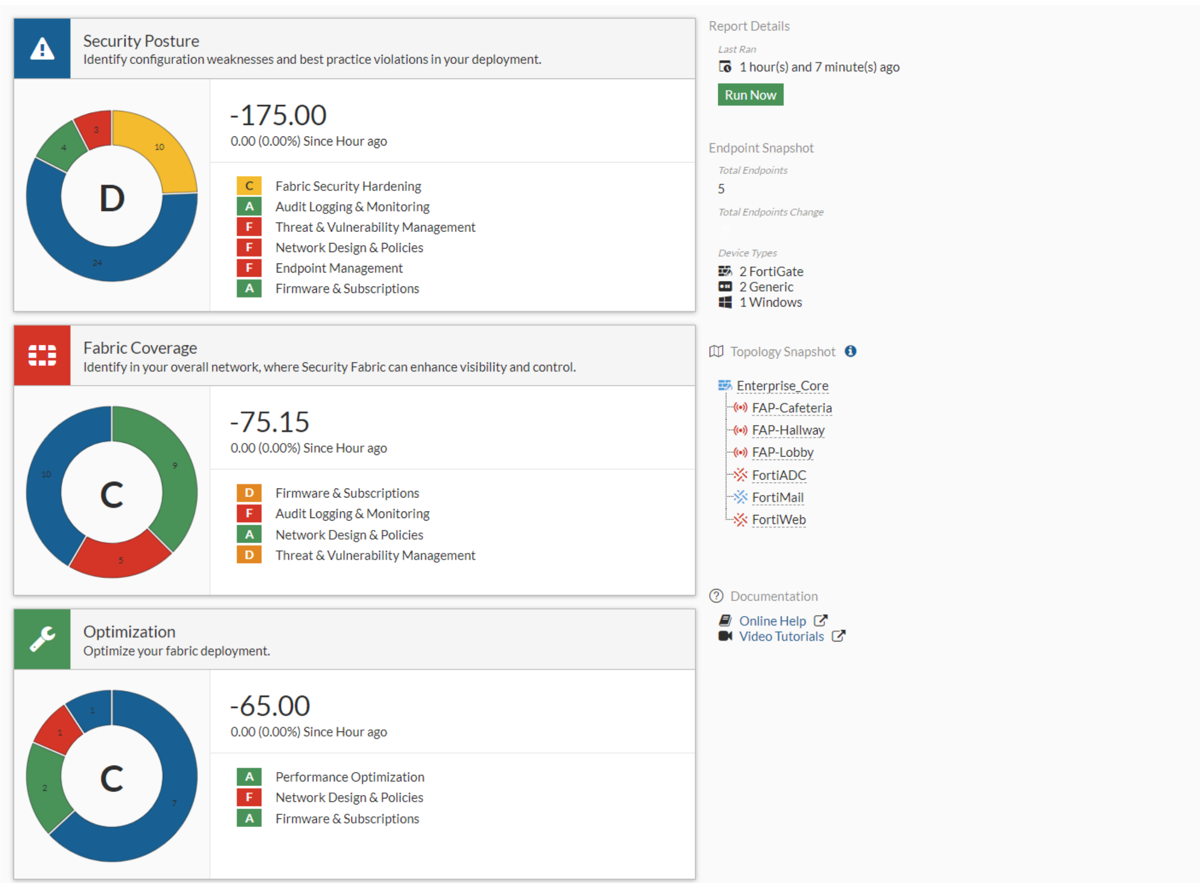
4. Edit the settings as needed.
5. Click *OK*.

Security rating

The security rating uses real-time monitoring to analyze your Security Fabric deployment, identify potential vulnerabilities, highlight best practices that can be used to improve the security and performance of your network, and calculate Security Fabric scores.

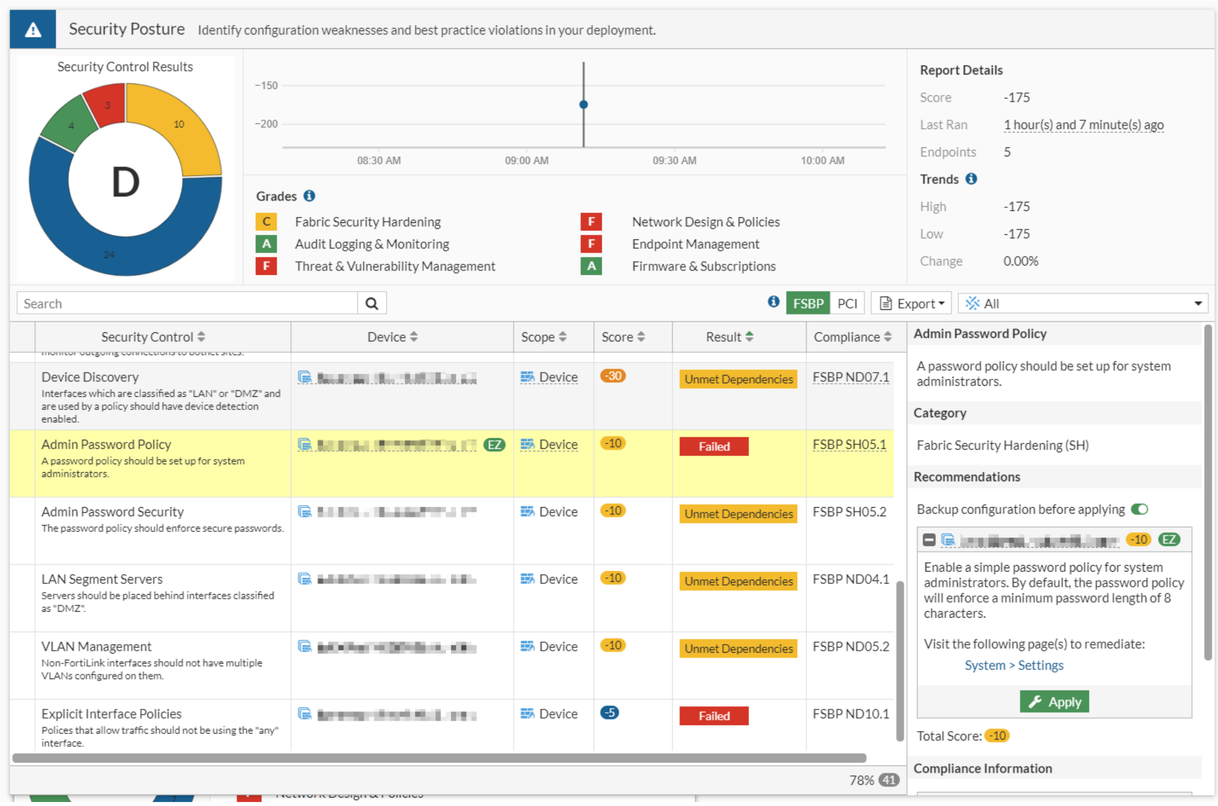
To view the security rating, go to *Security Fabric > Security Rating* on the root FortiGate.

The *Security Rating* page is separated into three major scorecards: *Security Posture*, *Fabric Coverage*, and *Optimization*, which provide an executive summary of the three largest areas of security focus in the Security Fabric.

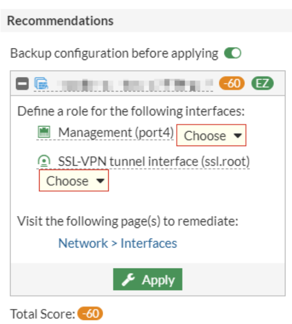


The scorecards show an overall letter grade and breakdown of the performance in sub-categories. Clicking a scorecard drills down to a detailed report of itemized results and compliance recommendations. The point score represents the net score for all passed and failed items in that area. In the drill down report, hover the cursor over a score to view the calculation breakdown.

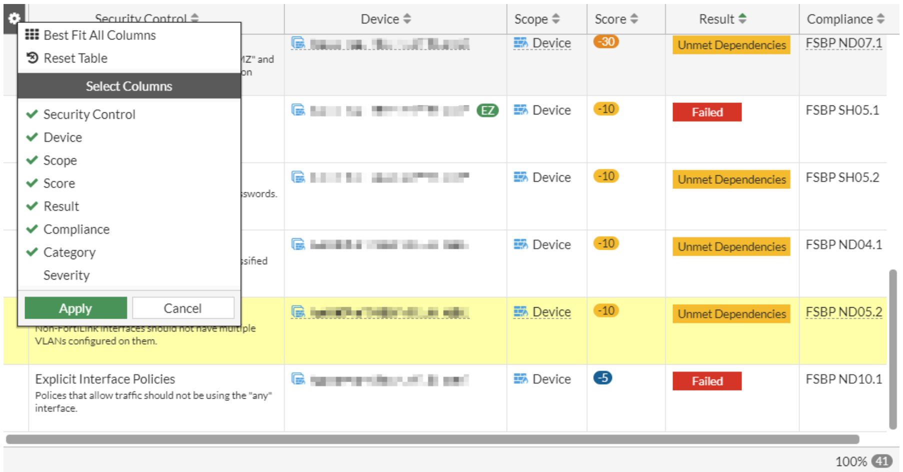
The report includes the security controls that were tested against, linking to specific FSBP or PCI compliance policies. Click the *FSBP* and *PCI* buttons to reference the corresponding standard. Users can search or filter the report results. If there is a failed check on the scorecard, there is a link in the *Recommendations* section that takes you to the page to resolve the problem.



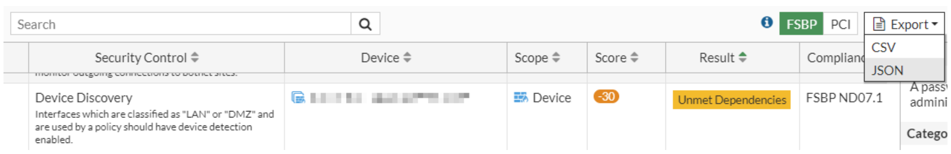
Certain remediations marked with an *EZ* symbol represent configuration recommendations that support *Easy Apply*. In the panel on the right, in the *Recommendations* section, click *Apply* to apply the changes to resolve the failed security control.



The report table can be customized by adding more columns, such as *Category*, to view, filter, or sort the results based on scorecard categories. Click the gear icon to customize the table.



Users can also export the reports as CSV or JSON files by clicking the *Export* dropdown.



To exit the current view, click the icon beside the scorecard title to return to the summary view.

For more information about security ratings, and details about each of the checks that are performed, go to Security Best Practices & Security Rating Feature.

The following licensing options are available for security rating checks:

- A base set of free checks
- A licensed set that requires a FortiGuard Security Rating Service subscription

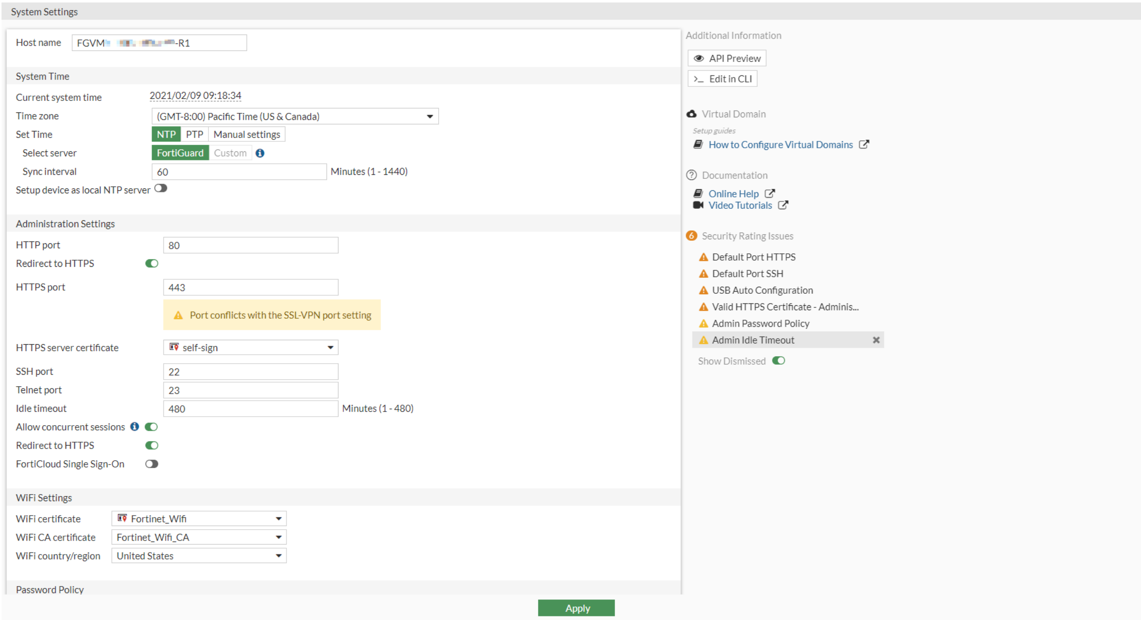
The base set can be run locally on any FortiGate and on all other devices in the Security Fabric. For a list of base and licensed security rating checks, see FortiGuard Security Rating Service.

Security rating notifications

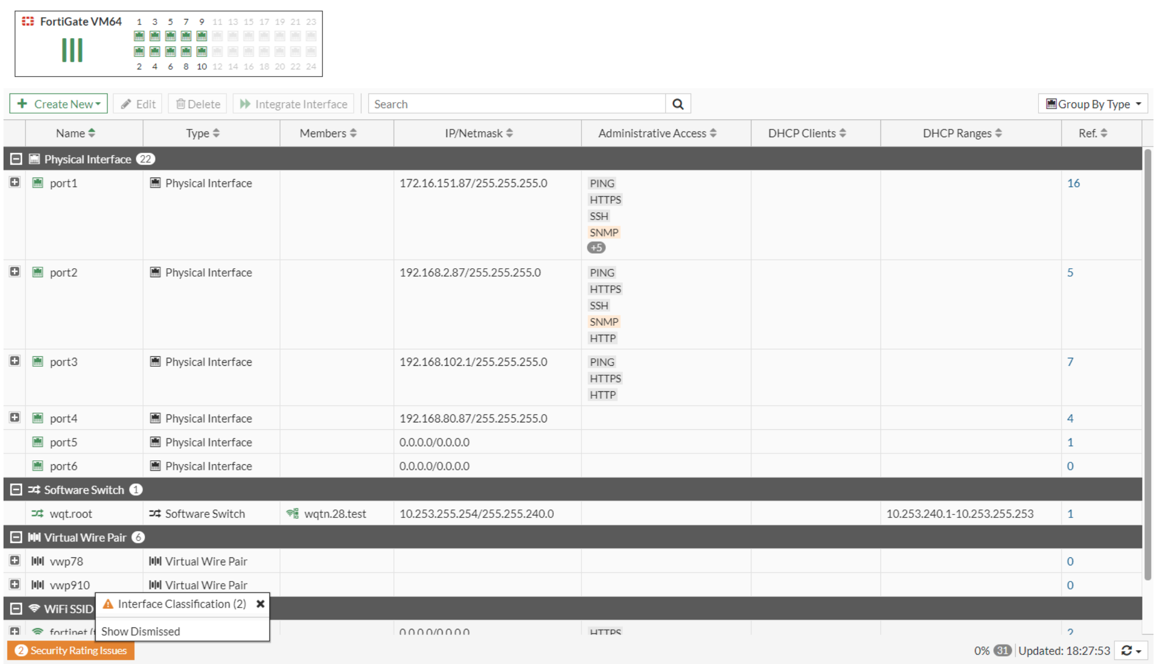
Security rating notifications are shown on settings pages, which list configuration issues determined by the security rating report. You can open the recommendations to see which items need to be fixed. Notifications can be dismissed in the GUI. Dismissed issues are unique for each administrator. Hashes for dismissed notifications are saved in local storage. If a user clears the local storage, all issues will show up again as not dismissed.

Notification locations

On the *System > Settings* page, there is a *Security Rating Issues* section in the right-side gutter. To dismiss a notification, hover over the issue and click the *X* beside it. To view dismissed notifications, enable *Show Dismissed*.

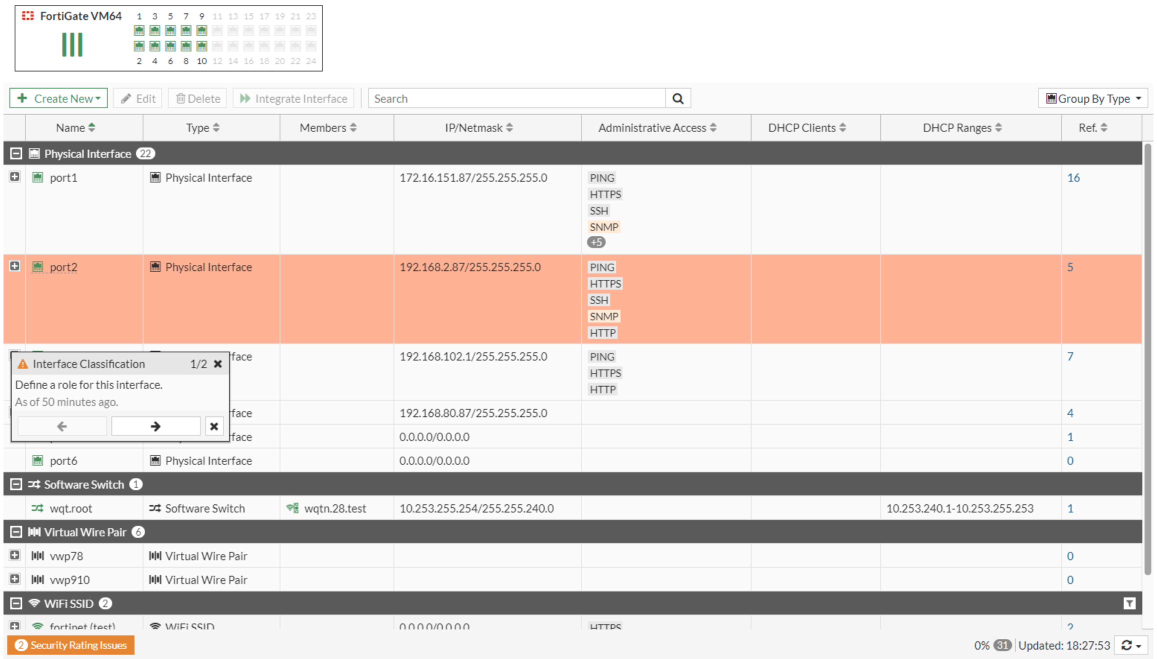


On the *Network > Interfaces* page, there is a *Security Rating Issues* section in the table footer. Click *Security Rating Issues* to view the list of issues. To dismiss a notification, click the *X* beside it. To view dismissed notifications, click *Show Dismissed*.



Notification pop-ups

When you click a security rating notification, a pop-up appears and the related setting is highlighted in the GUI. The pop-up contains a description of the problem and a timestamp of when the issue was found.



Once an issue is resolved, the notification disappears after the next security rating report runs.

Security rating check scheduling

Security rating checks by default are scheduled to run automatically every four hours.

To disable automatic security checks using the CLI:

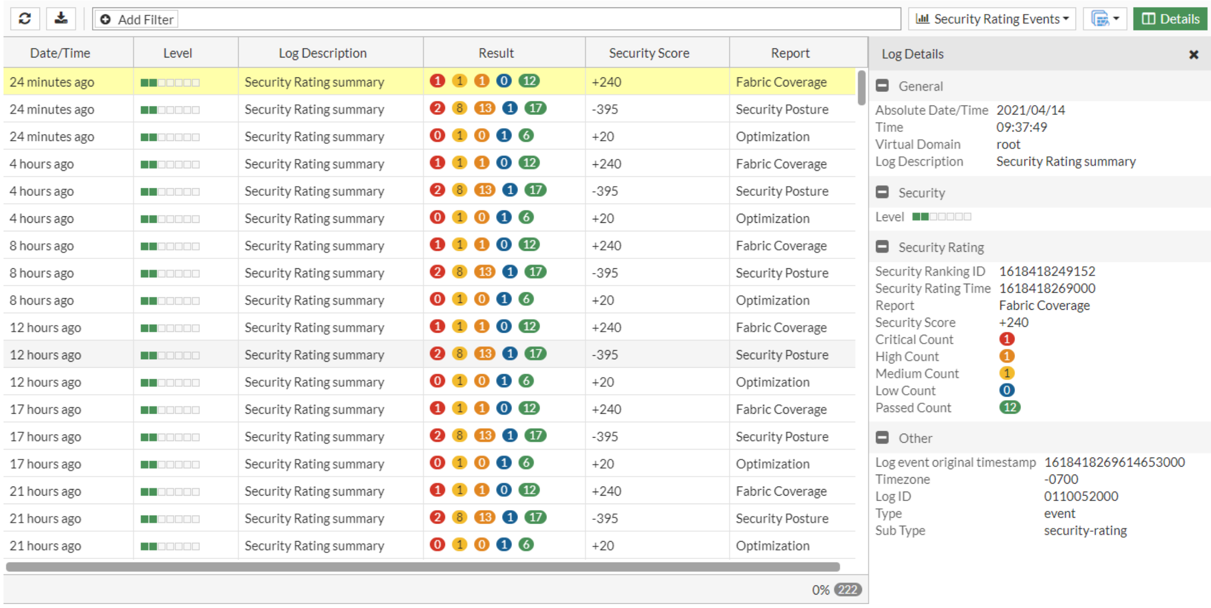
```
config system global
    security-rating-run-on-schedule disable
end
```

To manually run a report using the CLI:

```
# diagnose report-runner trigger
```

Logging the security rating

The results of past security checks are available on the *Log & Report > System Events* page. Click the *Security Rating Events* card to see the detailed log.



An event filter subtype can be created for the Security Fabric rating so event logs are created on the root FortiGate that summarize the results and show detailed information for the individual tests.

To configure security rating logging using the CLI:

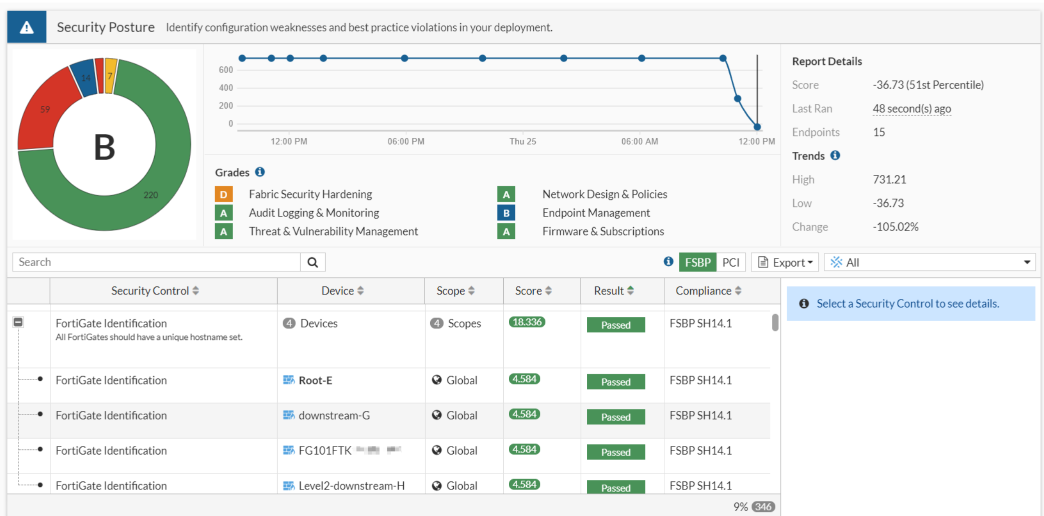
```
config log eventfilter
    set security-rating enable
end
```

Multi VDOM mode

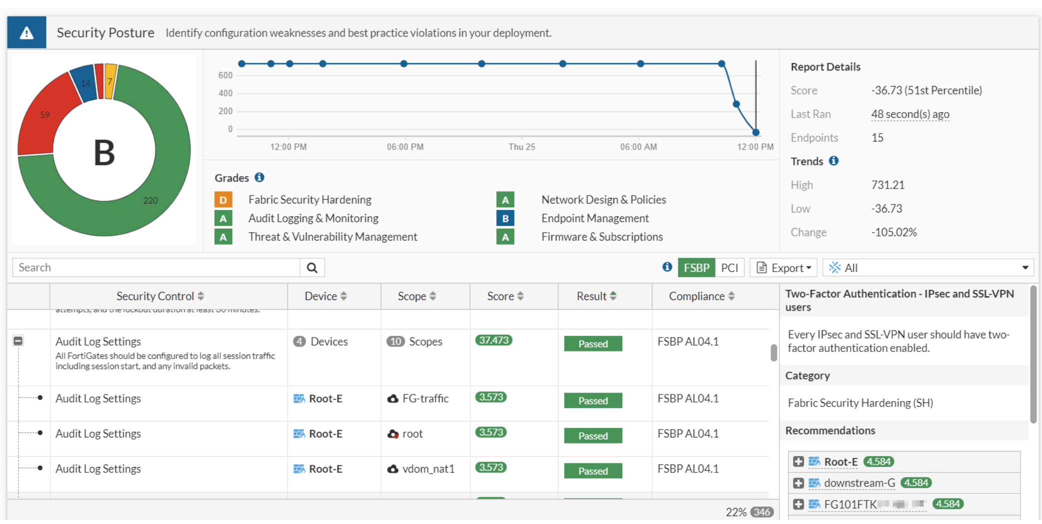
In multi VDOM mode, security rating reports can be generated in the Global VDOM for all of the VDOMs on the device. Administrators with read/write access can run the security rating report in the Global VDOM. Administrators with read-only access can only view the report.

On the report scorecards, the *Scope* column shows the VDOMs that the check was run on. On checks that support *Easy Apply*, the remediation can be run on all of the associated VDOMs.

Global scope:



VDOM scope:



The security rating event log is available on the root VDOM.

Security Fabric score

The Security Fabric score is calculated when a security rating check is run, based on the severity level of the checks that are passed or failed. A higher scores represents a more secure network. Points are added for passed checks and removed for failed checks.

Severity level	Weight (points)
Critical	50
High	25
Medium	10
Low	5

To calculate the number of points awarded to a device for a passed check, the following equation is used:

$$\text{score} = \frac{\text{<severity level weight>}}{\text{<\# of FortiGates>}} \times \text{<secure FortiGate multiplier>}$$

The secure FortiGate multiplier is determined using logarithms and the number of FortiGate devices in the Security Fabric.

For example, if there are four FortiGate devices in the Security Fabric that all pass the compatible firmware check, the score for each FortiGate device is calculated with the following equation:

$$\frac{50}{4} \times 1.292 = 16.15 \text{ points}$$

All of the FortiGate devices in the Security Fabric must pass the check in order to receive the points. If any one of the FortiGate devices fails a check, the devices that passed are not awarded any points. For the device that failed the check, the following equation is used to calculated the number of points that are lost:

$$\text{score} = \text{<severity level weight>} \;\; \text{x} \; \text{<secure FortiGate multiplier>}$$

For example, if the check finds two critical FortiClient vulnerabilities, the score is calculated with the following equation:

$$-50 \times 2 = -100 \text{ points}$$

Scores are not affected by checks that do not apply to your network. For example, if there are no FortiAP devices in the Security Fabric, no points will be added or subtracted for the FortiAP firmware version check.

Automation stitches

Automation stitches automate the activities between the different components in the Security Fabric, which decreases the response times to security events. Events from any source in the Security Fabric can be monitored, and action responses can be set up to any destination.

Automation stitches can also be used on FortiGate devices that are not part of a Security Fabric.

An automation stitch consists of two parts: the trigger and the actions. The trigger is the condition or event on the FortiGate that activates the action, for example, a specific log, or a failed log in attempt. The action is what the FortiGate does in response to the trigger.

Automation stitches that use cloud-based actions (AWS Lambda, Azure Function, Google Cloud Function, and AliCloud Function) have the option to delay an action after the previous action is completed.

Diagnose commands are available in the CLI to test, log, and display the stitch history and settings.

Automation stitches can only be created on the root FortiGate in a Security Fabric.

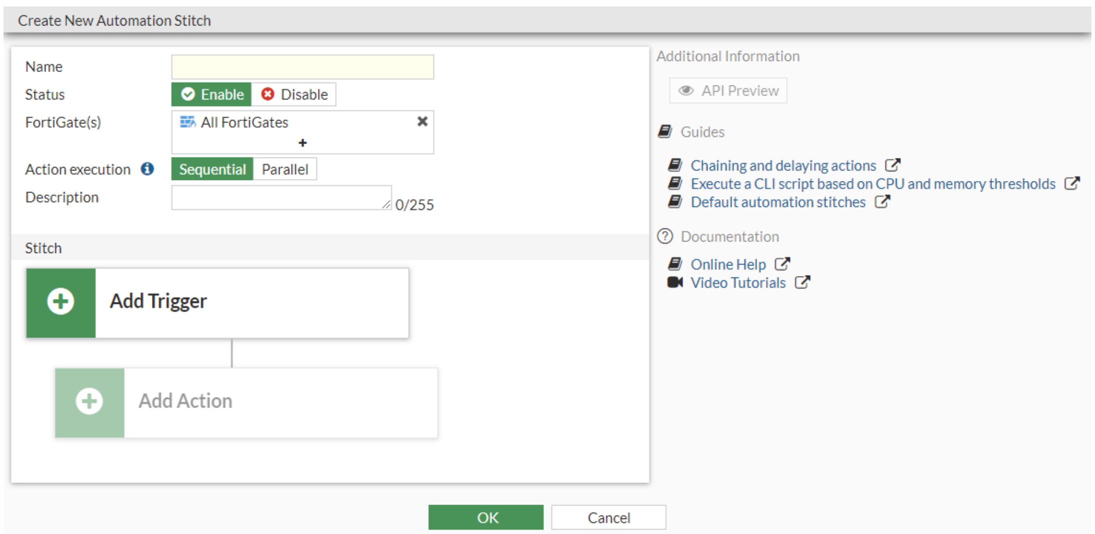
Creating automation stitches

To create an automation stitch, a trigger event and a response action or actions are selected. Automation stitches can be tested after they are created.

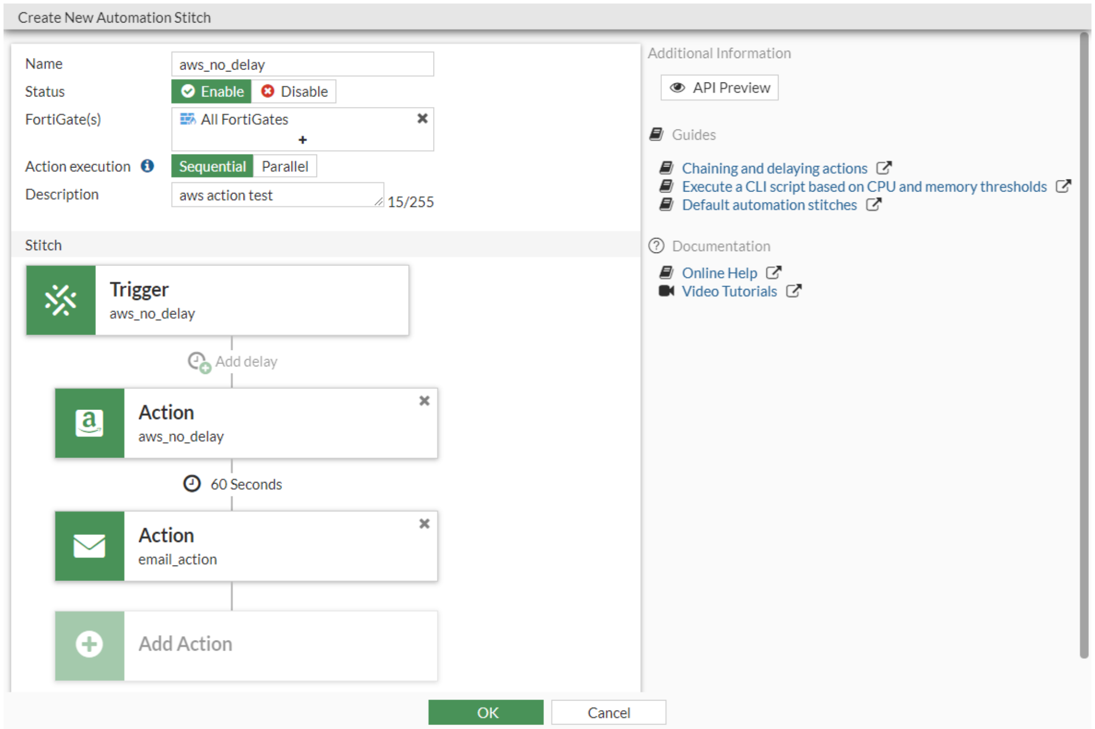
In the GUI, go to *Security Fabric > Automation* and click *Create New*. Automation stitches, actions, and triggers are configured in separate dialogs.

The stitch *Action execution* can be set to either *Sequential* or *Parallel*. In sequential execution, actions will execute one after another with a delay (if specified). If one action fails, then the action chain stops. This is the default setting. In parallel execution, all actions will execute immediately when the stitch is triggered.

When creating a stitch, clicking *Add Trigger* and *Add Action* displays a list of available triggers and actions, and the option to create a new one.



Once the stitch is configured, a process diagram of the trigger, actions, and delays is displayed. A delay can be added before an action if *Sequential* action execution is used. Executing the next action can be delayed by up to 3600 seconds (one hour).



Triggers and actions can be configured separately, and then added to an automation stitch.

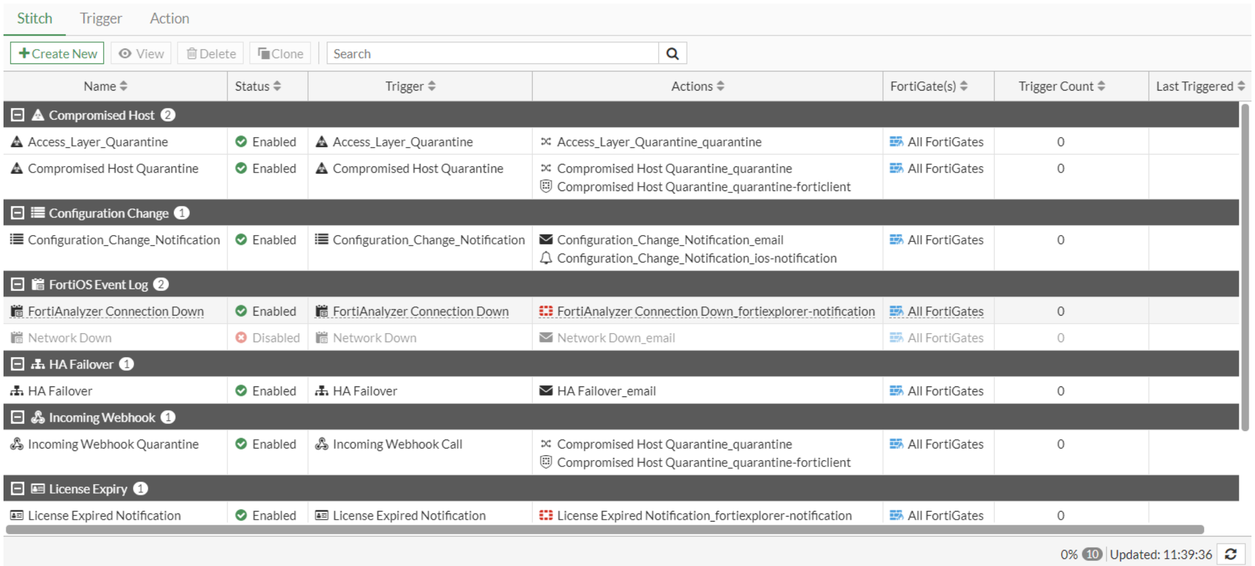
The maximum number of automation stitches that are allowed to run concurrently can be configured in the CLI (32 - 256, default = 128).

To configure the maximum number of concurrent automation stitches:

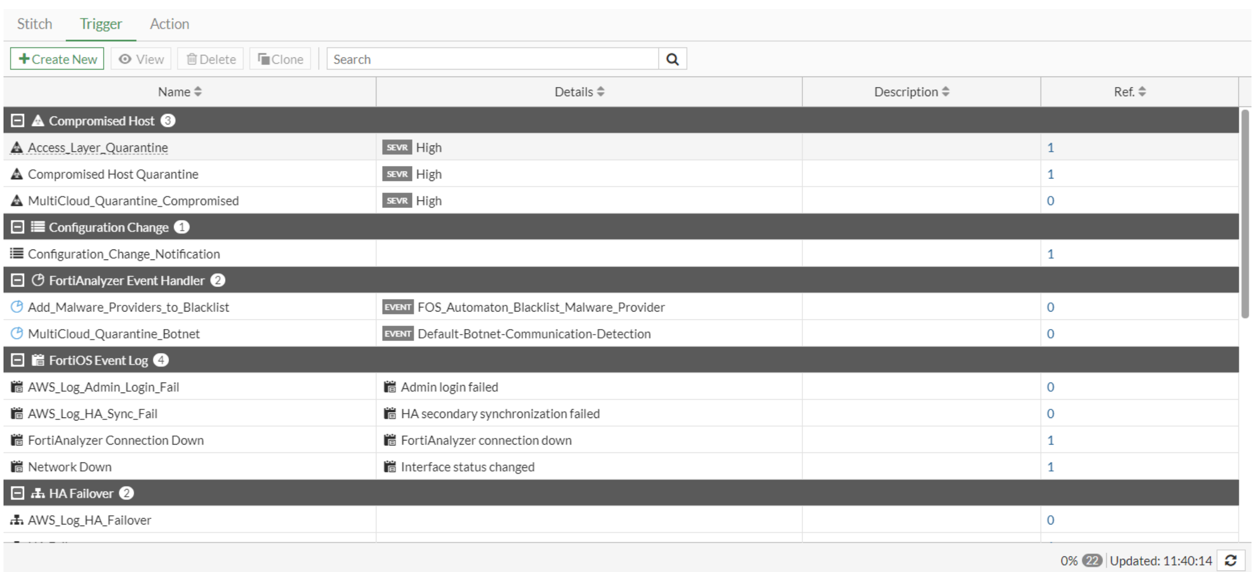
```
config automation setting
    set max-concurrent-stitches <integer>
end
```

Tabs on the Automation page

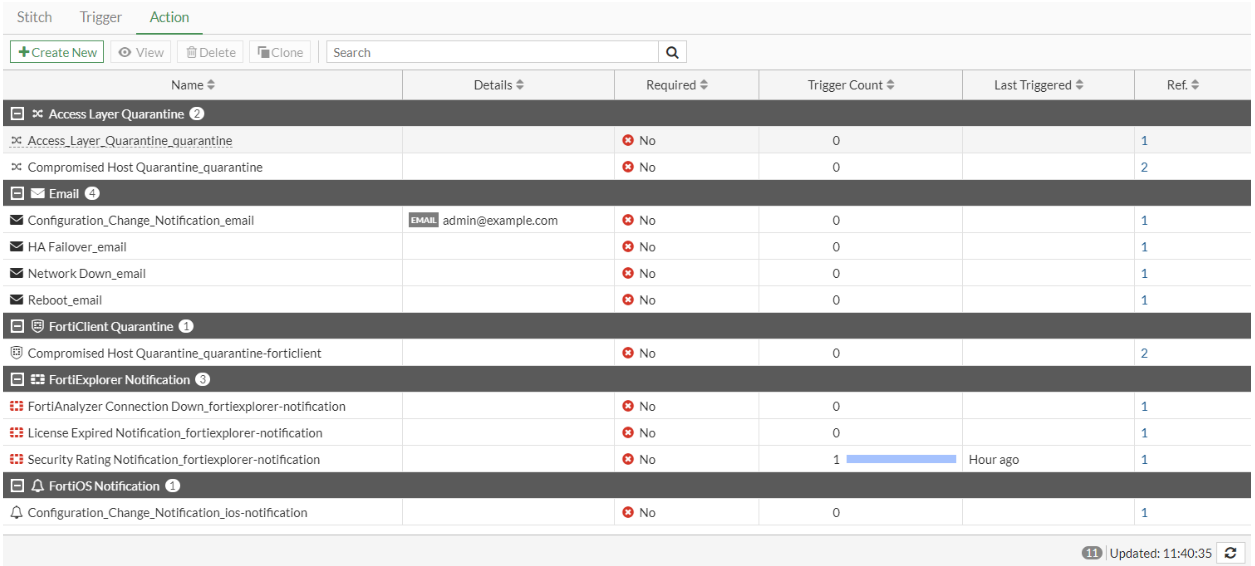
On the *Security Fabric* > *Automation* page, there are tabs for *Stitch*, *Trigger*, and *Action*. The *Stitch* tab is the default view that lists the trigger and actions used in each stitch. Individual triggers and actions can be created or edited in the corresponding tabs.



Click *Trigger* to view the list of triggers.



Click *Action* to view the list of actions.



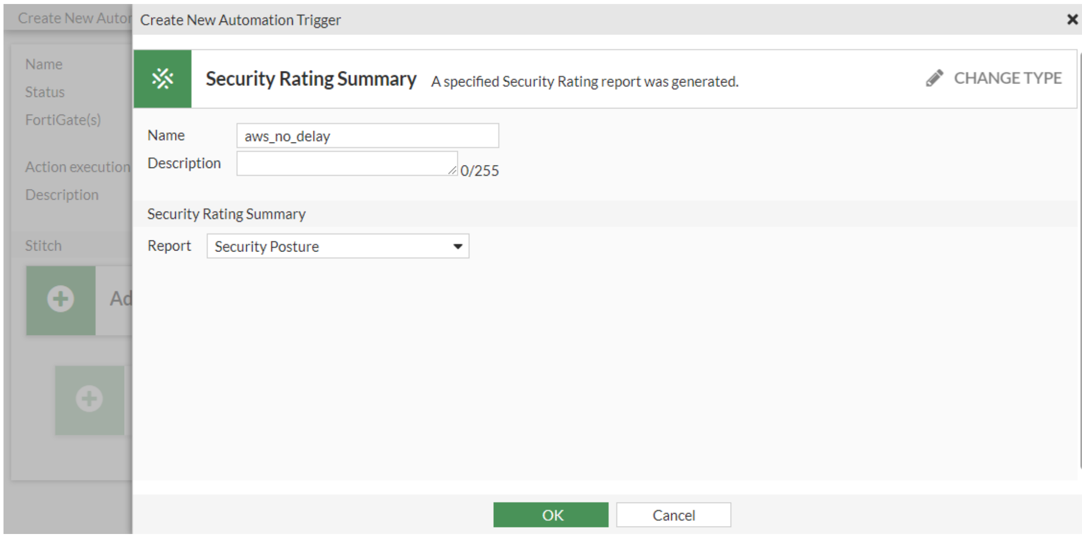
Sample configuration

The following example shows how to configure a Security Rating Summary automation stitch with AWS Lambda and Email actions. There is a 60-second delay before the Email action.

To configure the automation stitch in the GUI:

1. Go to *Security Fabric > Automation* and click *Create New*.
2. Enter the stitch name and description.
3. Configure the trigger:
   a. Click *Add Trigger*.
   b. Click *Create* and select *Security Rating Summary*.
   c. Enter the following:

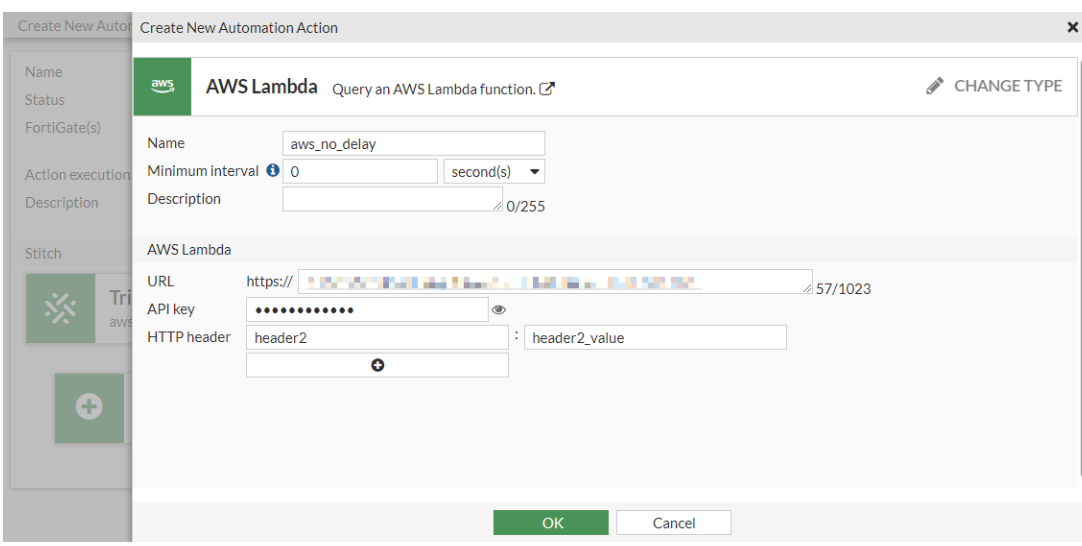
Name	*aws_no_delay*
Report	*Security Posture*



d. Click *OK*.
e. Select the trigger in the list and click *Apply*.

4. Configure the AWS Lambda function action:
   a. Click *Add Action*.
   b. Click *Create* and select *AWS Lambda*.
   c. Enter the following:

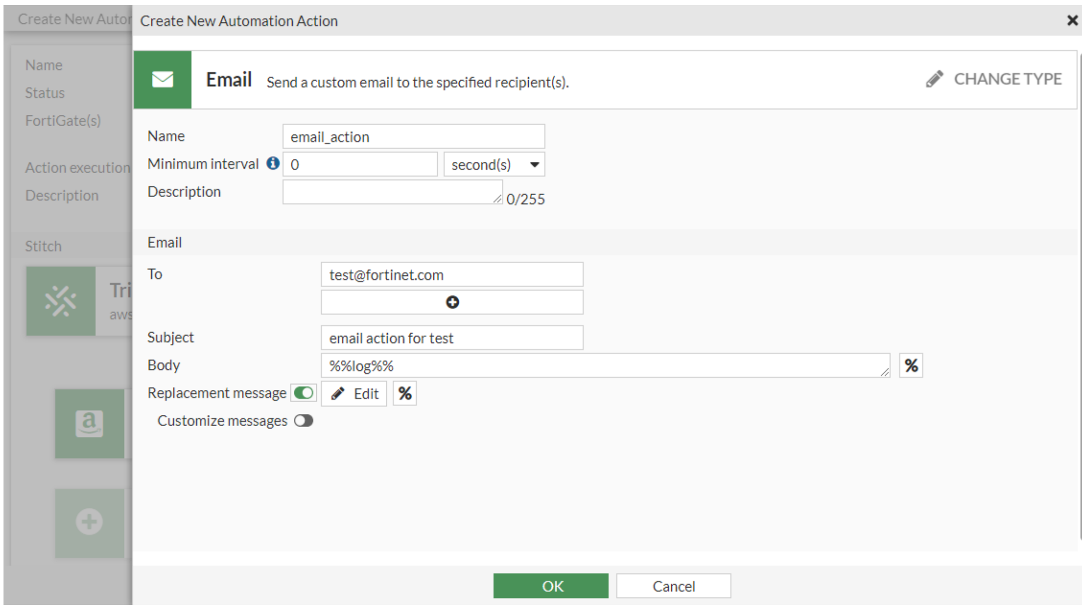
Name	*aws_no_delay*
URL	Enter the request API URI
API key	Enter the API key
HTTP header	*header2 : header2_value*



d. Click *OK*.
e. Select the action in the list and click *Apply*.

5. Configure the Email notification action:
   a. Click *Add Action*.
   b. Click *Create* and select *Email*.
   c. Enter the following:

Name	*email_action*
To	Enter an email address
Subject	*email action for test*
Replacement message	Enable



   d. Click *OK*.
   e. Select the action in the list and click *Apply*.
6. Click the *Add delay* located between both actions. Enter *60* and click *OK*.
7. Click *OK*.

To configure the automation stitch in the CLI:

1. Configure the trigger:

```
config system automation-trigger
    edit "aws_no_delay"
        set event-type security-rating-summary
    next
end
```

2. Configure the actions:

```
config system automation-action
    edit "aws_no_delay"
        set action-type aws-lambda
        set aws-api-key xxxxxxxxxxxx
        set uri "xxxxxxxxxx.execute-api.us-east-1.amazonaws.com/xxxxxxxxxx"
        set headers "header2:header2_value"
```

```
        next
        edit "email_action"
            set action-type email
            set email-to "test@fortinet.com"
            set email-subject "email action for test"
            set replacement-message enable
        next
    end
```

3. Configure the stitch:

```
    config system automation-stitch
        edit "aws_no_delay"
            set description "aws action test"
            set trigger "aws_no_delay"
            config actions
                edit 1
                    set action "aws_no_delay"
                    set required enable
                next
                edit 2
                    set action "email_action"
                    set delay 60
                    set required enable
                next
            end
        next
    end
```

Testing automation stitches

In the GUI, go to *Security Fabric > Automation*, right-click on the automation stitch and select *Test Automation Stitch*.

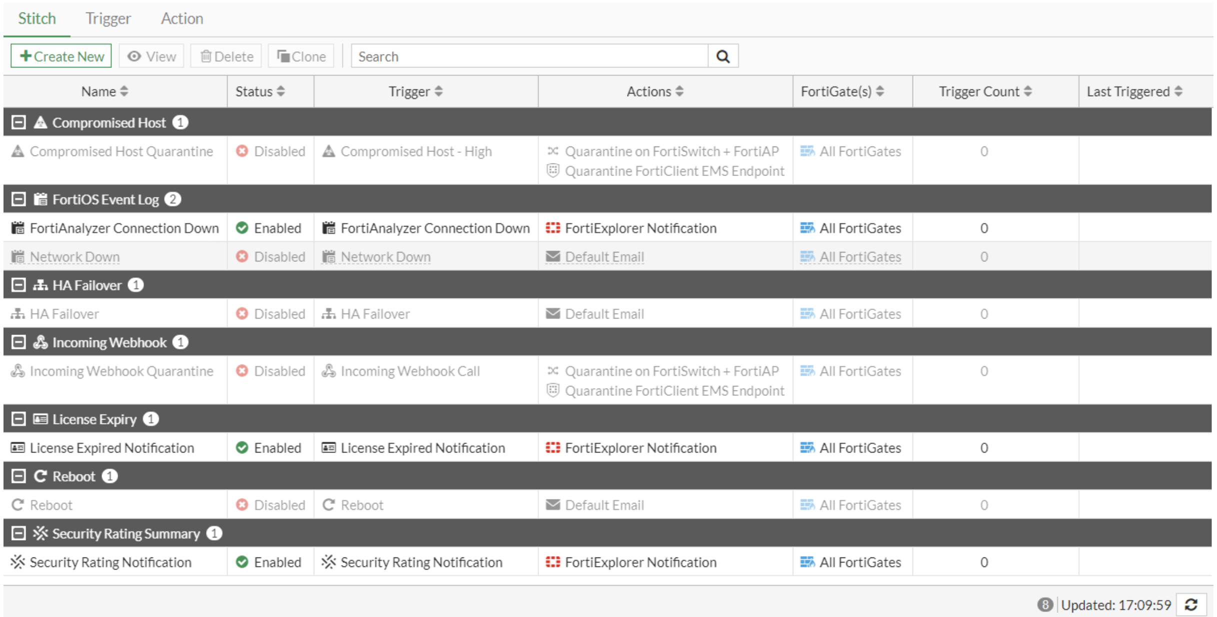
In the CLI, enter `diagnose automation test <automation-stitch name>`.

Default automation stitches

The following default automation stitches are included in FortiOS:

- Compromised Host Quarantine
- Incoming Webhook Quarantine
- HA Failover
- Network Down
- Reboot
- FortiAnalyzer Connection Down
- License Expired Notification
- Security Rating Notification

To view and edit the automation stitches in the GUI, go to *Security Fabric > Automation*.



CLI configurations

Compromised Host Quarantine

```
config system automation-action
    edit "Quarantine on FortiSwitch + FortiAP"
        set description "Default automation action configuration for quarantining a MAC
address on FortiSwitches and FortiAPs."
        set action-type quarantine
    next
    edit "Quarantine FortiClient EMS Endpoint"
        set description "Default automation action configuration for quarantining a
FortiClient EMS endpoint device."
        set action-type quarantine-forticlient
    next
end

config system automation-trigger
    edit "Compromised Host - High"
        set description "Default automation trigger configuration for when a high severity
compromised host is detected."
    next
end

config system automation-stitch
    edit "Compromised Host Quarantine"
        set description "Default automation stitch to quarantine a high severity compromised
host on FortiAPs, FortiSwitches, and FortiClient EMS."
        set status disable
        set trigger "Compromised Host - High"
        config actions
            edit 1
                set action "Quarantine on FortiSwitch + FortiAP"
            next
```

```
            edit 2
                set action "Quarantine FortiClient EMS Endpoint"
            next
        end
    next
end
```

FortiAnalyzer Connection Down

```
config system automation-action
    edit "FortiExplorer Notification"
        set description "Default automation action configuration for sending a notification
to any FortiExplorer mobile application."
        set action-type fortiexplorer-notification
    next
end
```

```
config system automation-trigger
    edit "FortiAnalyzer Connection Down"
        set description "Default automation trigger configuration for when the FortiAnalyzer
connection is lost."
        set event-type event-log
        set logid 22902
    next
end
```

```
config system automation-stitch
    edit "FortiAnalyzer Connection Down"
        set description "Default automation stitch to send a FortiExplorer notification when
the connection to FortiAnalyzer is lost."
        set trigger "FortiAnalyzer Connection Down"
        config actions
            edit 1
                set action "FortiExplorer Notification"
            next
        end
    next
end
```

Network Down

```
config system automation-action
     edit "Default Email"
        set description "Default automation action configuration for sending an email with
basic information on the log event."
        set action-type email
        set email-subject "%%log.logdesc%%"
    next
end
```

```
config system automation-trigger
    edit "Network Down"
        set description "Default automation trigger configuration for when a network
connection goes down."
        set event-type event-log
        set logid 20099
        config fields
```

```
            edit 1
                set name "status"
                set value "DOWN"
            next
        end
    next
end
```

```
config system automation-stitch
    edit "Network Down"
        set description "Default automation stitch to send an email when a network goes
down."
        set status disable
        set trigger "Network Down"
        config actions
            edit 1
                set action "Default Email"
            next
        end
    next
end
```

HA Failover

```
config system automation-action
    edit "Default Email"
        set description "Default automation action configuration for sending an email with
basic information on the log event."
        set action-type email
        set email-subject "%%log.logdesc%%"
    next
end
```

```
config system automation-trigger
    edit "HA Failover"
        set description "Default automation trigger configuration for when an HA failover
occurs."
        set event-type ha-failover
    next
end
```

```
config system automation-stitch
    edit "HA Failover"
        set description "Default automation stitch to send an email when a HA failover is
detected."
        set status disable
        set trigger "HA Failover"
        config actions
            edit 1
                set action "Default Email"
            next
        end
    next
end
```

Incoming Webhook Quarantine

```
config system automation-action
    edit "Quarantine on FortiSwitch + FortiAP"
        set description "Default automation action configuration for quarantining a MAC
address on FortiSwitches and FortiAPs."
        set action-type quarantine
    next
    edit "Quarantine FortiClient EMS Endpoint"
        set description "Default automation action configuration for quarantining a
FortiClient EMS endpoint device."
        set action-type quarantine-forticlient
    next
end
```

```
config system automation-trigger
    edit "Incoming Webhook Call"
        set description "Default automation trigger configuration for an incoming webhook."
        set event-type incoming-webhook
    next
end
```

```
config system automation-stitch
    edit "Incoming Webhook Quarantine"
        set description "Default automation stitch to quarantine a provided MAC address on
FortiAPs, FortiSwitches, and FortiClient EMS using an Incoming Webhook."
        set trigger "Incoming Webhook Call"
        config actions
            edit 1
                set action "Quarantine on FortiSwitch + FortiAP"
            next
            edit 2
                set action "Quarantine FortiClient EMS Endpoint"
            next
        end
    next
end
```

License Expired Notification

```
config system automation-action
    edit "FortiExplorer Notification"
        set description "Default automation action configuration for sending a notification
to any FortiExplorer mobile application."
        set action-type fortiexplorer-notification
    next
end
```

```
config system automation-trigger
    edit "License Expired Notification"
        set description "Default automation trigger configuration for when a license is near
expiration."
        set event-type license-near-expiry
        set license-type any
    next
end
```

```
config system automation-stitch
    edit "License Expired Notification"
        set description "Default automation stitch to send a FortiExplorer notification when
a license is near expiration."
        set trigger "License Expired Notification"
        config actions
            edit 1
                set action "FortiExplorer Notification"
            next
        end
    next
end
```

Reboot

```
config system automation-action
    edit "Default Email"
        set description "Default automation action configuration for sending an email with
basic information on the log event."
        set action-type email
        set email-subject "%%log.logdesc%%"
    next
end
```

```
config system automation-trigger
    edit "Reboot"
        set description "Default automation trigger configuration for when a FortiGate is
rebooted."
        set event-type reboot
    next
end
```

```
config system automation-stitch
    edit "Reboot"
        set description "Default automation stitch to send an email when a FortiGate is
rebooted."
        set status disable
        set trigger "Reboot"
        config actions
            edit 1
                set action "Default Email"
            next
        end
    next
end
```

Security Rating Notification

```
config system automation-action
    edit "FortiExplorer Notification"
        set description "Default automation action configuration for sending a notification
to any FortiExplorer mobile application."
        set action-type fortiexplorer-notification
    next
end
```

```
config system automation-trigger
    edit "Security Rating Notification"
        set description "Default automation trigger configuration for when a new Security 
Rating report is available."
        set event-type security-rating-summary
        set report-type any
    next
end
```

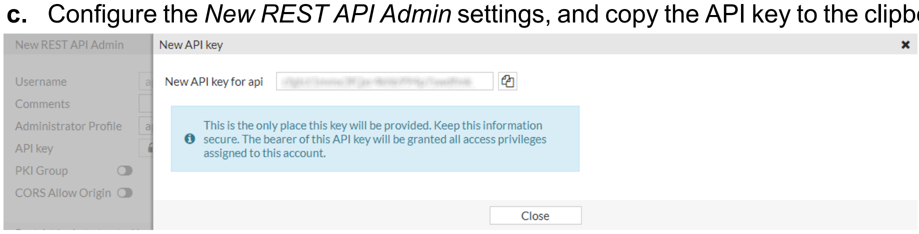
```
config system automation-stitch
    edit "Security Rating Notification"
        set description "Default automation stitch to send a FortiExplorer notification when 
a new Security Rating report is available."
        set trigger "Security Rating Notification"
        config actions
            edit 1
                set action "FortiExplorer Notification"
            next
        end
    next
end
```

Incoming Webhook Quarantine stitch

The Incoming Webhook Quarantine stitch for API calls to the FortiGate accepts multiple parameters (MAC address and FortiClient UUID) from an Incoming Webhook trigger, which enacts either the Access Layer Quarantine action (MAC address) or the FortiClient Quarantine action (FortiClient UUID). This is a default automation stitch included in FortiOS.

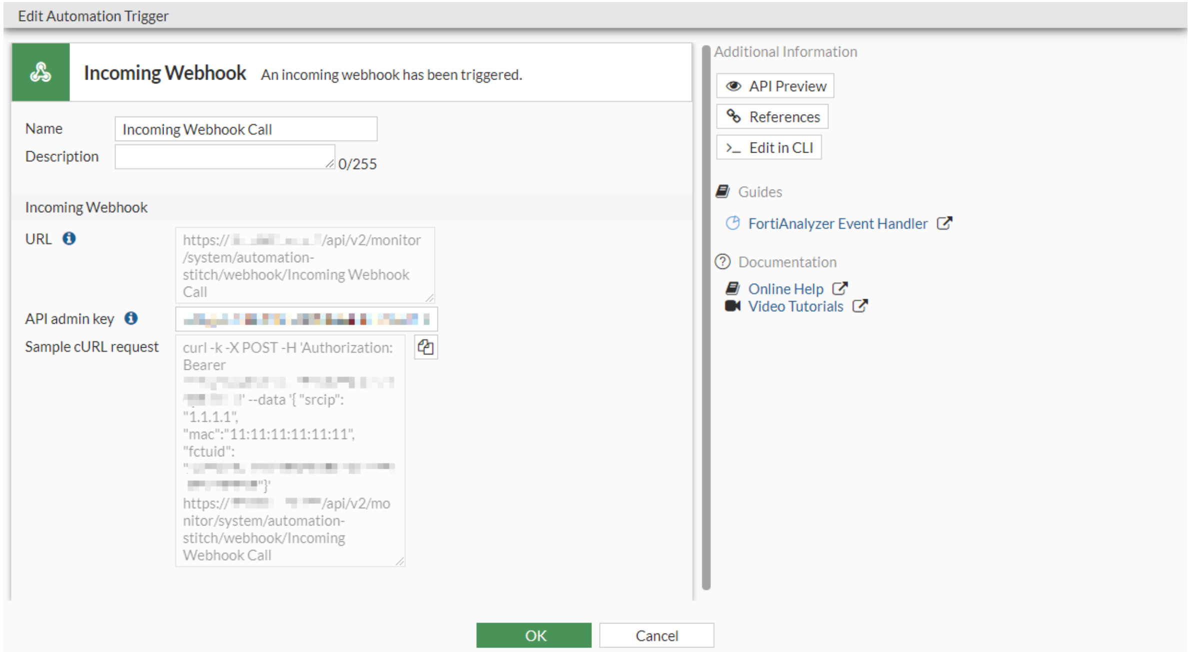
To trigger the Incoming Webhook Quarantine stitch in the GUI:

1. Create a new API user:
   a. Go to *System > Administrators*.
   b. Click *Create New > REST API Admin*.
   c. Configure the *New REST API Admin* settings, and copy the API key to the clipboard.



2. Enable the stitch:
   a. Go to *Security Fabric > Automation*.
   b. Under *Incoming Webhook*, right-click *Incoming Webhook Quarantine*, and select *Select Status > Enable*.
3. Get the sample cURL request:
   a. Click the *Trigger* trigger tab.
   b. Under *Incoming Webhook*, right-click *Incoming Webhook Call*, and select *Edit*.

c. In the *API admin key* field, enter the API key you recorded previously. The *Sample cURL request* field updates.



d. Copy the *Sample cURL request* to the clipboard.

e. Click *OK*.

4. Execute the request:

   a. Edit the sample cURL request you just copied.

   b. Add parameters to the `data` field (`"mac"` and `"fctuid"`), and then execute the request.

```
root@pc:~# curl -k -X POST -H 'Authorization: Bearer
cfgtct1mmx3fQxr4khb994p7swdfmk' --data '{ "mac":"0c:0a:00:0c:ce:b0", "fctuid":
"0000BB0B0ABD0D00B0D0A0B0E0F0B00B"}'
https://172.16.116.226/api/v2/monitor/system/automation-
stitch/webhook/Incoming%20Webhook%20Quarantine
{
   "http_method":"POST",
   "status":"success",
   "http_status":200,
   "serial":"FGT00E0Q00000000",
   "version":"v6.4.0",
   "build":1545
```

Encode spaces in the automation stitch name with `%20`. For example, `Incoming%20Webhook%20Quarantine`

Once the automation stitch is triggered, the MAC address is quarantined by the FortiGate, and an event log is created. The FortiClient UUID is quarantined on the EMS server side.

To trigger the Incoming Webhook Quarantine stitch in the CLI:

1. Create a new API user and note the API key:

```
config system api-user
```

2. Enable the automation stitch:

```
config system api-user
    edit "api"
        set api-key ************
        set accprofile "api_profile"
        set vdom "root"
        config trusthost
            edit 1
                set ipv4-trusthost 10.6.30.0 200.200.200.0
            next
        end
    next
end
```

3. Edit the cURL request to include parameters in the `data` field (`"mac"` and `"fctuid"`), then execute the request:

```
root@pc56:~# curl -k -X POST -H 'Authorization: Bearer
cfgtct1mmx0fQxr4khb000p70wdfmk' --data '{ "mac":"0c:0a:00:0c:ce:b0", "fctuid":
"3000BB0B0ABD0D00B0D0A0B0E0F0B00B"}'
https://100.10.100.200/api/v2/monitor/system/automation-
stitch/webhook/Incoming%20Webhook%20Quarantine
{
  "http_method":"POST",
  "status":"success",
  "http_status":200,
  "serial":"FGT80E0Q00000000",
  "version":"v6.4.0",
  "build":1545
```

Encode spaces in the automation stitch name with `%20`. For example, `Incoming%20Webhook%20Quarantine`.

Once the automation stitch is triggered, the MAC address is quarantined by the FortiGate, and an event log is created. The FortiClient UUID is quarantined on the EMS server side.

Sample log

```
date=2020-02-14 time=15:37:48 logid="0100046600" type="event" subtype="system"
level="notice" vd="root" eventtime=1581723468644200712 tz="-0800" logdesc="Automation
stitch triggered" stitch="Incoming Webhook Quarantine" trigger="Incoming Webhook
Quarantine" stitchaction="Compromised Host Quarantine_quarantine,Compromised Host
Quarantine_quarantine-forticlient" from="log" msg="stitch:Incoming Webhook Quarantine is
triggered."
```

Triggers

The following table outlines the available triggers.

Category	Trigger	Description
Security Fabric		

Category	Trigger	Description
	Compromised Host	An indicator of compromise (IoC) is detected on a host endpoint. The threat level must be selected and can be *Medium* or *High*. If *Medium* is selected, both medium and high level threats are included. Additional actions are available only for *Compromised Host* triggers: • Access Layer Quarantine • FortiClient Quarantine • VMware NSX Security Tag • IP Ban
	Security Rating Summary	A summary is available for a recently run Security Rating report. Options include: • Security Posture • Fabric Coverage • Optimization • Any
	FortiAnalyzer Event Handler	The specified FortiAnalyzer event handler has occurred. See FortiAnalyzer event handler trigger on page 2359 for details.
	FortiGate Cloud Event Handler	The specified FortiGate Cloud event handler has occurred. This option requires a FortiGate Cloud log retention license.
	Fabric Connector Event	An event has occurred on a specific Fabric connector. See Fabric connector event trigger on page 2364 for details.
	FortiGate Cloud-Based IOC	IOC detection from the FortiGate Cloud IOC service. This option requires an IOC license, a web filter license, and FortiCloud logging must be enabled.
System		
	Reboot	A FortiGate is rebooting.
	HA Failover	An HA failover is occurring.
	Conserve Mode	A FortiGate entered conserve mode due to low memory. See Execute a CLI script based on CPU and memory thresholds on page 2410 for an example.
	Configuration Change	A FortiGate configuration change has occurred.
	License Expiry	A FortiGuard license is expiring. The license type must be selected. Options include: • FortiCare Support • FortiGuard Web Filter • FortiGuard AntiSpam • FortiGuard AntiVirus • FortiGuard IPS • FortiGuard Management Service • FortiGate Cloud

Category	Trigger	Description
		• Any
	AV & IPS DB Update	The antivirus and IPS database is updating.
	High CPU	A FortiGate has high CPU usage. See Execute a CLI script based on CPU and memory thresholds on page 2410 for an example.
Event Log Category		
	Anomaly Logs	An anomalous event has occurred. See Event log category triggers on page 2370 for an example.
	IPS Logs	An IPS event has occurred.
	SSH Logs	An SSH event has occurred.
	Traffic Violation	A traffic policy has been violated.
	Virus Logs	A virus event has occurred.
	Web Filter Violation	A web filter policy has been violated.
Miscellaneous		
	FortiOS Event Log	The specified FortiOS log has occurred. Multiple event log IDs can be selected, and log field filters can be applied. See FortiOS event log trigger on page 2369 for an example.
	Incoming Webhook	An incoming webhook is triggered.
	Schedule	A scheduled monthly, weekly, daily, or hourly trigger. Set to occur on a specific minute of an specific hour on a specific day.

FortiAnalyzer event handler trigger

You can trigger automation stitches based on FortiAnalyzer event handlers. This allows you to define rules based on complex correlations across devices, log types, frequencies, and other criteria.

To set up a FortiAnalyzer event handler trigger:

1. Configure a FortiGate event handler on the FortiAnalyzer
2. Configure FortiAnalyzer logging on the FortiGate on page 2360
3. Configure an automation stitch that is triggered by a FortiAnalyzer event handler on page 2361

Configure a FortiGate event handler on the FortiAnalyzer

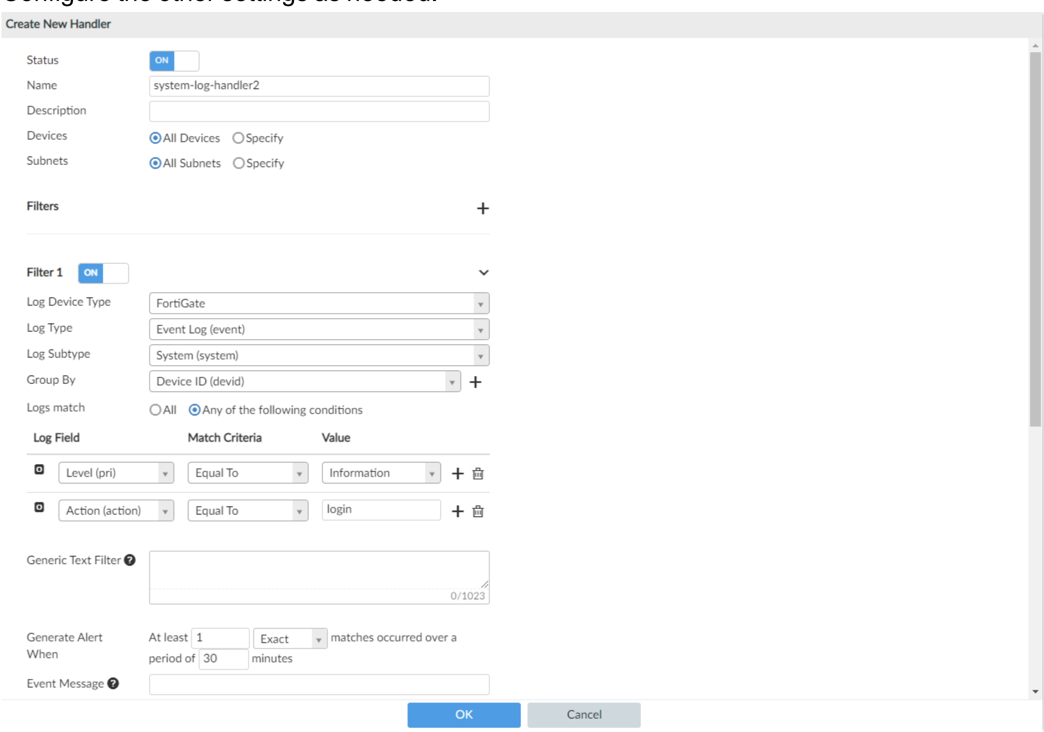
On the FortiAnalyzer, configure an event handler for the automation stitch. In this example, the event handler is triggered when an administrator logs in to the FortiGate. See Creating a custom event handler in the FortiAnalyzer Administration Guide for more information.

To configure an event handler on the FortiAnalyzer:

1. Go to *FortiSoC* > *Handlers* > *FortiGate Event Handlers*, and click *Create New*.
2. Configure an event handler with two conditions for the automation stitch:

Log Type	Event Log
Log Subtype	System
Group By	Device ID
Logs match	Any of the following conditions
Log Field	Level
Match Criteria	Equal To
Value	Information
Log Field	Action
Match Criteria	Equal To
Value	login

3. Configure the other settings as needed.



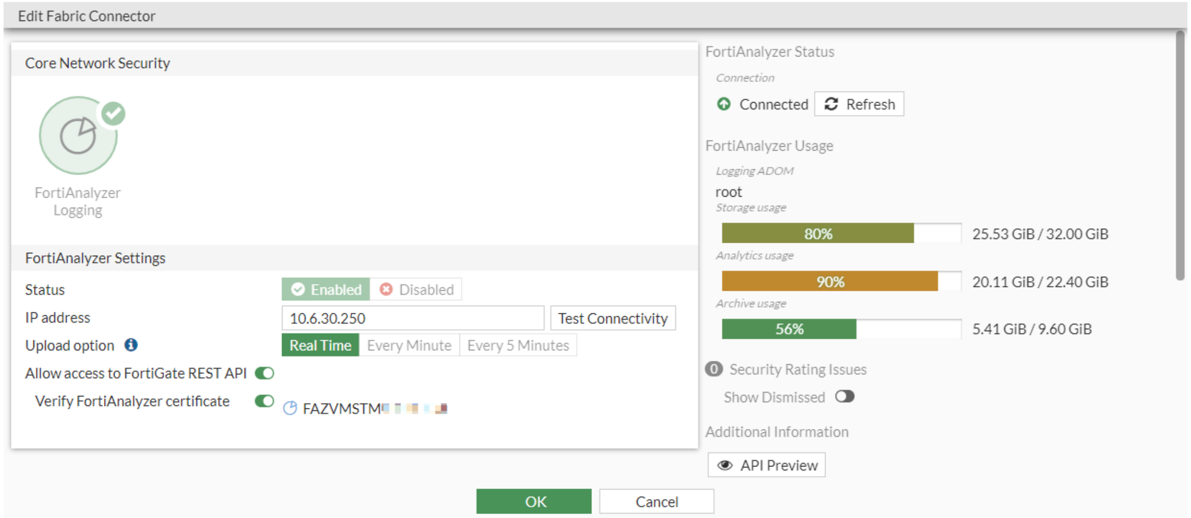
4. Click *OK*.

Configure FortiAnalyzer logging on the FortiGate

See Configuring FortiAnalyzer on page 2215 for more information.

To configure FortiAnalyzer logging in the GUI:

1. Go to *Security Fabric > Fabric Connectors* and double-click the *FortiAnalyzer Logging* card.
2. Ensure the *Status* is *Enabled*, and configure the settings as needed.



3. Click *OK*.

To configure FortiAnalyzer logging in the CLI:

```
config log fortianalyzer setting
    set status enable
    set server "10.6.30.250"
    set serial "FL-4HET000000000"
    set upload-option realtime
    set reliable enable
end
```

Configure an automation stitch that is triggered by a FortiAnalyzer event handler

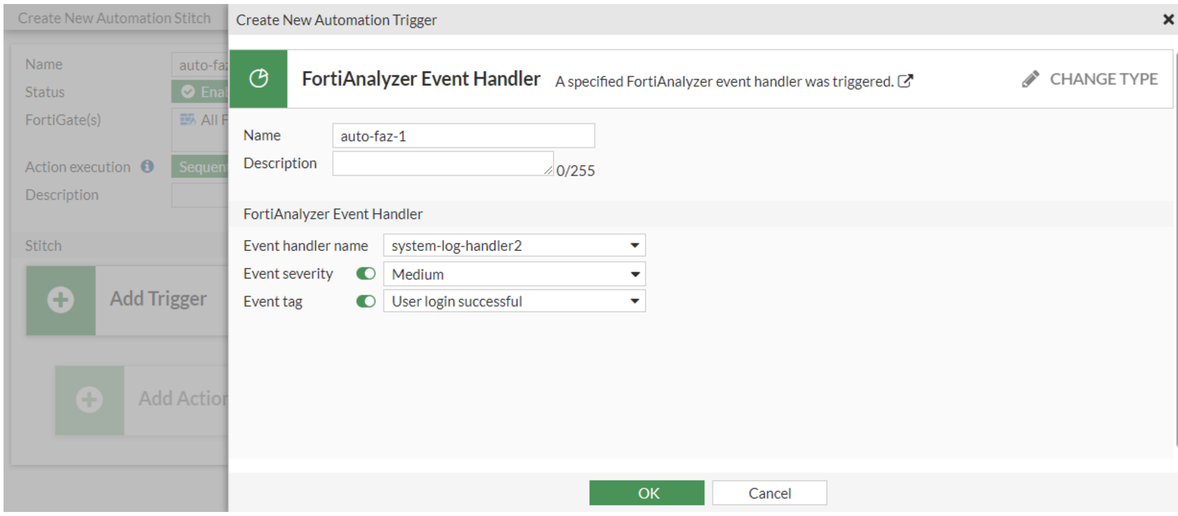
When a FortiAnalyzer event handler is triggered, it sends a notification to the FortiGate automation framework, which generates a log and triggers the automation stitch.

To configure an automation stitch that is triggered by a FortiAnalyzer event handler in the GUI:

1. Go to *Security Fabric > Automation* and click *Create New*.
2. Enter the stitch name, *auto-faz-1*.
3. Configure the trigger:
   a. Click *Add Trigger*.
   b. Click *Create* and select *FortiAnalyzer Event Handler*.
   c. Enter the following:

Name	auto-faz-1
Event handler name	system-log-handler2
Event severity	Medium

Event tag	User login successful

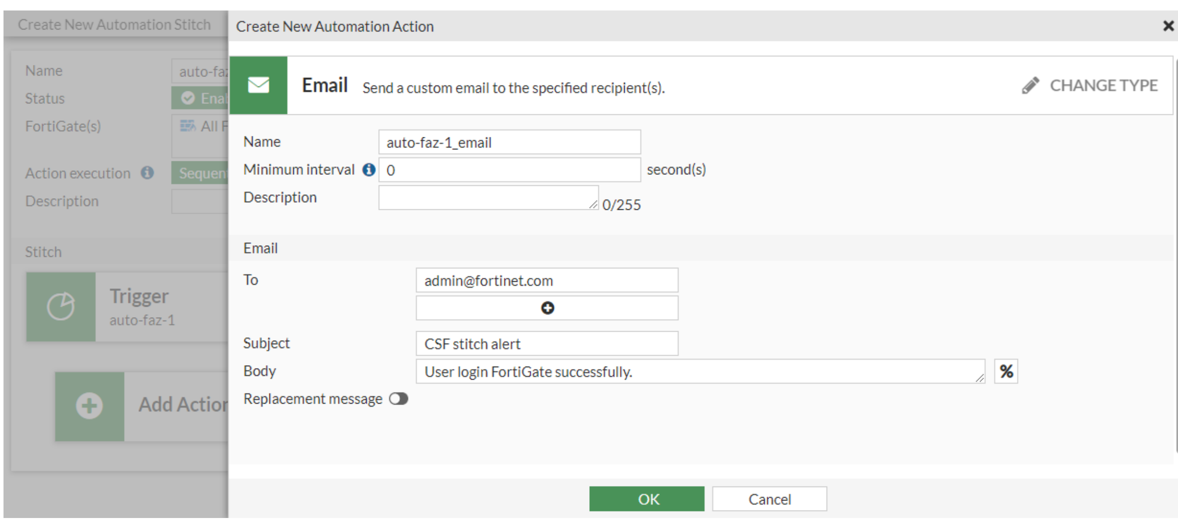


- **d.** Click *OK*.
- **e.** Select the trigger in the list and click *Apply*.

4. Configure the Email notification action:

- **a.** Click *Add Action*.
- **b.** Click *Create* and select *Email*.
- **c.** Enter the following:

Name	auto-faz-1_email
To	Enter an email address
Subject	CSF stitch alert
Body	User login FortiGate successfully.



- **d.** Click *OK*.
- **e.** Select the action in the list and click *Apply*.

5. Click *OK*.

To configure an automation stitch that is triggered by a FortiAnalyzer event handler in the CLI:

1. Create an automation trigger:

```
config system automation-trigger
    edit "auto-faz-1"
        set event-type faz-event
        set faz-event-name "system-log-handler2"
        set faz-event-severity "medium"
        set faz-event-tags "User log in successful"
    next
end
```

2. Create an automation action:

```
config system automation-action
    edit "auto-faz-1_email"
        set action-type email
        set email-to "admin@fortinet.com"
        set email-subject "CSF stitch alert"
        set message "User login FortiGate successfully."
    next
end
```

3. Create the automation stitch:

```
config system automation-stitch
    edit "auto-faz-1"
        set trigger "auto-faz-1"
        config actions
            edit 1
                set action "auto-faz-1_email"
                set required enable
            next
        end
    next
end
```

View the trigger event log

To view the trigger event log in the GUI:

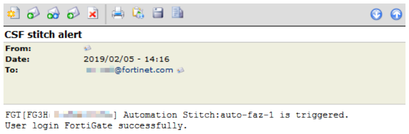
1. Log in to the FortiGate.
   The FortiAnalyzer sends a notification to the FortiGate automation framework, generates an event log on the FortiGate, and triggers the automation stitch.
2. Go to *Log & Report > System Events* and select *General System Events*. From the log location dropdown, select *FortiAnalyzer*.

To view the trigger event log in the CLI:

```
# execute log display
    ...
    date=2019-02-05 time=14:16:17 logid="0100046600" type="event" subtype="system"
level="notice" vd="root" eventtime=1549404977 logdesc="Automation stitch triggered"
stitch="auto-faz-1" trigger="auto-faz-1" from="log" msg="stitch:auto-faz-1 is triggered."
    ...
```

Sample email

The email sent by the action will look similar to the following:



Fabric connector event trigger

With the *Fabric Connector Event* trigger, any supported Fabric connector is able to trigger an automation stitch on the FortiGate based on a specific event defined on the Fabric connector. Currently, only FortiDeceptor 4.1 supports this trigger for the *Insider Threat*, *Notify Ban*, and *Notify Unban* events.

In the following example, an authorized FortiDeceptor in the Security Fabric deploys a decoy called ubuntu16 configured with SSH, SAMBA, HTTP, and HTTPS services.

This example assumes the Security Fabric is already configured. Refer to Configuring the root FortiGate and downstream FortiGates and FortiDeceptor for detailed configuration steps. On the root FortiGate, the *Allow downstream device REST API access* option must be enabled (`set downstream-access enable`). The minimum permission required for the selected *Administrator profile* is *Read/Write* for *User & Device* (`set authgrp read-write`).

Three stitches are configured, one for each FortiDeceptor trigger type:

Stitch name	Fabric connector event trigger	Actions
fortideceptor_threat	Insider threat	Email and IP ban
fortideceptor_ban	Notify ban	Email and IP ban
fortideceptor_unban	Notify unban	Email and CLI script

To configure stitches with the Fabric connector event trigger in the GUI:

1. Configure the triggers:
   a. Go to *Security Fabric > Automation*, select the *Trigger* tab, and click *Create New*.
   b. In the *Security Fabric* section, click *Fabric Connector Event* and enter the following:

Name	*fdc_Insider_Threat*
Description	*Insider_Threat*
Connector	Select the FortiDeceptor connector
Event Name	*Insider Threat*

   c. Click *OK*.
   d. Repeat these steps to create two more triggers with the following settings:

Name	*fdc_Notify_Ban*
Description	*Notify_Ban*

Connector	Select the FortiDeceptor connector
Event Name	*Notify Ban*

Name	*fdc_Notify_Unban*
Description	*Notify_Unban*
Connector	Select the FortiDeceptor connector
Event Name	*Notify Unban*

2. Configure the actions:
   a. Go to *Security Fabric > Automation*, select the *Action* tab, and click *Create New*.
   b. In the *Security Response* section, click *IP Ban* and enter the name, *fdc_ban-ip*.
   c. Click *OK*.
   d. Repeat these steps to create an *Email* (in the *Notifications* section) and a *CLI Script* (in the *General* section) action with the following settings:

Email	
Name	*email_log*
To	Enter an email address
Subject	*CSF stitch alert*

CLI Script	
Name	*fdc_unban*
Script	*diagnose user quarantine delete src4 %%log.srcip%%*
Administrator profile	*super_admin*

3. Configure the *fortideceptor_threat* stitch:
   a. Go to *Security Fabric > Automation*, select the *Stitch* tab, and click *Create New*.
   b. Enter the name, *fortideceptor_threat*.
   c. Click *Add Trigger*. Select *fdc_Insider_Threat* and click *Apply*.
   d. Click *Add Action*. Select *email_log* and click *Apply*.
   e. Click *Add Action*. Select *fdc_ban-ip* and click *Apply*.
   f. Click the *Add delay* located between both actions. Enter *5* and click *OK*.
   g. Click *OK*.
4. Configure the *fortideceptor_ban* stitch:
   a. Go to *Security Fabric > Automation*, select the *Stitch* tab, and click *Create New*.
   b. Enter the name, *fortideceptor_ban*.
   c. Click *Add Trigger*. Select *fdc_Notify_Ban* and click *Apply*.
   d. Click *Add Action*. Select *email_log* and click *Apply*.
   e. Click *Add Action*. Select *fdc_ban-ip* and click *Apply*.
   f. Click the *Add delay* located between both actions. Enter *5* and click *OK*.
   g. Click *OK*.

5. Configure the *fortideceptor_unban* stitch:
   a. Go to *Security Fabric > Automation*, select the *Stitch* tab, and click *Create New*.
   b. Enter the name, *fortideceptor_unban*.
   c. Click *Add Trigger*. Select *fdc_Notify_Unban* and click *Apply*.
   d. Click *Add Action*. Select *email_log* and click *Apply*.
   e. Click *Add Action*. Select *fdc_unban* and click *Apply*.
   f. Click the *Add delay* located between both actions. Enter *5* and click *OK*.
   g. Click *OK*.

To configure stitches with the Fabric connector event trigger in the CLI:

1. Configure the triggers:

```
config system automation-trigger
    edit "fdc_Insider_Threat"
        set description "Insider_Threat"
        set event-type fabric-event
        set serial "FDC-VMTM210000**"
        set fabric-event-name "insider_threat"
    next
    edit "fdc_Notify_Ban"
        set description "Notify_Ban"
        set event-type fabric-event
        set serial "FDC-VMTM210000**"
        set fabric-event-name "notify_ban"
    next
    edit "fdc_Notify_Unban"
        set description "Notify_Unban"
        set event-type fabric-event
        set serial "FDC-VMTM210000**"
        set fabric-event-name "notify_unban"
    next
end
```

2. Configure the actions:

```
config system automation-action
    edit "fdc_ban-ip"
        set action-type ban-ip
    next
    edit "fdc_unban"
        set action-type cli-script
        set script "diagnose user quarantine delete src4 %%log.srcip%%"
        set output-size 10
        set timeout 0
        set accprofile "super_admin"
    next
    edit "email_log"
        set action-type email
        set email-to "*******@fortinet.com"
        set email-subject "CSF stitch alert"
    next
end
```

3. Configure the stitches:

```
config system automation-stitch
    edit "fortideceptor_threat"
        set trigger "fdc_Insider_Threat"
        config actions
            edit 1
                set action "email_log"
                set required enable
            next
            edit 2
                set action "fdc_ban-ip"
                set delay 5
                set required enable
            next
        end
    next
    edit "fortideceptor_ban"
        set trigger "fdc_Notify_Ban"
        config actions
            edit 1
                set action "email_log"
                set required enable
            next
            edit 2
                set action "fdc_ban-ip"
                set delay 5
                set required enable
            next
        end
    next
    edit "fortideceptor_unban"
        set trigger "fdc_Notify_Unban"
        config actions
            edit 1
                set action "email_log"
                set required enable
            next
            edit 2
                set action "fdc_unban"
                set delay 5
                set required enable
            next
        end
    next
end
```

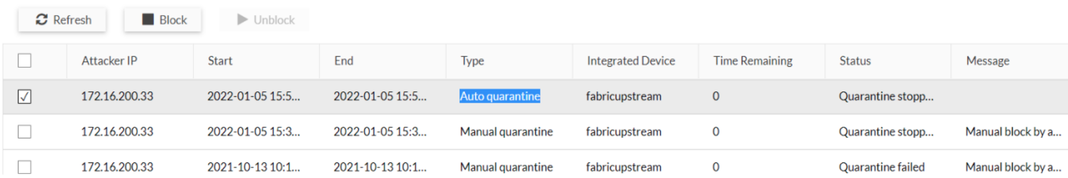
Verification

A device with IP 172.16.200.33 uses SSH to access the decoy (ubuntu16) deployed in the FortiDeceptor. The FortiDeceptor will detect the attacker IP 172.16.200.33, automatically quarantine it, and send the insider threat notification to the FortiGate. This notification will trigger the *fortideceptor_threat* stitch due to the insider threat event trigger, so an email alert is sent and the attacker IP (172.16.200.33) is banned.

In FortiDeceptor, if the attacker IP (172.16.200.33) is manually blocked or unblocked, the FortiDeceptor will send out the internal block or unblock notification to FortiGate (see Quarantine Status for more details). This notification will trigger the *fortideceptor_ban* or *fortideceptor_unban* stitch due the notify ban or unban event trigger. An email alert is sent, and based on the event, the IP is banned or the CLI script runs to unban the IP.

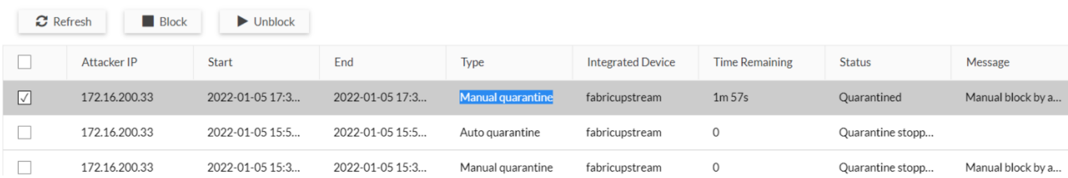
To view the quarantine details in FortiDeceptor:

1. Go to *Fabric* > *Quarantine Status*.
    a. Automatic quarantine:



	Attacker IP	Start	End	Type	Integrated Device	Time Remaining	Status	Message
☑	172.16.200.33	2022-01-05 15:5...	2022-01-05 15:5...	Auto quarantine	fabricupstream	0	Quarantine stopp...	
☐	172.16.200.33	2022-01-05 15:3...	2022-01-05 15:3...	Manual quarantine	fabricupstream	0	Quarantine stopp...	Manual block by a...
☐	172.16.200.33	2021-10-13 10:1...	2021-10-13 10:1...	Manual quarantine	fabricupstream	0	Quarantine failed	Manual block by a...

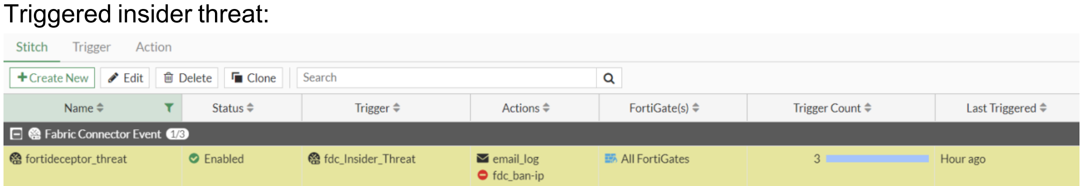
    b. Manual block or unblock:



	Attacker IP	Start	End	Type	Integrated Device	Time Remaining	Status	Message
☑	172.16.200.33	2022-01-05 17:3...	2022-01-05 17:3...	Manual quarantine	fabricupstream	1m 57s	Quarantined	Manual block by a...
☐	172.16.200.33	2022-01-05 15:5...	2022-01-05 15:5...	Auto quarantine	fabricupstream	0	Quarantine stopp...	
☐	172.16.200.33	2022-01-05 15:3...	2022-01-05 15:3...	Manual quarantine	fabricupstream	0	Quarantine stopp...	Manual block by a...

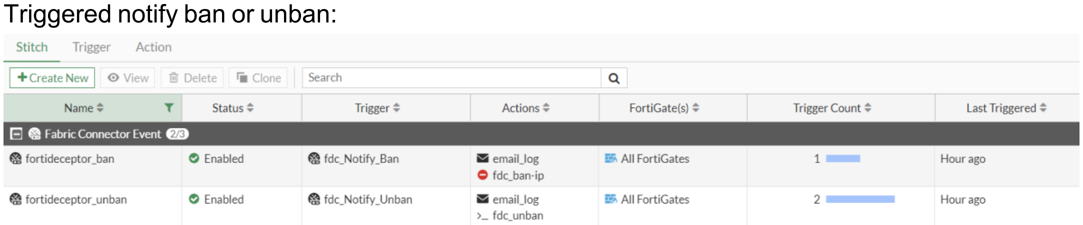
To confirm that the stitch was triggered in the FortiOS GUI:

1. Go to *Security Fabric* > *Automation* and select the *Stitch* tab.
    a. Triggered insider threat:



Name	Status	Trigger	Actions	FortiGate(s)	Trigger Count	Last Triggered
fortideceptor_threat	Enabled	fdc_Insider_Threat	email_log fdc_ban-ip	All FortiGates	3	Hour ago

    b. Triggered notify ban or unban:



Name	Status	Trigger	Actions	FortiGate(s)	Trigger Count	Last Triggered
fortideceptor_ban	Enabled	fdc_Notify_Ban	email_log fdc_ban-ip	All FortiGates	1	Hour ago
fortideceptor_unban	Enabled	fdc_Notify_Unban	email_log >_ fdc_unban	All FortiGates	2	Hour ago

To view the quarantined IP details in the FortiOS CLI:

```
# diagnose user quarantine list
src-ip-addr       created                   expires                  cause
172.16.200.33     Wed Jan  5 15:57:41 2022 indefinite               Administrative
```

If the IP is unbanned by the stitch, the list will be empty:

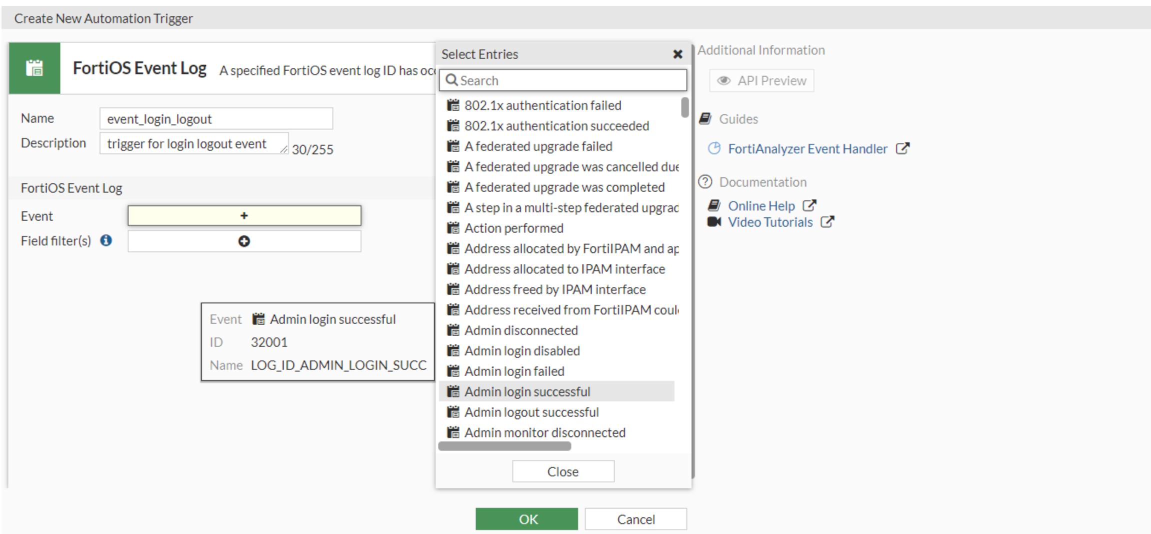
```
# diagnose user quarantine list
src-ip-addr       created                   expires                  cause
```

FortiOS event log trigger

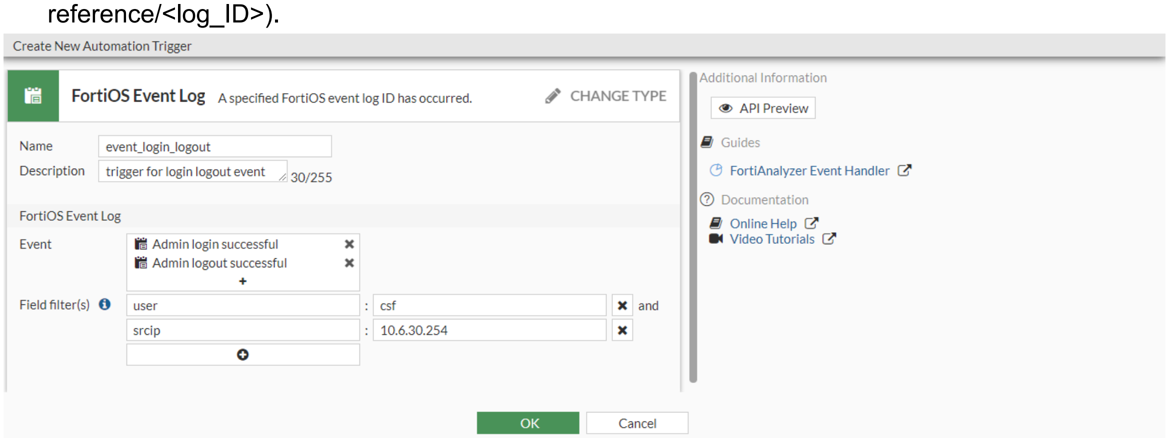
You can configure a FortiOS event log trigger for when a specific event log ID occurs. You can select multiple event log IDs, and apply log field filters.

To configure a FortiOS event log trigger in the GUI:

1. Go to *Security Fabric > Automation*, select the *Trigger* tab, and click *Create New*.
2. In the *Miscellaneous* section, click *FortiOS Event Log*.
3. Enter a name and description.
4. In the *Event* field, click the + to select multiple event log IDs.
   The *Event* options correspond to the *Message Meaning* listed in the FortiOS Log Message Reference. Hover over an entry to view the tooltip that includes the event ID and log name. In this example, the *Admin login successful* event in the GUI corresponds to log ID *32001*, which is *LOG_ID_ADMIN_LOGIN_SUCC*.



5. In the *Field filter(s)* field, click the + to add multiple field filters. The configured filters much match in order for the stitch to be triggered.
   a. To view the list of available fields for a log, refer to the FortiOS Log Message Reference by appending the log ID to the document URL (https://docs.fortinet.com/document/fortigate/7.2.0/fortios-log-message-reference/<log_ID>).



6. Click *OK*.

To configure a FortiOS event log trigger in the CLI:

```
config system automation-trigger
    edit "event_login_logout"
        set description "trigger for login logout event"
        set event-type event-log
        set logid 32001 32003
        config fields
            edit 1
                set name "user"
                set value "csf"
            next
            edit 2
                set name "srcip"
                set value "10.6.30.254"
            next
        end
    next
end
```

Event log category triggers

There are six automation triggers based on event log categories:

- Anomaly logs
- IPS logs
- SSH logs
- Traffic violations
- Virus logs
- Web filter violations

When multi VDOM mode is enabled, individual VDOMs can be specified so that the trigger is only applied to those VDOMs.

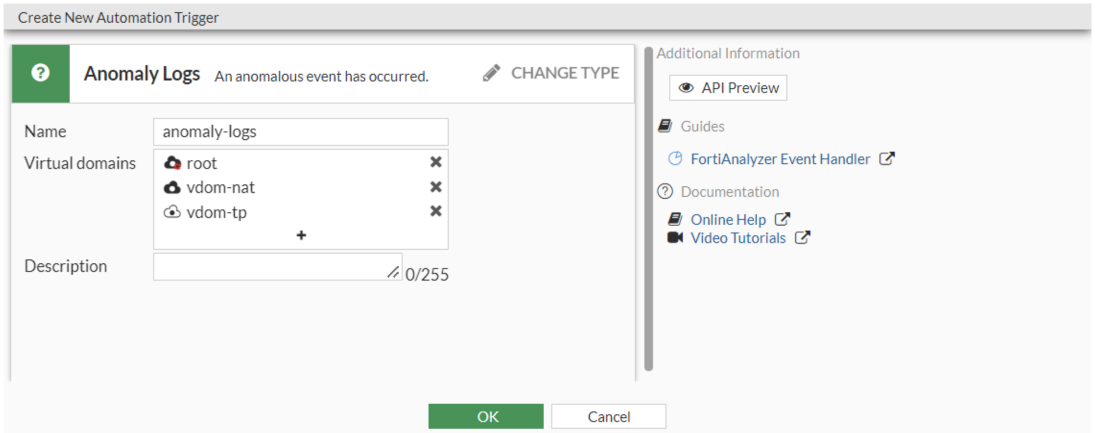
```
config system automation-trigger
    edit <name>
        set event-type {ips-logs | anomaly-logs | virus-logs | ssh-logs | webfilter-
violation | traffic-violation}
        set vdom <name>
    next
end
```

Example

In this example, an automation stitch is created that uses an anomaly logs trigger and an email notification action. The trigger specifies which VDOMs should be used. There is a three-second delay between the trigger and action.

To configure an automation stitch with the anomaly logs trigger in the GUI:

1. Configure the trigger:
   a. Go to *Security Fabric > Automation*, select the *Trigger* tab, and click *Create New*.
   b. In the *Event Log Category* section, click *Anomaly Logs*.
   c. Enter a name (*anomaly-logs*) and add the required VDOMs (*root*, *vdom-nat*, *vdom-tp*).



   d. Click *OK*.

2. Configure the action:
   a. Go to *Security Fabric > Automation*, select the *Action* tab, and click *Create New*.
   b. In the *Notifications* section, click *Email* and enter the following:

Name	*email_default_rep_message*
To	Enter an email address
Subject	*CSF stitch alert*
Replacement message	Enable

   c. Click *OK*.

3. Configure the stitch:
   a. Go to *Security Fabric > Automation*, select the *Stitch* tab, and click *Create New*.
   b. Enter the name, *anomaly-logs-stitch*.
   c. Click *Add Trigger*. Select *anomaly-logs* and click *Apply*.
   d. Click *Add Action*. Select *email_default_rep_message* and click *Apply*.
   e. Click *Add delay* (between the trigger and action). Enter *3* and click *OK*.
   f. Click *OK*.

To configure an automation stitch with the anomaly logs trigger in the CLI:

1. Configure the trigger:

```
config system automation-trigger
    edit "anomaly-logs"
        set event-type anomaly-logs
        set vdom "root" "vdom-nat" "vdom-tp"
    next
end
```

2. Configure the action:

```
config system automation-action
    edit "email_default_rep_message"
        set action-type email
        set email-to "admin@fortinet.com"
```

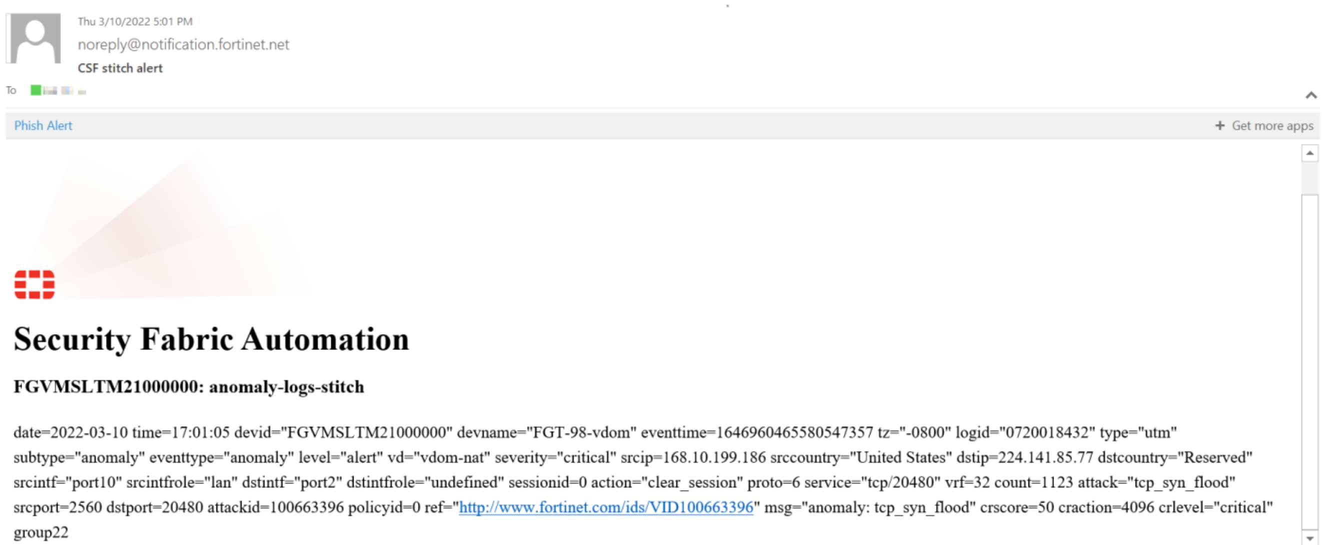
```
            set email-subject "CSF stitch alert"
            set replacement-message enable
        next
    end
```

3. Configure the stitch:

```
    config system automation-stitch
        edit "anomaly-logs-stitch"
            set description "anomaly-logs"
            set trigger "anomaly-logs"
            config actions
                edit 1
                    set action "email_default_rep_message"
                    set delay 3
                    set required enable
                next
            end
        next
    end
```

Verification

Once the anomaly log is generated, the automation stitch is triggered end the email notification is sent.



To confirm that the stitch was triggered in the GUI:

1. Go to *Security Fabric* > *Automation* and select the *Stitch* tab.
2. Verify the *Last Triggered* column.

To confirm that the stitch was triggered in the CLI:

```
# diagnose test application autod 2
...
stitch: anomaly-logs-stitch
        destinations: all
        trigger: anomaly-logs
```

```
            type:anomaly logs

            field ids:
                    (id:6)vd=root,vdom-nat,vdom-tp

    local hit: 1 relayed to: 0 relayed from: 0
    actions:
            email_default_rep_message type:email interval:0
                    delay:3 required:yes
                    subject: CSF stitch alert
                    body: %%log%%
                    sender:
                    mailto:admin@fortinet.com;
```

Actions

The following table outlines the available actions. Multiple actions can be added to an automation stitch. Actions can be reorganized in the *Edit Automation Stitch* page by dragging and dropping the actions in the diagram.

Category	Action	Description
Security Response		
	Access Layer Quarantine	This option is only available for Compromised Host triggers. Quarantine the MAC address on access layer devices (FortiSwitch and FortiAP).
	FortiClient Quarantine	This option is only available for Compromised Host triggers. Use FortiClient EMS to block all traffic from the source addresses that are flagged as compromised hosts. Quarantined devices are flagged on the Security Fabric topology views. Go to the *Dashboard* > *Users & Devices* > *Quarantine* widget to view and manage quarantined IP addresses.
	FortiNAC Quarantine	This option is only available for Compromised Host and Incoming Webhook triggers. Use FortiNAC to quarantine a client PC and disable its MAC address. See FortiNAC Quarantine action on page 2375 for details.
	VMware NSX Security Tag	This option is only available for Compromised Host triggers. If an endpoint instance in a VMware NSX environment is compromised, the configured security tag is assigned to the compromised endpoint. See VMware NSX security tag action on page 2378 and VMware NSX-T security tag action on page 2382 for details.
	IP Ban	This option is only available for Compromised Host triggers. Block all traffic from the source addresses flagged by the IoC. Go to the *Dashboard* > *Users & Devices* > *Quarantine* widget to view and manage quarantined IP addresses.

Category	Action	Description
Notifications		
	Email	Send a custom email message to the selected recipients. At least one recipient and an email subject must be specified. The email body can use parameters from logs or previous action results. Wrapping the parameter with %% will replace the expression with the JSON value for the parameter, for example: *%%results.source%%* is the source property from the previous action. Replacement messages can be enabled in the email body to create branded email alerts. See Replacement messages for email alerts on page 2386 for details.
	FortiExplorer Notification	Send push notifications to FortiExplorer. The FortiGate must be registered to FortiCare on the mobile app that will receive the notification.
	Slack Notification	Send a notification to a Slack channel. See Slack Notification action on page 2390 for details.
	Microsoft Teams Notification	Send a notification to channels in Microsoft Teams. See Microsoft Teams Notification action on page 2395 for details.
Cloud Compute		
	AWS Lambda	Send log data to an integrated AWS service. See AWS Lambda action on page 2400 for details.
	Azure Function	Send log data to an Azure function. See Azure Function action on page 2402 for details.
	Google Cloud Function	Send log data to a Google Cloud function. See Google Cloud Function action on page 2403 for details.
	AliCloud Function	Send log data to an AliCloud function. See AliCloud Function action on page 2405 for details.
General		
	CLI Script	Run one or more CLI scripts. See CLI script action on page 2407 for details. See Execute a CLI script based on CPU and memory thresholds on page 2410 for an example.
	Webhook	Send an HTTP request using a REST callback. See Webhook action on page 2418 for details, and Slack integration webhook on page 2422 and Microsoft Teams integration webhook on page 2424 for examples.
	Alert	Generate a FortiOS dashboard alert. This option is only available in the CLI.
	Disable SSID	Disable the SSID interface. This option is only available in the CLI.

FortiNAC Quarantine action

Users can configure an automation stitch with the FortiNAC Quarantine action with a Compromised Host or Incoming Webhook trigger. When the automation is triggered, the client PC will be quarantined and its MAC address is disabled in the configured FortiNAC.

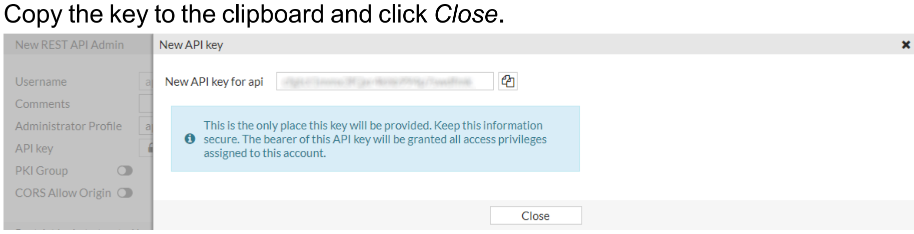
In this example, the FortiNAC has been configured to join an enabled Security Fabric. See Configuring FortiNAC on page 2250 for more information.

The FortiNAC must also be configured to isolate disabled hosts:

- Endpoints connecting to FortiWiFi or wired ports on FortiGate:
  - See the requisite *Configure FortiNAC* section in the FortiGate Endpoint Management Integration Guide.
- Endpoints connecting to FortiAP:
  - Set the *Dead End* VLAN. See Model configuration.
- Endpoints connecting to FortiSwitch:
  - Set the *Dead End* VLAN. See Model configuration.
  - Add the switch to the physical address filtering group. See Systems groups and Modify a group.

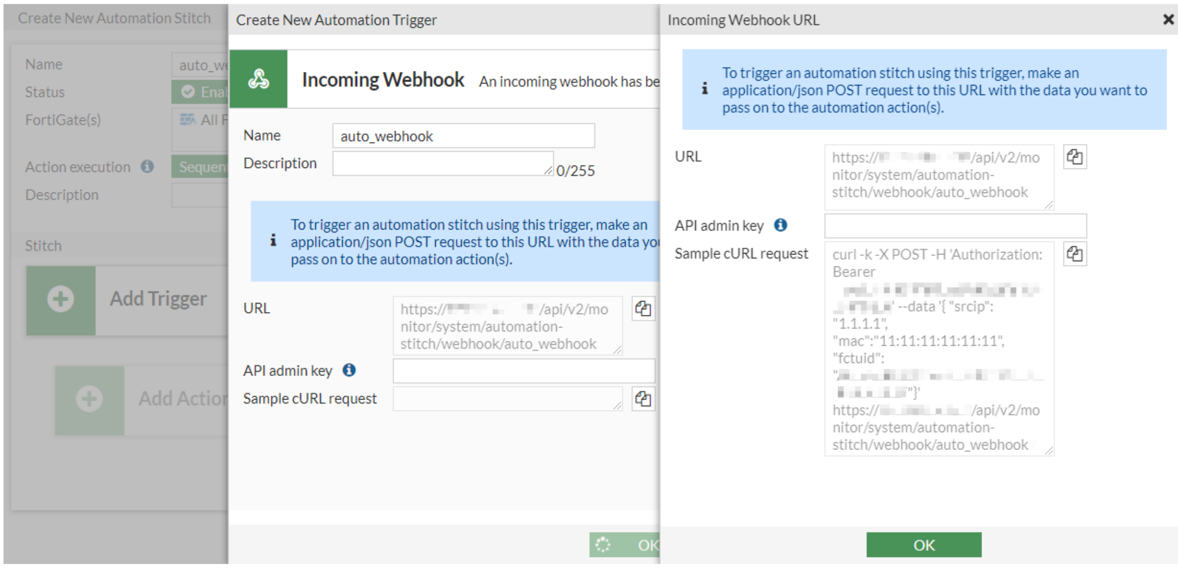
To configure an automation stitch with a FortiNAC quarantine action in the GUI:

1. Create a new API user and generate the API key:
   a. Go to *System > Administrators* and click *Create New > REST API Admin*.
   b. Configure the settings as needed.
   c. Click *OK*. The *New API key* window opens.
   d. Copy the key to the clipboard and click *Close*.



   e. Click *OK*.
2. Configure the automation stitch trigger:
   a. Go to *Security Fabric > Automation* and click *Create New*.
   b. Enter the stitch name (*auto_webhook*).
   c. Click *Add Trigger*.
   d. Click *Create* and select *Incoming Webhook*.
   e. Enter a name (*auto_webhook*).
   f. Click *OK*.

g. Paste the key in the *API admin key* field.



h. Click *OK*.

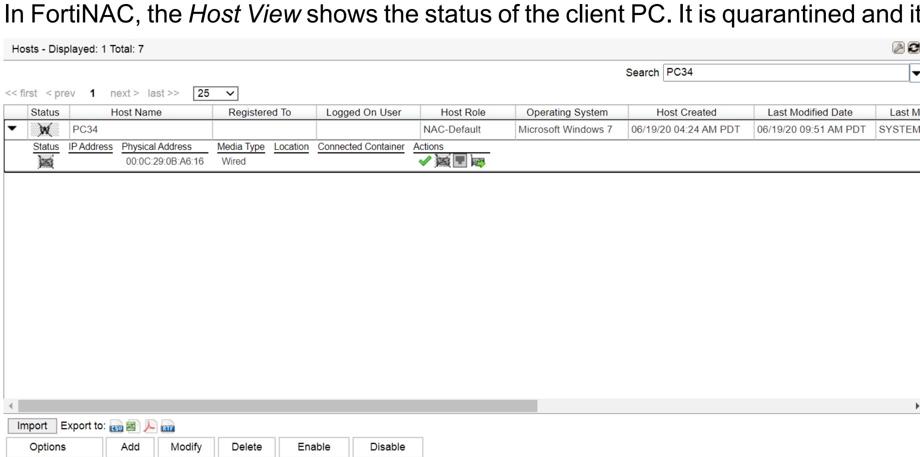
i. Select the trigger in the list and click *Apply*.

3. Configure the automation stitch action:
   a. Click *Add Action*.
   b. Click *Create* and select *FortiNAC Quarantine*.
   c. Enter an action name (*auto_webhook_quarantine-fortinac*) and click *OK*.
   d. Select the action in the list and click *Apply*.
   e. Click *OK*.
4. On a Linux PC accessible by the FortiGate, create a cURL request to trigger the automation stitch:

```
root@pc56:~# curl -k -X POST -H 'Authorization: Bearer ckx7d9xdzzx14Nztd1Ncr701dpwwy9' -
-data '{ "srcip": "1.1.1.1", "mac":"00:0C:29:0B:A6:16", "fctuid":
"A8BA0B12DA694E47BA4ADF24F8358E2F"}'
https://172.17.48.225:4431/api/v2/monitor/system/automation-stitch/webhook/auto_webhook
```

5. In FortiOS, verify the automation stitch is triggered and the action is executed:
   a. Go to *Log & Report > System Events* to confirm that the stitch was activated.
   b. Go to *Security Fabric > Automation* to see the last time that the stitch was triggered.

   In FortiNAC, the *Host View* shows the status of the client PC. It is quarantined and its MAC address is disabled.



To configure an automation stitch with a FortiNAC quarantine action in the CLI:

1. Create a new API user and generate the API key:

```
config system api-user
    edit "g-api-rw-user"
        set api-key ************
        set accprofile "super_admin"
        set vdom "root"
        config trusthost
            edit 1
                set ipv4-trusthost 10.6.30.0 255.255.255.0
            next
        end
    next
end
```

2. Configure the automation trigger:

```
config system automation-trigger
    edit "auto_webhook"
        set event-type incoming-webhook
    next
end
```

3. Configure the automation action:

```
config system automation-action
    edit "auto_webhook_quarantine-fortinac"
        set action-type quarantine-fortinac
    next
end
```

4. Configure the automation stitch:

```
config system automation-stitch
    edit "auto_webhook"
        set trigger "auto_webhook"
        config actions
            edit 1
                set action "auto_webhook_quarantine-fortinac"
                set required enable
            next
        end
    next
end
```

5. On a Linux PC accessible by the FortiGate, create a cURL request to trigger the automation stitch:

```
root@pc56:~# curl -k -X POST -H 'Authorization: Bearer ckx7d9xdzzx14Nztd1Ncr701dpwwy9' -
-data '{ "srcip": "1.1.1.1", "mac":"00:0C:29:0B:A6:16", "fctuid":
"A8BA0B12DA694E47BA4ADF24F8358E2F"}'
https://172.17.48.225:4431/api/v2/monitor/system/automation-stitch/webhook/auto_webhook
```

6. In FortiOS, verify that the automation stitch is triggered and the action is executed:

```
# diagnose test application autod 2
csf: enabled    root:yes
version:1592949233 sync time:Tue Jun 23 15:03:15 2020
```

```
total stitches activated: 1

stitch: auto_webhook
        destinations: all
        trigger: auto_webhook

                (id:15)service=auto_webhook

        local hit: 1 relayed to: 0 relayed from: 0
        actions:
                auto_webhook_quarantine-fortinac type:quarantine-fortinac interval:0

date=2020-06-23 time=15:25:44 logdesc="Internal Message" path="system" name="automation-
stitch" action="webhook" mkey="auto_webhook" srcip="1.1.1.1" mac="00:0C:29:0B:A6:16"
fctuid="A8BA0B12DA694E47BA4ADF24F8358E2F" vdom="root" service="auto_webhook"

date=2020-06-23 time=15:25:44 logid="0100046600" type="event" subtype="system"
level="notice" vd="root" eventtime=1592951144401490054 tz="-0700" logdesc="Automation
stitch triggered" stitch="auto_webhook" trigger="auto_webhook" stitchaction="auto_
webhook_quarantine-fortinac" from="log" msg="stitch:auto_webhook is triggered."
```

VMware NSX security tag action

If an endpoint instance in a VMware NSX environment is compromised, this action will assign the configured security tag to the compromised endpoint.

This action is only available when the automation trigger is set to compromised host.

To set up the NSX quarantine action, you need to:

1. Configure a VMware NSX SDN connector
2. Configure an NSX security tag automation stitch
3. Configure FortiAnalyzer logging on the FortiGate

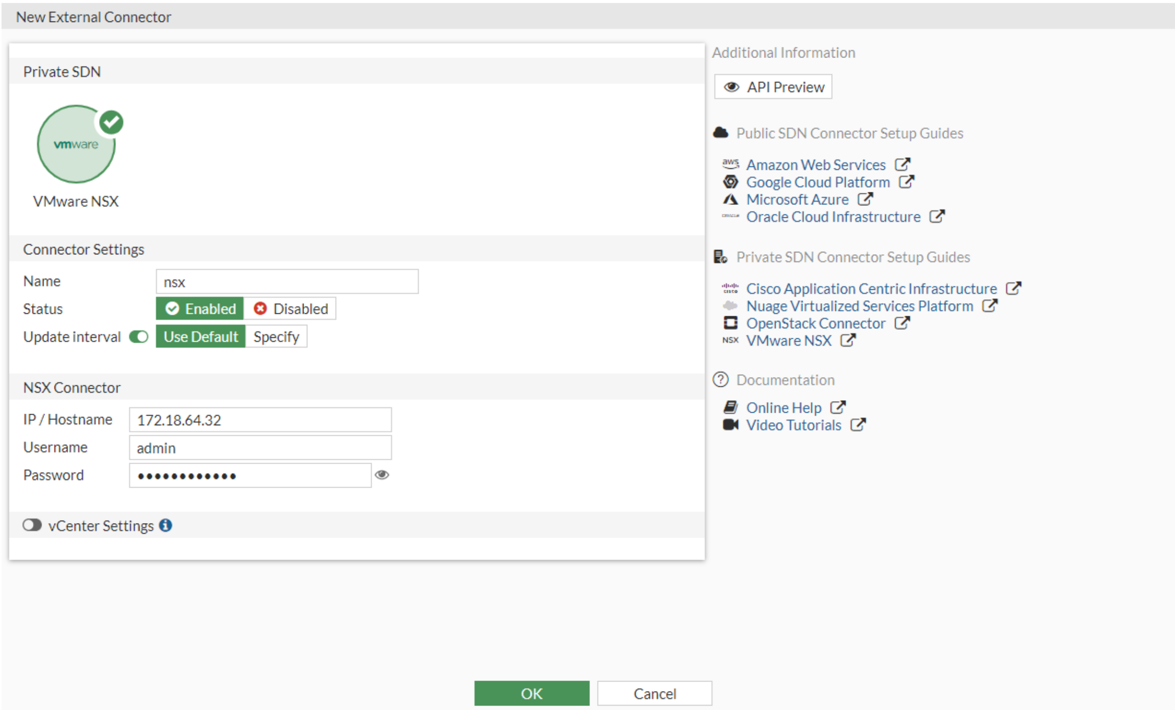
Configure a VMware NSX SDN connector

The FortiGate retrieves security tags from the VMware NSX server through the connector.

To configure a VMware NSX SDN connector in the GUI:

1. Go to *Security Fabric* > *External Connectors*.
2. Click *Create New*.
3. Select *VMware NSX*.

4. Configure the settings as needed.



5. Click *OK*.

To configure a VMware NSX SDN connector in the CLI:

```
config system sdn-connector
    edit "nsx"
        set type nsx
        set server "172.18.64.32"
        set username "admin"
        set password xxxxxxxxxxxx
    next
end
```

Configure an NSX security tag automation stitch

Security tags are retrieved from the VMware NSX server through the NSX SDN connector.

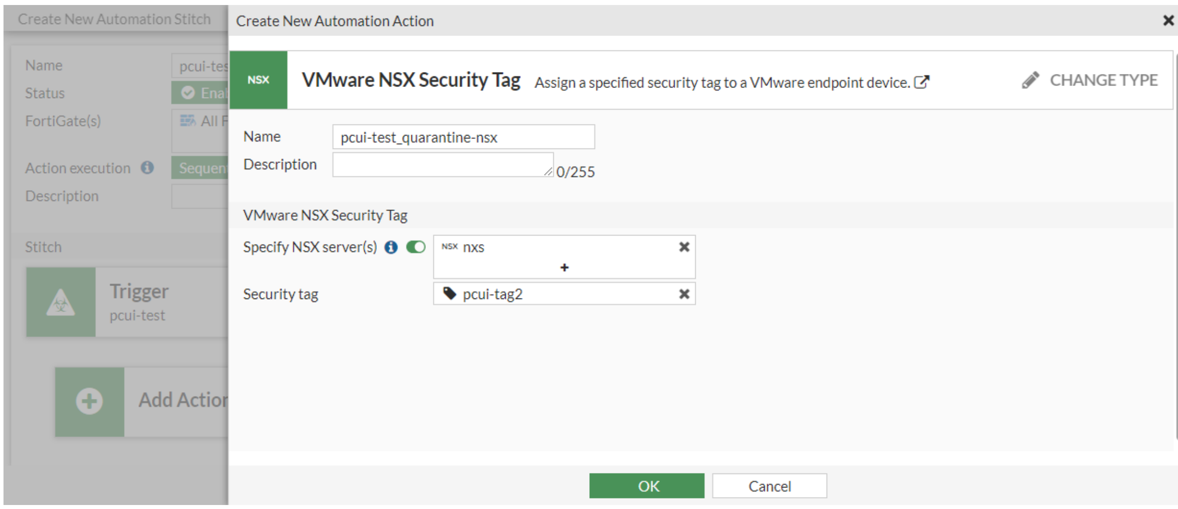
To configure an automation stitch with an NSX security tag in the GUI:

1. Go to *Security Fabric > Automation* and click *Create New*.
2. Enter the stitch name (*pcui-test*).
3. Configure the trigger:
    a. Click *Add Trigger*.
    b. Click *Create* and select *Compromised Host*.
    c. Enter the following:

Name	pcui-test
Threat level threshold	High

   d. Click *OK*.
   e. Select the trigger in the list and click *Apply*.
4. Configure the VMware NSX Security Tag action:
   a. Click *Add Action*.
   b. Click *Create* and select *VMware NSX Security Tag*.
   c. Enter the following:

Name	pcui-test_quarantine-nsx
Specify NSX server(s)	Enable and select the SDN connector
Security tag	Select an existing tag, or create a new one



   d. Click *OK*.
   e. Select the action in the list and click *Apply*.
5. Click *OK*.

To configure an automation stitch with an NSX security tag in the CLI:

1. Create an automation trigger:

```
config system automation-trigger
    edit "pcui-test"
        set ioc-level high
    next
end
```

2. Create an automation action:

```
config system automation-action
    edit "pcui-test_quarantine-nsx"
        set action-type quarantine-nsx
        set security-tag "pcui-tag2"
        set sdn-connector "nsx"
    next
end
```

3. Create the automation stitch:

```
config system automation-stitch
    edit "pcui-test"
        set trigger "pcui-test"
        config actions
            edit 1
                set action "pcui-test_quarantine-nsx"
                set required enable
            next
        end
    next
end
```

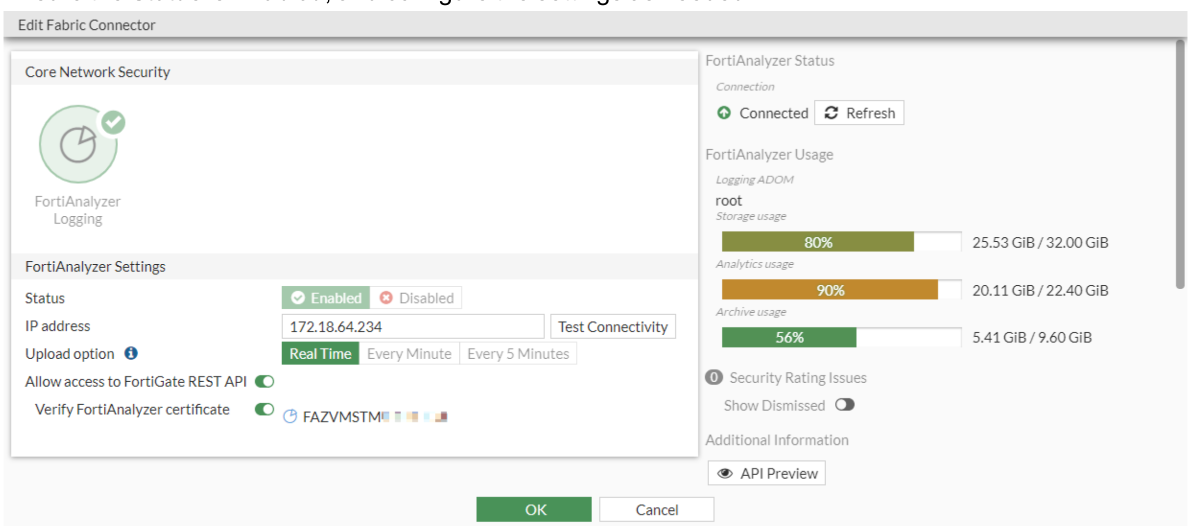
Configure FortiAnalyzer logging on the FortiGate

The FortiAnalyzer is used to send endpoint compromise notification to the FortiGate.

See Configuring FortiAnalyzer on page 2215 for more information.

To configure FortiAnalyzer logging in the GUI:

1. Go to *Security Fabric > Fabric Connectors* and double-click the *FortiAnalyzer Logging* card.
2. Ensure the *Status* is *Enabled*, and configure the settings as needed.



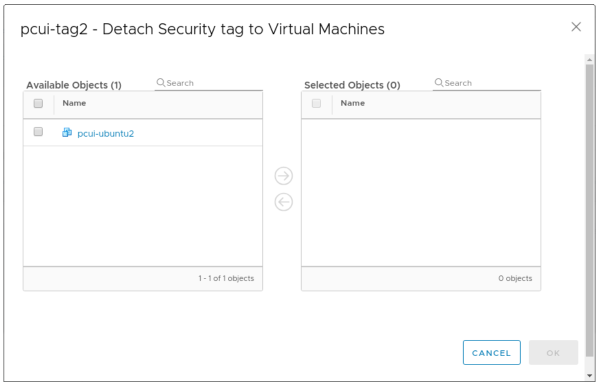
3. Click *Apply*.

To configure FortiAnalyzer logging in the CLI:

```
config log fortianalyzer setting
    set status enable
    set server "172.18.64.234"
    set serial "FL-8HFT000000000"
    set upload-option realtime
    set reliable enable
end
```

When an endpoint instance is compromised

When an endpoint instance, such as *pcui-ubuntu2*, in the VMware NSX environment is compromised, the automation stitch is triggered. The FortiGate then assigns the configured security tag, *pcui-tag2* in this example, to the compromised NSX endpoint instance.



VMware NSX-T security tag action

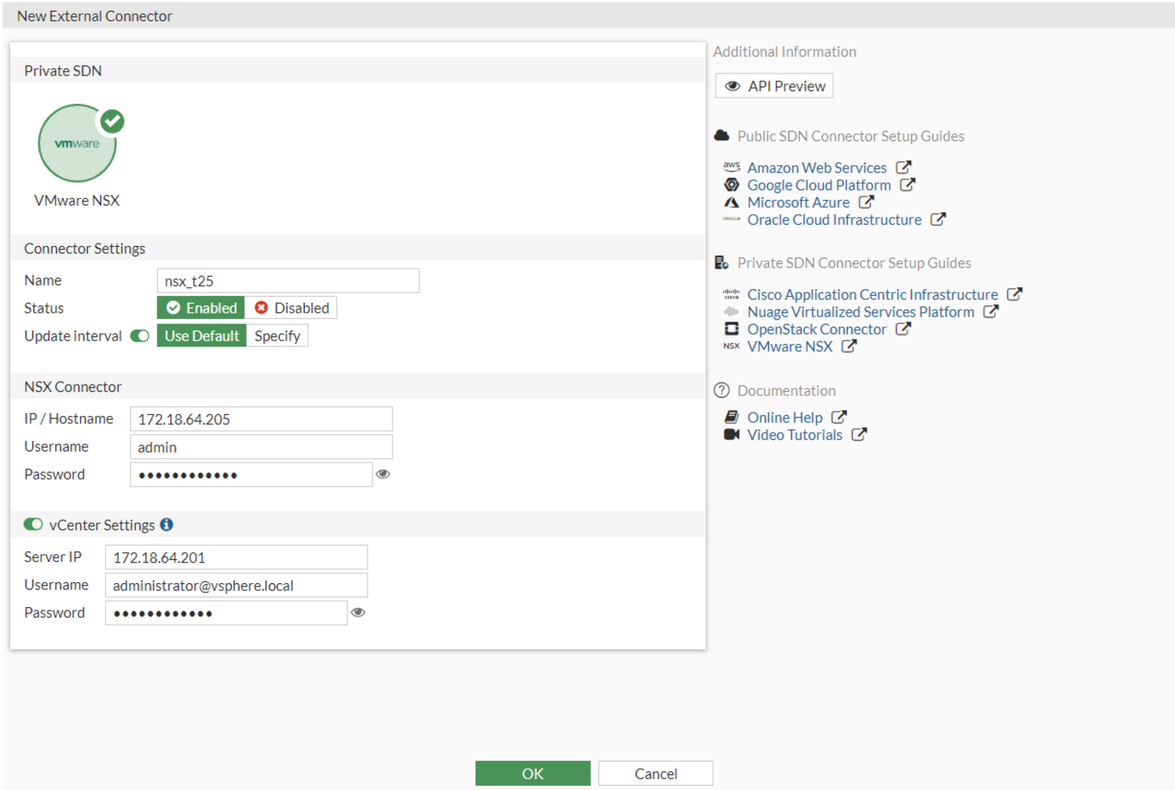
VMware NSX SDN connectors' vCenter server and credentials can be configured so the FortiGate resolves NSX-T VMs. The FortiGate uses the VMWare NSX Security Tag automation action to assign a tag to the VM through an automation stitch.

The FortiGate is notified of a compromised host on the NSX-T network by an incoming webhook or other means, such as FortiGuard IOC. An automation stitch can be configured to process this trigger and action it by assigning a VMware NSX security tag on the VM instance.

To configure an automation stitch to assign a security tag to NSX-T VMs in the GUI:

1. Configure the NSX SDN connector:
   a. Go to *Security Fabric > External Connectors* and click *Create New*.
   b. Select *VMware NSX*.
   c. Configure the connector settings.

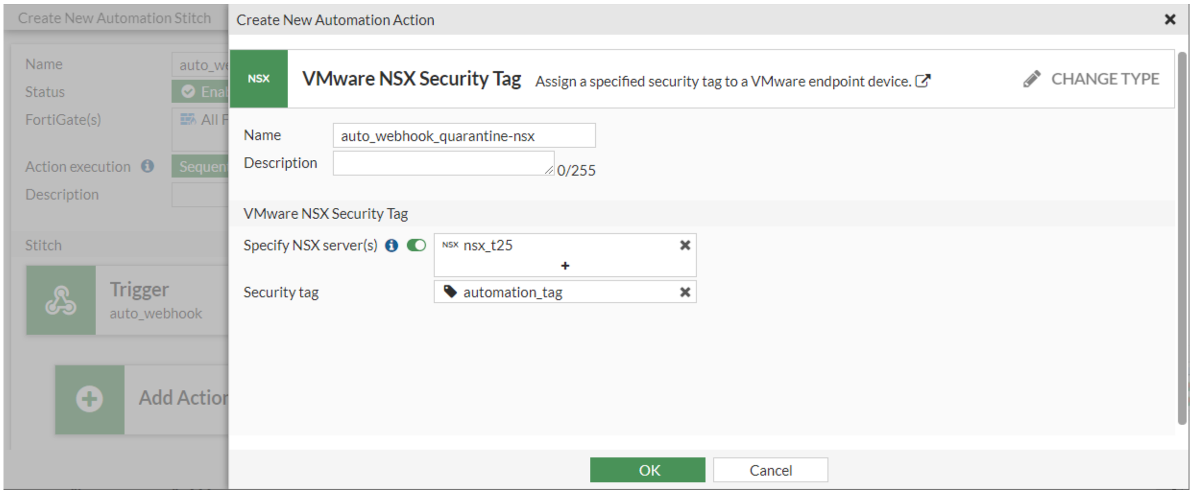
d. Enable *vCenter Settings* and configure as needed.



e. Click *OK*.

2. Configure the automation stitch trigger:
   a. Go to *Security Fabric > Automation* and click *Create New*.
   b. Enter the stitch name (*auto_webhook*).
   c. Click *Add Trigger*.
   d. Click *Create* and select *Incoming Webhook*.
   e. Enter a name (*auto_webhook*).
   f. Click *OK* to close the *Incoming Webhook URL* prompt.
   g. Select the trigger in the list and click *Apply*.
3. Configure the automation stitch action:
   a. Click *Add Action*.
   b. Click *Create* and select *VMware NSX Security Tag*.
   c. Enter the following:

Name	auto_webhook_quarantine-nsx
Specify NSX server(s)	Enable and select the SDN connector
Security tag	Select an existing tag, or create a new one



d. Click *OK*.

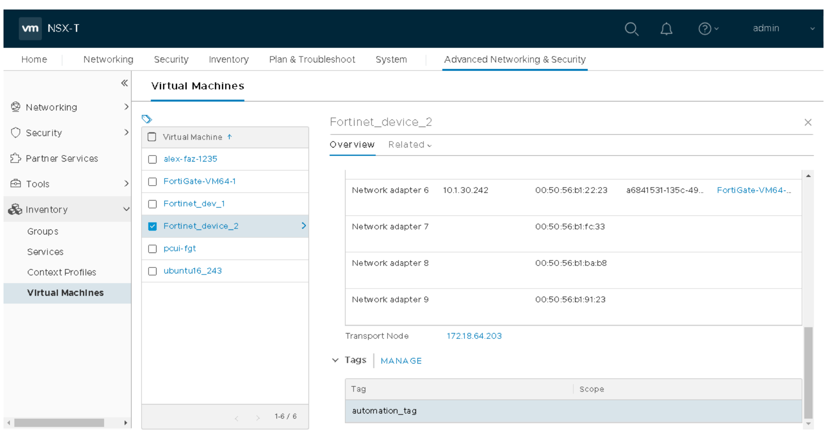
e. Select the action in the list and click *Apply*.

4. Click *OK*.

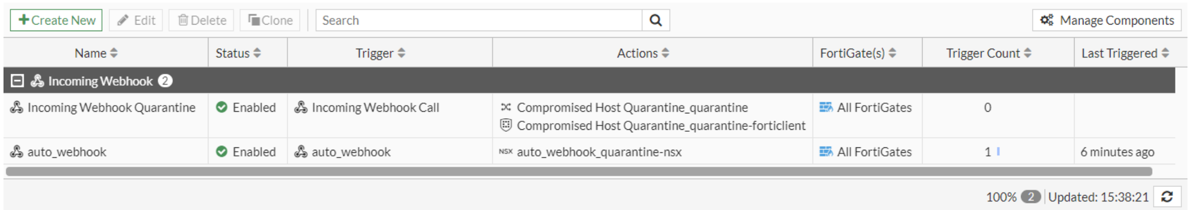
5. In NSX-T, create a cURL request to trigger the automation stitch on the FortiGate:

```
root@pc56:/home# curl -k -X POST -H 'Authorization: Bearer
3fdxNG08mgNg0fh4NQ51g1NQ1QHcxx' --data '{ "srcip": "10.1.30.242"}'
https://172.16.116.230/api/v2/monitor/system/automation-stitch/webhook/auto_webhook
{
  "http_method":"POST",
  "status":"success",
  "http_status":200,
  "serial":"FGVM08TM20000000",
  "version":"v6.4.0",
  "build":1608
}
```

The automation stitch is triggered and the configured tag is added to the NSX-T VM.



In FortiOS, the *Security Fabric > Automation* page shows the last trigger time.



To configure an automation stitch to assign a security tag to NSX-T VMs in the CLI:

1. Configure the NSX SDN connector:

```
config system sdn-connector
    edit "nsx_t25"
        set type nsx
        set server "172.18.64.205"
        set username "admin"
        set password xxxxxxxxxxxx
        set vcenter-server "172.18.64.201"
        set vcenter-username "administrator@vsphere.local"
        set vcenter-password xxxxxxxxxxxx
    next
end
```

2. Configure the automation stitch:

```
config system automation-trigger
    edit "auto_webhook"
        set trigger-type event-based
        set event-type incoming-webhook
    next
end
```

```
config system automation-action
    edit "auto_webhook_quarantine-nsx"
        set action-type quarantine-nsx
        set security-tag "automation_tag"
        set sdn-connector "nsx_t25"
    next
end
```

```
config system automation-stitch
    edit "auto_webhook"
        set trigger "auto_webhook"
        config actions
            edit 1
                set action "auto_webhook_quarantine-nsx"
                set required enable
            next
        end
    next
end
```

3. In NSX-T, create a cURL request to trigger the automation stitch on the FortiGate:

```
root@pc56:/home# curl -k -X POST -H 'Authorization: Bearer
3fdxNG08mgNg0fh4NQ51g1NQ1QHcxx' --data '{ "srcip": "10.1.30.242"}'
https://172.16.116.230/api/v2/monitor/system/automation-stitch/webhook/auto_webhook
```

```
{
  "http_method":"POST",
  "status":"success",
  "http_status":200,
  "serial":"FGVM08TM20000000",
  "version":"v6.4.0",
  "build":1608
}
```


To verify the automation stitch is triggered and the action is executed:

```
# diagnose test application autod 2

csf: enabled root:yes
version:1586883541 sync time:Tue Apr 14 11:04:05 2020

total stitches activated: 1

stitch: auto_webhook
destinations: all
trigger: auto_webhook

(id:15)service=auto_webhook

local hit: 1 relayed to: 0 relayed from: 0
actions:
auto_webhook_quarantine-nsx type:quarantine-nsx interval:0
security tag:automation_tag
sdn connector:
nsx_t25;
```

Replacement messages for email alerts

Automation stitches with an Email action can leverage the formatting options provided by replacement messages to create branded email alerts.

You can enable a replacement message and edit the message body or select a customized replacement message group when you configure the automation action. When the automation stitch is triggered, the FortiGate will send the email with the defined replacement message.

In this example, a Security Rating report triggers an Email notification action. The email uses a customized replacement message group.

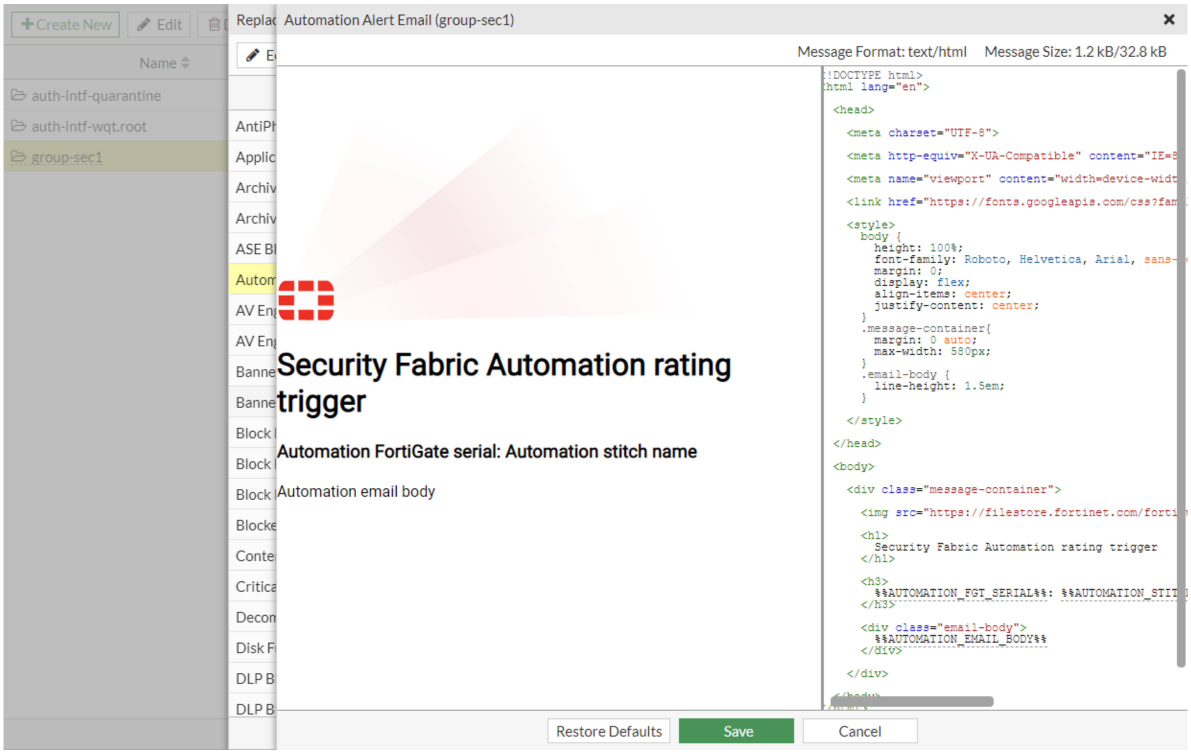
To configure the replacement message group in the GUI:

1. Go to *System > Replacement Message Groups* and click *Create New*.
2. Enter the following:

Name	group-sec1
Group Type	Security

3. Click *OK*.
4. Select the group in the list and click *Edit*.

5. Select *Automation Alert Email* and click *Edit*.

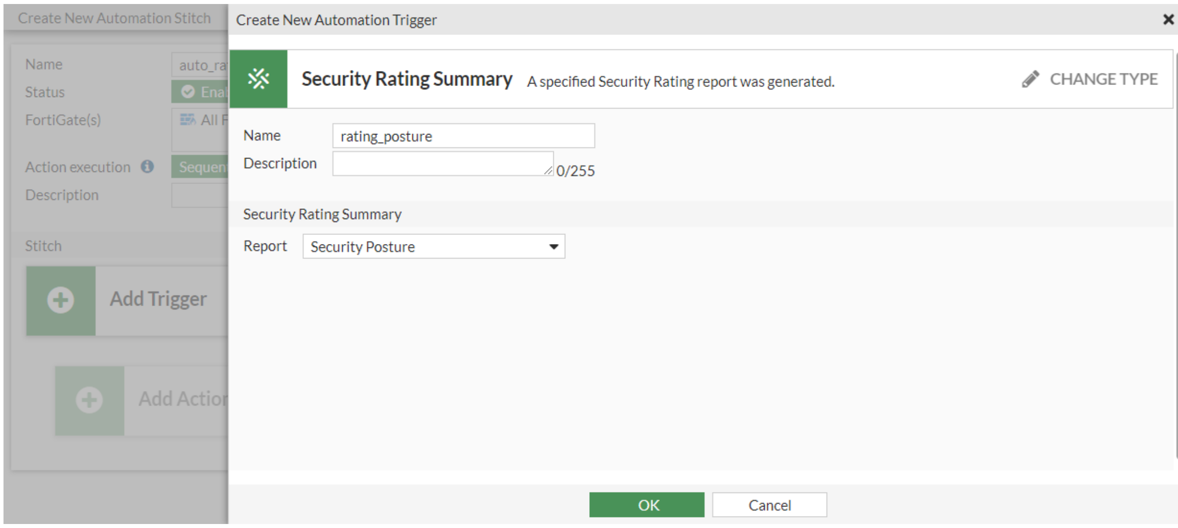


6. Edit the HTML code as needed, then click *Save*.

To configure the email action in the GUI:

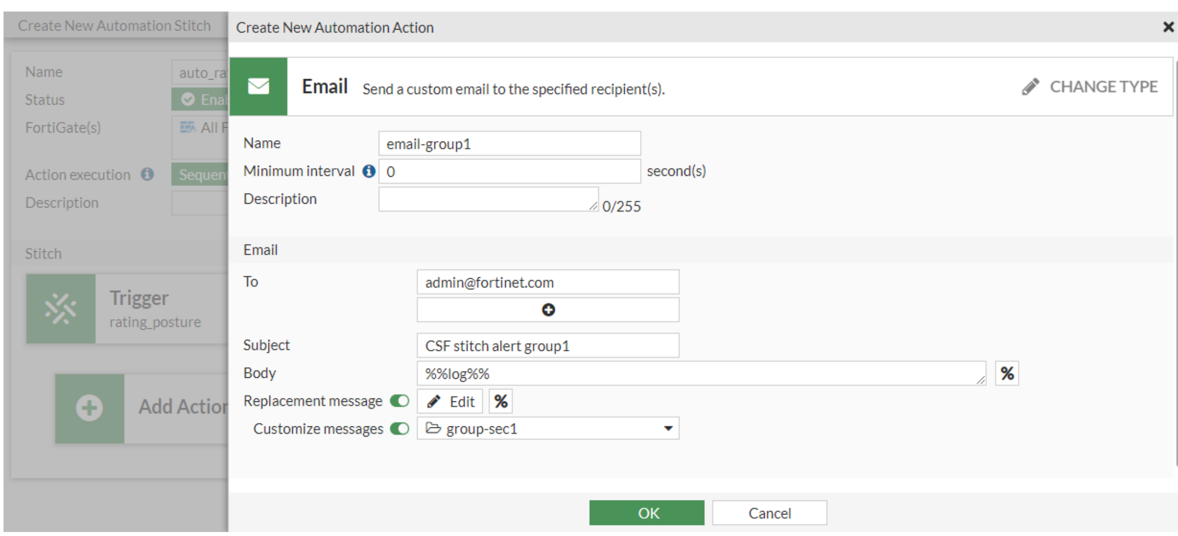
1. Go to *Security Fabric > Automation* and click *Create New*.
2. Enter the stitch name.
3. Configure the trigger:
   a. Click *Add Trigger*.
   b. Click *Create* and select *Security Rating Summary*.
   c. Enter the following:

Name	rating_posture
Description	rating test
Report	Security Posture



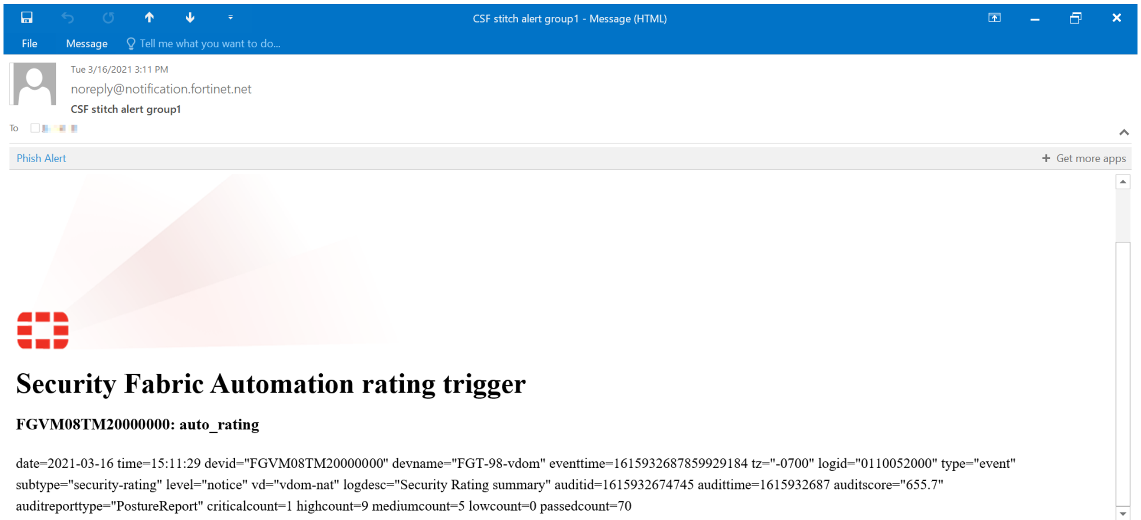
   d. Click *OK*.
   e. Select the trigger in the list and click *Apply*.
4. Configure the Email notification action:
   a. Click *Add Action*.
   b. Click *Create* and select *Email*.
   c. Enter the following:

Name	email-group1
To	Enter an email address
Subject	CSF stitch alert group1
Replacement message	Enable
Customize messages	Enable and select group-sec1 from the dropdown



   d. Click *OK*.
   e. Select the action in the list and click *Apply*.
5. Click *OK*.

6. Right-click the automation stitch, and click *Test Automation Stitch*.
   After the Security Rating report is finished, the automation is triggered, and the email is delivered with the customized replacement message in the email body.



To configure the replacement message group in the CLI:

```
config system replacemsg-group
    edit "group-sec1"
        set comment ""
        set group-type utm
        config automation
            edit "automation-email"
                set buffer "...<h1> Security Fabric Automation rating trigger </h1>..."
                ...
            next
        end
    next
end
```

To configure the email action in the CLI:

1. Configure the automation trigger:

```
config system automation-trigger
    edit "rating_posture"
        set description "rating test"
        set event-type security-rating-summary
    next
end
```

2. Configure the automation action:

```
config system automation-action
    edit "email-group1"
        set action-type email
        set email-to "admin@fortinet.com"
        set email-subject "CSF stitch alert group1"
        set replacement-message enable
```

```
        set replacemsg-group "group-sec1"
    next
end
```

3. Configure the automation stitch:

```
config system automation-stitch
    edit "auto_rating"
        set trigger "rating_posture"
        config actions
            edit 1
                set action "email-group1"
                set required enable
            next
        end
    next
end
```

4. To view the automation stitch information after it is triggered:

```
# diagnose test application autod 3
stitch: auto_rating
        local hit: 1 relayed to: 0 relayed from: 0
        last trigger:Tue Mar 16 15:11:29 2021
        last relay:
        actions:
                email-group1:
                        done: 1 relayed to: 0 relayed from: 0
                        last trigger:Tue Mar 16 15:11:29 2021
                        last relay:

logid2stitch mapping:
id:52000  local hit: 1 relayed hits: 0
        auto_rating
```

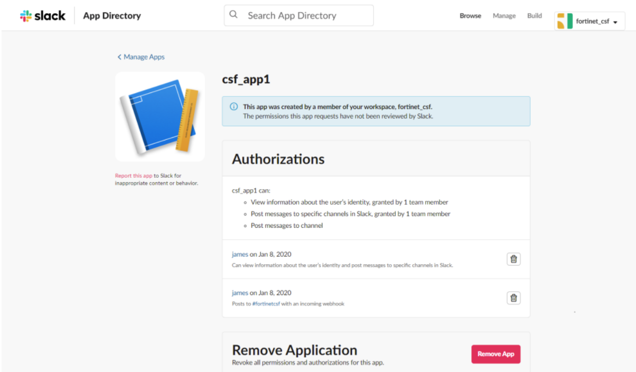
Slack Notification action

To configure an automation stitch with a Slack Notification action, you first need to configure an incoming webhook in Slack. Then you can enter the webhook URL when you configure the Slack Notification action.

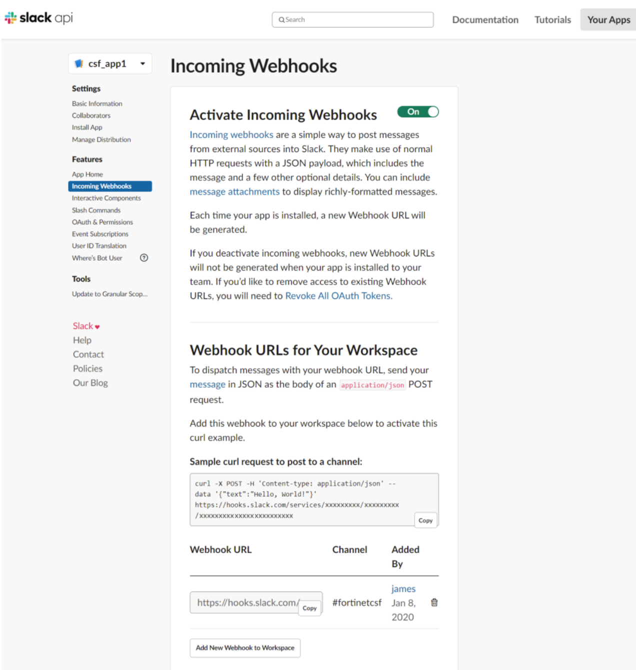
This example uses a Security Rating Summary trigger in the automation stitch with two Slack Notification actions with different notification messages. One message is a custom message, and the other is for the Security Rating Summary log with a 90 second delay.

To create an Incoming Webhook in Slack:

1. Go to the Slack website, and create a workspace.
2. Create a Slack application for the workspace.



3. Add an Incoming Webhook to a channel in the workspace (see Sending messages using Incoming Webhooks for more details).
4. Activate the Incoming Webhook, and copy the *Webhook URL* to the clipboard.



To configure an automation stitch with Slack Notification actions in the GUI:

1. Go to *Security Fabric > Automation* and click *Create New*.
2. Enter the stitch name.
3. Configure the Security Rating Summary trigger:
   a. Click *Add Trigger*.
   b. Click *Create* and select *Security Rating Summary*.

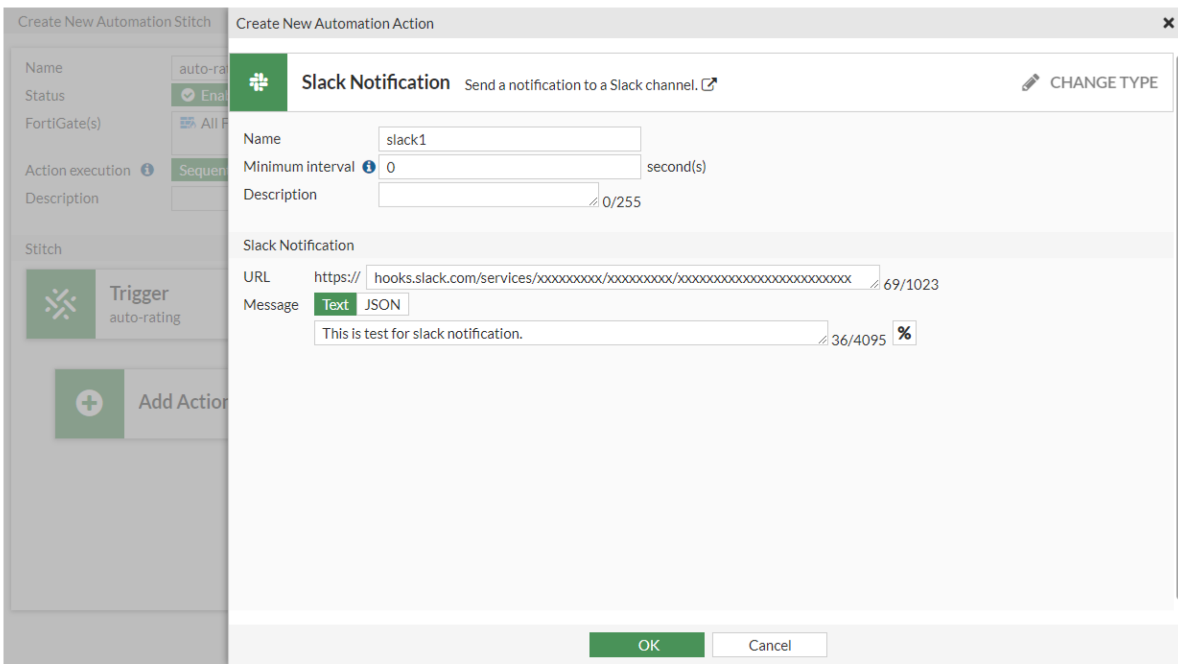
c. Enter the following:

Name	auto-rating
Report	Security Posture

d. Click *OK*.
e. Select the trigger in the list and click *Apply*.

4. Configure the first Slack Notification action:
   a. Click *Add Action*.
   b. Click *Create* and select *Slack Notification*.
   c. Enter the following:

Name	slack1
URL	Paste the webhook URL from the clipboard
Message	Text
Message text	This is test for slack notification.

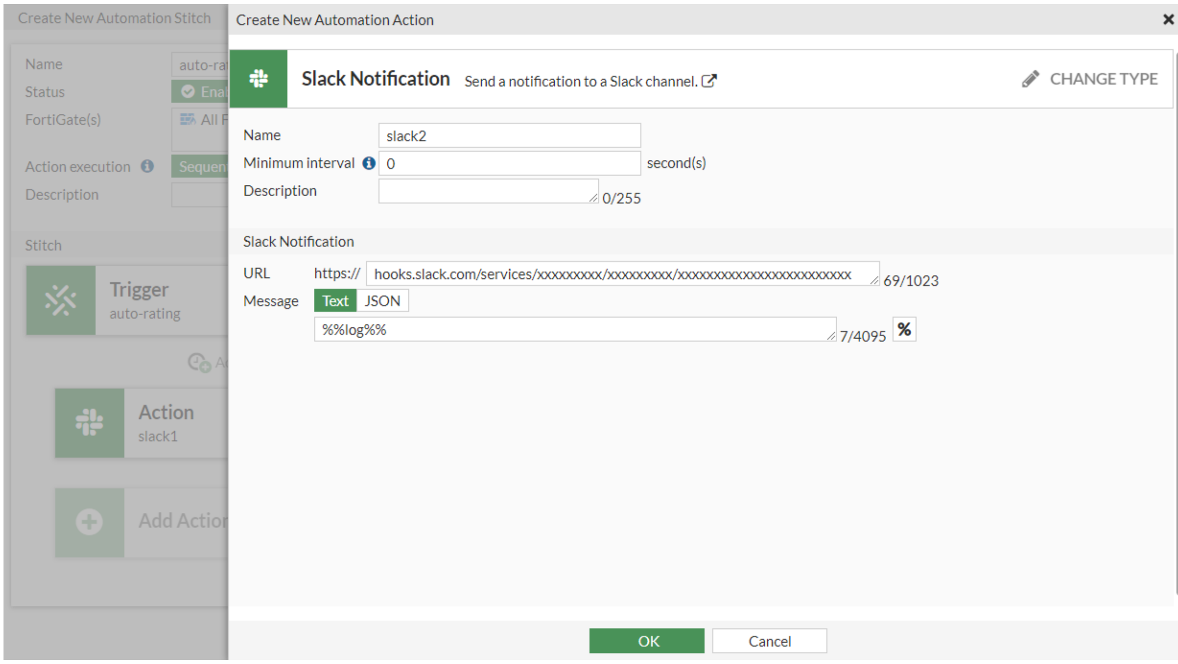


   d. Click *OK*.
   e. Select the action in the list and click *Apply*.

5. Configure the second Slack Notification action:
   a. Click *Add Action*.
   b. Click *Create* and select *Slack Notification*.
   c. Enter the following:

Name	slack2
URL	Paste the webhook URL from the clipboard

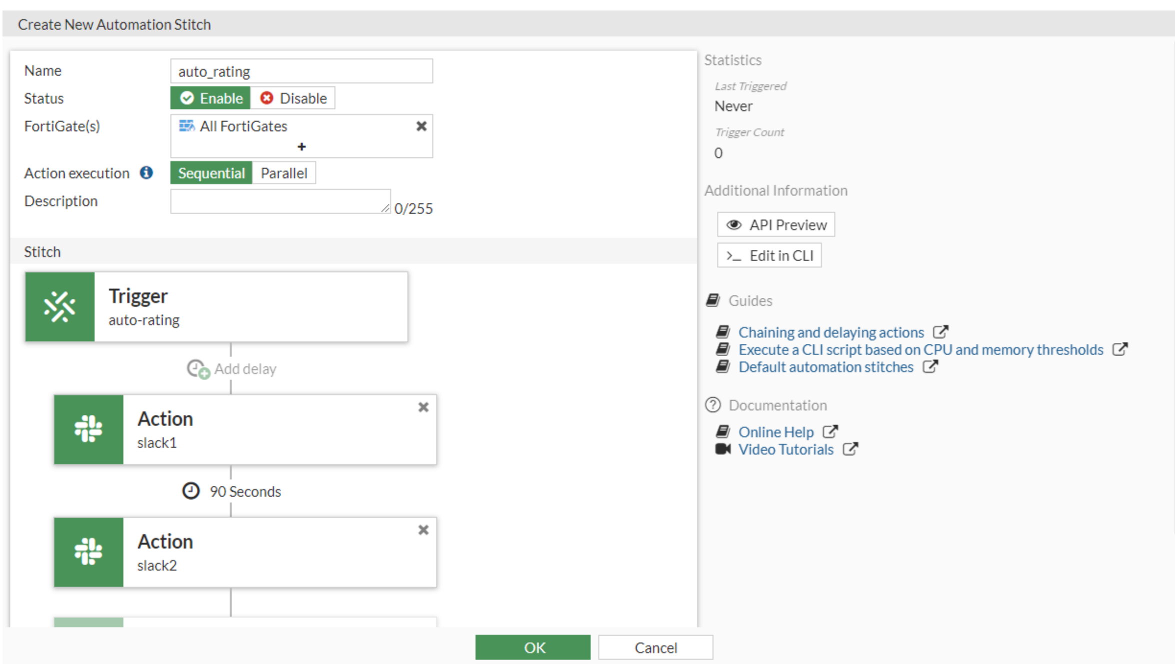
Message	Text
Message text	%%log%%



d. Click *OK*.

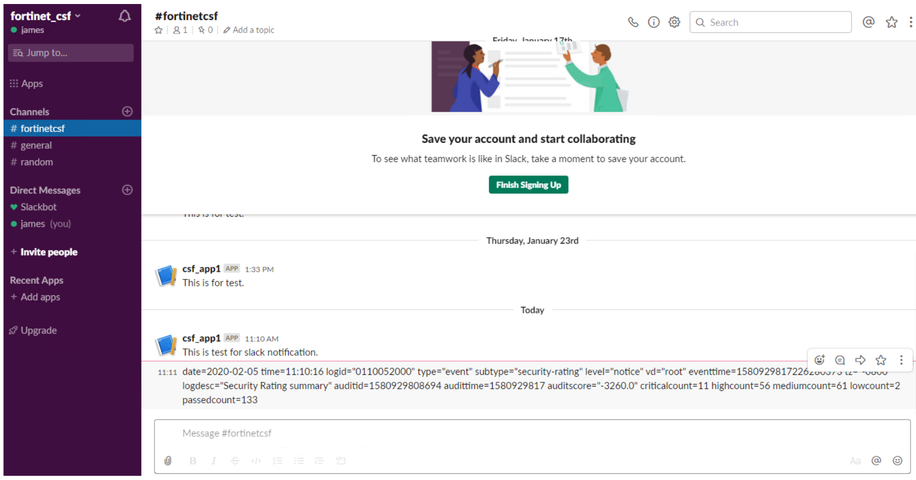
e. Select the action in the list and click *Apply*.

f. Click the *Add delay* located between both actions. Enter *90* and click *OK*.



6. Click *OK*.

7. Trigger the automation stitch:
   a. Right-click the automation stitch and select *Test Automation Stitch*.
   After the Security Rating report is finished, the automation is triggered and an event log is created by the FortiGate. The two notifications are sent to the Slack channel.



To configure an automation stitch with Slack Notification actions in the CLI:

1. Configure the automation trigger:

```
config system automation-trigger
    edit "auto-rating"
        set event-type security-rating-summary
    next
end
```

2. Configure the automation actions:

```
config system automation-action
    edit "slack1"
        set action-type slack-notification
        set message "This is test for slack notification."
        set uri "hooks.slack.com/services/xxxxxxxxx/xxxxxxxxx/xxxxxxxxxxxxxxxxxxxxxxxx"
    next
    edit "slack2"
        set action-type slack-notification
        set uri "hooks.slack.com/services/xxxxxxxxx/xxxxxxxxx/xxxxxxxxxxxxxxxxxxxxxxxx"
    next
end
```

3. Configure the automation stitch:

```
config system automation-stitch
    edit "auto_rating"
        set trigger "auto-rating"
        config actions
            edit 1
                set action "slack1"
                set required enable
            next
            edit 2
                set action "slack2"
```

```
                set delay 90
                set required enable
            next
        end
    next
end
```

4. Verify that the automation action was triggered:

```
# diagnose test application autod 3
stitch: auto-rating
    local hit: 1 relayed to: 0 relayed from: 0
    last trigger:Wed Feb 05 11:10:23 2020
    last relay:
    actions:
        slack1:
            done: 1 relayed to: 0 relayed from: 0
            last trigger:Wed Feb 11:10:23 2020
            last relay:
        slack2:
            done: 1 relayed to: 0 relayed from: 0
            last trigger:Wed Feb 05 11:10:23 2020
            last relay:
```

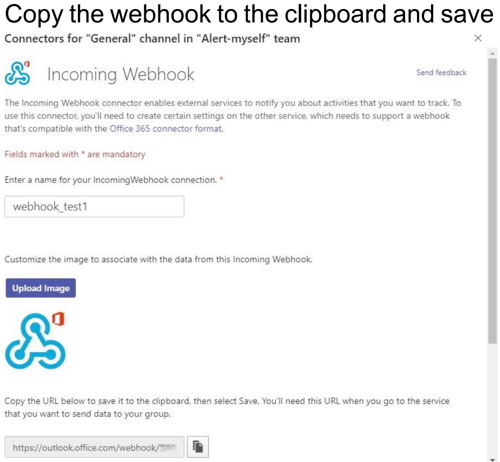
Microsoft Teams Notification action

Microsoft Teams Notification actions can be configured to send notifications to channels in Microsoft Teams. To trigger the notifications, you need to add an Incoming Webhook connector to a channel in Microsoft Teams, then you can configure the automation stitch with the webhook URL.

In the following example, you will configure an automation stitch with a Security Rating Summary trigger and two Microsoft Teams Notification actions with different notification messages. One message is for the Security Rating Summary log, and the other is a custom message with a ten second delay.

To add the Incoming Webhook connector in a Microsoft Teams channel:

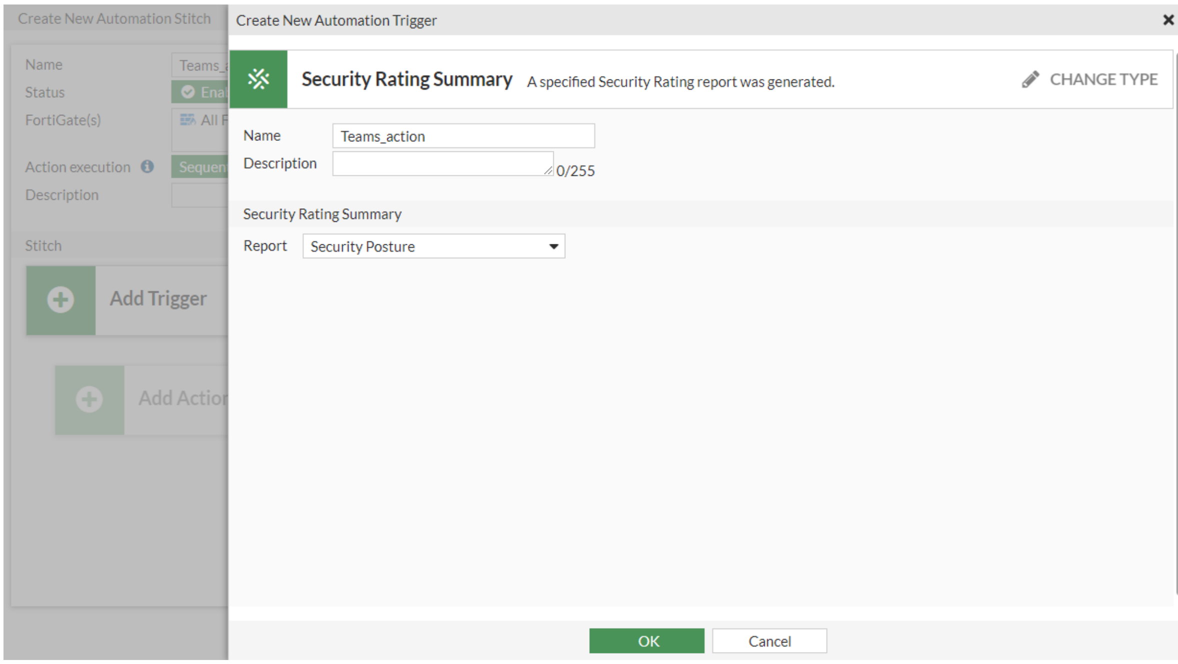
1. In Microsoft Teams, click the *...* (*More options*) beside the channel name, and select *Connectors*.
2. Search for *Incoming Webhook* and click *Configure*.
3. Enter a name for the webhook, upload an image for the webhook, and click *Create*.
4. Copy the webhook to the clipboard and save it.



5. Click *Done*.

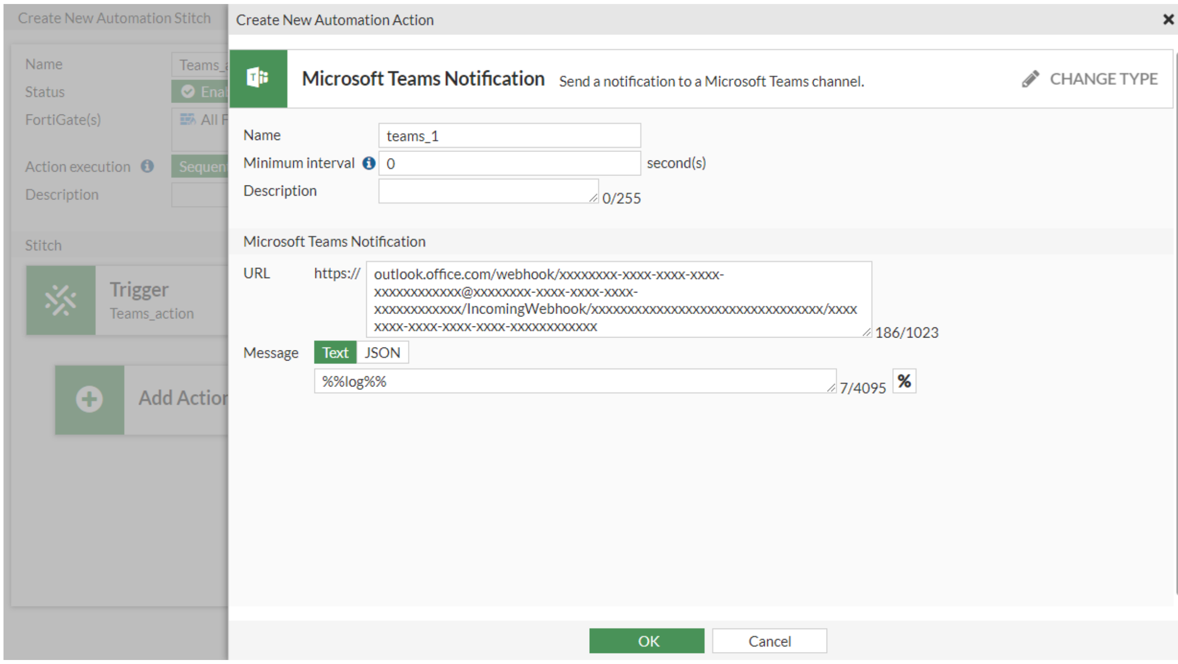
To configure an automation stitch with Microsoft Teams Notification actions in the GUI:

1. Go to *Security Fabric* > *Automation* and click *Create New*.
2. Enter the stitch name.
3. Configure the Security Rating Summary trigger:
   a. Click *Add Trigger*.
   b. Click *Create* and select *Security Rating Summary*.
   c. Enter a name, and for *Report*, select *Security Posture*.



   d. Click *OK*.
   e. Select the trigger in the list and click *Apply*.
4. Configure the first Microsoft Teams Notification action:
   a. Click *Add Action*.
   b. Click *Create* and select *Microsoft Teams Notification*.
   c. Enter the following:

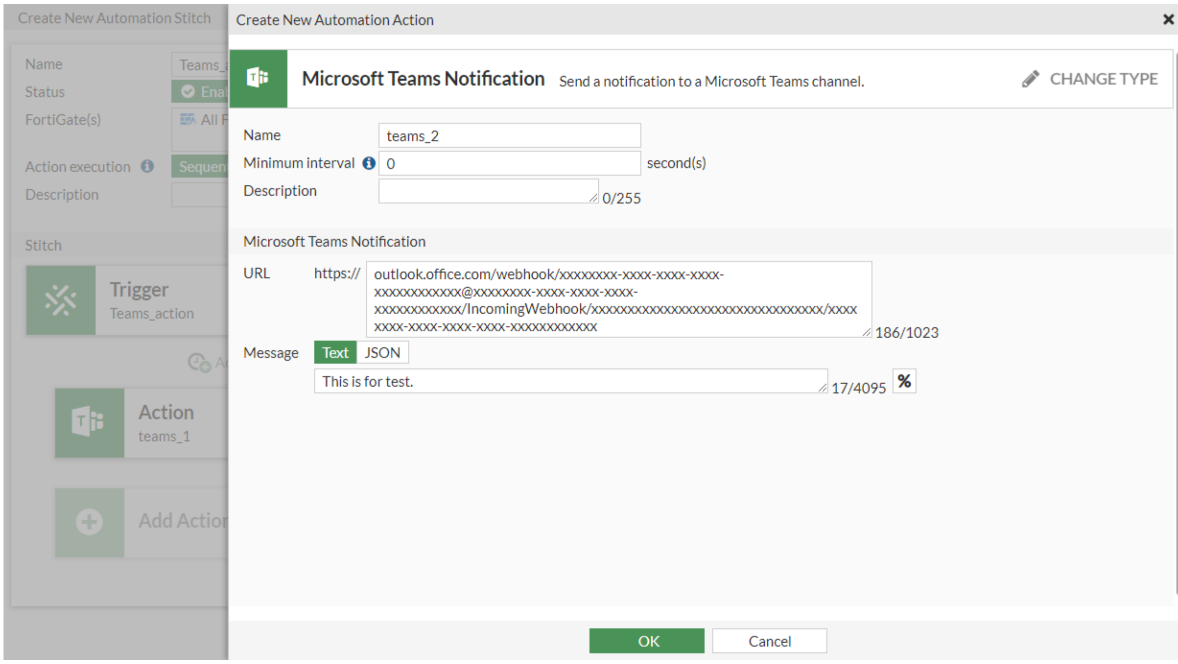
Name	teams_1
URL	Paste the webhook URI from the clipboard
Message	Text
Message text	%%log%%



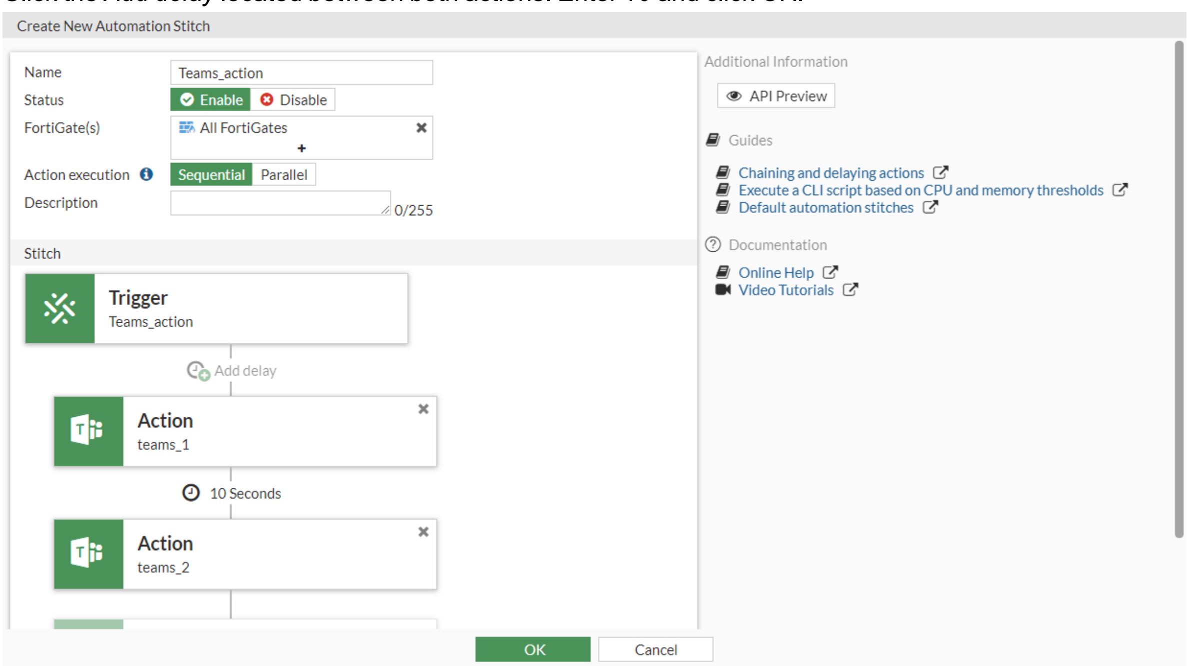
   d. Click *OK*.
   e. Select the action in the list and click *Apply*.

5. Configure the second Microsoft Teams Notification action:
   a. Click *Add Action*.
   b. Click *Create* and select *Microsoft Teams Notification*.
   c. Enter the following:

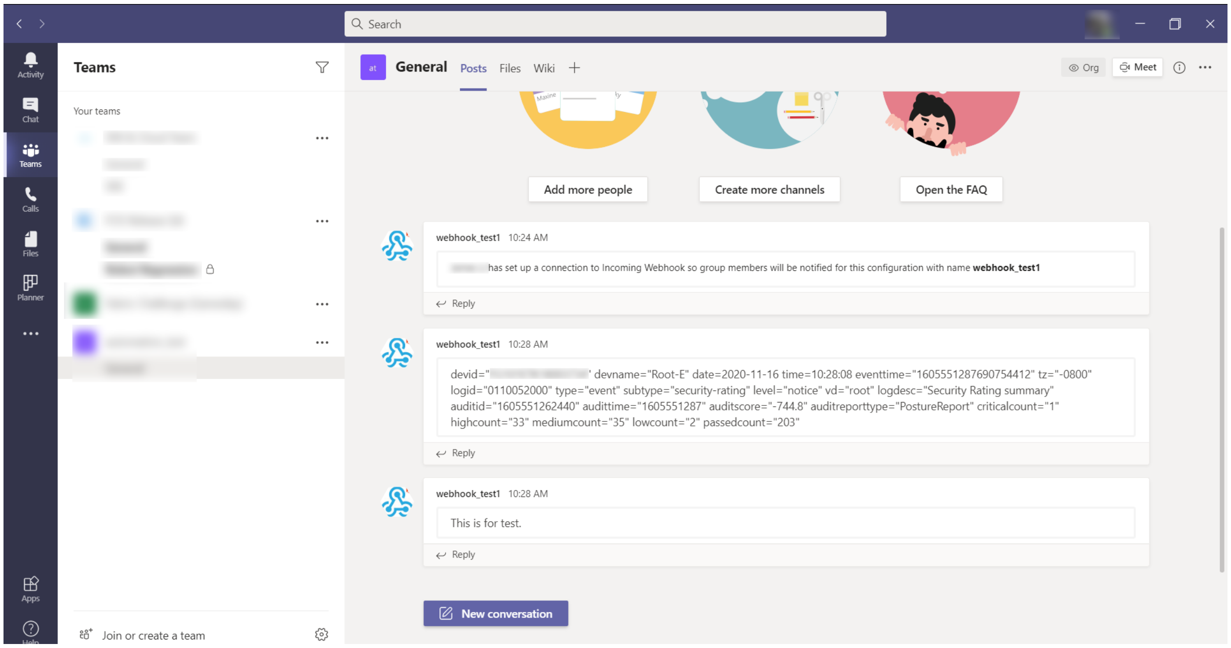
Name	teams_2
URL	Paste the webhook URI from the clipboard
Message	Text
Message text	This is for test.



d. Click *OK*.
e. Select the action in the list and click *Apply*.
f. Click the *Add delay* located between both actions. Enter *10* and click *OK*.



6. Click *OK*.
7. Trigger the automation stitch:
   a. Right-click the automation stitch and select *Test Automation Stitch*.
      After the Security Rating report is finished, the automation is triggered and an event log is created by the FortiGate. The two notifications are sent to the Microsoft Teams channel.



To configure an automation stitch with Microsoft Teams Notification actions in the CLI:

1. Configure the automation trigger:

```
config system automation-trigger
    edit "Teams_action"
        set event-type security-rating-summary
    next
end
```

2. Configure the automation actions:

```
config system automation-action
    edit "teams_1"
        set action-type microsoft-teams-notification
        set uri "outlook.office.com/webhook/xxxxxxxx-xxxx-xxxx-xxxx-
xxxxxxxxxxxx@xxxxxxxx-xxxx-xxxx-xxxx-
xxxxxxxxxxxx/IncomingWebhook/xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx/xxxxxxxx-xxxx-xxxx-xxxx-
xxxxxxxxxxxx"
    next
    edit "teams_2"
        set action-type microsoft-teams-notification
        set message "This is for test."
        set uri "outlook.office.com/webhook/xxxxxxxx-xxxx-xxxx-xxxx-
xxxxxxxxxxxx@xxxxxxxx-xxxx-xxxx-xxxx-
xxxxxxxxxxxx/IncomingWebhook/xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx/xxxxxxxx-xxxx-xxxx-xxxx-
xxxxxxxxxxxx"
    next
end
```

3. Configure the automation stitch:

```
config system automation-stitch
    edit "Teams_action"
        set trigger "Teams_action"
```

```
        config actions
            edit 1
                set action "teams_1"
                set required enable
            next
            edit 2
                set action "teams_2"
                set delay 10
                set required enable
            next
        end
    next
end
```

4. Verify that the automation action was triggered:

```
# diagnose test application autod 3
stitch: Teams_action
   local hit: 2 relayed to: 0 relayed from: 0
   last trigger:Mon Nov 16 10:28:08 2020
   last relay:
   actions:
      teams_1:
      done: 2 relayed to: 0 relayed from: 0
         last trigger:Mon Nov 16 10:28:08 2020
         last relay:
      teams_2:
      done: 2 relayed to: 0 relayed from: 0
         last trigger:Mon Nov 16 10:28:08 2020
         last relay:
logid2stitch mapping:
id:52000 local hit: 22 relayed hits: 0
Teams_action
```

AWS Lambda action

AWS Lambda functions can be called when an automation stitch is triggered. This example uses a Security Rating Summary trigger in the automation stitch.

To configure an AWS Lambda function automation stitch in the GUI:

1. Go to *Security Fabric > Automation* and click *Create New*.
2. Enter the stitch name.
3. Configure the trigger:
   a. Click *Add Trigger*.
   b. Click *Create* and select *Security Rating Summary*.
   c. Enter the following:

Name	auto-aws
Report	Security Posture

   d. Click *OK*.
   e. Select the trigger in the list and click *Apply*.

4. Configure the AWS Lambda function action:
   a. Click *Add Action*.
   b. Click *Create* and select *AWS Lambda*.
   c. Enter the following:

Name	aws-action-1
URL	Enter the request API URI
API key	Enter the API key
HTTP header	header2 : header2_value

   d. Click *OK*.
   e. Select the action in the list and click *Apply*.
5. Click *OK*.

To configure an AWS Lambda function automation stitch in the CLI:

1. Create the automation trigger:

```
config system automation-trigger
    edit "auto-aws"
        set event-type security-rating-summary
    next
end
```

2. Create the automation action:

```
config system automation-action
    edit "aws-action-1"
        set action-type aws-lambda
        set aws-api-key *************
        set uri "0100000000.execute-api.us-east-2.amazonaws.com/default/xxxxx-
autobatoon-XXX-lambdaXXX"
        set headers "header2:header2_value"
    next
end
```

3. Create the automation stitch:

```
config system automation-stitch
    edit "auto-aws"
        set trigger "auto-aws"
        config actions
            edit 1
                set action "aws-action-1"
                set required enable
            next
        end
    next
end
```

When the automation stitch is triggered, the *Security Fabric* > *Automation* page shows the stitch trigger time. In AWS, the log shows that the function was called, executed, and finished.

Azure Function action

Azure functions can be called when an automation stitch is triggered. This example uses a Security Rating Summary trigger in the automation stitch.

To configure an Azure function automation stitch in the GUI:

1. Go to *Security Fabric > Automation* and click *Create New*.
2. Enter the stitch name.
3. Configure the trigger:
   a. Click *Add Trigger*.
   b. Click *Create* and select *Security Rating Summary*.
   c. Enter the following:

Name	auto-azure
Report	Security Posture

   d. Click *OK*.
   e. Select the trigger in the list and click *Apply*.
4. Configure the Azure Function action:
   a. Click *Add Action*.
   b. Click *Create* and select *Azure Function*.
   c. Enter the following:

Name	azure_function
URL	Enter the request API URI
Authorization	Function
API key	Enter the API key
HTTP header	header1 : value1

   d. Click *OK*.
   e. Select the action in the list and click *Apply*.
5. Click *OK*.

To configure an Azure function automation stitch in the CLI:

1. Create an automation trigger:

```
config system automation-trigger
    edit "auto-azure"
        set event-type security-rating-summary
    next
end
```

2. Create an automation action:

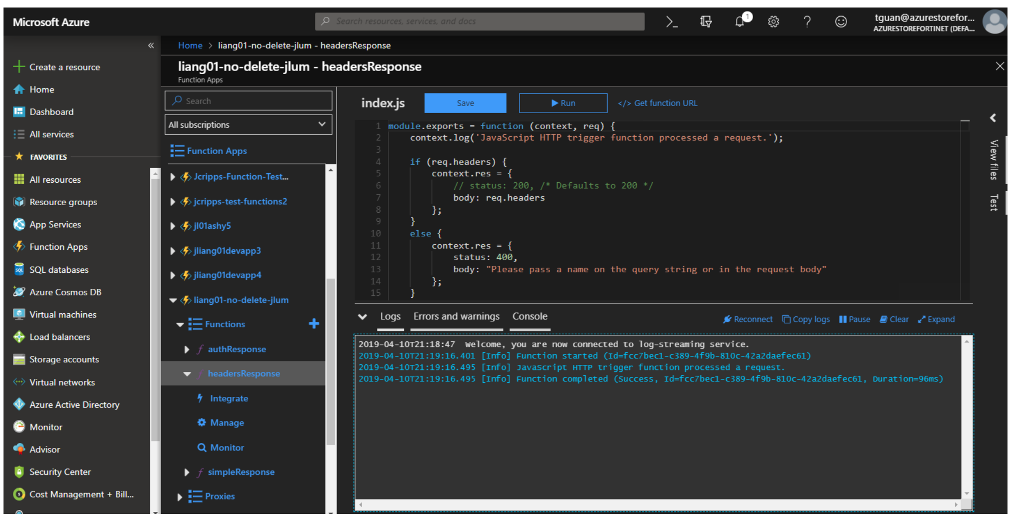
```
config system automation-action
    edit "azure_function"
        set action-type azure-function
```

```
            set azure-function-authorization function
            set azure-api-key **********
            set uri "xxxxx00-no-delete-xxxx.azurewebsites.net/api/headersResponse"
            set headers "header1:value1"
        next
    end
```

3. Create the automation stitch:

```
config system automation-stitch
    edit "auto-azure"
        set trigger "auto-azure"
        config actions
            edit 1
                set action "azure_function"
                set required enable
            next
        end
    next
end
```

When the automation stitch is triggered, the *Security Fabric > Automation* page shows the stitch trigger time. In Azure, the function log shows that the function was called, executed, and finished:



Google Cloud Function action

Google Cloud functions can be called when an automation stitch is triggered. This example uses a Security Rating Summary trigger in the automation stitch.

To configure a Google Cloud function automation stitch in the GUI:

1. Go to *Security Fabric > Automation* and click *Create New*.
2. Enter the stitch name.
3. Configure the trigger:
    a. Click *Add Trigger*.
    b. Click *Create* and select *Security Rating Summary*.

c. Enter the following:

Name	auto-google1
Report	Security Posture

d. Click *OK*.

e. Select the trigger in the list and click *Apply*.

4. Configure the Google Cloud Function action:
   a. Click *Add Action*.
   b. Click *Create* and select *Google Cloud Function*.
   c. Enter the following:

Name	google-echo
URL	Enter the request API URI
HTTP header	echo-header : echo-value

   d. Click *OK*.
   e. Select the action in the list and click *Apply*.
5. Click *OK*.

To configure a Google Cloud function automation stitch in the CLI:

1. Create an automation trigger:

```
config system automation-trigger
    edit "auto-google1"
        set event-type security-rating-summary
    next
end
```

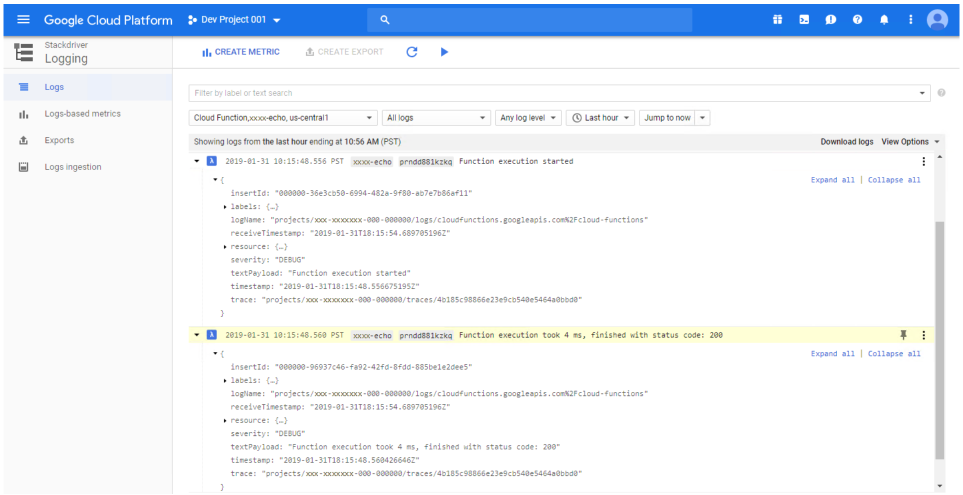
2. Create an automation action:

```
config system automation-action
    edit "google-echo"
        set action-type google-cloud-function
        set uri "us-central1-xxx-xxxxxxx-000-000000.cloudfunctions.net/xxxx-echo"
        set headers "echo-header:echo-value"
    next
end
```

3. Create the automation stitch:

```
config system automation-stitch
    edit "auto-google1"
        set trigger "auto-google1"
        config actions
            edit 1
                set action "google-echo"
                set required enable
            next
        end
    next
end
```

When the automation stitch is triggered, the *Security Fabric > Automation* page shows the stitch trigger time. In Google Cloud, go to *Logs* to see the function log showing that the configured function was called, executed, and finished:



AliCloud Function action

AliCloud functions can be called when an automation stitch is triggered. This example uses a Security Rating Summary trigger in the automation stitch.

To configure an AliCloud function automation stitch in the GUI:

1. Go to *Security Fabric > Automation* and click *Create New*.
2. Enter the stitch name.
3. Configure the trigger:
    a. Click *Add Trigger*.
    b. Click *Create* and select *Security Rating Summary*.
    c. Enter the following:

Name	auto-ali
Report	Security Posture

    d. Click *OK*.
    e. Select the trigger in the list and click *Apply*.
4. Configure the AliCloud Function action:
    a. Click *Add Action*.
    b. Click *Create* and select *AliCloud Function*.
    c. Enter the following:

Name	Ali-Action-1
URL	Enter the request API URI
Authorization	Function
AccessKey ID	Enter the access key ID

AccessKey Secret	Enter the access key secret

   d. Click *OK*.
   e. Select the action in the list and click *Apply*.
5. Click *OK*.

To configure an AliCloud function automation stitch in the CLI:

1. Create an automation trigger:

```
config system automation-trigger
    edit "auto-ali"
        set event-type security-rating-summary
    next
end
```

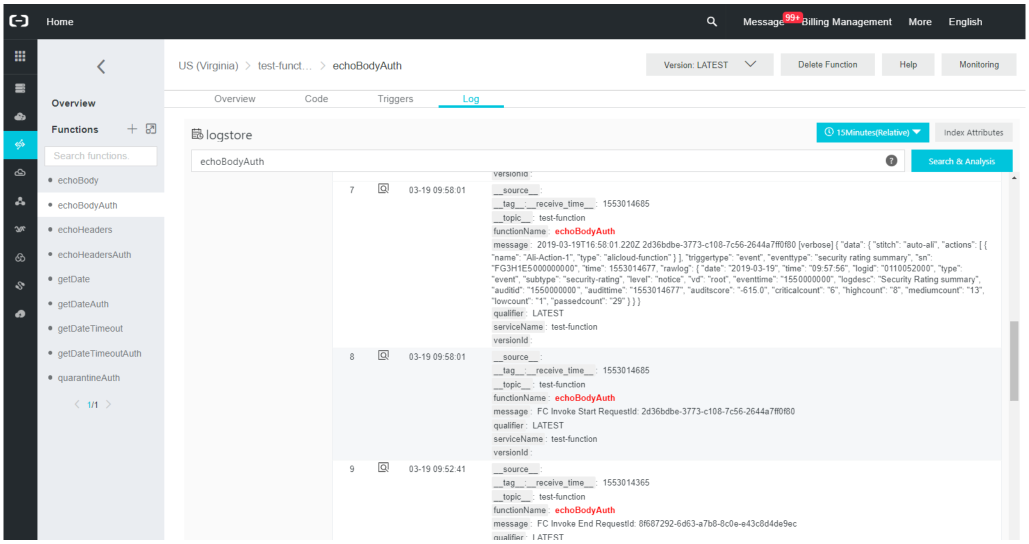
2. Create an automation action:

```
config system automation-action
    edit "Ali-Action-1"
        set action-type alicloud-function
        set alicloud-function-authorization function
        set alicloud-access-key-id "XXXXXxXXXXxxxxxx"
        set alicloud-access-key-secret xxxxxx
        set uri "0000000000000000.us-east-1.fc.aliyuncs.com/2099-99-99/proxy/test-
function/echoBodyAuth/"
    next
end
```

3. Create the automation stitch:

```
config system automation-stitch
    edit "auto-ali"
        set trigger "auto-ali"
        config actions
            edit 1
                set action "Ali-Action-1"
                set required enable
            next
        end
    next
end
```

When the automation stitch is triggered, the *Security Fabric > Automation* page shows the stitch trigger time. In AliCloud, the function log shows that the function was called, executed, and finished:



CLI script action

CLI scripts can run when an automation stitch is triggered. The scripts can be entered manually, uploaded as a file, or recorded in the CLI console. The output of the script can be sent as an email action.

The maximum length of the CLI script action output is 16 thousand characters. The CLI script can be configured in the GUI and CLI; however, the output size and timeout settings can only be configured in the CLI.

```
config system automation-action
    edit <name>
        set action-type cli-script
        set output-size <integer>
        set timeout <integer>
    next
end
```

`output-size <integer>`	Set the size to limit the script output, in megabytes (1 - 1024, default = 10).
`timeout <integer>`	Set the maximum running time for this script, in seconds (0 - 300, 0 = no timeout).

Example

In this example, the script sets the idle timeout value to 479 minutes, and sends an email with the script output.

To configure a stitch with a CLI script action in the GUI:

1. Go to *Security Fabric > Automation* and click *Create New*.
2. Enter the stitch name (*auto-cli-1*).
3. Configure the trigger:
   a. Click *Add Trigger*.
   b. Click *Create* and select *Security Rating Summary*.

c. Enter the following:

Name	auto-cli-1
Report	Security Posture

d. Click *OK*.

e. Select the trigger in the list and click *Apply*.

4. Configure the CLI Script action:

a. Click *Add Action*.

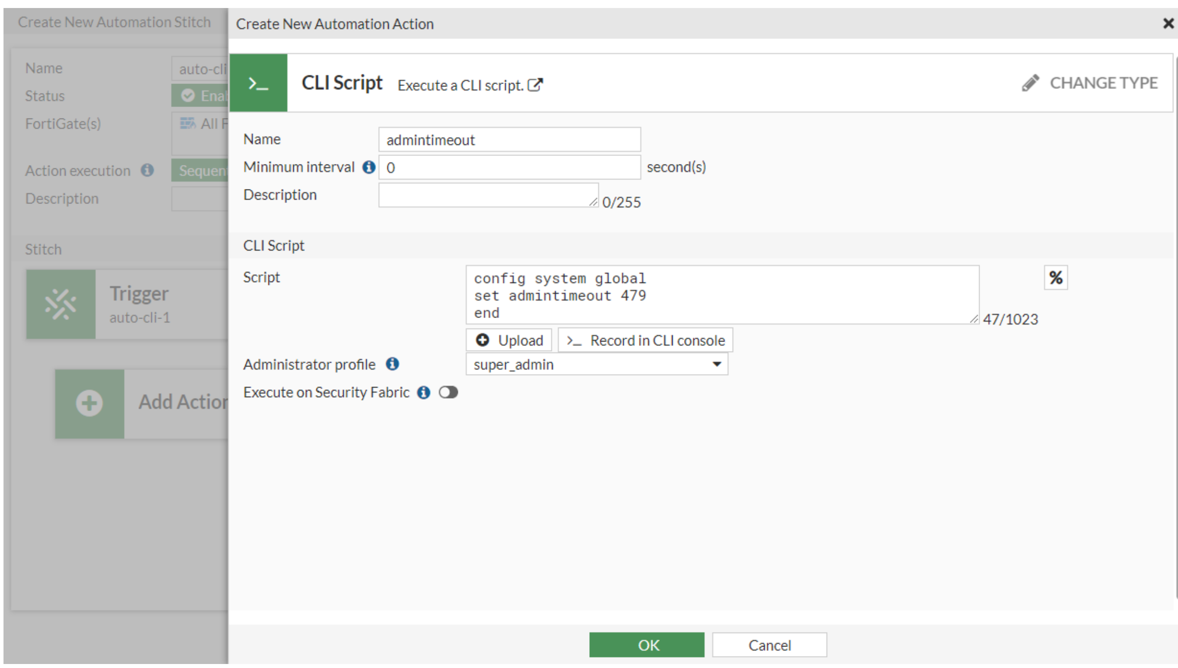
b. Click *Create* and select *CLI Script*.

c. Enter the following:

Name	admintimeout

Script	`config system global`
	`    set admintimeout 479`
	`end`
	
	Alternatively, click *Upload* to upload a file, or click *>_Record in CLI console* and enter the CLI commands.

Administrator profile	Select a profile



d. Click *OK*.

e. Select the action in the list and click *Apply*.

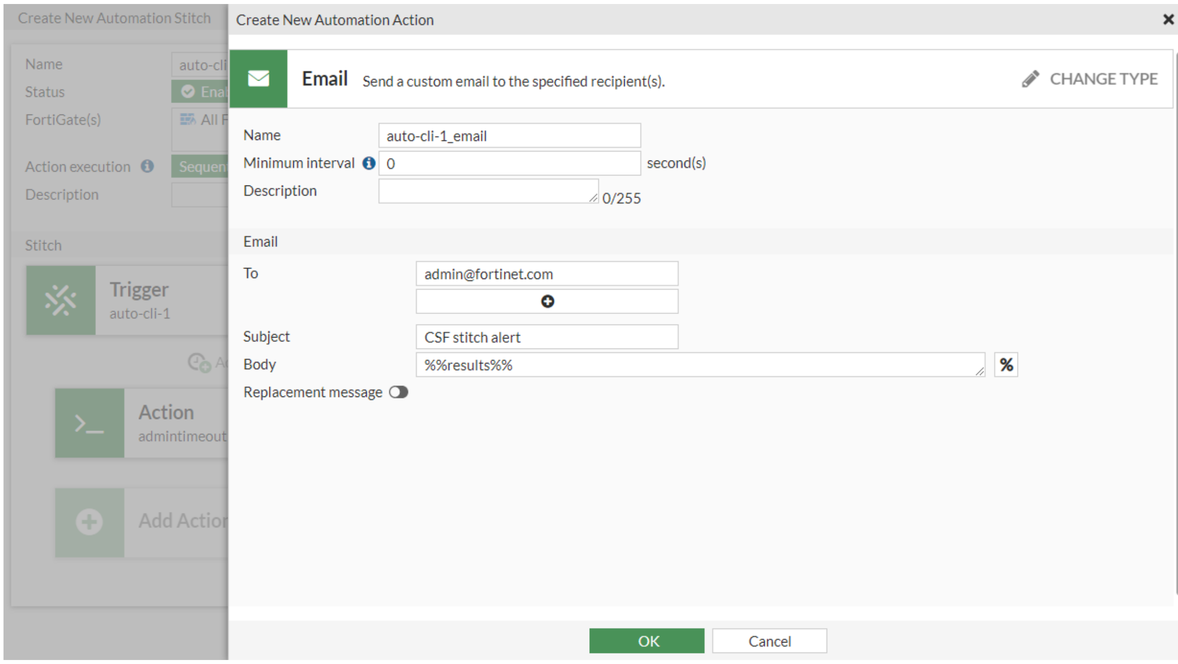
5. Configure the Email notification action:

a. Click *Add Action*.

b. Click *Create* and select *Email*.

c. Enter the following:

Name	auto-cli-1_email
To	Enter an email address
Subject	CSF stitch alert
Body	%%results%%



d. Click *OK*.

e. Select the action in the list and click *Apply*.

6. Click *OK*.

To configure a stitch with a CLI script action in the CLI:

1. Create the automation trigger:

```
config system automation-trigger
    edit "auto-cli-1"
        set event-type security-rating-summary
    next
end
```

2. Create the automation actions:

```
config system automation-action
    edit "admintimeout"
        set action-type cli-script
        set script "config system global
            set admintimeout 479
            end"
        set output-size 10
        set timeout 0
        set accprofile "super_admin"
```

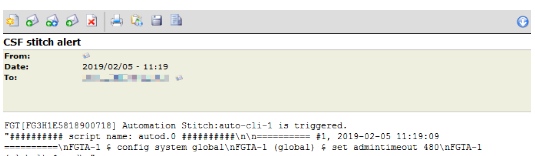
```
    next
    edit "auto-cli-1_email"
        set action-type email
        set email-to "admin@fortinet.com"
        set email-subject "CSF stitch alert"
        set message "%%results%%"
    next
end
```

3. Create the automation stitch:

```
config system automation-stitch
    edit "auto-cli-1"
        set trigger "auto-cli-1"
        config actions
            edit 1
                set action "admintimeout"
                set required enable
            next
            edit 2
                set action "auto-cli-1_email"
                set required enable
            next
        end
    next
end
```

Sample email

The email sent by the action will look similar to the following:



Execute a CLI script based on CPU and memory thresholds

Automation stitches can be created to run a CLI script and send an email message when CPU or memory usage exceeds specified thresholds.

In this example, two automation stitches are created that run a CLI script to collect debug information, and then email the results of the script to a specified email address when the CPU usage threshold is exceeded, or memory usage causes the FortiGate to enter conserve mode.

The maximum size of the CLI script action output is 16K characters.

To define CPU and memory usage thresholds:

```
config system global
    set cpu-use-threshold <percent>
    set memory-use-threshold-extreme <percent>
    set memory-use-threshold-green <percent>
    set memory-use-threshold-red <percent>
end
```

Where:

`cpu-use-threshold`	Threshold at which CPU usage is reported, in percent of total possible CPU utilization (default = 90).
`memory-use-threshold-extreme`	Threshold at which memory usage is considered extreme, and new sessions are dropped, in percent of total RAM (default = 95).
`memory-use-threshold-green`	Threshold at which memory usage forces the FortiGate to exit conserve mode, in percent of total RAM (default = 82).
`memory-use-threshold-red`	Threshold at which memory usage forces the FortiGate to enter conserve mode, in percent of total RAM (default = 88).

Configuring the automation stitches

High CPU usage stitch

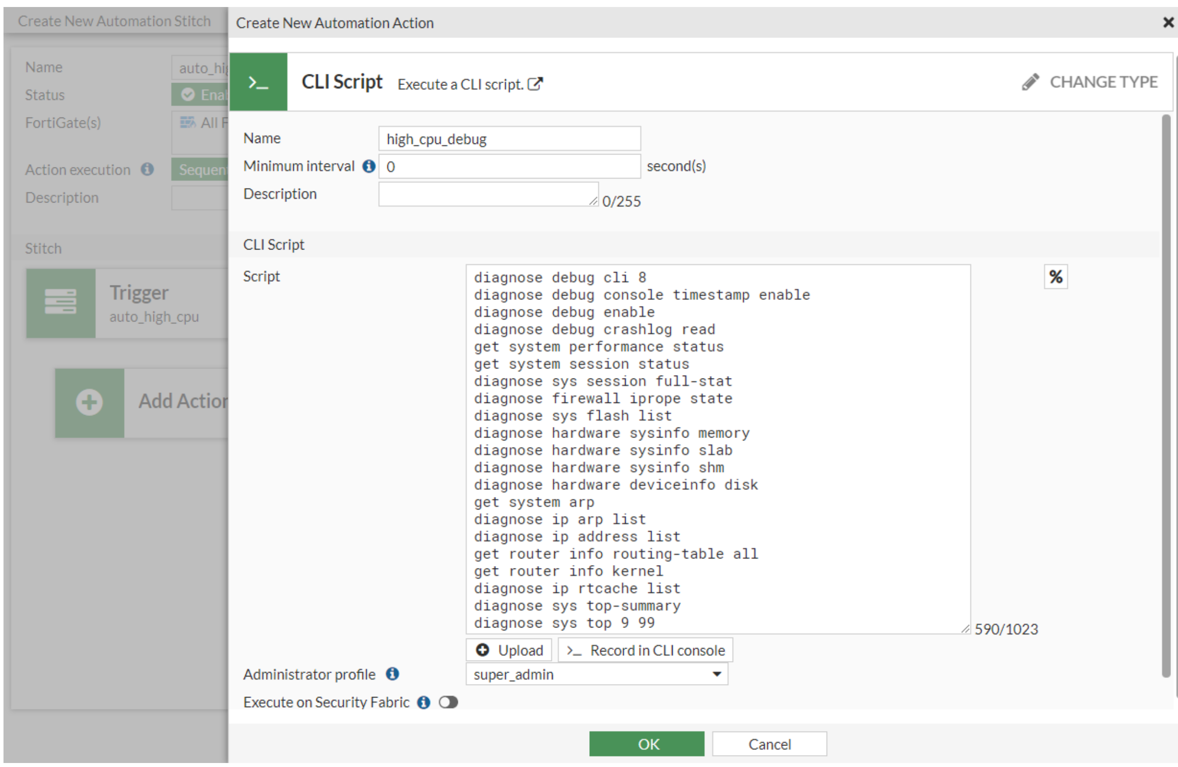
To create an automation stitch for high CPU usage in the GUI:

1. Go to *Security Fabric > Automation* and click *Create New*.
2. Enter the stitch name (*auto_high_cpu*).
3. Configure the trigger:
   a. Click *Add Trigger*.
   b. Click *Create* and select *High CPU*.
   c. Enter the name, *auto_high_cpu*.
   d. Click *OK*.
   e. Select the trigger in the list and click *Apply*.
4. Configure the CLI Script action:
   a. Click *Add Action*.
   b. Click *Create* and select *CLI Script*.
   c. Enter the following:

Name	high_cpu_debug

Script	`diagnose debug cli 8`
	`diagnose debug console timestamp enable`
	`diagnose debug enable`
	`diagnose debug crashlog read`
	`get system performance status`
	`get system session status`

	diagnose sys session full-stat diagnose firewall iprope state diagnose sys flash list diagnose hardware sysinfo memory diagnose hardware sysinfo slab diagnose hardware sysinfo shm diagnose hardware deviceinfo disk get system arp diagnose ip arp list diagnose ip address list get router info routing-table all get router info kernel diagnose ip rtcache list diagnose sys top-summary diagnose sys top 9 99
Administrator profile	Select a profile



    d. Click *OK*.
    e. Select the action in the list and click *Apply*.
5. Configure the Email notification action:
    a. Click *Add Action*.
    b. Click *Create* and select *Email*.
    c. Enter the following:

Name	auto_high_cpu_email

To	Enter an email address
Subject	CSF stitch alert: high_cpu
Body	%%results%%

   d. Click *OK*.
   e. Select the action in the list and click *Apply*.
6. Click *OK*.

To create an automation stitch for high CPU usage in the CLI:

1. Create the automation trigger:

```
config system automation-trigger
    edit "auto_high_cpu"
        set event-type high-cpu
    next
end
```

2. Create the automation actions:

```
config system automation-action
    edit "high_cpu_debug"
        set action-type cli-script
        set script "diagnose debug cli 8
diagnose debug console timestamp enable
diagnose debug enable
diagnose debug crashlog read
get system performance status
get system session status
diagnose sys session full-stat
diagnose firewall iprope state
diagnose sys flash list
diagnose hardware sysinfo memory
diagnose hardware sysinfo slab
diagnose hardware sysinfo shm
diagnose hardware deviceinfo disk
get system arp
diagnose ip arp list
diagnose ip address list
get router info routing-table all
get router info kernel
diagnose ip rtcache list
diagnose sys top-summary
diagnose sys top 9 99"
        set output-size 10
        set timeout 0
        set accprofile "super_admin"
    next
    edit "auto_high_cpu_email"
        set action-type email
        set email-to "person@fortinet.com"
        set email-subject "CSF stitch alert: high_cpu"
        set message "%%results%%"
    next
end
```

3. Create the automation stitch:

```
config system automation-stitch
    edit "auto_high_cpu"
        set trigger "auto_high_cpu"
        config actions
            edit 1
                set action "high_cpu_debug"
                set required enable
            next
            edit 2
                set action "auto_high_cpu_email"
                set required enable
            next
        end
    next
end
```

High memory usage stitch

To create an automation stitch for high memory usage in the GUI:

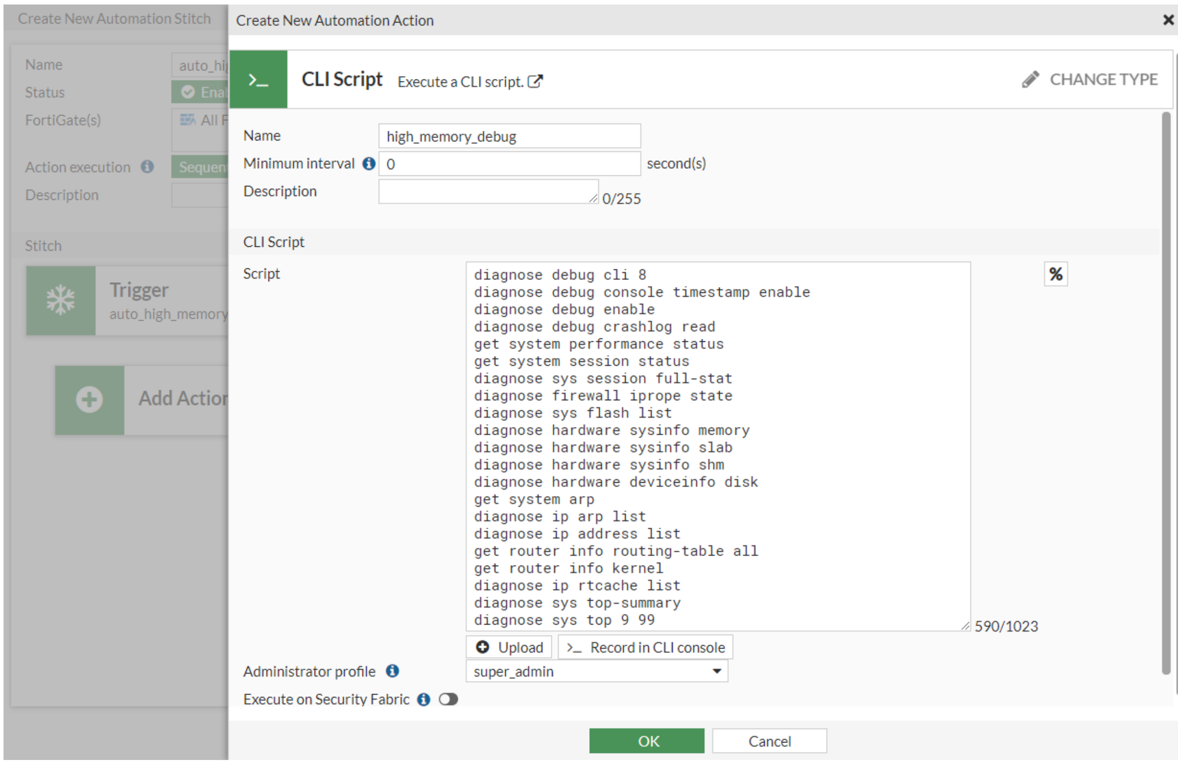
1. Go to *Security Fabric* > *Automation* and click *Create New*.
2. Enter the stitch name (*auto_high_memory*).
3. Configure the trigger:
   a. Click *Add Trigger*.
   b. Click *Create* and select *Conserve Mode*.
   c. Enter the name, *auto_high_memory*.
   d. Click *OK*.
   e. Select the trigger in the list and click *Apply*.
4. Configure the CLI Script action:
   a. Click *Add Action*.
   b. Click *Create* and select *CLI Script*.
   c. Enter the following:

Name	high_memory_debug

Script	`diagnose debug cli 8`
	`diagnose debug console timestamp enable`
	`diagnose debug enable`
	`diagnose debug crashlog read`
	`get system performance status`
	`get system session status`
	`diagnose sys session full-stat`
	`diagnose firewall iprope state`
	`diagnose sys flash list`
	`diagnose hardware sysinfo memory`
	`diagnose hardware sysinfo slab`
	`diagnose hardware sysinfo shm`
	`diagnose hardware deviceinfo disk`
	`get system arp`

	`diagnose ip arp list`
	`diagnose ip address list`
	`get router info routing-table all`
	`get router info kernel`
	`diagnose ip rtcache list`
	`diagnose sys top-summary`
	`diagnose sys top 9 99`

Administrator profile	Select a profile



   d. Click *OK*.
   e. Select the action in the list and click *Apply*.
5. Configure the Email notification action:
   a. Click *Add Action*.
   b. Click *Create* and select *Email*.
   c. Enter the following:

Name	auto_high_memory_email
To	Enter an email address
Subject	CSF stitch alert: high_memory
Body	%%results%%

   d. Click *OK*.
   e. Select the action in the list and click *Apply*.
6. Click *OK*.

To create an automation stitch for high memory usage in the CLI:

1. Create the automation trigger:

```
config system automation-trigger
    edit "auto_high_memory"
        set event-type low-memory
    next
end
```

2. Create the automation actions:

```
config system automation-action
    edit "high_memory_debug"
        set action-type cli-script
        set script "diagnose debug cli 8
diagnose debug console timestamp enable
diagnose debug enable
diagnose debug crashlog read
get system performance status
get system session status
diagnose sys session full-stat
diagnose firewall iprope state
diagnose sys flash list
diagnose hardware sysinfo memory
diagnose hardware sysinfo slab
diagnose hardware sysinfo shm
diagnose hardware deviceinfo disk
get system arp
diagnose ip arp list
diagnose ip address list
get router info routing-table all
get router info kernel
diagnose ip rtcache list
diagnose sys top-summary
diagnose sys top 9 99"
        set output-size 10
        set timeout 0
        set accprofile "super_admin"
    next
    edit "auto_high_memory_email"
        set action-type email
        set email-to "person@fortinet.com"
        set email-subject "CSF stitch alert: high_memory"
        set message "%%results%%"
    next
end
```

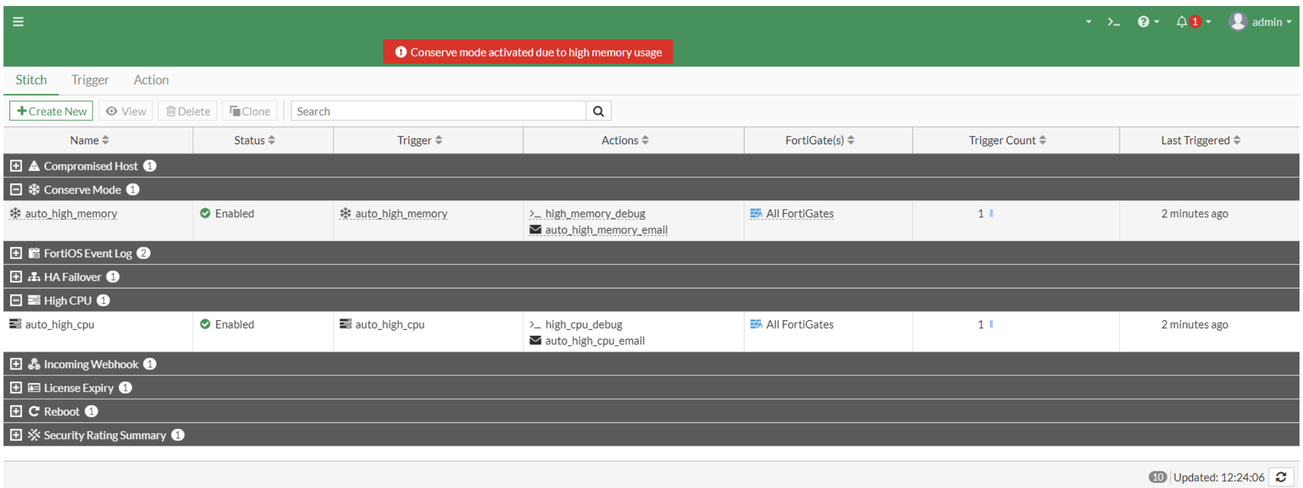
3. Create the automation stitch:

```
config system automation-stitch
    edit "auto_high_memory"
        set trigger "auto_high_memory"
        config actions
            edit 1
                set action "high_memory_debug"
                set required enable
            next
```

```
                edit 2
                    set action "auto_high_memory_email"
                    set required enable
                next
            end
        next
    end
```

Results

When the FortiGate enters conserve mode due to the `memory-use-threshold-red` being exceeded, the GUI displays a notice, and the *auto_high_memory* automation stitch is triggered. This causes the CLI script to run and the script results are emailed to the specified address.



Here is sample text from the email message:

```
CSF stitch alert: high_memory
noreply@notification.fortinet.net
Thu 11/21/2019 11:06 AM
John Doe
FGT[FGVM16TM19000000] Automation Stitch:auto_high_memory is triggered.
########## script name: autod.47 ##########
========== #1, 2019-11-21 11:07:24 ==========
FGVM16TM19000000 $  diag deb cli 8
Debug messages will be on for 25 minutes.
FGVM16TM19000000 $  diag deb console timestamp enable
FGVM16TM19000000 $  diag deb enable
FGVM16TM19000000 $  diag deb crashlog read
1: 2019-08-08 11:35:25 the killed daemon is /bin/dhcpcd: status=0x0
2: 2019-08-08 17:52:47 the killed daemon is /bin/pyfcgid: status=0x0
3: 2019-08-23 11:32:31 from=license status=INVALID
4: 2019-08-23 11:32:32 from=license status=INVALID
5: 2019-11-21 09:53:31 from=license status=VALID
...
```

Webhook action

The webhook automation stitch action makes HTTP and HTTPS requests to a specified server, with custom headers, bodies, ports, and methods. It can be used to leverage the ubiquity of HTML requests and APIs to integrate with other tools.

The URI and HTTP body can use parameters from logs or previous action results. Wrapping the parameter with %% will replace the expression with the JSON value for the parameter, for example: *%%results.source%%* is the source property from the previous action.

In this example, a specific log message (failed administrator log in attempt) triggers the FortiGate to send the contents of the log to a server. The server responds with a generic reply. This example assumes that the server is already configured and able to communicate with the FortiGate.

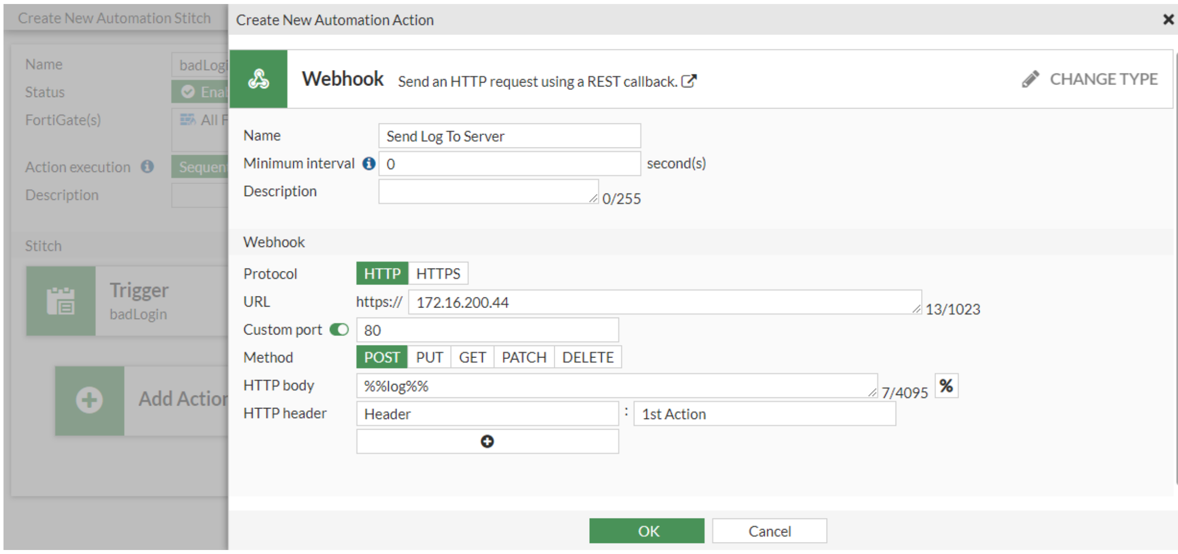
To configure the webhook automation stitch in the GUI:

1. Go to *Security Fabric > Automation* and click *Create New*.
2. Enter the stitch name (*badLogin*).
3. Configure the trigger:
   a. Click *Add Trigger*.
   b. Click *Create* and select *FortiOS Event Log*.
   c. Enter the following:

Name	badLogin
Event	Admin login failed

   d. Click *OK*.
   e. Select the trigger in the list and click *Apply*.
4. Configure the automation stitch action:
   a. Click *Add Action*.
   b. Click *Create* and select *Webhook*.
   c. Enter the following:

Name	Send Log To Server
Protocol	HTTP
URL	172.16.200.44
Custom port	Enable and enter 80
Method	POST
HTTP body	%%log%%
HTTP header	Header : 1st Action



   d. Click *OK*.
   e. Select the action in the list and click *Apply*.

5. Click *OK*.

To configure the webhook automation stitch in the CLI:

1. Create an automation trigger:

```
config system automation-trigger
    edit "badLogin"
        set event-type event-log
        set logid 32002
    next
end
```

2. Create the automation action:

```
config system automation-action
    edit "Send Log To Server"
        set action-type webhook
        set uri "172.16.200.44"
        set http-body "%%log%%"
        set port 80
        set headers "Header:1st Action"
    next
end
```

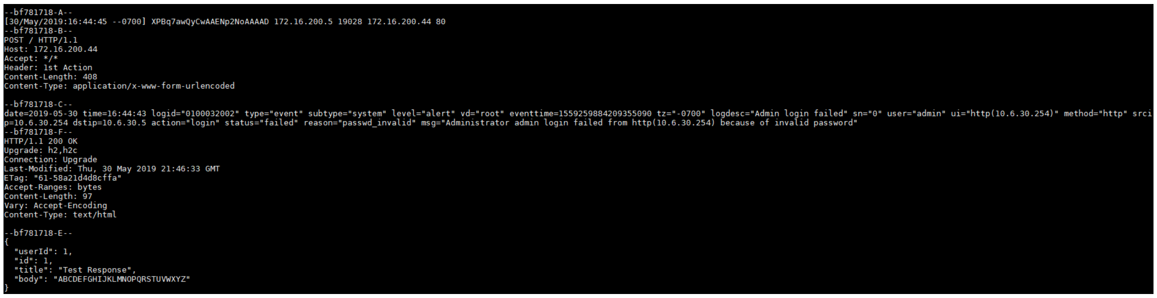
3. Create the automation stitch:

```
config system automation-stitch
    edit "badLogin"
        set trigger "badLogin"
        config actions
            edit 1
                set action "Send Log To Server"
                set required enable
            next
        end
```

```
        next
    end
```

To test the automation stitch:

1. Attempt to log in to the FortiGate with an incorrect username or password.
2. On the server, check the log to see that its contents were sent by the FortiGate.



   The body content is replaced with the log from the trigger.
3. On the FortiGate, go to *Log & Report* > *System Events* to confirm that the stitch was activated.
4. Go to *Security Fabric* > *Automation* to see the last time that the stitch was triggered.

Diagnose commands

To enable log dumping:

```
# diagnose test application autod 1
autod dumped total:1 logs, num of logids:1
autod log dumping is enabled

vdom:root(0) logid:32002 len:408 log:
date=2019-05-30 time=17:41:03 logid="0100032002" type="event" subtype="system" level="alert"
vd="root" eventtime=1559263263858888451 tz="-0700" logdesc="Admin login failed" sn="0"
user="admin" ui="http(10.6.30.254)" method="http" srcip=10.6.30.254 dstip=10.6.30.5
action="login" status="failed" reason="passwd_invalid" msg="Administrator admin login failed
from http(10.6.30.254) because of invalid password"
autod log dumping is disabled

autod logs dumping summary:
        logid:32002 count:1

autod dumped total:1 logs, num of logids:1
```

To show the automation settings:

```
# diagnose test application autod 2
csf: enabled   root:yes
total stitches activated: 2

stitch: badLogin
       destinations: all
       trigger: badLogin

       local hit: 6 relayed to: 6 relayed from: 6
       actions:
```

```
                Send Log To Server type:webhook interval:0
                        delay:0 required:no
                        proto:0 method:0 port:80
                        uri: 172.16.200.44
                        http body: %%log%%
                        headers:
                        0. Header:1st Action
```

To show the automation statistics:

```
# diagnose test application autod 3

stitch: badLogin

        local hit: 1 relayed to: 1 relayed from: 1
        last trigger:Wed Jul 10 12:14:14 2019
        last relay:Wed Jul 10 12:14:14 2019

        actions:
                Send Log To Server:
                        done: 1 relayed to: 1 relayed from: 1
                        last trigger:Wed Jul 10 12:14:14 2019
                        last relay:Wed Jul 10 12:14:14 2019

logid2stitch mapping:
id:32002  local hit: 3 relayed to: 3 relayed from: 3
        badLogin

action run cfg&stats:
total:55 cur:0 done:55 drop:0
        email:
                flags:10
                stats: total:4 cur:0 done:4 drop:0
        fortiexplorer-notification:
                flags:1
                stats: total:0 cur:0 done:0 drop:0
        alert:
                flags:0
                stats: total:0 cur:0 done:0 drop:0
        disable-ssid:
                flags:7
                stats: total:0 cur:0 done:0 drop:0
        quarantine:
                flags:7
                stats: total:0 cur:0 done:0 drop:0
        quarantine-forticlient:
                flags:4
                stats: total:0 cur:0 done:0 drop:0
        quarantine-nsx:
                flags:4
                stats: total:0 cur:0 done:0 drop:0
        ban-ip:
                flags:7
                stats: total:0 cur:0 done:0 drop:0
        aws-lambda:
                flags:11
```

```
                stats: total:21 cur:0 done:21 drop:0
        webhook:
                flags:11
                stats: total:6 cur:0 done:6 drop:0
        cli-script:
                flags:10
                stats: total:4 cur:0 done:4 drop:0
        azure-function:
                flags:11
                stats: total:0 cur:0 done:0 drop:0
        google-cloud-function:
                flags:11
                stats: total:0 cur:0 done:0 drop:0
        alicloud-function:
                flags:11
                stats: total:20 cur:0 done:20 drop:0
```

To enable debug output and turn on automation debug messages for about 30 minutes:

```
# diagnose debug enable
# diagnose debug application autod -1
__auto_generate_generic_curl_request()-358: Generating generic automation CURL request for
action (Send Log To Server).
__auto_generate_generic_curl_request()-406: Generic automation CURL request POST data for
action (Send Log To Server):
date=2019-05-30 time=16:44:43 logid="0100032002" type="event" subtype="system" level="alert"
vd="root" eventtime=1559259884209355090 tz="-0700" logdesc="Admin login failed" sn="0"
user="admin" ui="http(10.6.30.254)" method="http" srcip=10.6.30.254 dstip=10.6.30.5
action="login" status="failed" reason="passwd_invalid" msg="Administrator admin login failed
from http(10.6.30.254) because of invalid password"

__auto_generic_curl_request_close()-512: Generic CURL request response body from
http://172.16.200.44:
{
  "userId": 1,
  "id": 1,
  "title": "Test Response",
  "body": "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
}
```

Slack integration webhook

A webhook can be created to post messages and notifications to Slack.

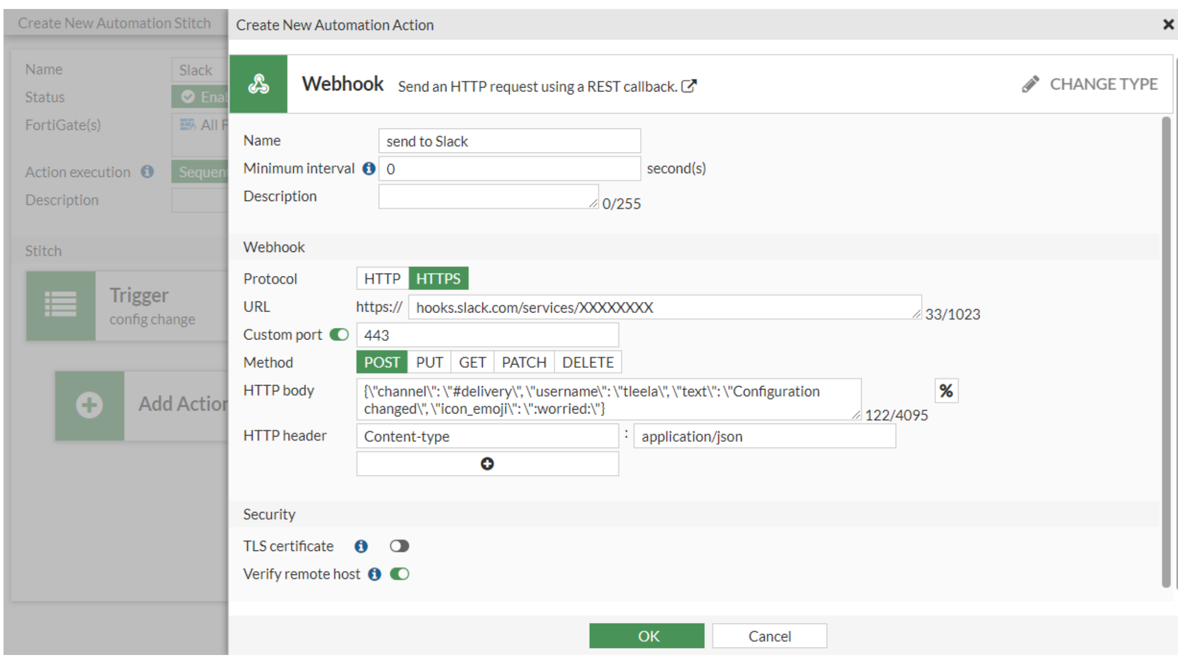
In this example, a configuration change triggers the FortiGate to post a message to Slack.

To create a webhook automation stitch for Slack integration in the GUI:

1. Create an incoming webhook in Slack. See Sending messages using Incoming Webhooks for more information.
2. Go to *Security Fabric > Automation* and click *Create New*.
3. Enter the stitch name.
4. Configure the trigger:
   a. Click *Add Trigger*.
   b. Click *Create* and select *Configuration Change*.

   c. Enter a name (*config change*).
   d. Click *OK*.
   e. Select the trigger in the list and click *Apply*.
5. Configure the action:
   a. Click *Add Action*.
   b. Click *Create* and select *Webhook*.
   c. Enter the following:

Name	send to Slack
Protocol	HTTPS
URL	Enter the incoming webhook URL created in Slack
Custom port	Enable and enter 443
Method	POST
HTTP body	{\"channel\": \"#delivery\", \"username\": \"tleela\", \"text\": \"Configuration changed\", \"icon_emoji\": \":worried:\"}
HTTP header	Content-type : application/json



   d. Click *OK*.
   e. Select the action in the list and click *Apply*.
6. Click *OK*.

To create a webhook automation stitch for Slack integration in the CLI:

1. Create an incoming webhook in Slack. See Sending messages using Incoming Webhooks for more information.
2. Create the automation trigger:

```
config system automation-trigger
    edit "config change"
        set event-type config-change
    next
end
```

3. Create the automation action:

```
config system automation-action
    edit "send to Slack"
        set action-type webhook
        set protocol https
        set uri "hooks.slack.com/services/XXXXXXXX"
        set http-body "{\"channel\": \"#delivery\", \"username\": \"tleela\", \"text\":
\"Configuration changed\", \"icon_emoji\": \":worried:\"}"
        set port 443
        set headers "Content-type:application/json"
    next
end
```

4. Create the automation stitch:

```
config system automation-stitch
    edit "Slack"
        set trigger "config change"
        config actions
            edit 1
                set action "send to Slack"
                set required enable
            next
        end
    next
end
```

Microsoft Teams integration webhook

A webhook can be created to post messages and notifications to Microsoft Teams.

In this example, a configuration change triggers the FortiGate to post a message to Teams.

To create a webhook automation stitch for Teams integration in the GUI:

1. Create an incoming webhook in Teams. See Create an incoming webhook for information.
2. Go to *Security Fabric > Automation* and click *Create New*.
3. Enter the stitch name.
4. Configure the trigger:
   a. Click *Add Trigger*.
   b. Click *Create* and select *Configuration Change*.
   c. Enter a name (*Teams*).

d. Click *OK*.

e. Select the trigger in the list and click *Apply*.

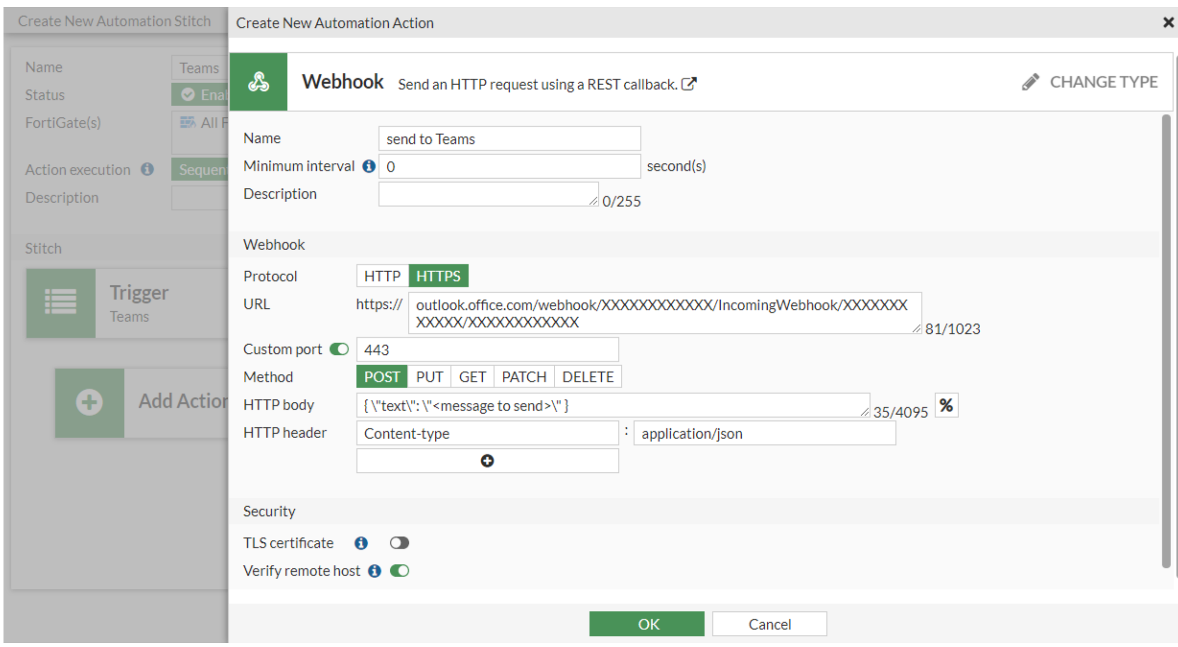
5. Configure the action:

a. Click *Add Action*.

b. Click *Create* and select *Webhook*.

c. Enter the following:

Name	send to Teams
Protocol	HTTPS
URL	Enter the incoming webhook URL created in Teams
Custom port	Enable and enter 443
Method	POST
HTTP body	{ \"text\": \"<message to send>\" }
HTTP header	Content-type : application/json



d. Click *OK*.

e. Select the action in the list and click *Apply*.

6. Click *OK*.

To create a webhook automation stitch for Teams integration in the CLI:

1. Create an incoming webhook in Teams. See Create an incoming webhook for information.

2. Create the automation trigger:

```
config system automation-trigger
    edit "Teams"
        set event-type config-change
```

```
    next
end
```

3. Create the automation action:

```
config system automation-action
    edit "send to Teams"
        set action-type webhook
        set protocol https
        set uri
"outlook.office.com/webhook/XXXXXXXXXXXX/IncomingWebhook/XXXXXXXXXXXX/XXXXXXXXXXXX"
        set http-body "{ \"text\": \"<message to send>\" }"
        set port 443
        set headers "Content-type:application/json"
    next
end
```

4. Create the automation stitch:

```
config system automation-stitch
    edit "Teams"
        set trigger "Teams"
        config actions
            edit 1
                set action "send to Teams"
                set required enable
            next
        end
    next
end
```

For information about more advanced messages that can be configured and sent to the Teams incoming webhook, see Sending messages to connectors and webhooks.

Public and private SDN connectors

Cloud SDN connectors provide integration and orchestration of Fortinet products with public and private cloud solutions. In a typical cloud environment, resources are dynamic and often provisioned and scaled on-demand. By using an SDN connector, you can ensure that changes to cloud environment attributes are automatically updated in the Security Fabric.

To protect the East-West or North-South traffic in these environments, the FortiGate uses the SDN connector to sync the dynamic addresses that these volatile environments use. You can then configure the dynamic address objects as sources or destinations for firewall policies. When you make changes to cloud environment resources, such as moving them to a new location or assigning different IP addresses to them, you do not need to modify the policy in FortiOS, as the SDN connector syncs changes to the cloud address objects.

These configurations consist of three primary steps:

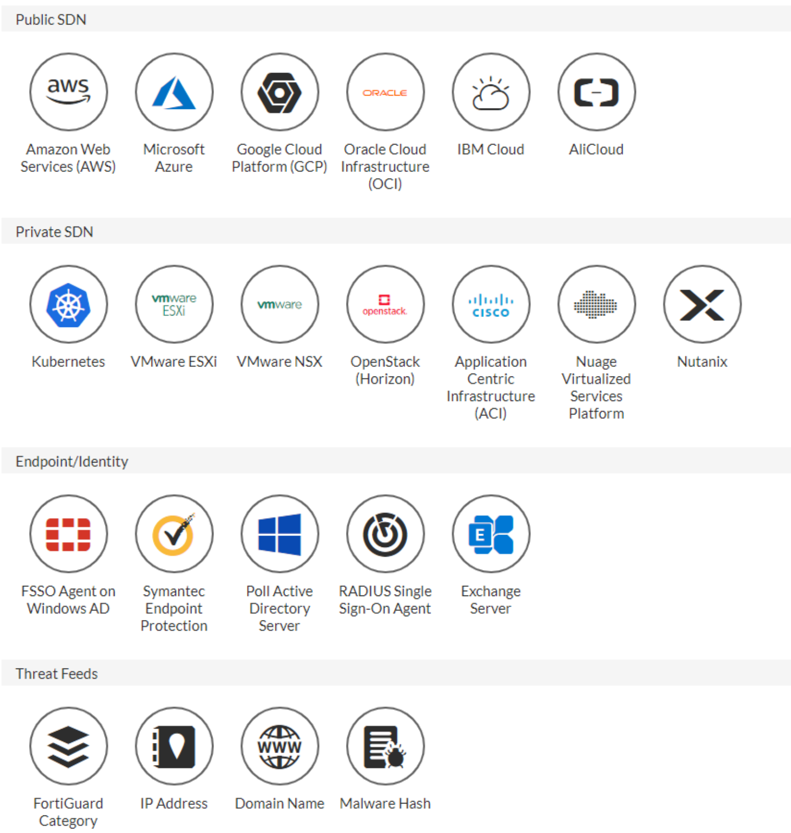
1. Configure the cloud SDN connector to connect your FortiGate and public or private cloud account.
2. Create dynamic address objects to use the SDN connector. Use filters to sync only cloud address objects that you

require.

3. Apply the dynamic address objects to your firewall policy to protect your traffic.

This chapter explores the steps in detail and describes how to connect to each currently supported cloud platform. This chapter does not discuss cloud account role-based or permission requirements. The respective cloud documents contain this information.

The following external connector categories are available in the Security Fabric: Public SDN, Private SDN, Endpoint/Identity, and Threat Feeds.



If VDOMs are enabled, SDN and Threat Feeds connectors are in the global settings, and Endpoint/Identity connectors are per VDOM.

Getting started with public and private SDN connectors

You can use SDN connectors to connect your FortiGate to public and private cloud solutions. By using an SDN connector, you can ensure that changes to cloud environment attributes are automatically updated in the Security Fabric. You can use SDN connector address objects to create policies that provide dynamic access control based on cloud environment attribute changes. There is no need to manually reconfigure addresses and policies whenever changes to the cloud environment occur.

There are four steps to creating and using an SDN connector:

1. Gather the required information. The required information depends on which public or private cloud solution SDN connector you are configuring.
2. Creating the SDN connector on page 2428
3. Creating an SDN connector address on page 2428
4. Adding the address to a firewall policy on page 2430

The following provides general instructions for creating an SDN connector and using the dynamic address object in a firewall policy. For instructions for specific public and private cloud solutions, see the relevant topic in this guide. For advanced scenarios regarding SDN connectors, see the appropriate FortiOS 7.2 cloud platform guide.

Creating the SDN connector

To create an SDN connector in the GUI:

1. Go to *Security Fabric > External Connectors*.
2. Click *Create New*.
3. Click the desired public or private cloud.
4. Enter the *Name*, *Status*, and *Update Interval* for the connector.
5. Enter previously collected information for the connector as needed.
6. Click *OK*.

To create an SDN connector in the CLI:

```
config system sdn-connector
    edit <name>
        set status {enable | disable}
        set type {connector type}
        ...
        set update-interval <integer>
    next
end
```

The available CLI commands vary depending on the selected SDN connector type.

Creating an SDN connector address

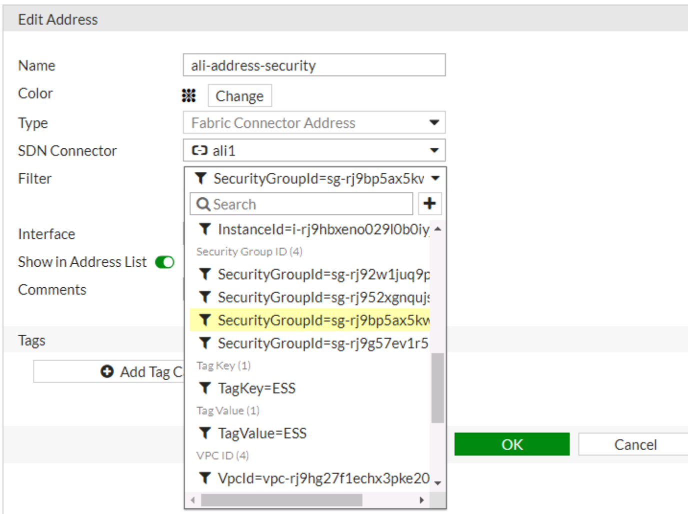
You can use an SDN connector address in the following ways:

- As the source or destination address for firewall policies.
- To automatically update changes to addresses in the public or private cloud environment, based on specified filters.
- To automatically apply changes to firewall policies that use the address, based on specified filters.

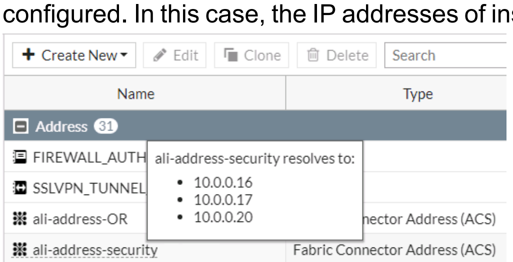
To create an SDN connector address in the GUI:

1. Go to *Policy & Objects > Addresses*.
2. Click *Create New > Address*.

3. Configure the address:
   a. Set the *Type* to *Dynamic*.
   b. From the *Sub Type* dropdown list, select *Fabric Connector Address*.
   c. From the *SDN Connector* dropdown list, select the desired SDN connector.
   d. From the *Filter* dropdown list, configure the desired filter. The filters available depend on the selected SDN connector type. The SDN connector automatically populates and updates IP addresses only for instances that satisfy the filter requirements. In this example, the address will automatically populate and update IP addresses only for AliCloud instances that belong to the specified security group:



   You can set filtering conditions using multiple entries with AND ("&") or OR ("|"). When both AND and OR are specified, AND is interpreted first, then OR.
   e. Configure other settings as desired.
   f. Click *OK*.
4. Ensure that the SDN connector resolves dynamic firewall IP addresses as configured:
   a. Go to *Policy & Objects > Addresses*.
   b. Hover over the address that you created to see a list of IP addresses for instances that satisfy the filter that you configured. In this case, the IP addresses of instances that belong to the specified security group display:



To create an SDN connector address in the CLI:

1. Create the address:

```
config firewall address
   edit <name>
      set type dynamic
      set sdn <sdn_connector>
      set visibility enable
      set associated-interface <interface_name>
      set color <integer>
```

```
        ...
        set comment <comment>
        config tagging
            edit <name>
                set category <string>
                set tags <strings>
            next
        end
    next
end
```

2. Ensure that the SDN connector resolves dynamic firewall IP addresses as configured by running `show`. The following shows example output:

```
config firewall address
    edit "ali-address-security"
        set type dynamic
        config list
            edit "10.0.0.16"
            next
            edit "10.0.0.17"
            next
            edit "10.0.20.20"
            next
        end
        ...
    next
end
```

The available CLI commands vary depending on the selected SDN connector type.

Adding the address to a firewall policy

You can use an SDN connector address as the source or destination address in a policy.

To add the address to a firewall policy in the GUI:

1. Go to *Policy & Objects > Firewall Policy*.
2. Click *Create New*.
3. Use the SDN connector address as the source or destination address.
4. Configure the remaining settings as needed.
5. Click *OK*.

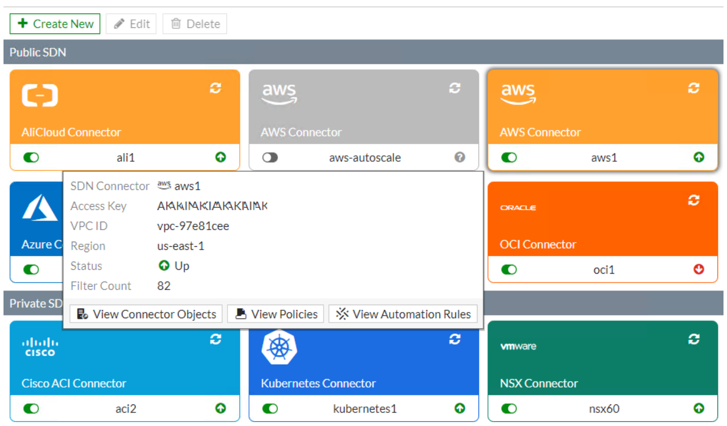
To add the address to a firewall policy in the CLI:

```
config firewall policy
    edit 0
        set name <name>
        set srcintf <port_name>
        set dstintf <port_name>
        set srcaddr <firewall_address>
```

```
        set dstaddr <firewall_address>
        set action accept
        set schedule <schedule>
        set service <service>
    next
end
```

Connector tooltips

In *Security Fabric > External Connectors*, hover over an SDN connector to view a tooltip that shows basic configuration information.



Three buttons provide additional information:

Button	Information
View Connector Objects	Connector's dynamic objects, such as filters and instances.
View Policies	List of policies that use the dynamic addresses from the connector.
View Automation Rules	List of automation actions that use the connector.

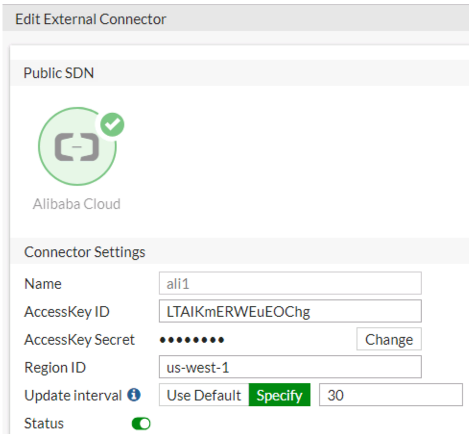
AliCloud SDN connector using access key

FortiOS automatically updates dynamic addresses for AliCloud using an AliCloud SDN connector, including mapping the following attributes from AliCloud instances to dynamic address groups in FortiOS:

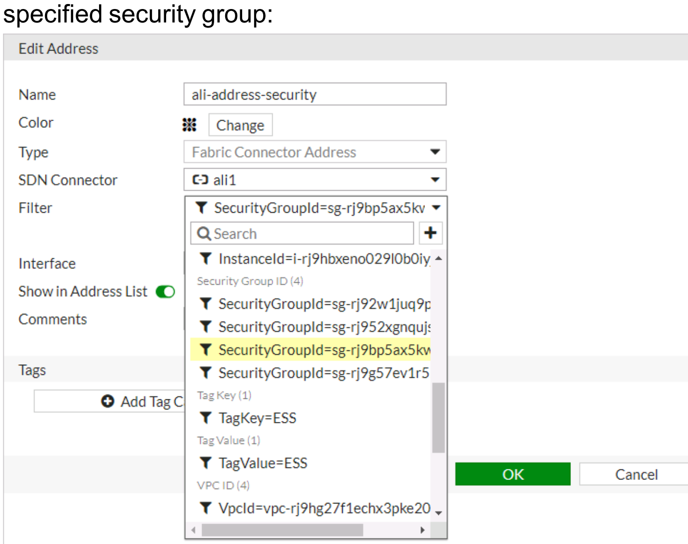
- ImageId
- InstanceId
- SecurityGroupId
- VpcId
- VSwitchId
- TagKey
- TagValue

To configure AliCloud SDN connector using the GUI:

1. Configure the AliCloud SDN connector:
   a. Go to *Security Fabric > External Connectors*.
   b. Click *Create New*, and select *AliCloud*.
   c. Configure as shown, substituting the access key, secret, and region ID for your deployment. The update interval is in seconds.

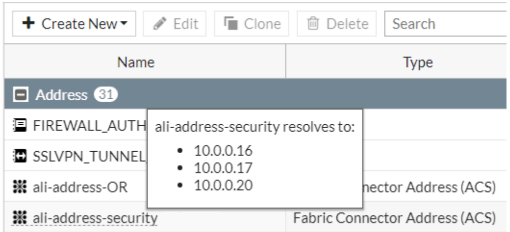


2. Create a dynamic firewall address for the configured AliCloud SDN connector:
   a. Go to *Policy & Objects > Addresses*.
   b. Click *Create New*, then select *Address*.
   c. Configure the address as shown, selecting the desired filter in the *Filter* dropdown list. In this example, the address will automatically populate and update IP addresses only for AliCloud instances that belong to the specified security group:



3. Ensure that the AliCloud SDN connector resolves dynamic firewall IP addresses:
   a. Go to *Policy & Objects > Addresses*.
   b. Hover over the address created in step 2 to see a list of IP addresses for instances that belong to the security

group configured in step 2:



To configure AliCloud SDN connector using CLI commands:

1. Configure the AliCloud SDN connector:
```
config system sdn-connector
   edit "ali1"
      set type acs
      set access-key "LTAIKmERWEuEOChg"
      set secret-key xxxxx
      set region "us-west-1"
      set update-interval 30
   next
end
```
2. Create a dynamic firewall address for the configured AliCloud SDN connector with the supported AliCloud filter. In this example, the AliCloud SDN Connector will automatically populate and update IP addresses only for instances that belong to the specified security group:
```
config firewall address
   edit "ali-address-security"
      set type dynamic
      set sdn "ali1"
      set filter "SecurityGroupId=sg-rj9bp5ax5kwy3gqdizqb"
   next
end
```
3. Confirm that the AliCloud SDN connector resolves dynamic firewall IP addresses using the configured filter:
```
config firewall address
   edit "ali-address-security"
      set type dynamic
      set sdn "ali1"
      set filter "SecurityGroupId=sg-rj9bp5ax5kwy3gqdizqb"
      config list
         edit "10.0.0.16"
         next
         edit "10.0.0.17"
         next
         edit "10.0.0.20"
         next
      end
   next
end
```

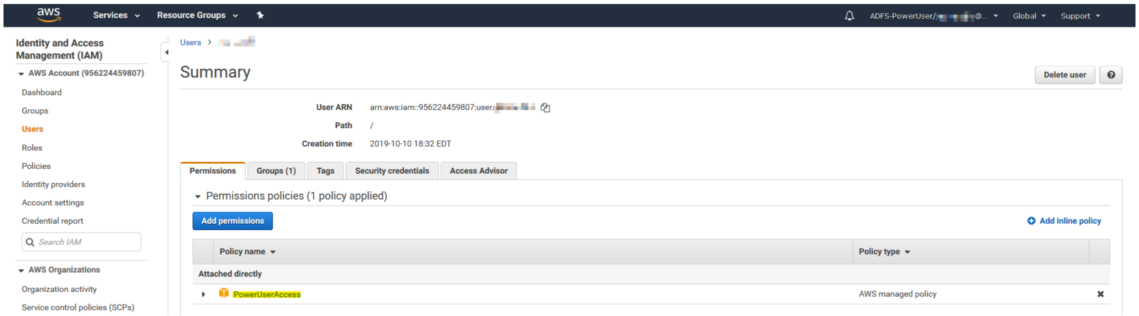
AWS SDN connector using certificates

FortiOS automatically updates dynamic addresses for AWS using an AWS SDN connector, including mapping attributes from AWS instances to dynamic address groups in FortiOS.

Configuring the SDN connector using the GUI, then checking the configuration using the CLI is recommended.

To configure an AWS SDN connector using the GUI:

1. Configure the AWS SDN connector:
   a. Go to *Security Fabric > External Connectors*.
   b. Click *Create New*, and select *Amazon Web Services (AWS)*.
   c. In the *Access key ID* field, enter the key created in the AWS management portal.
   d. In the *Secret access key* field, enter the secret access key accompanying the above access key.
   e. In the *Region name* field, enter the region name. Refer to AWS Regions and Endpoints for the desired region name.
   f. In the *VPC ID* field, enter the VPC ID within the specified region you desire to cover with the SDN connector.
   g. Click *OK*.
2. Check the configuration using the CLI:
```
config system sdn-connector
  edit "<connector-name>"
    show
```
   The output resembles the following:
```
config system sdn-connector
  edit "<connector-name>"
    set access-key "<example-access-key>"
    set secret-key ENC <example-secret-key>
    set region "us-west-2"
    set vpc-id "vpc-e1e4b587"
    set update-interval 1
  next
end
```
   If you see that the SDN connector is not enabled in *Security Fabric > External Connectors* in the GUI, run the following commands to enable the SDN connector:
```
diagnose deb application awsd -1
diagnose debug enable
```
   The output may display an error like the following:
```
FGT # awsd sdn connector AWS_SDN prepare to update
awsd sdn connector AWS_SDN start updating
aws curl response err, 403
<?xml version="1.0" encoding="UTF-8"?>
<Response><Errors><Error><Code>UnauthorizedOperation</Code><Message>You are not
     authorized to perform this
     operation.</Message></Error></Errors><RequestID>8403cc11-b185-41da-ad6d-
     23bb4db7d00a</RequestID></Response>
awsd curl failed 403
awsd sdn connector AWS_SDN failed to get instance list
aws curl response err, 403
{"Message":"User: arn:aws:iam::956224459807:user/jcarcavallo is not authorized to
     perform: eks:ListClusters on resource: arn:aws:eks:us-east-
     1:956224459807:cluster/*"}
awsd sdn connector AWS_SDN get EKS cluster list failed
awsd sdn connector AWS_SDN list EKS cluster failed
awsd sdn connector AWS_SDN start updating IP addresses
awsd sdn connector AWS_SDN finish updating IP addresses
awsd reap child pid: 569
```
   In this case, you must configure power user access for the current administrator in the AWS management console:



After configuring power user access, run the following commands:

```
diagnose deb application awsd -1
diagnose debug enable
```

The output should display without error, as follows:

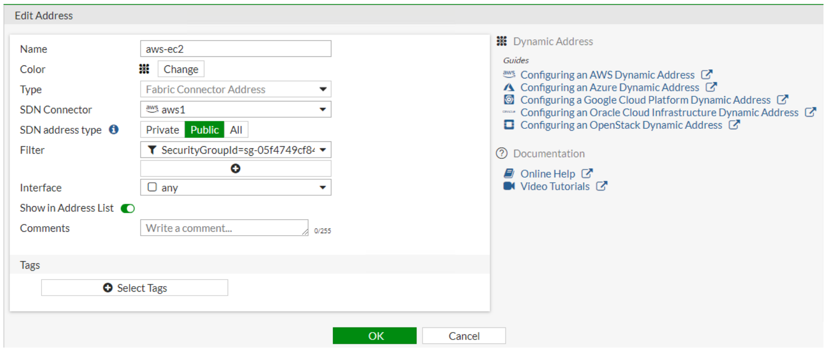
```
FGT # AWSD: update sdn connector AWS_SDN status to enabled
awsd sdn connector AWS_SDN prepare to update
awsd sdn connector AWS_SDN start updating
awsd get ec2 instance info successfully
awsd sdn connector AWS_SDN start updating IP addresses
awsd sdn connector AWS_SDN finish updating IP addresses
awsd reap child pid: 893
```

The AWS connector is now enabled.

3. Create a dynamic firewall address for the configured AWS SDN connector:
   a. Go to *Policy & Objects* > *Addresses*.
   b. Click *Create New*, then select *Address*.
   c. From the *Type* dropdown list, select *Dynamic*.
   d. From the *Sub Type* dropdown list, select *Fabric Connector Address*.

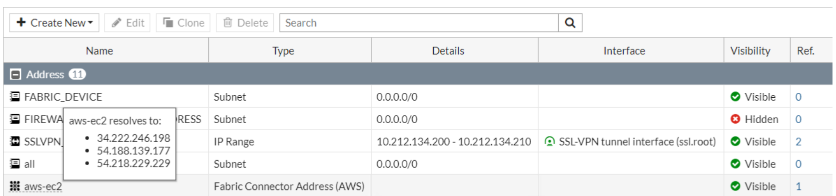
e. In the *Filter* field, add the desired filters. The following filters are supported:

Description	Key	Example value
Architecture	architecture	x86
Autoscaling group	AutoScaleGroup	10703c-4f731e90-fortigate-payg-auto-scaling-group
AZ	placement.availabilityzone	us-east-1a
Group name	placement.groupname	
Image ID	imageId	ami-123456
Instance ID	instanceId	i-12345678
Instance type	instanceType	t2.micro
Key name	keyName	
Kubernetes cluster	k8s_cluster	
Kubernetes label and its name	k8s_label.Name	
Kubernetes namespace	k8s_namespace	
Kubernetes node name	k8s_nodename	
Kubernetes pod name	k8s_podname	
Kubernetes region	k8s_region	
Kubernetes service name	k8s_servicename	
Kubernetes zone	k8s_zone	
Private DNS name	privateDnsName	ip-172-31-10-211.us-west-2.compute.internal
Public DNS name	publicDnsName	ec2-54-202-168-254.us-west-2.compute.amazonaws.com
Security group ID	SecurityGroupId	
Subnet ID	subnetId	sub-123456
Tag and its name. This key supports a maximum of eight tags.	tag.Name	
Tenancy placement	placement.tenancy	
VPC ID	VpcId	

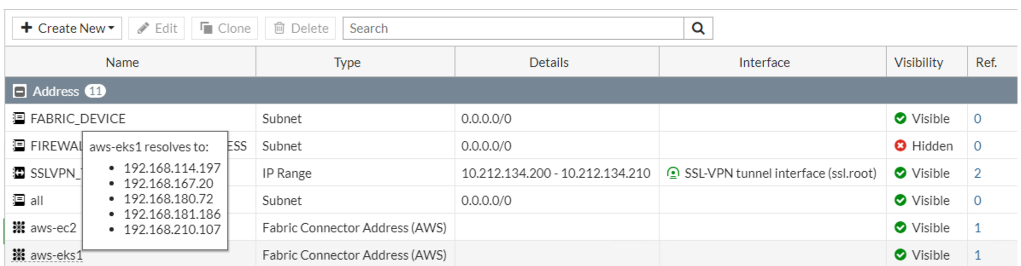


4. Ensure that the AWS SDN connector resolves dynamic firewall IP addresses:
   a. Go to *Policy & Objects* > *Addresses*.
   b. Hover over the address created in step 2 to see a list of IP addresses for instances that belong to the security group configured in step 2.

   The following is an example for a public SDN address type:



   The following is an example for a private SDN address type:



To configure AWS SDN connector using CLI commands:

1. Configure the AWS connector:

```
config system sdn-connector
   edit "<connector-name>"
      set access-key "<example-access-key>"
      set secret-key ENC <example-secret-key>
      set region "us-west-2"
      set vpc-id "vpc-e1e4b587"
      set update-interval 1
   next
end
```

2. Create a dynamic firewall address for the configured AWS SDN connector with the supported filter:

```
config firewall address
   edit "aws-ec2"
      set type dynamic
      set sdn "<connector-name>"
      set filter "SecurityGroupId=sg-05f4749cf84267548"
      set sdn-addr-type public
```

```
  next
  edit "aws-eks1"
    set type dynamic
    set sdn "<connector-name>"
    set filter "K8S_Region=us-west-2"
  next
end
```

3. Confirm that the AWS SDN connector resolves dynamic firewall IP addresses using the configured filter:

```
config firewall address
  edit "aws-ec2"
    set type dynamic
    set sdn "<connector-name>"
    set filter "SecurityGroupId=sg-05f4749cf84267548"
    set sdn-addr-type public
    config list
      edit "34.222.246.198"
      next
      edit "54.188.139.177"
      next
      edit "54.218.229.229"
      next
    end
  next
  edit "aws-eks1"
    set type dynamic
    set sdn "<connector-name>"
    set filter "K8S_Region=us-west-2"
    config list
      edit "192.168.114.197"
      next
      edit "192.168.167.20"
      next
      edit "192.168.180.72"
      next
      edit "192.168.181.186"
      next
      edit "192.168.210.107"
      next
    end
  next
end
```

To add an EC2 instance to test automatic address population:

1. Assume that you want to boot up another instance with an IP address of 34.222.246.178, which is currently stopped. This instance belongs to the security group that the aws-ec2 address is filtering for. In the AWS management portal, start the instance.
2. Verify that the instance is running.
3. At this point, running `show` again shows the SDN connector has automatically populated and added the 34.222.246.178 instance.

```
config firewall address
  edit "aws-ec2"
    set type dynamic
    set sdn "<connector-name>"
    set filter "SecurityGroupId=sg-05f4749cf84267548"
```

```
        set sdn-addr-type public
        config list
            edit "34.222.246.198"
            next
            edit "54.188.139.177"
            next
            edit "54.218.229.229"
            next
            edit "34.222.246.178"
            next
        end
    next
end
```

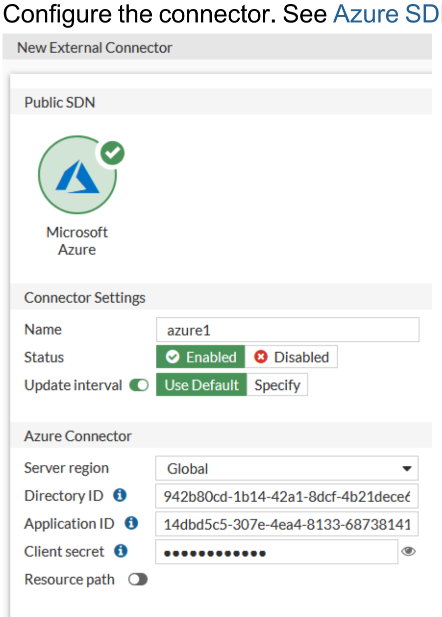
Therefore, administrators do not need to add this instance to the address manually. When a firewall policy is applied to this address, 34.222.246.178 is automatically covered.

Azure SDN connector using service principal

FortiOS automatically updates dynamic addresses for Azure using Azure SDN connector, including mapping attributes from Azure instances to dynamic address groups in FortiOS.

To configure the Azure SDN connector using service principal:

1. Create an Azure SDN connector:
   a. Go to *Security Fabric > External Connectors* and click *Create New*.
   b. Select *Microsoft Azure*.
   c. Configure the connector. See Azure SDN connector service principal configuration requirements:



   d. Click *OK*.
2. Create a dynamic firewall address for the Azure connector.
   a. Go to *Policy & Objects > Addresses* and click *Create New > Address*.
   b. From the *Type* dropdown list, select *Dynamic*.
   c. From the *Sub Type* dropdown list, select *Fabric Connector Address*.

d. From the *SDN Connector* dropdown list, select the Azure SDN connector.

e. In the *Filter* field, add filters as desired. The Azure SDN connector supports the following filters:
   - vm=<VM name>
   - securitygroup=<nsg id>
   - vnet=<VNet id>
   - subnet=<subnet id>
   - vmss=<VM scale set>
   - tag.<key>=<value>
   - servicetag=<value>
   - tag.<key>=<value>

f. Click *OK*.

g. Hover the cursor over the address name to see the dynamic IP addresses that the connector resolves.

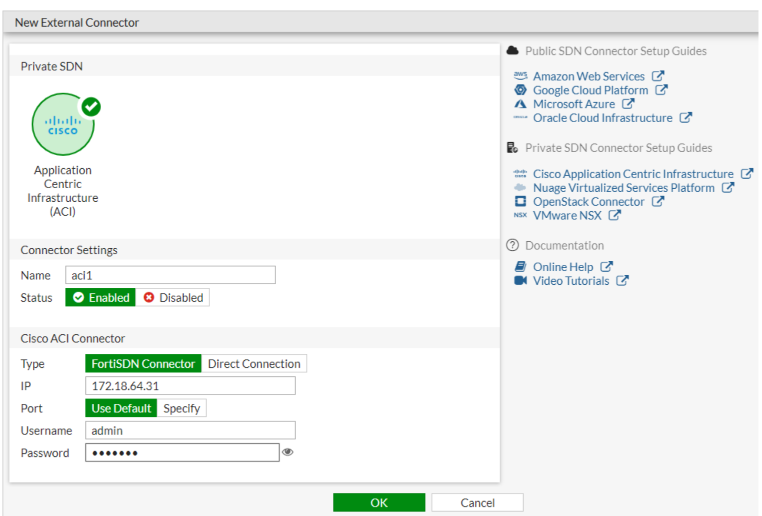
Cisco ACI SDN connector using a standalone connector

Cisco ACI (Application Centric Infrastructure) SDN connectors can be used in dynamic firewall addresses.

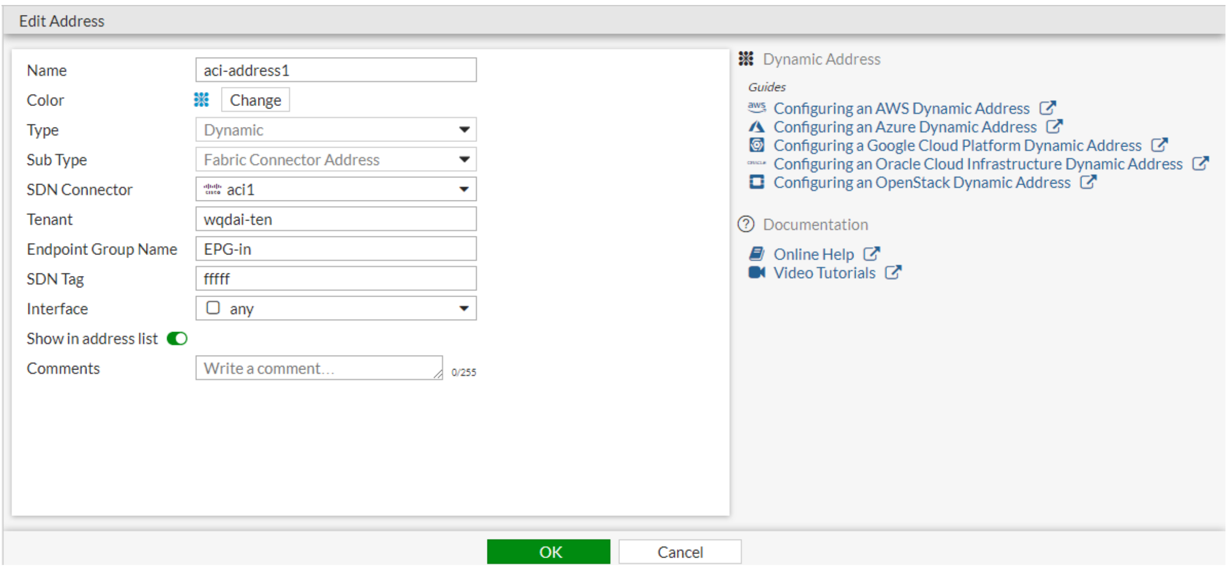
The Fortinet SDN Connector for Cisco ACI and Nuage Networks is a standalone connector that connects to SDN controllers within Cisco ACI and Nuage Networks. You must configure a connection to the Fortinet SDN connector in FortiOS to query the dynamic addresses.

To configure a Cisco ACI connector in the GUI:

1. Create the Cisco ACI SDN connector:
   a. Go to *Security Fabric > External Connectors* and click *Create New*.
   b. In the *Private SDN* section, click *Application Centric Infrastructure (ACI)*.
   c. In the *Cisco ACI Connector* section, for *Type*, select *Fortinet SDN Connector* and configure the remaining settings as needed.
   d. Click *OK*.



2. Create the dynamic firewall address for the connector:
   a. Go to *Policy & Objects > Addresses* and click *Create New > Address*.
   b. Configure the following settings:
      i. For *Type*, select *Dynamic*.
      ii. For *Sub Type*, select *Fabric Connector Address*.
      iii. For *SDN Connector*, select the first ACI connector.
      iv. Configure the remaining settings as needed.
   c. Click *OK*.



To verify the dynamic firewall IPs are resolved by the SDN connector in the GUI:

1. Go to *Policy & Objects > Addresses*.
2. In the address table, hover over the address to view which IPs it resolves to.

To configure a Cisco ACI connector in the CLI:

1. Create the SDN connector:

```
config system sdn-connector
    edit "aci1"
        set type aci
        set server "172.18.64.31"
        set username "admin"
        set password xxxxxxx
    next
end
```

2. Create the dynamic firewall address for the connector:

```
config firewall address
    edit "aci-address1"
        set type dynamic
        set sdn "aci1"
        set color 17
        set tenant "wqdai-ten"
```

```
            set epg-name "EPG-in"
            set sdn-tag "fffff"
        next
    end
```

To verify the dynamic firewall IPs are resolved by the SDN connector in the CLI:

```
# diagnose firewall dynamic list

List all dynamic addresses:
aci1.aci.wqdai-ten.EPG-in.fffff: ID(171)
        ADDR(192.168.100.20)
```

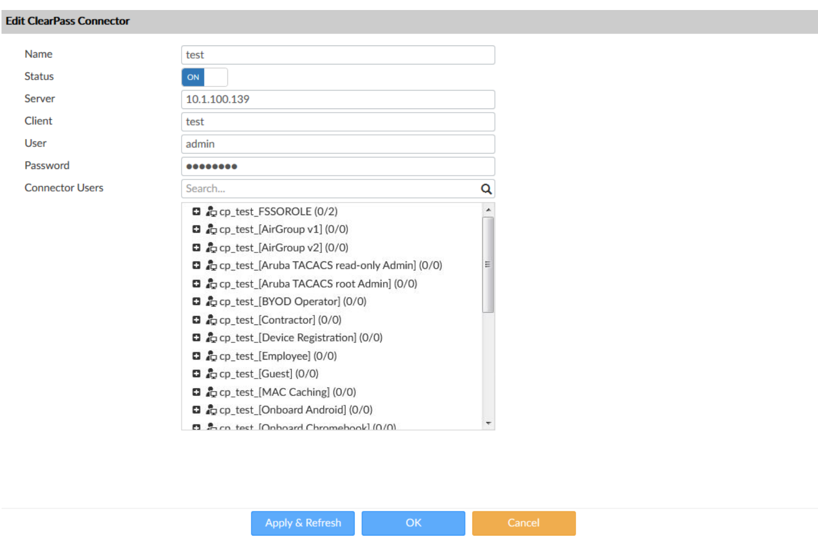
ClearPass endpoint connector via FortiManager

ClearPass Policy Manager (CPPM) is a network access system that can send information about authenticated users to third party systems, such as a FortiGate or FortiManager.

In this example, communications are established between CPPM and FortiManager, and then the FortiManager forwards information to a managed FortiGate. On the FortiGate, the user information can be used in firewall policies and added to FSSO dynamic addresses.

Configure the FortiManager

Establish communications between FortiManager and CPPM so that FortiManager can synchronize CPPM user groups. See Creating a ClearPass connector in the FortiManager Administration Guide.

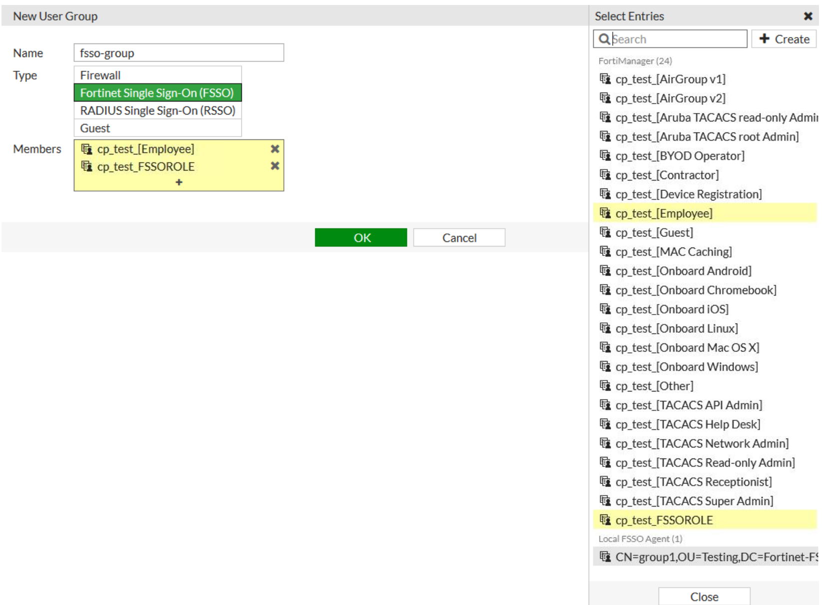


FortiManager forwards the group information to managed FortiGates.

Adding CPPM FSSO user groups to a local user group

To add CPPM user groups to a local user group in the GUI:

1. On the FortiGate, go to *User & Authentication > User Groups*.
2. Click *Create New*.
3. Enter a name for the group and set *Type* to *Fortinet Single Sign-On (FSSO)*.
4. Click the *Members* field, and add one or more FSSO groups.
   FSSO groups can come from multiple sources; CPPM FSSO groups are prefixed with *cp_* and are listed under the *FortiManager* heading.



5. Click *OK*.

To add CPPM user groups to a local user group in the CLI:

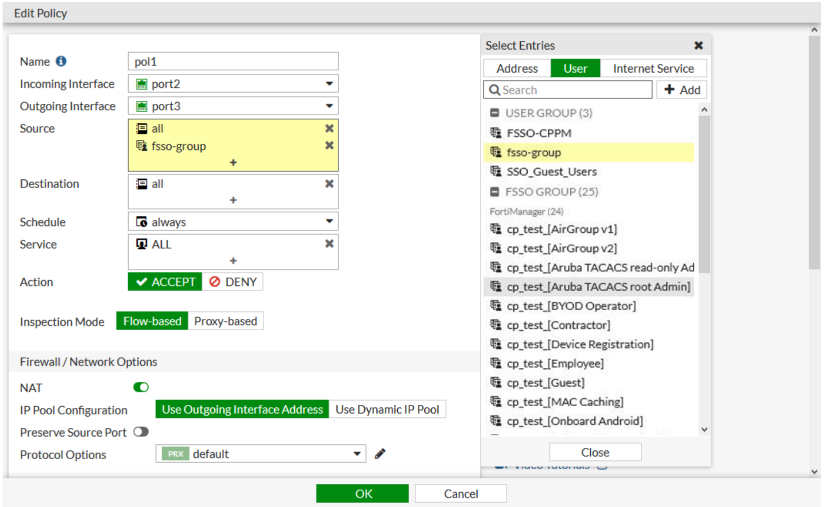
```
config user group
    edit fsso-group
        set group-type fsso-service
        set member "cp_test_[Employee]" "cp_test_FSSOROLE"
    next
end
```

Using the local FSSO user group in a firewall policy

To add the local FSSO user group to a firewall policy in the GUI:

1. Go to *Policy & Objects > Firewall Policy*.
2. Create a new policy, or edit an existing one.

3. Click in the *Source* field and add the *fsso-group* user group.



CPPM user groups can also be added directly to the policy.

4. Click *OK*.

To add the local FSSO user group to a firewall policy in the CLI:

```
config firewall policy
    edit 1
        set name "pol1"
        set srcintf "port2"
        set dstintf "port3"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set logtraffic all
        set groups "fsso-group"
        set nat enable
    next
end
```

Verification

To verify that a user was added to the FSSO list on the FortiGate:

1. Log on to the client and authenticate with CPPM.
   After successful authentication, the user is added to the FSSO list on the FortiGate.
2. On the FortiGate, go to *Dashboard > Users & Devices* and look at the *Firewall Users* widget to verify that the user was added.

The user group *cp_test_FSSOROLE* is listed separately because the user is a member of that group on the CPPM.

To verify that traffic can pass the firewall:

1. Log on to the client and browse to an external website.
2. On the FortiGate, go to *Dashboard > FortiView Sources*.
3. Double-click on the user and select the *Destinations* tab to verify that traffic is being passed by the firewall.

To verify the user address groups:

```
show user adgrp
    config user adgrp
        edit "cp_test_FSSOROLE"
            set server-name "FortiManager"
        next
        edit "cp_test_[AirGroup v1]"
            set server-name "FortiManager"
        next
        edit "cp_test_[AirGroup v2]"
            set server-name "FortiManager"
        next
        edit "cp_test_[Aruba TACACS read-only Admin]"
            set server-name "FortiManager"
        next
        edit "cp_test_[Aruba TACACS root Admin]"
            set server-name "FortiManager"
        next
        edit "cp_test_[BYOD Operator]"
            set server-name "FortiManager"
        next
        edit "cp_test_[Contractor]"
            set server-name "FortiManager"
        next
        edit "cp_test_[Device Registration]"
            set server-name "FortiManager"
        next
        ...
        edit "CN=group1,OU=Testing,DC=Fortinet-FSSO,DC=COM"
            set server-name "Local FSSO Agent"   <----- !!!
        next
    end
```

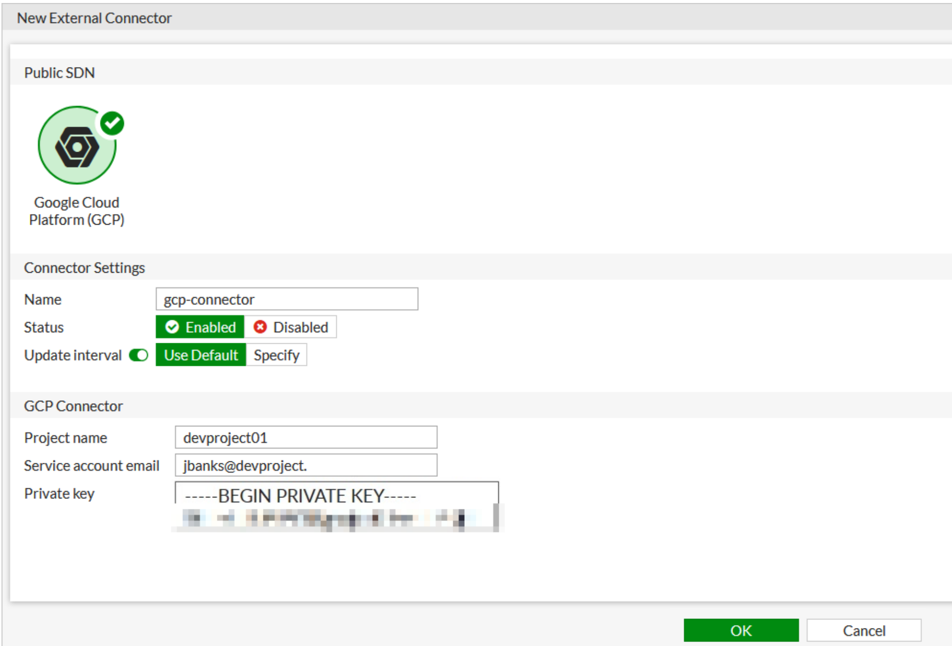
GCP SDN connector using service account

FortiOS automatically updates dynamic addresses for GCP using a GCP SDN connector, including mapping attributes from GCP instances to dynamic address groups in FortiOS.

To configure GCP connector using the GUI:

1. In FortiOS, go to *Security Fabric > External Connectors*.
2. Click *Create New*, and select *Google Cloud Platform (GCP)*.
   Note you can create only one SDN Connector per connector type. For example, you can create one entry for GCP.

3. Configure the connector as follows:
   a. *Project name*: Enter the name of the GCP project. The VMs whose IP addresses you want to populate should be running within this project.
   b. *Service account email*: Enter the email address associated with the service account that will call APIs to the GCP project specified above.
   c. *Private key*: Enter the private key statement as shown in the text box. For details, see Creating a GCP service account.



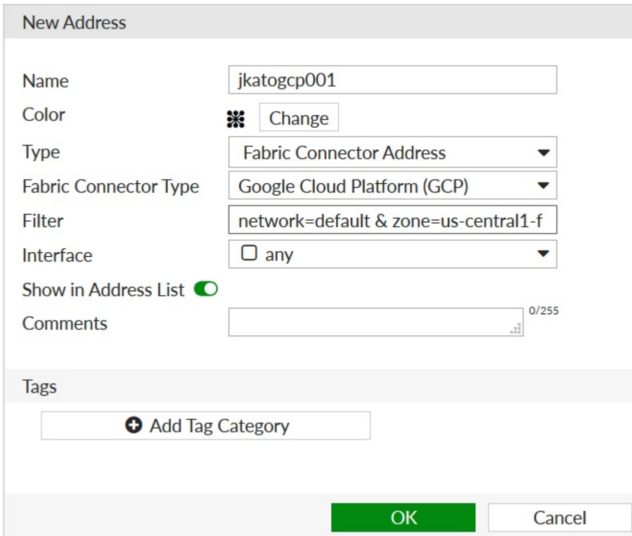
   Once the connector is successfully configured, a green indicator appears at the bottom right corner. If the indicator is red, the connector is not working. See Troubleshooting GCP SDN Connector.

4. Create a dynamic firewall address for the configured GCP SDN connector:
   a. Go to *Policy & Objects* > *Addresses*. Click *Create New*, then select *Address*.
   b. Configure the address:
      i. *Name*: Enter the desired name.
      ii. *Type*: Select *Dynamic*.
      iii. *Fabric Connector Type*: Select *Google Cloud Platform (GCP)*.
      iv. *Filter*: The SDN connector automatically populates and updates only instances that match this filtering condition. Currently GCP supports the following filters:
         - `id=<instance id>` : This matches an VM instance ID.
         - `name=<instance name>` : This matches a VM instance name.
         - `zone=<gcp zones>` : This matches a zone name.
         - `network=<gcp network name>` : This matches a network name.
         - `subnet=<gcp subnet name>` : This matches a subnet name.
         - `tag=<gcp network tags>` : This matches a network tag.
         - `label.<gcp label key>=<gcp label value>` : This matches a free form GCP label key and its value.

         In the example, the filter is set as `'network=default & zone=us-central-1f'`. This configuration populates all IP addresses that belong to the default network in the zone us-central-1f.

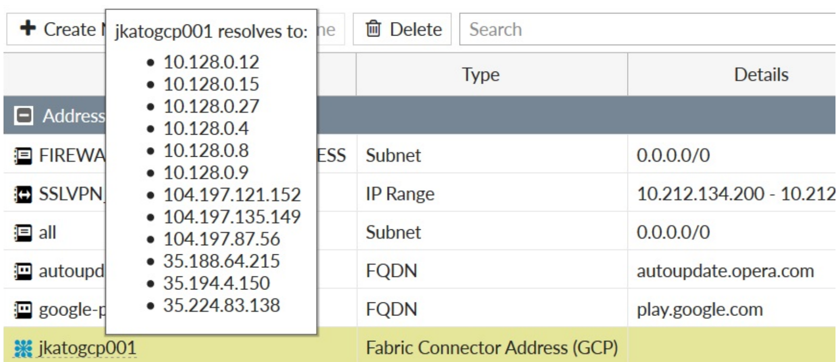
         You can set filtering conditions using multiple entries with AND ("&") or OR ("|"). When both AND and OR are specified, AND is interpreted first, then OR.

Note that wildcards (such as the asterisk) are not allowed in filter values.



v. Click *OK*.

The address has been created. Wait for a few minutes before the setting takes effect. You will know that the address is in effect when the exclamation mark disappears from the address entry. When you hover over the address, you can see the list of populated IP addresses.



If the exclamation mark does not disappear, check the address settings.

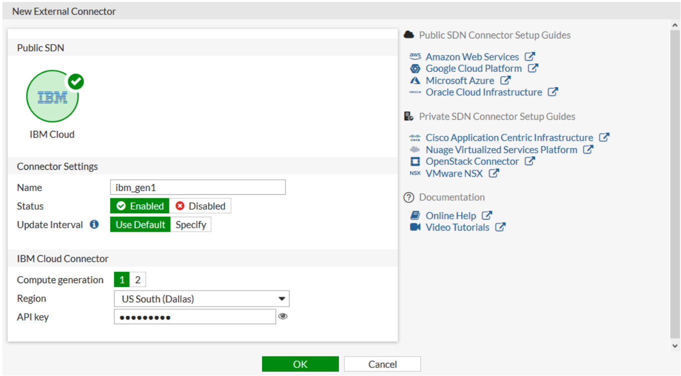
IBM Cloud SDN connector using API keys

FortiOS can automatically update dynamic addresses for IBM Cloud using an SDN connector.

To configure IBM Cloud SDN connectors using the GUI:

1. Create SDN connectors for compute generation 1 and 2:
   a. Go to *Security Fabric > External Connectors*.
   b. Click *Create New*, then select *IBM Cloud*.

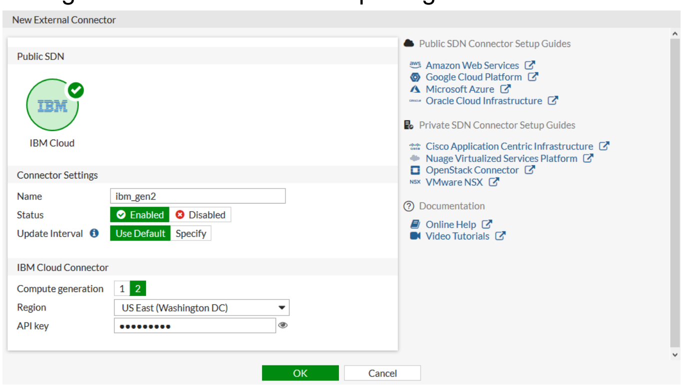
c. Configure the connector for computer generation 1:



d. Click *OK*.

e. Click *Create New*, then select *IBM Cloud*.

f. Configure the connector for computer generation 2:



g. Click *OK*.

2. Create dynamic firewall addresses for the configured connectors:

a. Go to *Policy & Objects* > *Addresses*.

b. Click *Create New* > *Address*.

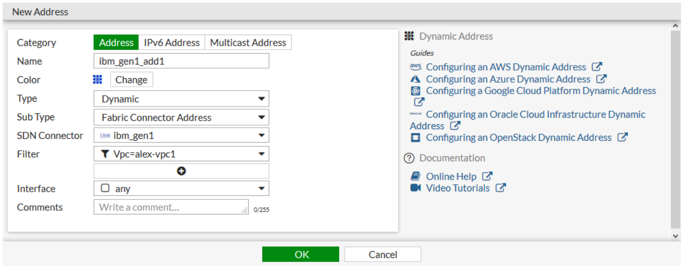
c. From the *Type* dropdown list, select *Dynamic*.

d. From the *Sub Type* dropdown list, select *Fabric Connector Address*.

e. From the *SDN Connector* dropdown list, select the IBM SDN connector.

f. In the *Filter* field, add the desired filters. The following filters are supported:

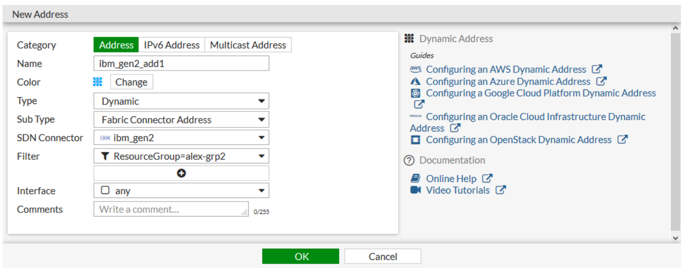
- <InstanceId>
- <InstanceName>
- <ImageId>
- <ImageName>
- <Architecture>
- <Profile>
- <Vpc>
- <Zone>
- <Subnet>
- <ResourceGroup>



g. Click *OK*.

h. Click *Create New* > *Address*.

i. Repeat the process for computer generation 2:

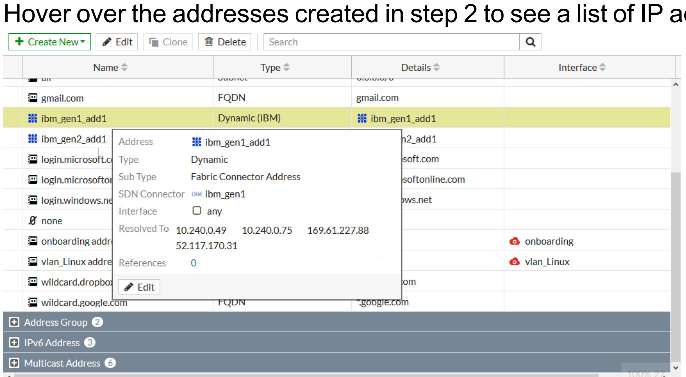


j. Click *OK*.

3. Ensure that the connectors resolve dynamic firewall IP addresses:

a. Go to *Policy & Objects* > *Addresses*.

b. Hover over the addresses created in step 2 to see a list of IP addresses that the connector has resolved:



To configure IBM Cloud SDN connectors using the CLI:

1. Create SDN connectors for compute generation 1 and 2:

```
config system sdn-connector
    edit "ibm_gen1"
        set status enable
        set type ibm
        set api-key xxxxxx
        set compute-generation 1
        set ibm-region us-south
        set update-interval 60
    next
    edit "ibm_gen2"
        set status enable
        set type ibm
```

```
        set api-key xxxxxx
        set compute-generation 2
        set ibm-region us-east
        set update-interval 60
    next
end
```

2. Create dynamic firewall addresses for the configured connectors:

```
config firewall address
    edit "ibm_gen1_add1"
        set type dynamic
        set sdn "ibm_gen1"
        set color 19
        set filter "Vpc=alex-vpc1"
    next
    edit "ibm_gen2_add1"
        set type dynamic
        set sdn "ibm_gen2"
        set color 19
        set filter "ResourceGroup=alex-grp2"
    next
end
```

3. Ensure that the connectors resolve dynamic firewall IP addresses:

```
# show firewall address ibm_gen1_add1
config firewall address
    edit "ibm_gen1_add1"
        set uuid 586841c4-7f46-51ea-dc66-dbf840af03d3
        set type dynamic
        set sdn "ibm_gen1"
        set color 19
        set filter "Vpc=alex-vpc1"
        config list
            edit "10.240.0.49"
            next
            edit "10.240.0.75"
            next
            edit "169.61.227.88"
            next
            edit "52.117.170.31"
            next
        end
    next
end
```

```
# show firewall address ibm_gen2_add1
config firewall address
    edit "ibm_gen2_add1"
        set uuid 5868c4f0-7f46-51ea-2b79-b5170fbfd4a8
        set type dynamic
        set sdn "ibm_gen2"
        set color 19
        set filter "ResourceGroup=alex-grp2"
        config list
            edit "10.241.128.4"
            next
```

```
                edit "10.241.128.5"
                next
                edit "10.241.129.4"
                next
                edit "52.117.126.69"
                next
            end
        next
    end
```

Kubernetes (K8s) SDN connectors

The following topics provide information about configuring Kubernetes SDN connectors:

AliCloud Kubernetes SDN connector using access key

When an AliCloud SDN connector is configured, dynamic address objects can support Kubernetes filters based on cluster, service, node, pod, and more.

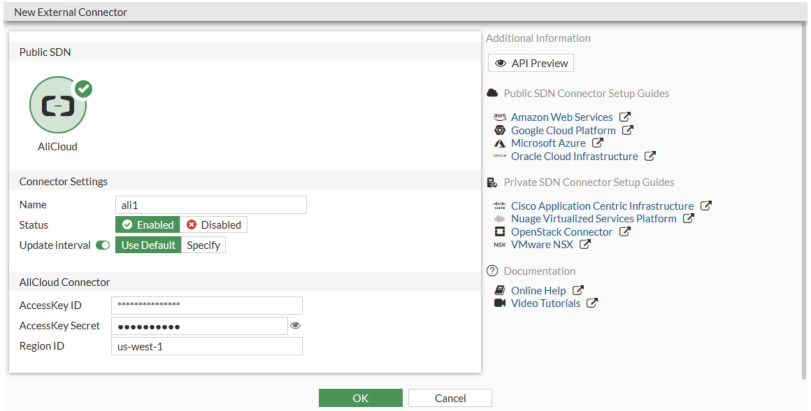
The following address filters can be applied:

- K8S_Cluster
- K8S_Namespace
- K8S_ServiceName
- K8S_NodeName
- K8S_PodName
- K8S_Region
- K8S_Zone
- K8S_Label

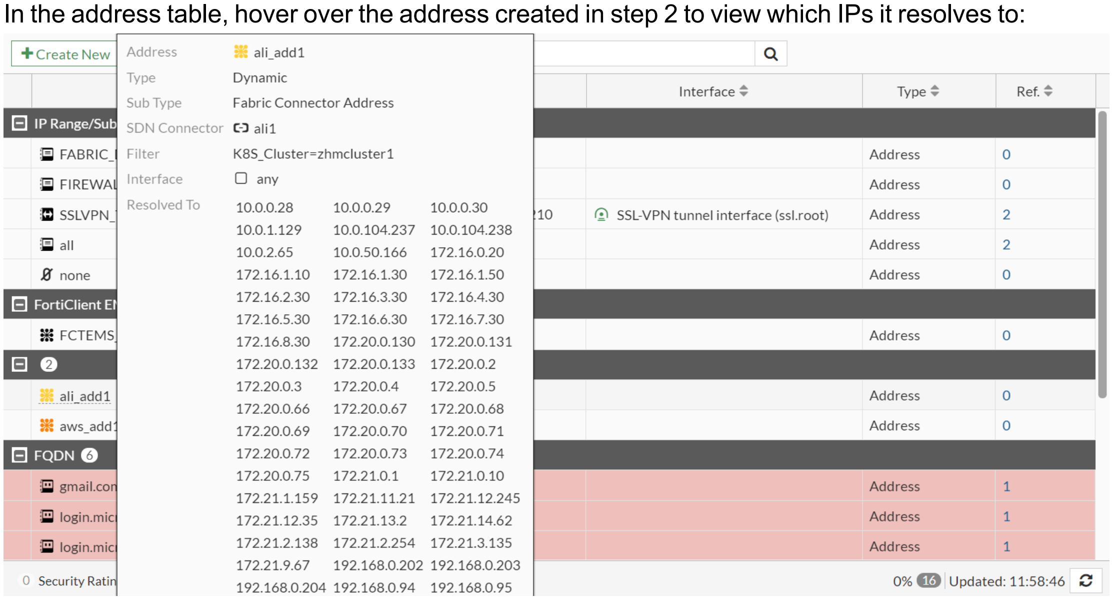
To configure an AliCloud SDN connector with a Kubernetes filter in the GUI:

1. Configure the AliCloud SDN connector:
   a. Go to *Security Fabric > External Connectors*.
   b. Click *Create New*, and select *AliCloud*.

c. Configure the settings as needed and click *OK*.



2. Create a dynamic firewall address with the supported Kubernetes filter:
   a. Go to *Policy & Objects > Addresses*.
   b. Click *Create New > Address* and enter a name.
   c. Configure the following settings:
      i. For *Type*, select *Dynamic*.
      ii. For *Sub Type*, select *Fabric Connector Address*.
      iii. For *SDN Connector*, select the connector created in step 1.
      iv. For *SDN address type*, select *Private*.
      v. For *Filter*, select *K8S_Cluster=zhmcluster*.
   d. Click *OK*.
      The corresponding IP addresses are dynamically updated and resolved after applying the Kubernetes filter.
3. Confirm that the AliCloud SDN connector resolves dynamic firewall IP addresses using the configured filter:
   a. Go to *Policy & Objects > Addresses*.
   b. In the address table, hover over the address created in step 2 to view which IPs it resolves to:



To configure an AliCloud SDN connector with a Kubernetes filter in the CLI:

1. Configure the AliCloud SDN connector:

```
config system sdn-connector
    edit "ali1"
        set type alicloud
        set access-key "****************"
        set secret-key xxxxxxxx
        set region "us-west-1"
    next
end
```

2. Create a dynamic firewall address with the supported Kubernetes filter:

```
config firewall address
    edit "ali_add1"
        set type dynamic
        set sdn "ali1"
        set color 10
        set filter "K8S_Cluster=zhmcluster1"
    next
end
```

3. Confirm that the AliCloud SDN connector resolves dynamic firewall IP addresses using the configured filter:

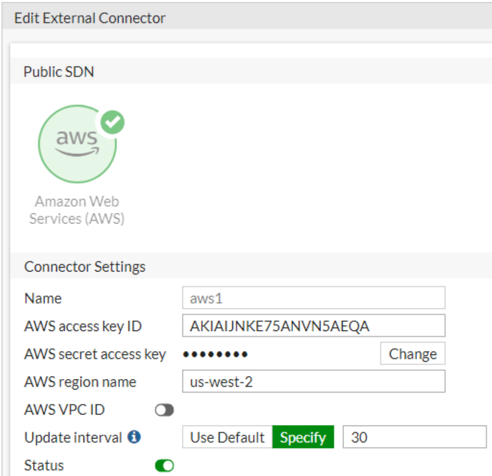
```
config firewall address
    edit "ali_add1"
        show
            config firewall address
                edit "ali_add1"
                    set uuid c48e4f00-5435-51eb-0547-aced5cf80f1f
                    set type dynamic
                    set sdn "ali1"
                    set color 10
                    set filter "K8S_Cluster=zhmcluster1"
                    config list
                        edit "10.0.0.28"
                        next
                        edit "10.0.0.29"
                        next
                        edit "10.0.0.30"
                        next
                        ...
                    end
                next
            end
    next
end
```

AWS Kubernetes (EKS) SDN connector using access key

AWS SDN connectors support dynamic address groups based on AWS Kubernetes (EKS) filters.

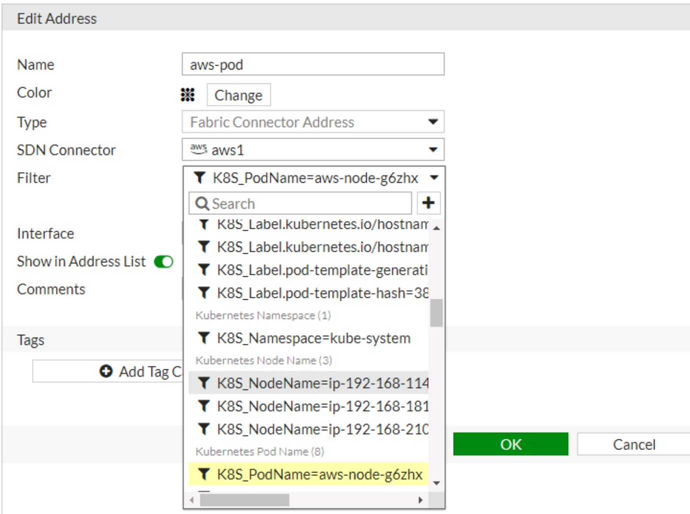
To enable an AWS SDN connector to fetch IP addresses from AWS Kubernetes:

1. Go to *Security Fabric > External Connectors*. Click *Create New*, then select *Amazon Web Services (AWS)*. Configure the SDN connector as desired. See AWS SDN connector using certificates on page 2433

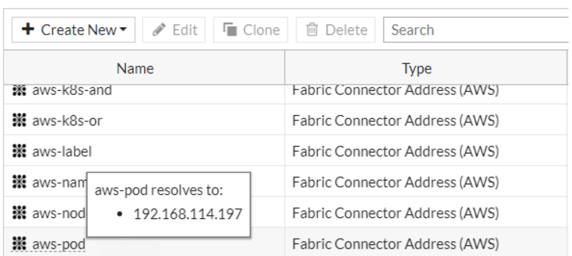


2. Go to *Policies & Objects > Addresses*. Click *Create New > Address* to create a dynamic firewall address for the configured SDN connector using the supported Kubernetes filter.
3. From the *Type* dropdown list, select *Dynamic*.
4. From the *Sub Type* dropdown list, select *Fabric Connector Address*.
5. From the *SDN Connector* dropdown list, select the desired SDN connector.
6. In the *Filter* field, add the desired filters. The following filters are supported:

Filter	Description
k8s_cluster	Name of Kubernetes cluster.
k8s_namespace	Namespace of a Kubernetes service or pod.
k8s_svcname	Name of a Kubernetes service.
k8s_nodename	Name of a Kubernetes node.
k8s_zone	Zone of a Kubernetes node.
k8s_region	Region of a Kubernetes node.
k8s_podname	Name of a Kubernetes pod.
k8s_label.xxx	Name of label of a Kubernetes resource (cluster/service/node/pod).



7. Configure the rest of the settings, then click *OK*.
8. Ensure that the SDN connector resolves the dynamic firewall address IP addresses by going to *Policy & Objects > Addresses* and hovering over the newly created address.



To configure an AWS Kubernetes SDN connector through the CLI:

1. Configure the SDN connector:

```
config system sdn-connector
   edit "aws1"
      set type aws
      set access-key "AKIAIJNKE75ANVN5AEQA"
      set secret-key xxxxx
      set region "us-west-2"
      set update-interval 30
   next
end
```

2. Create a dynamic firewall address for the SDN connector with a supported Kubernetes filter:

```
config firewall address
   edit "aws-pod"
      set type dynamic
      set sdn "aws1"
      set filter "K8S_PodName=aws-node-g6zhx"
   next
end
```

The SDN connector resolves the dynamic firewall address IP address:

```
config firewall address
   edit "aws-pod"
      set type dynamic
      set sdn "aws1"
```

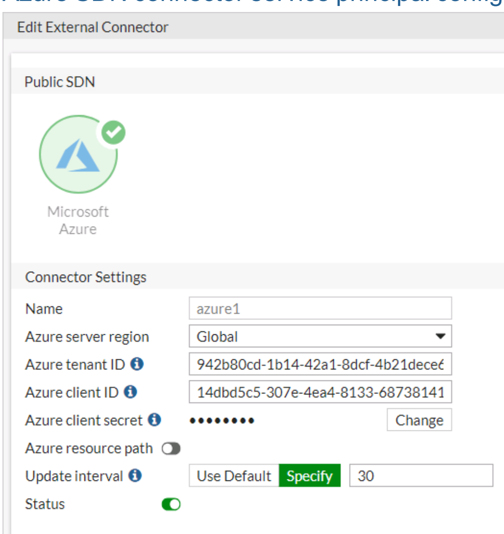
```
            set filter "K8S_PodName=aws-node-g6zhx"
            config list
                edit "192.168.114.197"
                next
            end
        next
    end
```

Azure Kubernetes (AKS) SDN connector using client secret

Azure SDN connectors support dynamic address groups based on Azure Kubernetes (AKS) filters.

To enable an Azure SDN connector to fetch IP addresses from Azure Kubernetes:

1. Configure the Azure SDN connector:
   a. Go to *Security Fabric > External Connectors*.
   b. Click *Create New*, and select *Azure*.
   c. Configure as shown substituting the region, tenant and client IDs, and client secret for your deployment. See Azure SDN connector service principal configuration requirements.

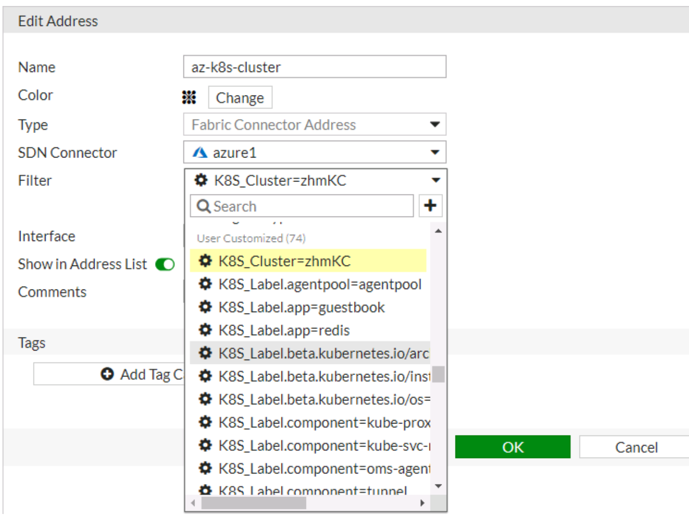


2. Create a dynamic firewall address for the configured K8s SDN connector:
   a. Go to *Policy & Objects > Addresses*.
   b. Click *Create New*, then select *Address*.
   c. From the *Type* dropdown list, select *Dynamic*.
   d. From the *Sub Type* dropdown list, select *Fabric Connector Address*.
   e. From the *SDN Connector* dropdown list, select the desired SDN connector.
   f. In the *Filter* field, add the desired filter. The following filters are supported:

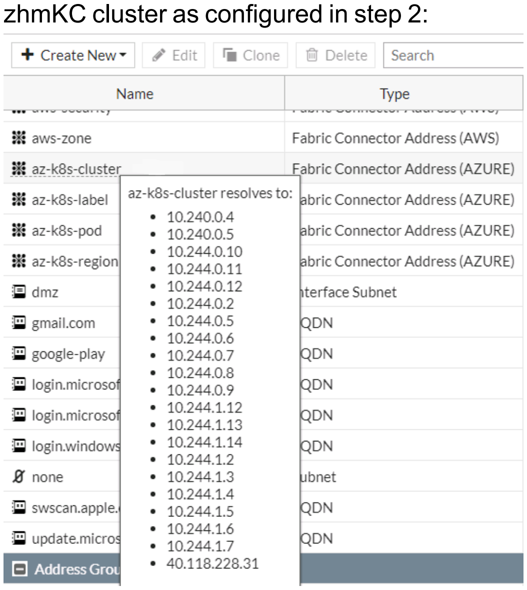
Filter	Description
k8s_cluster	Name of Kubernetes cluster.
k8s_namespace	Namespace of a Kubernetes service or pod.

Filter	Description
k8s_svcname	Name of a Kubernetes service.
k8s_nodename	Name of a Kubernetes node.
k8s_zone	Zone of a Kubernetes node.
k8s_region	Region of a Kubernetes node.
k8s_podname	Name of a Kubernetes pod.
k8s_label.xxx	Name of label of a Kubernetes resource (cluster/service/node/pod).

In this example, the address is configured to automatically populate and update IP addresses only for instances that belong to the zhmKC cluster:



3. Ensure that the K8s SDN connector resolves dynamic firewall IP addresses:
   a. Go to *Policy & Objects > Addresses*.
   b. Hover over the address created in step 2 to see a list of IP addresses for instances that belong to the zhmKC cluster as configured in step 2:



To configure an Azure Kubernetes SDN connector through the CLI:

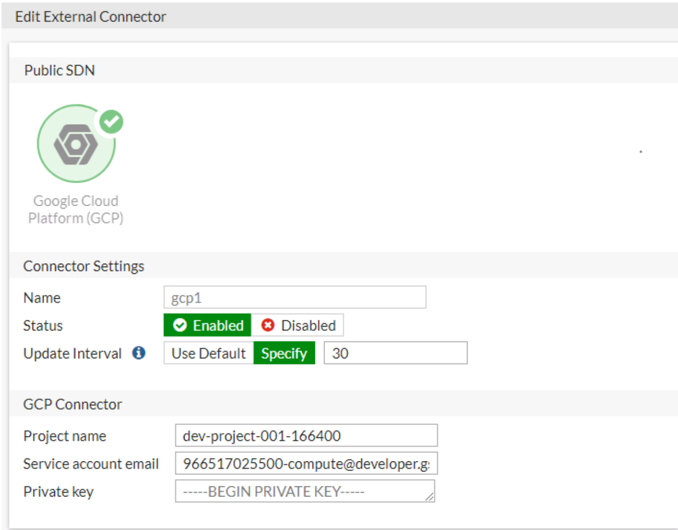
1. Configure the SDN connector:
```
config system sdn-connector
    edit "azure1"
        set type azure
        set tenant-id "942b80cd-1b14-42a1-8dcf-4b21dece61ba"
        set client-id "14dbd5c5-307e-4ea4-8133-68738141feb1"
        set client-secret xxxxx
        set update-interval 30
    next
end
```
2. Create a dynamic firewall address for the SDN connector with a supported Kubernetes filter. In this example, the address will automatically populate and update IP addresses only for instances that belong to the zhmKC cluster:
```
config firewall address
    edit "az-k8s-cluster"
        set type dynamic
        set sdn "azure1"
        set filter "K8S_Cluster=zhmKC"
    next
end
```
3. Confirm that the Azure SDN connector resolves dynamic firewall IP addresses using the configured filter:
```
config firewall address
    edit "az-k8s-cluster"
        set type dynamic
        set sdn "azure1"
        set filter "K8S_Cluster=zhmKC"
        config list
            edit "10.240.0.4"
            next
            edit "10.240.0.5"
            next
            edit "10.244.0.10"
            next
        end
    next
end
```

GCP Kubernetes (GKE) SDN connector using service account

Google Cloud Platform (GCP) SDN connectors support dynamic address groups based on GCP Kubernetes Engine (GKE) filters.

To enable a GCP SDN connector to fetch IP addresses from GKE:

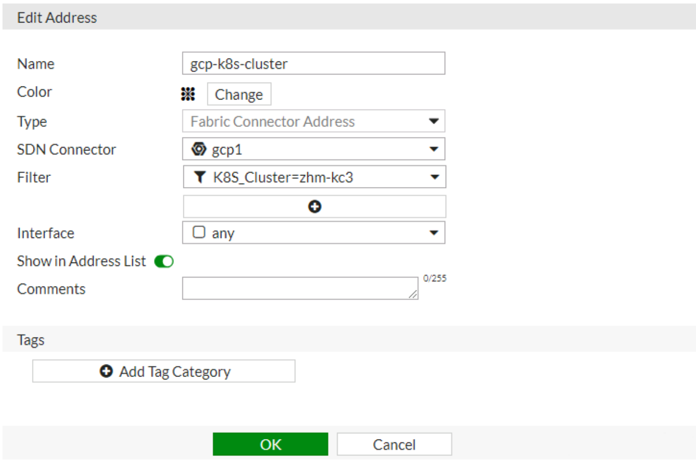
1. Go to *Security Fabric* > *External Connectors*, and configure an SDN connector for GCP.



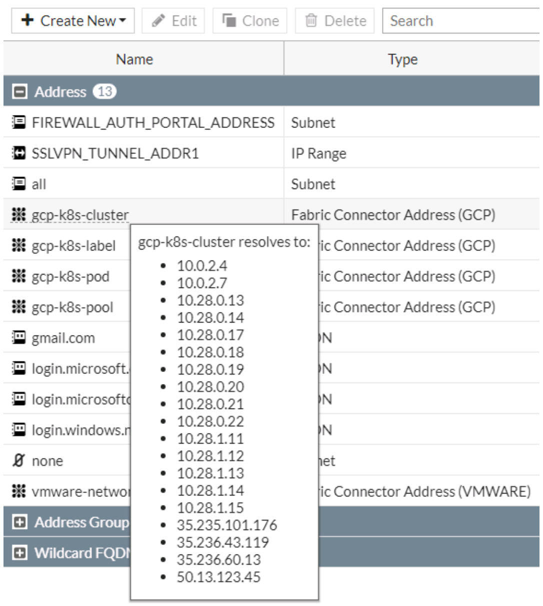
2. Go to *Policies & Objects* > *Addresses* and create a dynamic firewall address for the configured SDN connector using the supported Kubernetes filter.
3. To filter out the Kubernetes IP addresses, select the address filter or filters. The following filters are supported:

Filter	Description
k8s_cluster	Name of Kubernetes cluster.
k8s_namespace	Namespace of a Kubernetes service or pod.
k8s_svcname	Name of a Kubernetes service.
k8s_nodename	Name of a Kubernetes node.
k8s_zone	Zone of a Kubernetes node.
k8s_region	Region of a Kubernetes node.
k8s_podname	Name of a Kubernetes pod.
k8s_label.xxx	Name of label of a Kubernetes resource (cluster/service/node/Pod).

In this example, the GCP SDN connector will automatically populate and update IP addresses only for instances that belong to the zhm-kc3 cluster:



4. Configure the rest of the settings, then click *OK*.
   The dynamic firewall address IP is resolved by the SDN connector.



To configure a GCP Kubernetes SDN connector through the CLI:

1. Configure an SDN connector for Kubernetes:

```
config system sdn-connector
   edit "gcp1"
      set type gcp
      set gcp-project "dev-project-001-166400"
      set service-account "966517025500-compute@developer.gserviceaccount.com"
      set update-interval 30
   next
end
```

2. Create a dynamic firewall address for the SDN connector with a supported Kubernetes filter:

```
config firewall address
   edit "gcp-k8s-cluster"
      set type dynamic
      set sdn "gcp1"
```

```
        set filter "K8S_Cluster=zhm-kc3"
    next
end
```

The dynamic firewall address IP is resolved by the SDN connector:

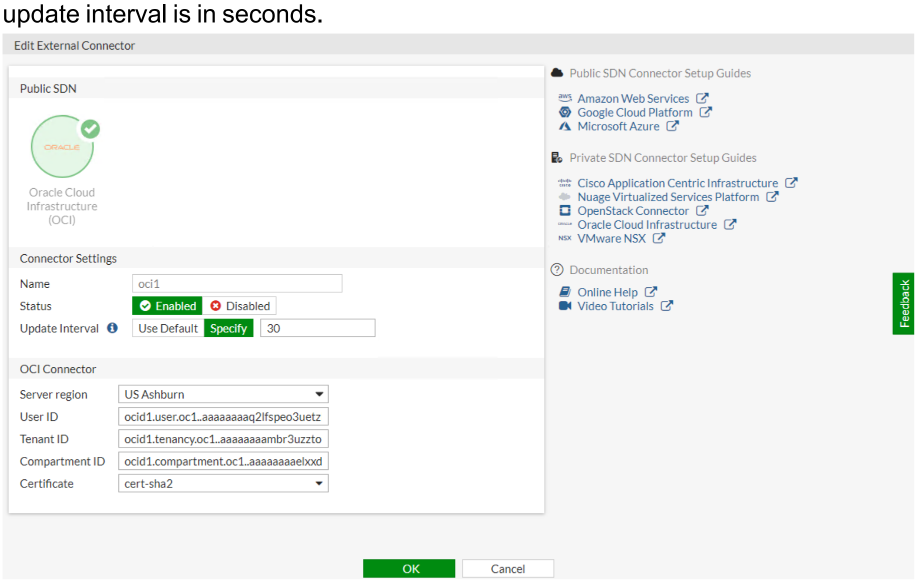
```
config firewall address
    edit "gcp-k8s-cluster"
        set type dynamic
        set sdn "gcp1"
        set filter "K8S_Cluster=zhm-kc3"
        config list
            edit "10.0.2.4"
            next
            edit "10.0.2.7"
            next
            edit "10.28.0.13"
            next
        end
    next
end
```

Oracle Kubernetes (OKE) SDN connector using certificates

OCI SDN connectors support dynamic address groups based on Oracle Kubernetes (OKE) filters.

To enable an OCI SDN connector to fetch IP addresses from Oracle Kubernetes:

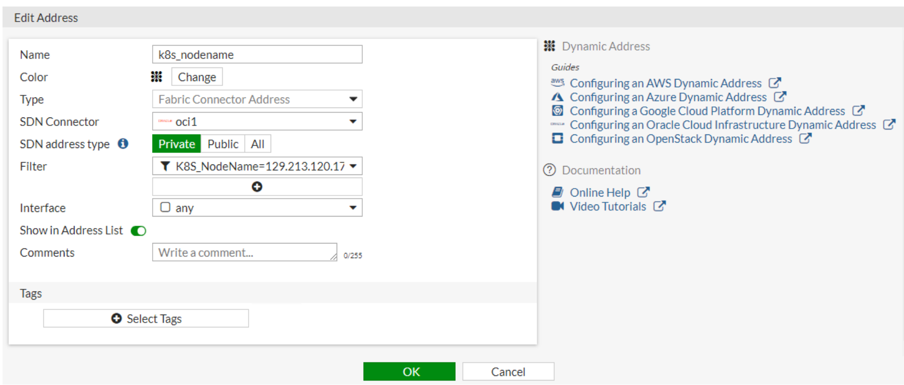
1. Configure the OCI SDN connector:
   a. Go to *Security Fabric > External Connectors*.
   b. Click *Create New*, and select *Oracle Cloud Infrastructure (OCI)*.
   c. Configure as shown substituting the region, tenant and client IDs, and client secret for your deployment. The update interval is in seconds.



2. Create dynamic firewall addresses for the configured SDN connector with supported Kubernetes filter:
   a. Go to *Policy & Objects > Addresses*.
   b. Click *Create New*, then select *Address*.

c. In the *Filter* field, select the desired filters. The following filters are supported:

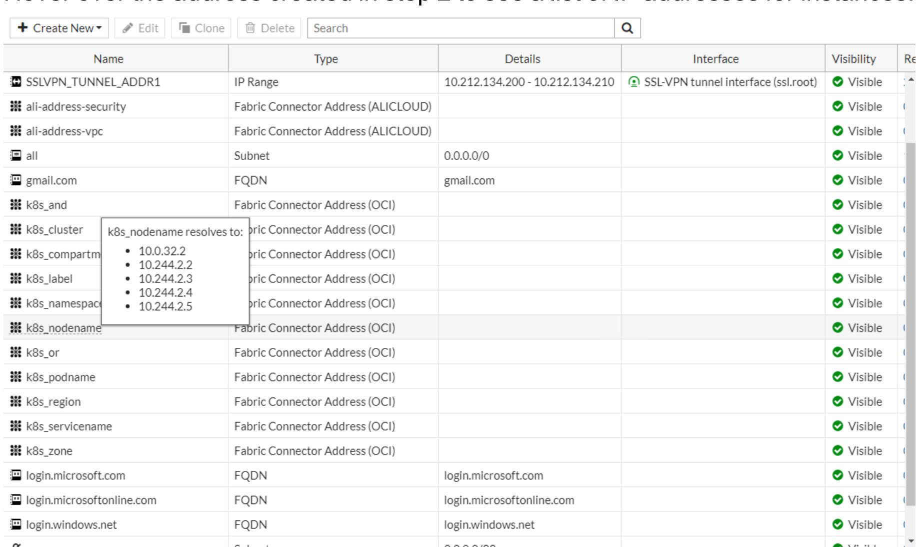
Filter	Description
k8s_cluster	Name of Kubernetes cluster.
k8s_namespace	Namespace of a Kubernetes service or pod.
k8s_svcname	Name of a Kubernetes service.
k8s_nodename	Name of a Kubernetes node.
k8s_zone	Zone of a Kubernetes node.
k8s_region	Region of a Kubernetes node.
k8s_podname	Name of a Kubernetes pod.
k8s_label.xxx	Name of label of a Kubernetes resource (cluster/service/node/Pod).



3. Confirm that the SDN connector resolves dynamic firewall IP addresses:

a. Go to *Policy & Objects > Addresses*.

b. Hover over the address created in step 2 to see a list of IP addresses for instances:



To configure an SDN connector through the CLI:

1. Configure the OCI SDN connector:
```
config system sdn-connector
   edit "oci1"
      set type oci
      set tenant-id
           "ocid1.tenancy.oc1..aaaaaaaambr3uzztoyhweohbzqqdo775h7d3t54zpmzkp4b2cf35vs5
           5cxxx"
      set user-id
           "ocid1.user.oc1..aaaaaaaaq2lfspeo3uetzbzpiv2pqvzzevozccnys347stwssvizqlatfx
           xx"
      set compartment-id
           "ocid1.compartment.oc1..aaaaaaaaelxxdjazqo7nzczpgypyiqcgkmytjry6nfq5345vw7e
           avpwnmxxx"
      set oci-region ashburn
      set oci-cert "cert-sha2"
      set update-interval 30
   next
end
```
2. Create dynamic firewall addresses for the configured SDN connector with supported Kubernetes filter:
```
config firewall address
   edit "k8s_nodename"
      set type dynamic
      set sdn "oci1"
      set filter "K8S_NodeName=129.213.120.172"
   next
end
```
3. Confirm that the SDN connector resolves dynamic firewall IP addresses:
```
config firewall address
   edit "k8s_nodename"
      set type dynamic
      set sdn "oci1"
      set filter "K8S_NodeName=129.213.120.172"
      config list
         edit "10.0.32.2"
         next
         edit "10.244.2.2"
         next
         edit "10.244.2.3"
         next
         edit "10.244.2.4"
         next
         edit "10.244.2.5"
         next
      end
   next
end
```

Private cloud K8s SDN connector using secret token

FortiOS automatically updates dynamic and cluster IP addresses for Kubernetes (K8s) by using a K8s SDN connector, enabling FortiOS to manage K8s pods as global address objects, as with other connectors. This includes mapping the following attributes from K8s instances to dynamic address groups in FortiOS:

Filter	Description
Namespace	Filter service IP addresses in a given namespace.
ServiceName	Filter service IP addresses by the given service name.
NodeName	Filter node IP addresses by the given node name.
PodName	Filter IP addresses by the pod name.
Label.XXX	Filter service or node IP addresses with the given label XXX. For example: `K8S_Label.app=nginx`.

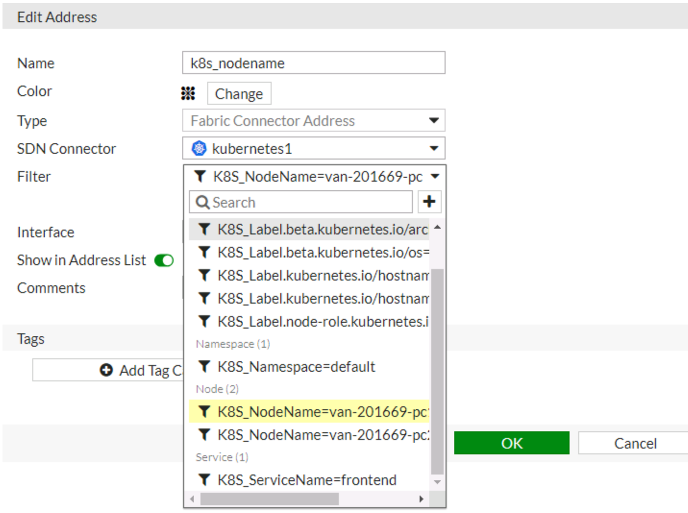
FortiOS 6.2.3 and later collect cluster IP addresses in addition to external IP addresses for exposed K8s services.

There is no maximum limit for the number of IP addresses populated with the filters.

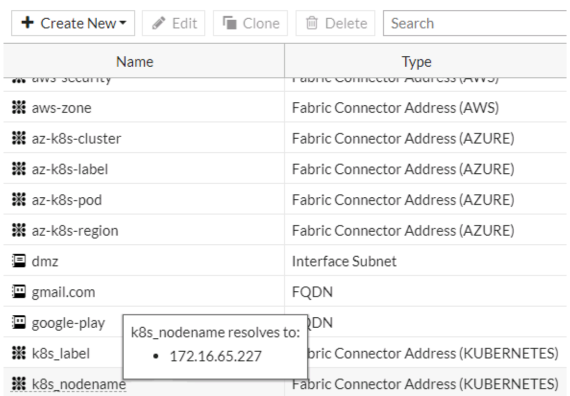
To configure K8s SDN connector using the GUI:

1. Configure the K8s SDN connector:
   a. Go to *Security Fabric > External Connectors > Create New Connector*.
   b. Select *Kubernetes*.
   c. In the *IP* field, enter the IP address that you obtained in Obtaining the IP address, port, and secret token in Kubernetes.
   d. In the *Port* field, select *Specify*, then enter the port that you obtained in Obtaining the IP address, port, and secret token in Kubernetes.
   e. In the *Secret token* field, enter the token that you obtained in Obtaining the IP address, port, and secret token in Kubernetes.
   f. Configure the other fields as desired.
2. Create a dynamic firewall address for the configured K8S SDN connector:
   a. Go to *Policy & Objects > Addresses*.
   b. Click *Create New*, then select *Address*.
   c. Configure the address as shown, selecting the desired filter in the *Filter* dropdown list. In this example, the K8s SDN connector will automatically populate and update IP addresses only for node instances that match

the specified node name:



3. Ensure that the K8s SDN connector resolves dynamic firewall IP addresses:
   a. Go to *Policy & Objects > Addresses*.
   b. Hover over the address created in step 2 to see a list of IP addresses for node instances that match the node name configured in step 2:



To configure K8s SDN connector using CLI commands:

1. Configure the K8s SDN connector:

```
config system sdn-connector
   edit "kubernetes1"
      set type kubernetes
      set server "<IP address obtained in Obtaining the IP address, port, and secret
           token in Kubernetes>"
      set server-port <Port obtained in Obtaining the IP address, port, and secret token
           in Kubernetes>
      set secret-token <Secret token obtained in Obtaining the IP address, port, and
           secret token in Kubernetes>
      set update-interval 30
   next
end
```

2. Create a dynamic firewall address for the configured K8s SDN connector with the supported K8s filter. In this example, the K8s SDN connector will automatically populate and update IP addresses only for node instances that match the specified node name:

```
config firewall address
   edit "k8s_nodename"
      set type dynamic
      set sdn "kubernetes1"
      set filter "K8S_NodeName=van-201669-pc1"
   next
end
```

3. Confirm that the K8s SDN connector resolves dynamic firewall IP addresses using the configured filter:

```
config firewall address
   edit "k8s_nodename"
      set type dynamic
      set sdn "kubernetes1"
      set filter "K8S_NodeName=van-201669-pc1"
      config list
         edit "172.16.65.227"
         next
      end
   next
end
```

To troubleshoot the connection:

1. In FortiOS, run the following commands:

```
diagnose deb application kubed -1
diagnose debug enable
```

2. Reset the connection on the web UI to generate logs and troubleshoot the issue. The following shows the output in the case of a failure:

```
fortigate # diagnose deb application kubed -1
Debug messages will be on for 30 minutes.

fortigate # diagnose debug enable

fortigate # k8s: update sdn connector kubernetes1 status to enabled
k8s: update sdn connector kubernetes2 status to disabled
kubed sdn connector kubernetes1 prepare to update
getting token
kubed sdn connector kubernetes1 start updating
kube url: https://172.17.215.10:6443/api/v1/services
kube host: 172.17.215.10:6443:172.17.215.10
{"kind":"Status","apiVersion":"v1","metadata":{},"status":"Failure","message":"s
ervices is forbidden: User \"system:serviceaccount:default:fortigateconnector\"
cannot list resource \"services\" in API group \"\" at the cluster scope","reaso
n":"Forbidden","details":{"kind":"services"},"code":403}

kubed failed to list kubernetes services.
kubed failed to get IPs from kubedrnetes services.
kubed failed to get ip addr list
kubed reap child pid: 1226
```

The following shows the output in the case of a success:

```
 kube-system
k8s pod ip: 10.180.1.2, podname: metrics-server-v0.3.6-64655c969-djt8s, namespac
e: kube-system
k8s pod ip: 10.138.0.6, podname: netd-4qvvn, namespace: kube-system
k8s pod ip: 10.138.0.5, podname: netd-756ch, namespace: kube-system
k8s pod ip: 10.138.0.4, podname: netd-hr75d, namespace: kube-system
k8s pod ip: 10.138.0.6, podname: prometheus-to-sd-59trp, namespace: kube-system
k8s pod ip: 10.138.0.4, podname: prometheus-to-sd-q6qv5, namespace: kube-system
k8s pod ip: 10.138.0.5, podname: prometheus-to-sd-rq2zm, namespace: kube-system
k8s pod ip: 10.180.1.3, podname: stackdriver-metadata-agent-cluster-level-6c4f64
f8cc-zgnp5, namespace: kube-system
k8s pod ip: 10.180.0.3, podname: nginx-deployment-c68885cbb-sf6f5, namespace: ng
inx
k8s pod ip: 10.180.1.4, podname: nginx-deployment-c68885cbb-w5w2b, namespace: ng
inx
kubed get IP address list from Kubernetes:
kubed sdn connector kubernetes2 start updating IP addresses
kubed checking firewall address object gcp-address, vd 0
 address num change 0/3, new ip list:
  10.180.0.3
  10.180.1.4
  10.184.0.1
kubed sdn connector kubernetes2 finish updating IP addresses
kubed reap child pid: 1252
```

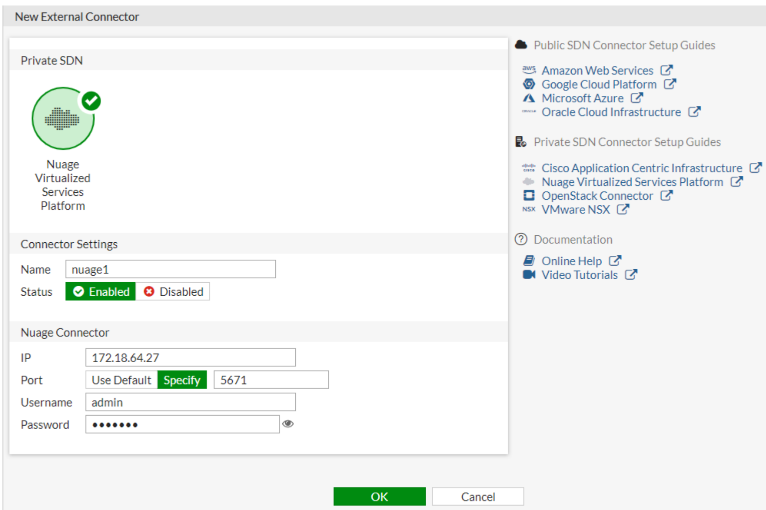
Nuage SDN connector using server credentials

You can use Nuage SDN connectors in dynamic firewall addresses.

The Fortinet SDN Connector for Cisco ACI and Nuage Networks is a standalone connector that connects to SDN controllers within Cisco ACI and Nuage Networks. You must configure a connection to the Fortinet SDN connector in FortiOS to query the dynamic addresses.

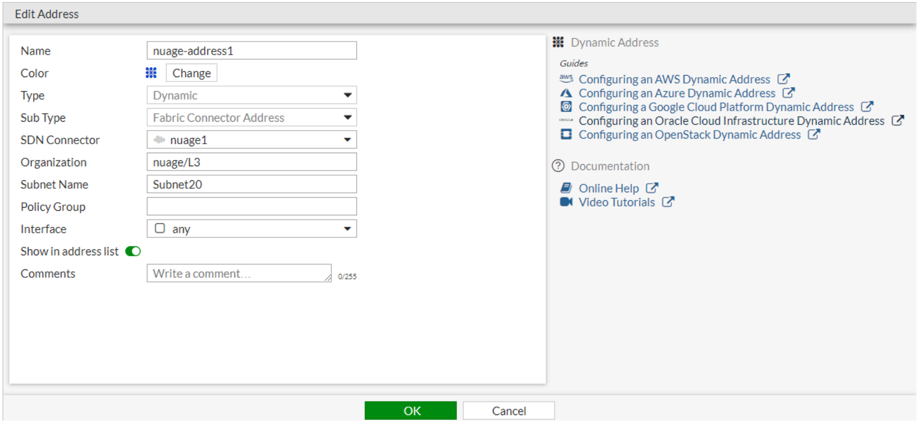
To configure a Nuage connector in the GUI:

1. Create the Nuage SDN connector:
   a. Go to *Security Fabric > External Connectors* and click *Create New*.
   b. In the *Private SDN* section, click *Nuage Virtualized Services Platform*.
   c. Configure the settings as needed.
   d. Click *OK*.



2. Create the dynamic firewall address for the connector:
   a. Go to *Policy & Objects > Addresses* and click *Create New > Address*.
   b. Configure the following settings:
      i. For *Type*, select *Dynamic*.
      ii. For *Sub Type*, select *Fabric Connector Address*.
      iii. For *SDN Connector*, select the Nuage connector.
      iv. Configure the remaining settings as needed.

c. Click *OK*.



To verify the SDN connector resolves the dynamic firewall IP addresses in the GUI:

1. Go to *Policy & Objects > Addresses*.
2. In the address table, hover over an address to view which IP addresses it resolves to.

To configure a Nuage connector in the CLI:

1. Create the SDN connector:

```
config system sdn-connector
    edit "nuage1"
        set type nuage
        set server "172.18.64.27"
        set server-port 5671
        set username "admin"
        set password xxxxxxx
    next
end
```

2. Create the dynamic firewall address for the connector:

```
config firewall address
    edit "nuage-address1"
        set type dynamic
        set sdn "nuage1"
        set color 19
        set organization "nuage/L3"
        set subnet-name "Subnet20"
    next
end
```

To verify the SDN connector resolves the dynamic firewall IP addresses in the CLI:

```
# diagnose firewall dynamic list

List all dynamic addresses:
nuage1.nuage.nuage/L3.Subnet20.*: ID(196)
        ADDR(192.168.20.92)
        ADDR(192.168.20.240)
```

Nutanix SDN connector using server credentials

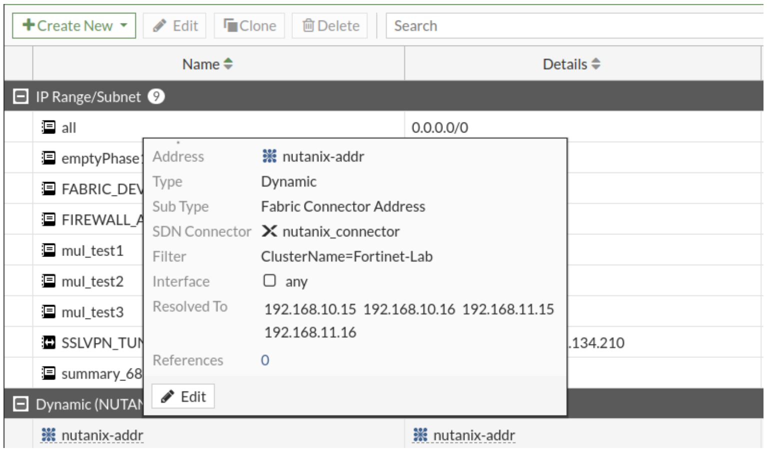
FortiOS automatically updates dynamic addresses for Nutanix using an Nutanix SDN connector, including mapping the following attributes from Nutanix instances to dynamic address groups in FortiOS:

- Cluster name
- Cluster UUID
- Description
- Host name
- Host UUID
- Hypervisor type
- Image name
- Image UUID
- Subnet name
- Subnet UUID
- VM name
- VM UUID

To configure a Nutanix connector using the GUI:

1. Configure the Nutanix SDN connector:
   a. Go to *Security Fabric > External Connectors*.
   b. Select *Nutanix*.
   c. In the *IP address* field, enter the IP address for your Nutanix environment.
   d. In the *Port* field, enter the desired port.
   e. In the *Username* and *Password* fields, enter the credentials for your Nutanix environment.
   f. Click *OK*.
2. Create a dynamic firewall address for the configured Nutanix SDN connector:
   a. Go to *Policy & Objects > Addresses*.
   b. Click *Create New*, then select *Address*.
   c. From the *Type* dropdown list, select *Dynamic*.
   d. From the *Sub Type* dropdown list, select *Fabric Connector Address*.
   e. From the *SDN Connector* dropdown list, select the Nutanix connector.
   f. From the *Filter* dropdown list, select the desired filters.
   g. Click *OK*.
3. Ensure that the Nutanix SDN connector resolves dynamic firewall IP addresses:
   a. Go to *Policy & Objects > Addresses*.
   b. Hover over the address created in step 2 to see a list of IP addresses for instances that satisfy the filter

requirements configured in step 2. In this example, the configured filter is "ClusterName=Fortinet-Lab":



To configure a Nutanix connector using the CLI:

1. Configure the Nutanix SDN connector:
```
config system sdn-connector
   edit "nutanix_connector"
      set status disable
      set type nutanix set server "172.18.33.59"
      set server-port 9440
      set username "admin"
      set password **********
      set update-interval 60
   next
end
```
2. Create a dynamic firewall address for the configured Nutanix SDN connector:
```
config firewall address
   edit "nutanix-addr"
      set uuid 382ceafe-8e72-51eb-7300-0807ee907946
      set type dynamic
      set sdn "nutanix_connector"
      set color 2
      set filter "ClusterName=Fortinet-Lab"
   next
end
```
3. Ensure that the Nutanix SDN connector resolves dynamic firewall IP addresses:
```
config firewall address
   edit "nutanix-addr"
      set uuid 382ceafe-8e72-51eb-7300-0807ee907946
      set type dynamic
      set sdn "nutanix_connector"
      set color 2
      set filter "ClusterName=Fortinet-Lab"
      config list
         edit "192.168.10.15"
         next
         edit "192.168.10.16"
         next
         edit "192.168.11.15"
         next
         edit "192.168.11.16"
```

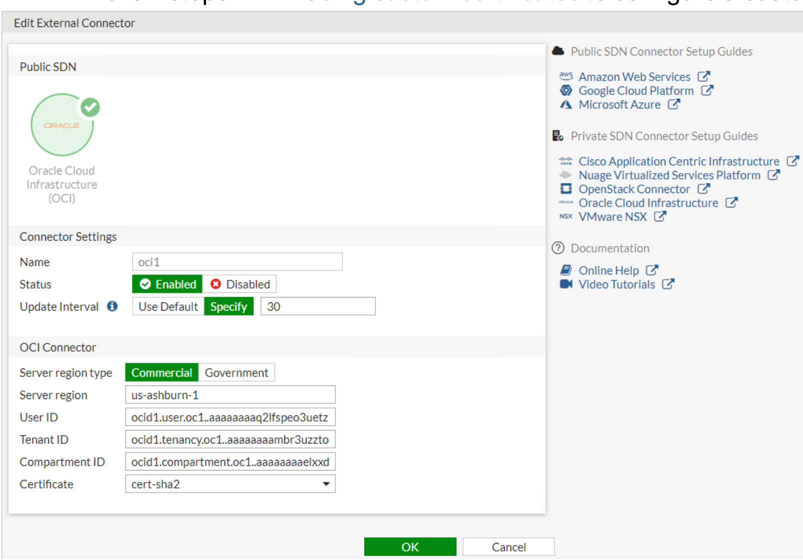
```
            next
        end
    next
end
```

OCI SDN connector using certificates

You can configure SDN connector integration with Oracle Cloud Infrastructure (OCI).

To configure an OCI SDN connector in the GUI:

1. Go to *Security Fabric > External Connectors* and click *Create New*.
2. In the *Public SDN* section, select *Oracle Cloud Infrastructure (OCI)*.
3. Configure the connector as desired:
   a. *User ID*: Enter the OCID of the OCI user who belongs to the administrator group. See Certificate-based SDN connector requirements.
   b. For the *OCI Certificate* field, you must select a certificate that satisfies OCI key size limits. The minimum size is 2048 bits. Do one of the following:
      i. Select the built-in default certificate called Fortinet_Factory.
      ii. Follow steps 1-2 in Using custom certificates to configure a custom certificate.



4. Click *OK*.
5. At this stage, you must register the certificate's fingerprint to the specified OCI user.
   a. Go to the OCI user, then *API Keys > Add Public Key*.
   b. If you selected the Fortinet_Factory certificate in step 2f, do the following:
      i. In FortiOS, go to *System > Certificate*. Select Fortinet_Factory, then click *Download*.
      ii. You now have the Fortinet_Factory.cer file. Create a public key file in PEM format from it, using a freely available tool of your choice such as OpenSSL.
   c. Copy and paste the content of the certificate PEM key file in the *Add Public Key* window in OCI. Click *Add*.

d. You now see the fingerprint.

You can configure the following for the fingerprint:

1. *Update Interval*: The default value is 60 seconds. You can change the value to between 1 and 3600 seconds.
2. *Status*: Green means that the connector is enabled. You can disable it at any time by toggling the switch.

e. Click *OK*.

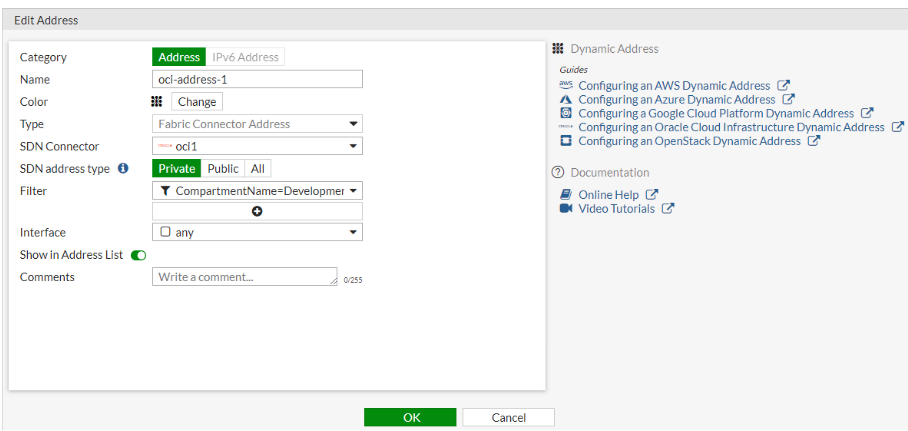
6. Go to *Policy & Objects* > *Addresses* and click *Create New* > *Address*.
7. Configure the address as needed, selecting the OCI connector in the *SDN Connector* field. The following filters are supported:

`'vm_name=<vm name>'`: matches VM instance name.

`'instance_id=<instance id>'`: matches instance OCID.

`'tag.<key>=<value>'`: matches freeform tag key and its value.

`'definedtag.<namespace>.<key>=<value>'`: matches a tag namespace, tag key, and its value.



8. Click *OK*.

To configure an OCI SDN connector in the CLI:

1. Configure an SDN connector:

```
config system sdn-connector
    edit "oci1"
        set status enable
        set type oci
        set tenant-id
"ocid1.tenancy.oc1..aaaaaaaaaaa3aaaaaaaaaaaaaaaa77xxxxxx54bbbbbb4xxxx35xx55xxxx"
        set user-id
"ocid1.user.oc1..aaaaaaaaa2laaaaa3aaaaaaaaaabbbbbbbbbbcccc3ccccccccccxxxxxxxx"
        set compartment-id
"ocid1.compartment.oc1..aaaaaaaaaaaaaaaaaa7bbbbbbbbbbcccccccccc6xxx53xxxx7xxxxxxxxxx"
        set oci-region "us-ashburn-1"
```

```
        set oci-region-type commercial
        set oci-cert "cert-sha2"
        set update-interval 30
    next
end
```

2. Create a dynamic firewall address for the SDN connector with a supported filter:

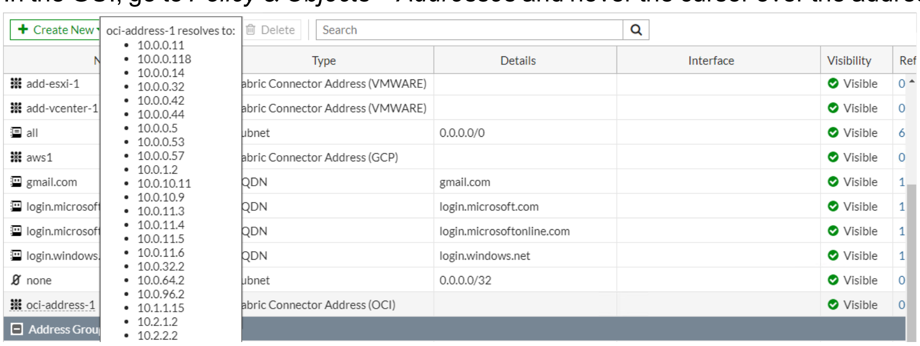
```
config firewall address
    edit "oci-address-1"
        set type dynamic
        set sdn "oci1"
        set filter "CompartmentName=DevelopmentEngineering"
    next
end
```

To confirm that dynamic firewall addresses are resolved by the SDN connector:

1. In the CLI, check that the addresses are listed:

```
config firewall address
    edit "oci-address-1"
        set type dynamic
        set sdn "oci1"
        set filter "CompartmentName=DevelopmentEngineering"
        config list
            edit "10.0.0.11"
            next
            edit "10.0.0.118"
            next
            ...
            next
        end
    next
end
```

2. In the GUI, go to *Policy & Objects* > *Addresses* and hover the cursor over the address name.

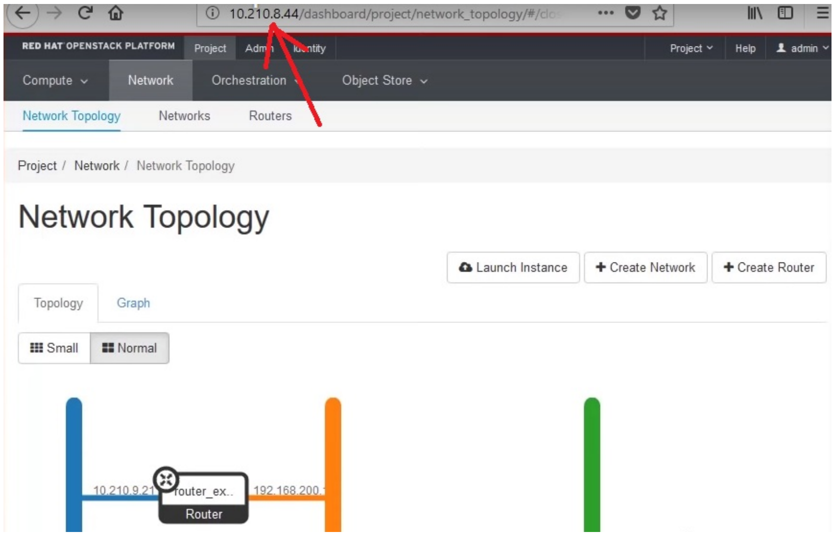


OpenStack SDN connector using node credentials

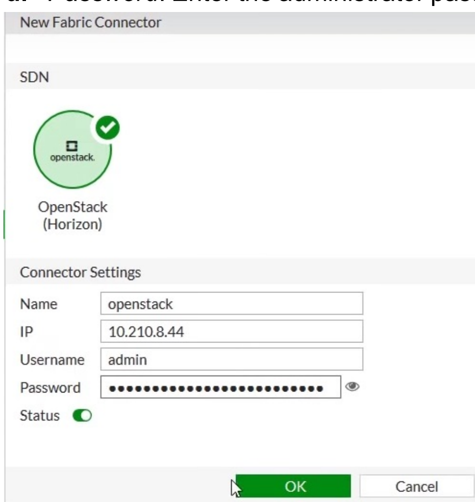
To configure OpenStack SDN connector using node credentials:

1. Go to *Security Fabric* > *External Connectors*.
2. Click *Create New*, then select *OpenStack (Horizon)*.

3. Configure the fields as follows:
   a. *Name*: Name the connector as desired.
   b. *IP*: Enter the OpenStack management component's IP address. Generally you can find it in the OpenStack identity.



   c. *User name*: Enter the specified node's administrator name.
   d. *Password*: Enter the administrator password.



4. Click *OK*. The SDN connector is now configured.

To configure a dynamic firewall address:

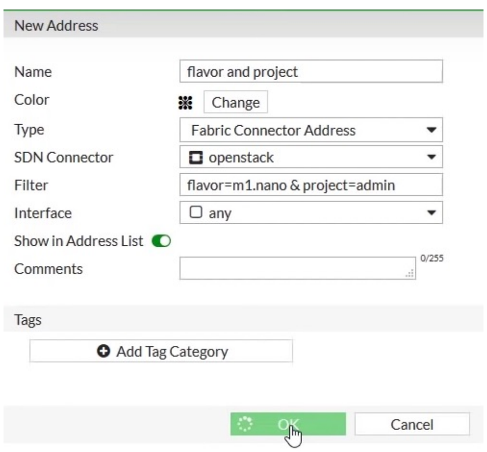
The next step is to create an address that will be used as an address group or single address that acts as the source/destination for firewall policies. The address is based on IP addresses and contains VM instances' IP addresses.

No matter what changes occur to the instances, the SDN connector populates and updates the changes automatically based on the specified filtering condition so that administrators do not need to reconfigure the address content manually. Appropriate firewall policies using the address are applied to instances that are members of the address.

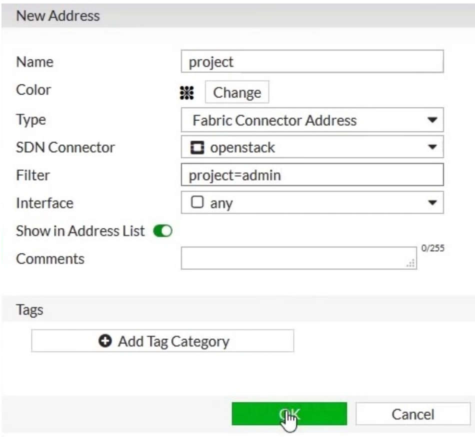
1. Go to *Policy & Objects > Address*. Click *Create New*, then select *Address*.
2. Configure the address as follows:
   a. *Name*: Name the address as desired.
   b. *Type*: Select *Dynamic*.
   c. *Sub Type*: Select *Fabric Connector Address*.
   d. *SDN Connector*: Select *openstack*.
   e. *Filter*: The SDN connector automatically populates and updates only IP addresses belonging to the specified filter that matches the condition. OpenStack Horizon connectors support the following filters:
      i. `id=<instance id>`: This matches a VM instance ID.
      ii. name=<instance name>: This matches a VM instance name.
      iii. `flavor=<instance flavor name>`: This matches an instance flavor name.
      iv. `keypair=<key pair name>`: This matches a key pair name.
      v. `network=<net name>`: This matches a network name.
      vi. `project=<project name>`: This matches a project name.
      vii. `availabilityzone=<zone name>`: This matches an availability zone name.
      viii. `servergroup=<group name>`: This matches a server group name.
      ix. `securitygroup=<security group name>`: This matches a security group name.
      x. `metadata.<key>=<value>`: This matches metadata with its key and value pair.

   You can set filtering conditions using multiple entries with AND ("&") or OR ("|"). When both AND and OR are specified, AND is interpreted first, then OR.

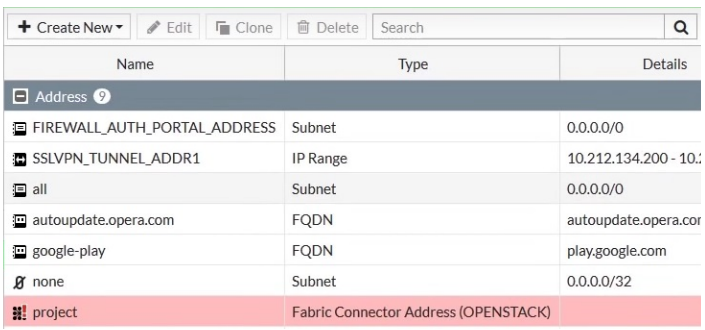
   For example, you could enter `flavor=m1.nano&project=admin`. In this case, IP addresses of instances that match both the flavor name and project name are populated. Wildcards (asterisks) are not allowed in values.



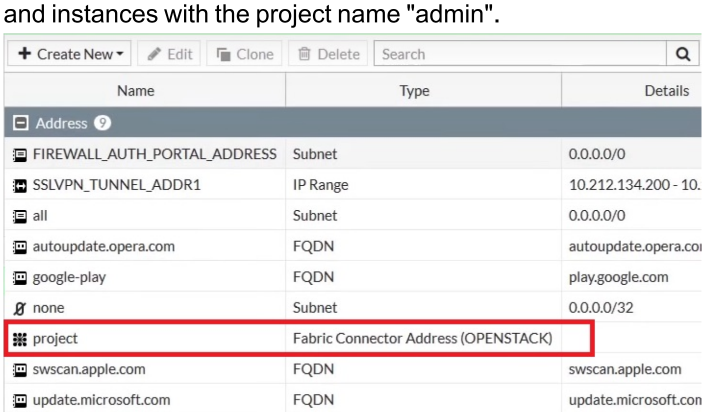
   In this example, let's use project=admin, assuming the project name is admin.

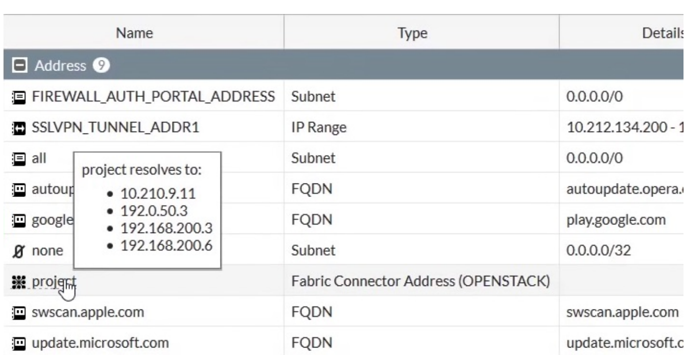


3. Click *OK* after completing all required fields.
4. Ensure that the address was created.



5. After a few minutes, the new address takes effect. Hover your cursor on the address to see a list of IP addresses and instances with the project name "admin".





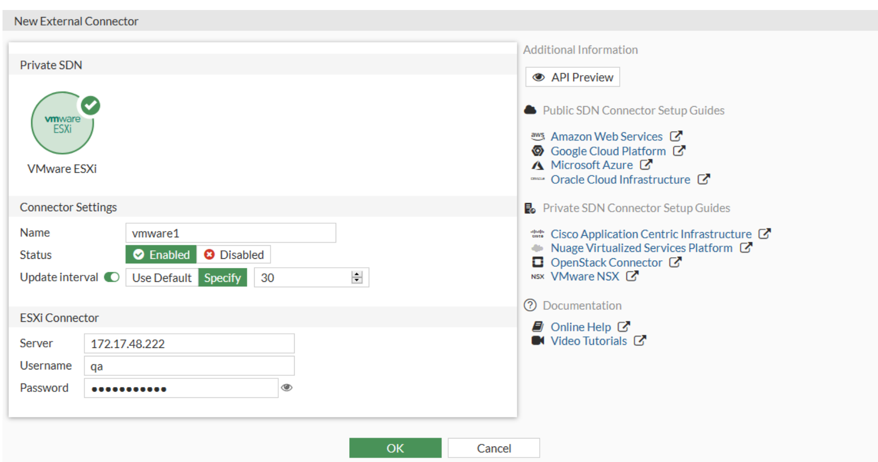
VMware ESXi SDN connector using server credentials

Dynamic addresses for VMware ESXi and vCenter servers can be automatically updated by using a VMware ESXi SDN connector, including mapping the following attributes from VMware ESXi and vCenter objects to dynamic address groups in FortiOS:

- vmid
- host
- name
- uuid
- vmuuid
- vmnetwork
- guestid
- guestname
- annotation
- datacenter
- tag

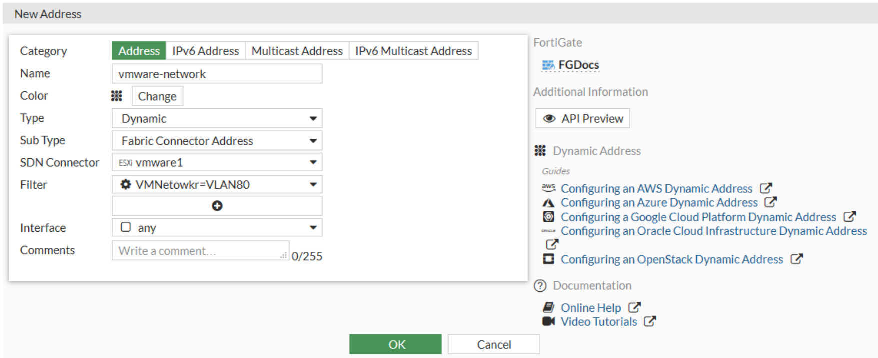
To configure VMware ESXi SDN connector using the GUI:

1. Configure the VMware ESXi SDN connector:
   a. Go to *Security Fabric > External Connectors*.
   b. Click *Create New*, and select *VMware ESXi*.
   c. Configure as shown, substituting the server IP address, username, and password for your deployment. The update interval is in seconds. The password cannot contain single or double quotes.



2. Create a dynamic firewall address for the configured VMware ESXi SDN connector:
   a. Go to *Policy & Objects > Addresses* and click *Create New > Address*.
   b. Configure the address as shown, selecting the desired filter in the *Filter* dropdown list. In this example, the VMware ESXi fabric connector will automatically populate and update IP addresses only for instances that

belong to VLAN80:



3. Ensure that the VMware ESXi SDN connector resolves dynamic firewall IP addresses:
   a. Go to *Policy & Objects > Addresses*.
   b. Hover over the address created in step 2 to see a list of IP addresses for instances that belong to VLAN80 as configured in step 2:

To configure VMware ESXi SDN connector using CLI commands:

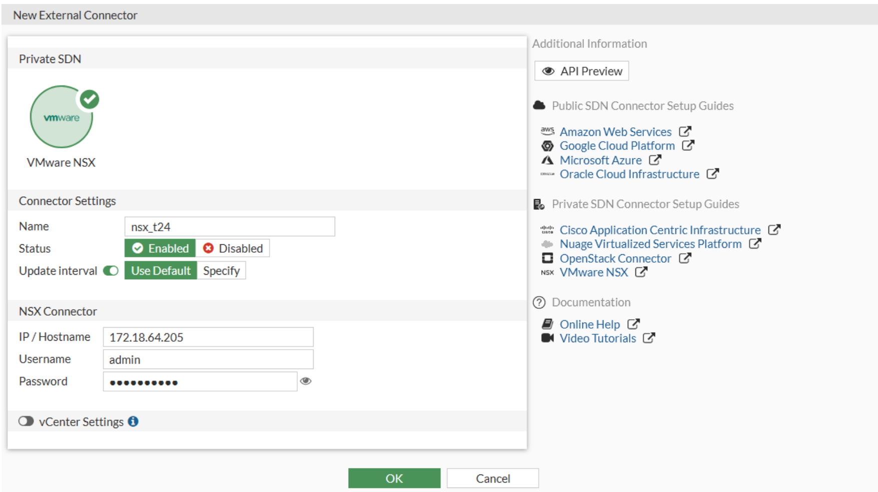
1. Configure the VMware ESXi SDN connector:
```
config system sdn-connector
   edit "vmware1"
      set type vmware
      set server "172.17.48.222"
      set username "example_username"
      set password xxxxx
      set update-interval 30
   next
end
```
2. Create a dynamic firewall address for the configured VMware ESXi SDN connector with the supported VMware ESXi filter. In this example, the VMware ESXi SDN connector will automatically populate and update IP addresses only for instances that belong to the specified VLAN:
```
config firewall address
   edit "vmware-network"
      set type dynamic
      set sdn "vmware1"
      set filter "vmnetwork=VLAN80"
   next
end
```
3. Confirm that the VMware ESXi SDN connector resolves dynamic firewall IP addresses using the configured filter:
```
config firewall address
   edit "vmware-network"
      set type dynamic
      set sdn "vmware1"
      set filter "vmnetwork=VLAN80"
      config list
         edit "192.168.8.240"
         next
      end
   next
end
```

VMware NSX-T Manager SDN connector using NSX-T Manager credentials

This feature provides SDN connector configuration for VMware NSX-T manager. You can import specific groups, or all groups from the NSX-T Manager.

To configure SDN connector for NSX-T Manager in the GUI:

1. Go to *Security Fabric > External Connectors* and click *Create New*.
2. In the *Private SDN* section, click *VMware NSX*.



3. Configure the settings and click *OK*.

To configure SDN connector for NSX-T Manager in the CLI:

```
config system sdn-connector
    edit "nsx_t24"
        set type nsx
        set server "172.18.64.205"
        set username "admin"
        set password xxxxxx
    next
end
```

To import a specific group from the NSX-T Manager:

```
# execute nsx group import nsx_t24 root csf_ns_group
[1] 336914ba-0660-4840-b0f1-9320f5c5ca5e csf_ns_group:
  Name:csf_ns_group
  Address:1.1.1.0
  Address:1.1.1.1
  Address:172.16.10.104
  Address:172.16.20.104
  Address:172.16.30.104
  Address:2.2.2.0
  Address:2.2.2.2
  Address:4.4.4.0
  Address:5.5.5.0
```

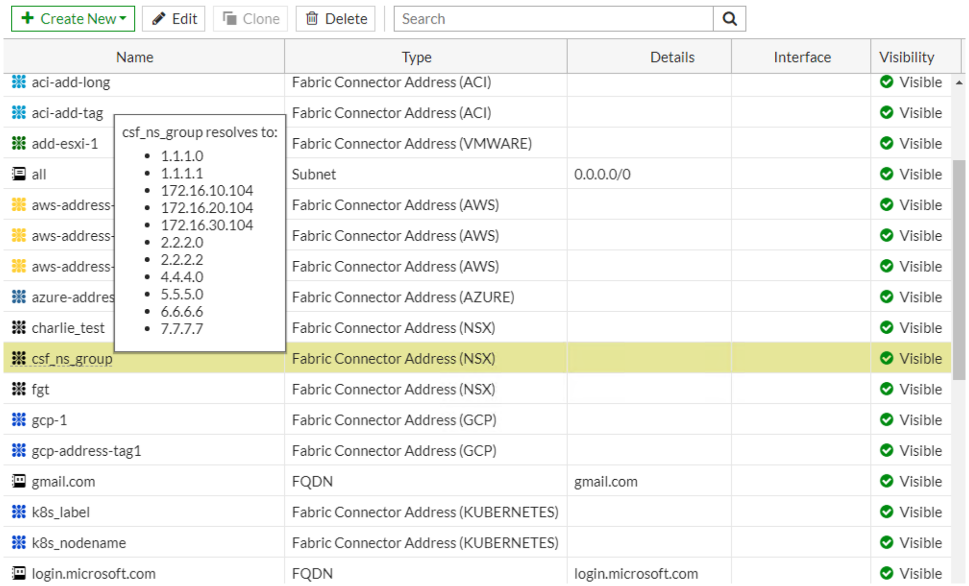
```
  Address:6.6.6.6
  Address:7.7.7.7
```

To import all groups from NSX-T Manager:

```
# execute nsx group import nsx_t24 root
[1] 663a7686-b9a3-4659-b06f-b45c908349a0 ServiceInsertion_NSGroup:
  Name:ServiceInsertion_NSGroup
  Address:10.0.0.2
[2] 336914ba-0660-4840-b0f1-9320f5c5ca5e csf_ns_group:
  Name:csf_ns_group
  Address:1.1.1.0
  Address:1.1.1.1
  Address:172.16.10.104
  Address:172.16.20.104
  Address:172.16.30.104
  Address:2.2.2.0
  Address:2.2.2.2
  Address:4.4.4.0
  Address:5.5.5.0
  Address:6.6.6.6
  Address:7.7.7.7
[3] c462ec4d-d526-4ceb-aeb5-3f168cecd89d charlie_test:
  Name:charlie_test
  Address:1.1.1.1
  Address:2.2.2.2
  Address:6.6.6.6
  Address:7.7.7.7
[4] ff4dcb08-53cf-46bd-bef4-f7aeda9c0ad9 fgt:
  Name:fgt
  Address:172.16.10.101
  Address:172.16.10.102
  Address:172.16.20.102
  Address:172.16.30.103
[5] 3dd7df0d-2baa-44e0-b88f-bd21a92eb2e5 yongyu_test:
  Name:yongyu_test
  Address:1.1.1.0
  Address:2.2.2.0
  Address:4.4.4.0
  Address:5.5.5.0
```

To view the dynamic firewall IP addresses that are resolved by the SDN connector in the GUI:

1. Go to *Policy & Objects > Addresses* to view the IP addresses resolved by an SDN connector.



To view the dynamic firewall IP addresses that are resolved by the SDN connector in the CLI:

```
# show firewall address csf_ns_group
config firewall address
    edit "csf_ns_group"
        set uuid ee4a2696-bacd-51e9-f828-59457565b880
        set type dynamic
        set sdn "nsx_t24"
        set obj-id "336914ba-0660-4840-b0f1-9320f5c5ca5e"
        config list
            edit "1.1.1.0"
            next
            edit "1.1.1.1"
            next
            edit "172.16.10.104"
            next
            edit "172.16.20.104"
            next
            edit "172.16.30.104"
            next
            edit "2.2.2.0"
            next
            edit "2.2.2.2"
            next
            edit "4.4.4.0"
            next
            edit "5.5.5.0"
            next
            edit "6.6.6.6"
            next
            edit "7.7.7.7"
            next
```

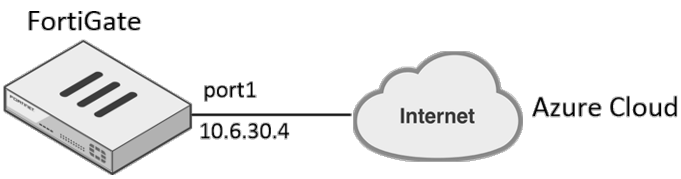
```
        end
    next
end
```

Multiple concurrent SDN connectors

This guide shows how to configure SDN connectors and resolve dynamic firewall addresses through the configured SDN connector in FortiOS.

FortiOS supports multiple SDN connectors including public connectors (AWS, Azure, GCP, OCI, AliCloud) and private connectors (Kubernetes, VMware ESXi, VMware NSX, OpenStack, Cisco ACI, Nuage). FortiOS also supports multiple instances for each type of SDN connector.

This guide uses an Azure SDN connector as an example. The configuration procedure for all supported SDN connectors is the same. In the following topology, the FortiGate accesses the Azure public cloud through the Internet:



This process consists of the following:

1. Configure the interface.
2. Configure a static route to connect to the Internet.
3. Configure two Azure SDN connectors with different client IDs.
4. Check the configured SDN connectors.
5. Create two firewall addresses.
6. Check the resolved firewall addresses after the update interval.
7. Run diagnose commands.

To configure the interface:

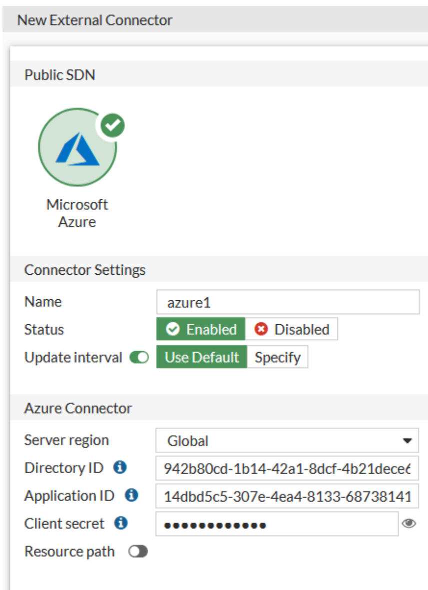
1. In FortiOS, go to *Network* > *Interfaces*.
2. Edit port1:
   a. From the *Role* dropdown list, select *WAN*.
   b. In the *IP/Network Mask* field, enter 10.6.30.4/255.255.255.0 for the interface connected to the Internet.

To configure a static route to connect to the Internet:

1. Go to *Network* > *Static Routes*. Click *Create New*.
2. In the *Destination* field, enter 0.0.0.0/0.0.0.0.
3. From the *Interface* dropdown list, select *port1*.
4. In the *Gateway Address* field, enter 10.60.30.254.

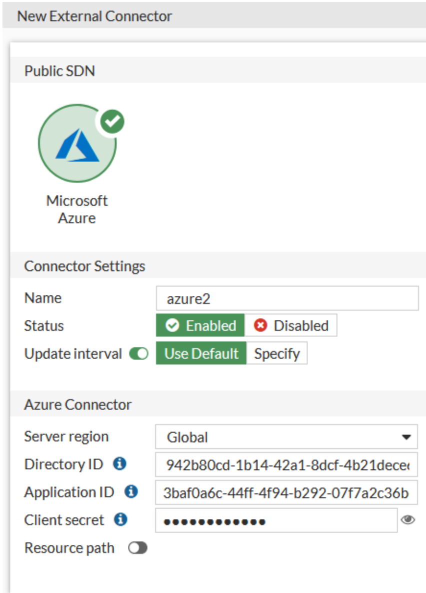
To configure two Azure SDN connectors with different client IDs:

1. Go to *Security Fabric > External Connectors*.
2. Click *Create New*. Configure the first SDN connector:
   a. Select *Microsoft Azure*.
   b. In the *Name* field, enter azure1.
   c. In the *Status* field, select *Enabled*.
   d. From the *Server* region dropdown list, select *Global*.
   e. In the *Directory ID* field, enter the directory ID. In this example, it is 942b80cd-1b14-42a1-8dcf-4b21dece61ba.
   f. In the *Application ID* field, enter the application ID. In this example, it is 14dbd5c5-307e-4ea4-8133-68738141feb1.
   g. In the *Client secret* field, enter the client secret.
   h. Leave the *Resource path* disabled.
   i. Click *OK*.



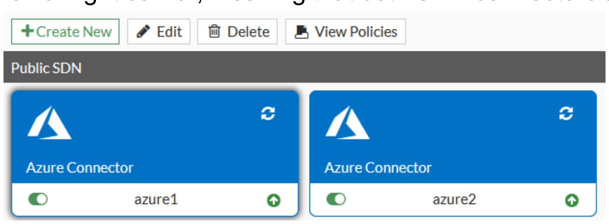
3. Click *Create New*. Configure the second SDN connector:
   a. Select *Microsoft Azure*.
   b. In the *Name* field, enter azure2.
   c. In the *Status* field, select *Enabled*.
   d. From the *Server* region dropdown list, select *Global*.
   e. In the *Directory ID* field, enter the directory ID. In this example, it is 942b80cd-1b14-42a1-8dcf-4b21dece61ba.
   f. In the *Application ID* field, enter the application ID. In this example, it is 3baf0a6c-44ff-4f94-b292-07f7a2c36be6.
   g. In the *Client secret* field, enter the client secret.
   h. Leave the *Resource path* disabled.

i. Click *OK*.



To check the configured SDN connectors:

1. Go to *Security Fabric > External Connectors*.
2. Click the *Refresh* icon in the upper right corner of each configured SDN connector. A green up arrow appears in the lower right corner, meaning that both SDN connectors are connected to the Azure cloud using different client IDs.

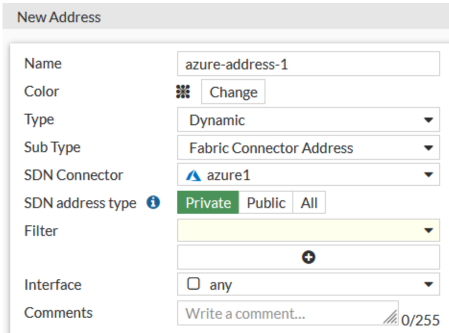


To create two firewall addresses:

This process creates two SDN connector firewall addresses to associate with the configured SDN connectors.

1. Go to *Policy & Objects > Addresses*.
2. Click *Create New > Address*. Configure the first SDN connector firewall address:
   a. In the *Name* field, enter azure-address-1.
   b. From the *Type* dropdown list, select *Dynamic*.
   c. From the *Sub Type* dropdown list, select *Fabric Connector address*.
   d. From the *SDN Connector* dropdown list, select *azure1*.
   e. For *SDN address type*, select *Private*.
   f. From the *Filter* dropdown list, select the desired filter.
   g. For *Interface*, select *any*.

h. Click *OK*.

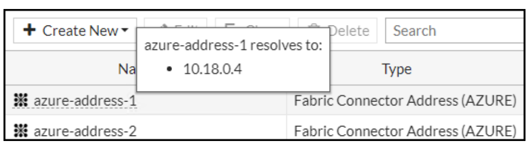


3. Click *Create New* > *Address*. Configure the second SDN connector firewall address:
   a. In the *Name* field, enter azure-address-1.
   b. From the *Type* dropdown list, select *Dynamic*.
   c. From the *Sub Type* dropdown list, select *Fabric Connector address*.
   d. From the *SDN Connector* dropdown list, select *azure2*.
   e. For *SDN address type*, select *Private*.
   f. From the *Filter* dropdown list, select the desired filter.
   g. For *Interface*, select *any*.
   h. Click *OK*.

To check the resolved firewall addresses after the update interval:

By default, the update interval is 60 seconds.

1. Go to *Policy & Objects* > *Addresses*.
2. Hover over the created addresses. The firewall address that the configured SDN connectors resolved display.



To run diagnose commands:

Run the `show sdn connector status` command. Both SDN connectors should appear with a status of `connected`.

Run the `diagnose debug application azd -1` command. The output should look like the following:

```
Level2-downstream-D # diagnose debug application azd -1
...
azd sdn connector azure1 start updating IP addresses
azd checking firewall address object azure-address-1, vd 0
IP address change, new list:
10.18.0.4
...
```

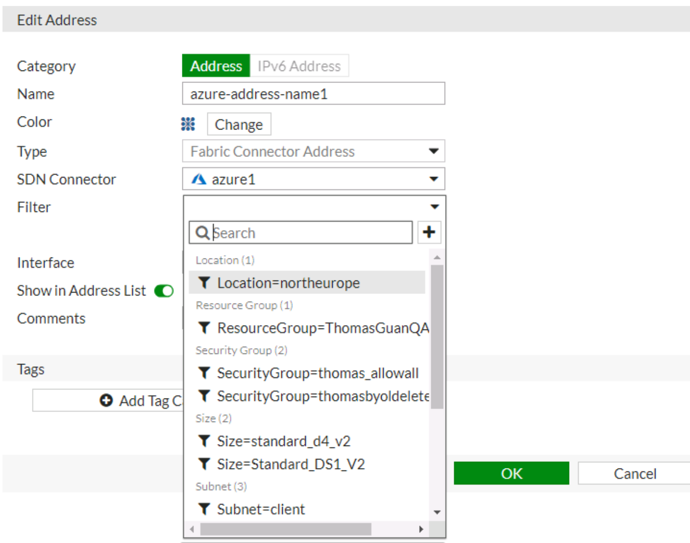
To restart the Azure SDN connector daemon, run the `diagnose test application azd 99` command.

Filter lookup in SDN connectors

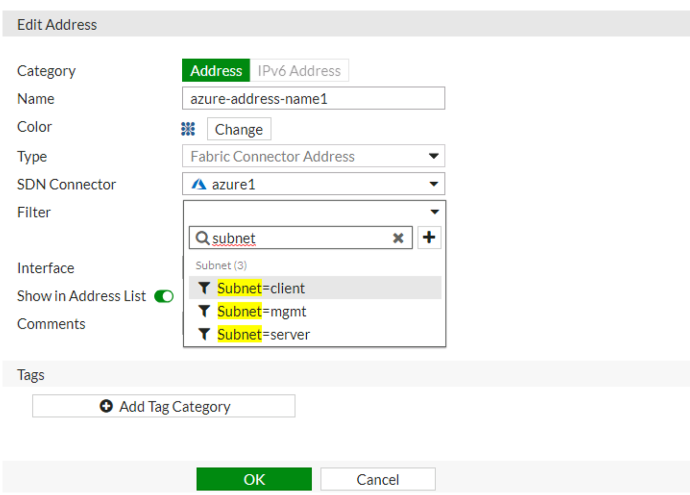
When configuring dynamic address mappings for filters in SDN connectors for Azure, GCP, OpenStack, Kubernetes, and AliCloud, FortiGate can query the filters automatically.

To use the filter lookup:

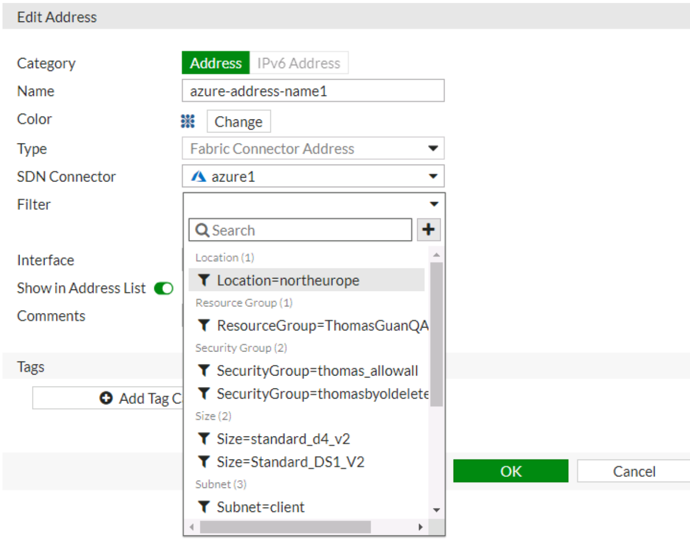
1. Navigate to *Policy & Objects > Addresses*.
2. Create or edit an SDN connector type dynamic IP address.
   Supported SDN connector types include: AWS, Azure, GCP, OpenStack, Kubernetes, and AliCloud. The example below is for an Azure SDN connector.
3. In the address *Filter* field, you can perform the following actions:
   - List all available filters.

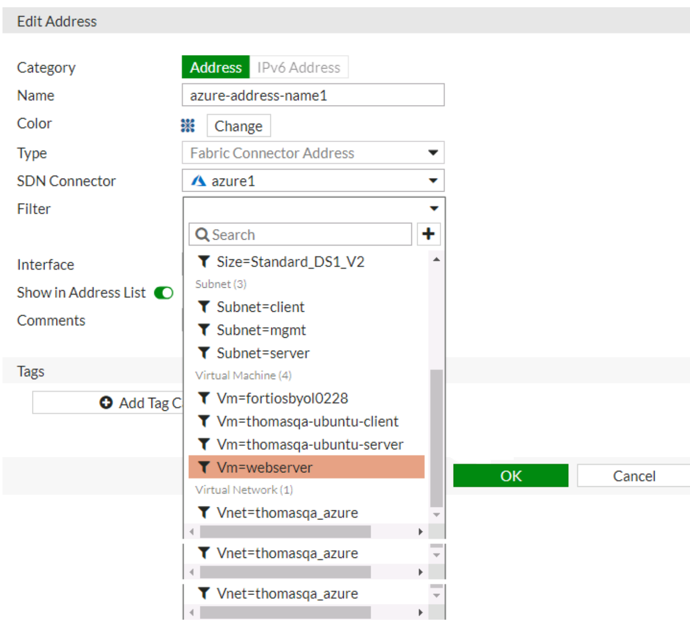


   - Search the available filters.

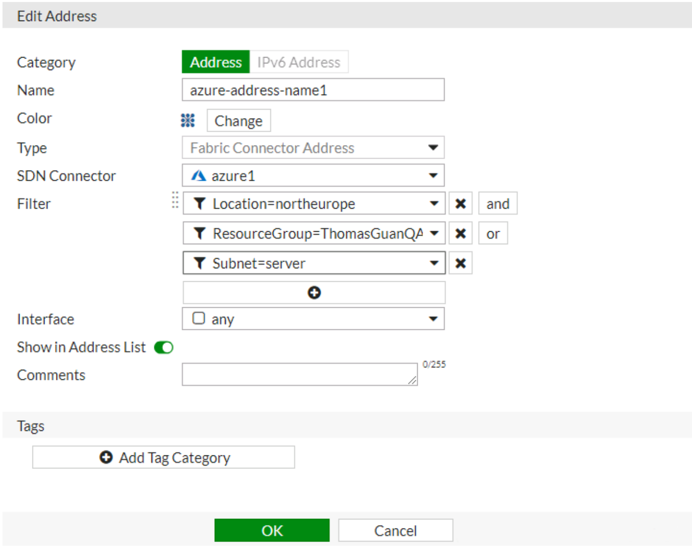


- Create custom filters.





- Set filter logic [and|or].



Support for wildcard SDN connectors in filter configurations

Wildcards are supported for SDN connectors when configuring dynamic address filters.

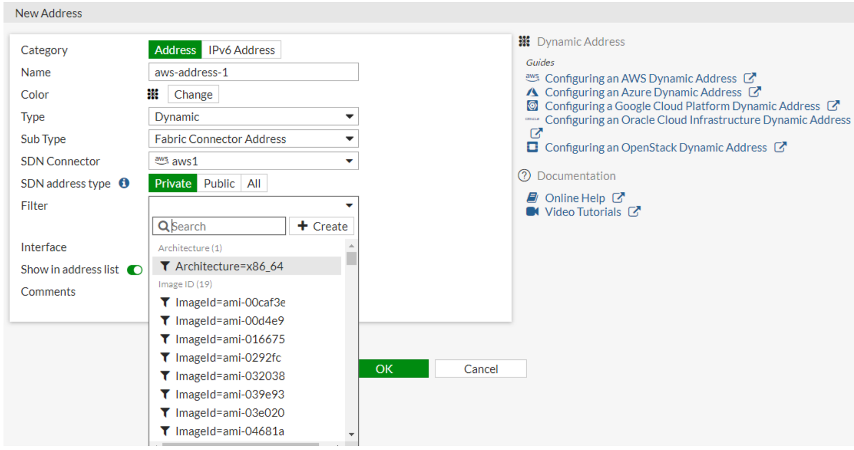
The following SDN connector types are currently supported:

- AWS
- Azure
- Google Cloud Platform
- Kubernetes
- OpenStack
- Oracle Cloud Infrastructure
- VMware ESXi

To configure a dynamic address filter for AWS in the GUI:

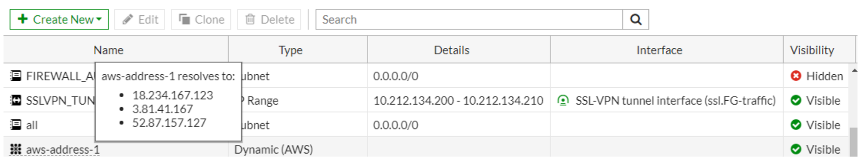
1. Create the SDN connector:
   a. Go to *Security Fabric > External Connectors*.
   b. Click *Create New*.
   c. In the *Public SDN* section, click *Amazon Web Services (AWS)*.
   d. Configure the settings as needed.
   e. Click *OK*.
2. Create the dynamic firewall address:
   a. Go to *Policy & Objects > Addresses*.
   b. Click *Create New > Address*
   c. Enter a name for the address, then configure the following settings:
      - Set *Type* to *Dynamic*.
      - Set *Sub Type* to *Fabric Connector Address*.
      - Set *SDN Connector* to *aws1*.
      - Set *SDN address type* to *Private*.

- For *Filter*, click *Create*, enter `Tag.Name=aws*`, the click *OK*.

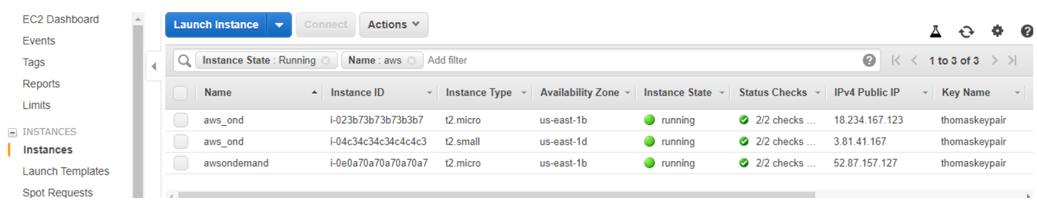


d. Click *OK*.

3. In the address table, hover over the address to view what IPs it resolves to.



4. In AWS, verify to confirm the IP addresses match.



To configure a dynamic address filter for AWS in the CLI:

1. Configure the SDN connector:

```
config firewall address
    edit "aws-address-1"
        set type dynamic
        set sdn "aws1"
        set filter "Tag.Name=aws*"
        set sdn-addr-type public
    next
end
```

2. Create the dynamic firewall address and verify where the IP addresses resolve to:

```
config firewall address
    edit "aws-address-1"
        set type dynamic
        set sdn "aws1"
        set filter "Tag.Name=aws*"
        set sdn-addr-type public
        config list
            edit "18.234.167.123"
```

```
                next
                edit "3.81.41.167"
                next
                edit "52.87.157.127"
                next
            end
        next
    end
```

3. In AWS, verify that the IP addresses match.

Endpoint/Identity connectors

SSO fabric connectors integrate SSO authentication into the network. This allows users to enter their credentials only once, and have those credentials reused when accessing other network resources through the FortiGate.

The following fabric connectors are available:

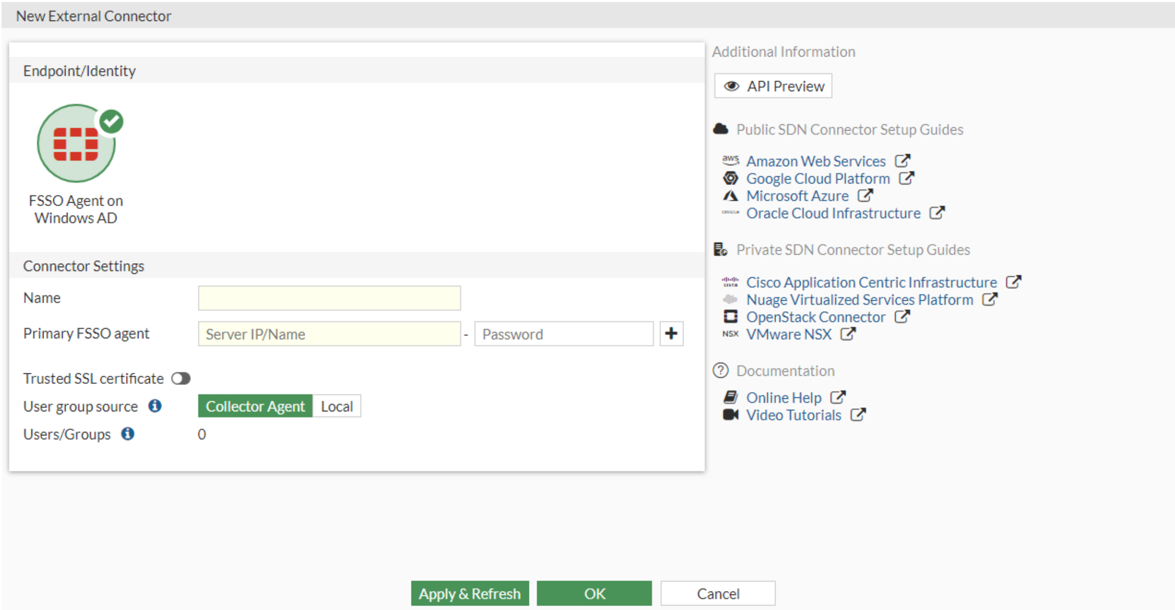
- Fortinet single sign-on agent on page 2490
- Poll Active Directory server on page 2491
- Symantec endpoint connector on page 2492
- RADIUS single sign-on agent on page 2498
- Exchange Server connector on page 2501

Fortinet single sign-on agent

To create an FSSO agent connector in the GUI:

1. Go to *Security Fabric > External Connectors*.
2. Click *Create New*.

3. In the *Endpoint/Identity* section, click *FSSO Agent on Windows AD*.



4. Fill in the *Name*, and *Primary FSSO Agent* server IP address or name and *Password*.
5. Optionally, add more FSSO agents by clicking the plus icon.
6. Optionally, enable *Trusted SSL certificate* and select or import a certificate.
7. Select the *User group source*:
   - *Collector Agent*: User groups will be pushed to the FortiGate from the collector agent. Click *Apply & Refresh* to fetch group filters from the collector agent.
   - *Local*: User groups will be specified in the FortiGate unit's configuration. Select the LDAP server from the list, then click *Edit* to select the *Users*, *Groups*, and *Organizational Units*. Optionally, enable *Proactively retrieve from LDAP server* and configure the *Search filter* and *Interval*.
8. Click *OK*.

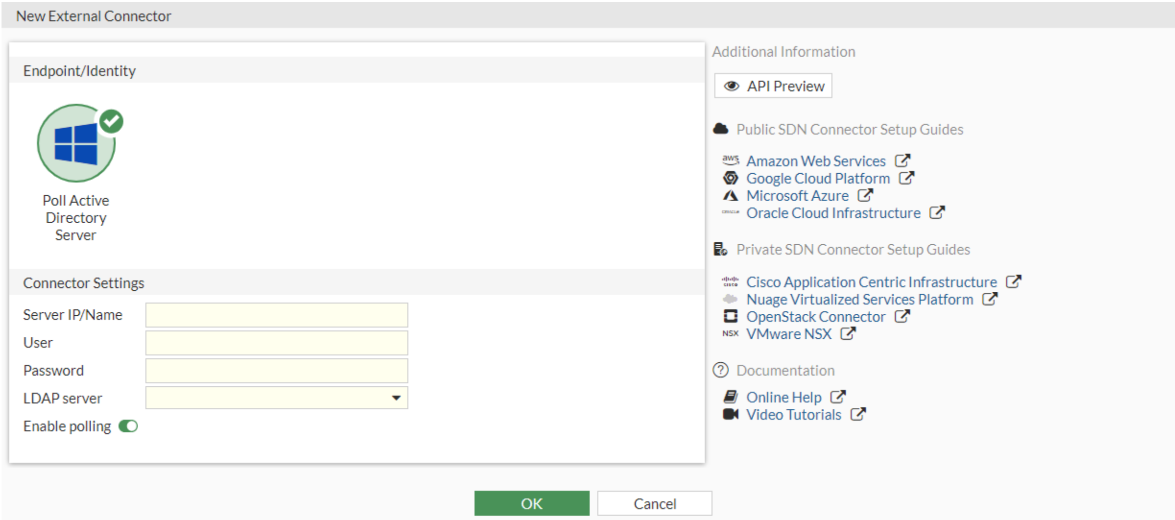
Poll Active Directory server

The FortiGate unit can authenticate users and allow them network access based on groups membership in Windows Active Directory (AD).

To create an AD server connector in the GUI:

1. Go to *Security Fabric > External Connectors*.
2. Click *Create New*.

3. In the *Endpoint/Identity* section, click *Poll Active Directory Server*.



4. Fill in the *Server IP/Name*, *User*, and *Password* for the AD server.
5. Select the LDAP server from the list.
6. If necessary, disable *Enable Polling*. This can be used to temporarily stop the FortiGate from polling security event logs on the Windows logon server, for troubleshooting purposes.
7. Click *OK*.

Symantec endpoint connector

With the Fabric connector for Symantec Endpoint Protection Manager (SEPM), you can use the client IP information from SEPM to assign to dynamic IP addresses on FortiOS.

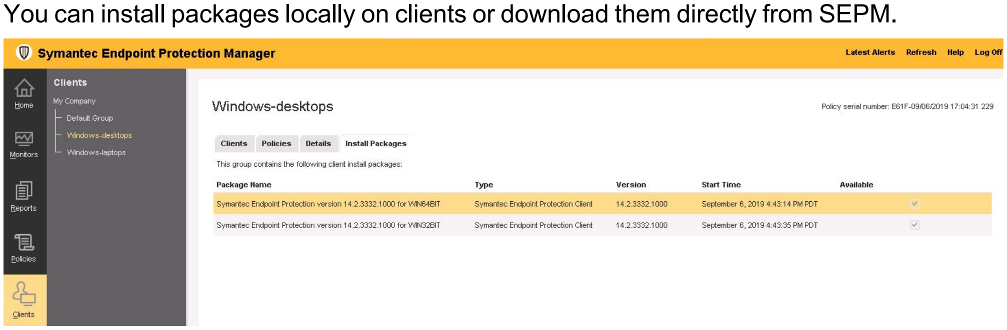
When communication between FortiGate and SEPM is established, FortiGate polls every minute for updates via TLS over port 8446. You can use the CLI to change the default one minute polling interval.

For example, you can create a dynamic Fabric Connector IP address subtype and use it in firewall policies as the source address. The dynamic IP address contains all IP addresses sent by SEPM.

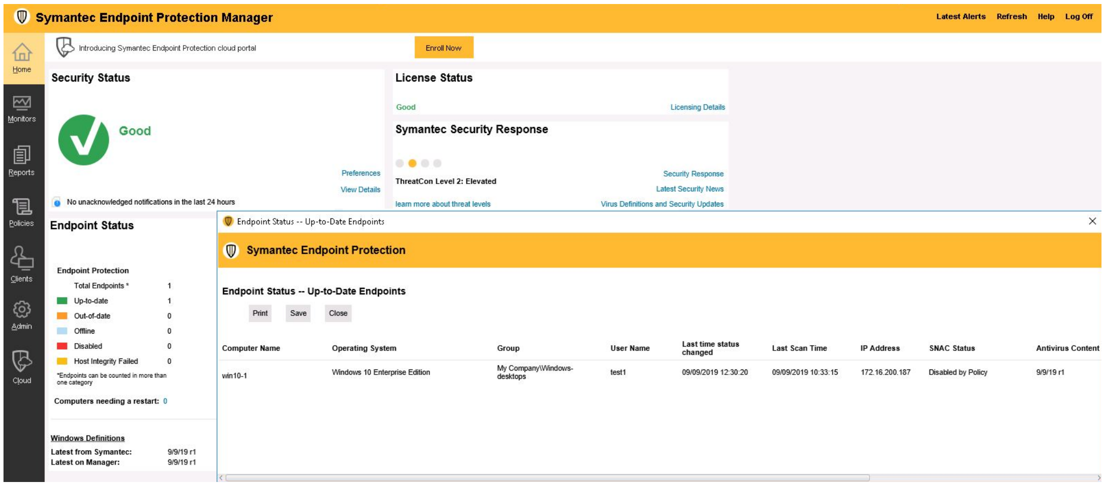
This example shows a dynamic IP address with SEPM and one client PC managed by SEPM using FortiGate as the default gateway.

To configure SEPM on a managed client PC:

1. In SEPM, create client packages for client hosts and group them into SEPM groups.
   You can install packages locally on clients or download them directly from SEPM.

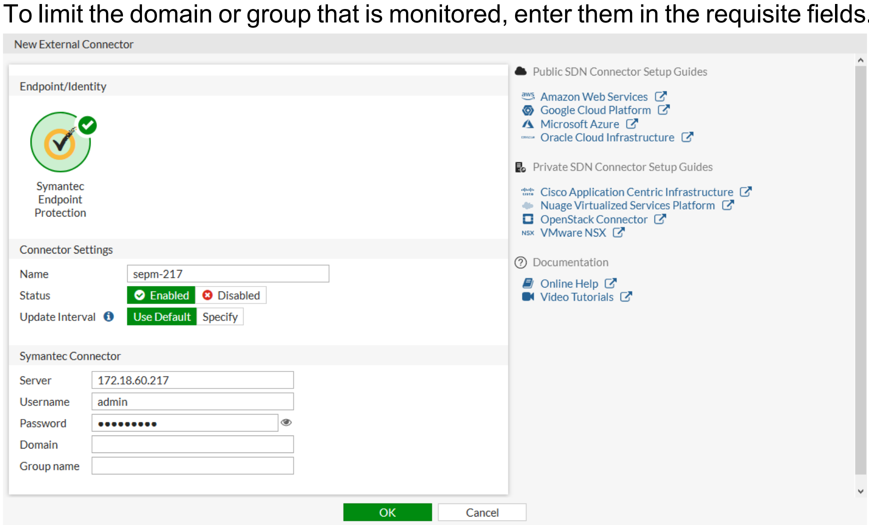


2. When a package is installed on the client host, the host is considered managed by SEPM.
   Even if the host has multiple interfaces, only one IP per host is displayed.



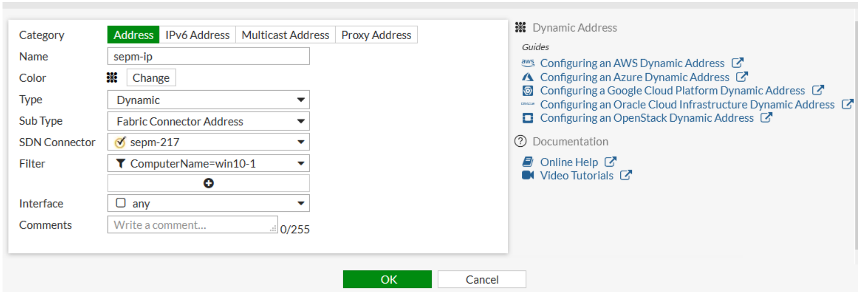
To configure Symantec endpoint connector on FortiGate in the GUI:

1. Go to *Security Fabric > External Connectors* and click *Create New*:
   a. In the *Endpoint/Identity* section, click *Symantec Endpoint Protection*.
   b. Fill in the *Name*, and set the *Status* and *Update Interval*.
   c. Set *Server* to the SEPM IP address.
   d. Enter the *Username* and *Password* for the server.
   e. To limit the domain or group that is monitored, enter them in the requisite fields.



   f. Click *OK*.
      When the connection is established, you can see a green up arrow in the bottom right of the card. You might need to refresh your browser to see the established connection.
2. Go to *Policy & Objects > Addresses* and click *Create New > Address*:
   a. Fill in the address *Name*.
   b. Set *Type* to *Dynamic*.
   c. Set *Sub Type* to *Fabric Connector Address*.
   d. Set *SDN Connector* to the fabric connector that you just created.

e. Add *Filters* as needed.



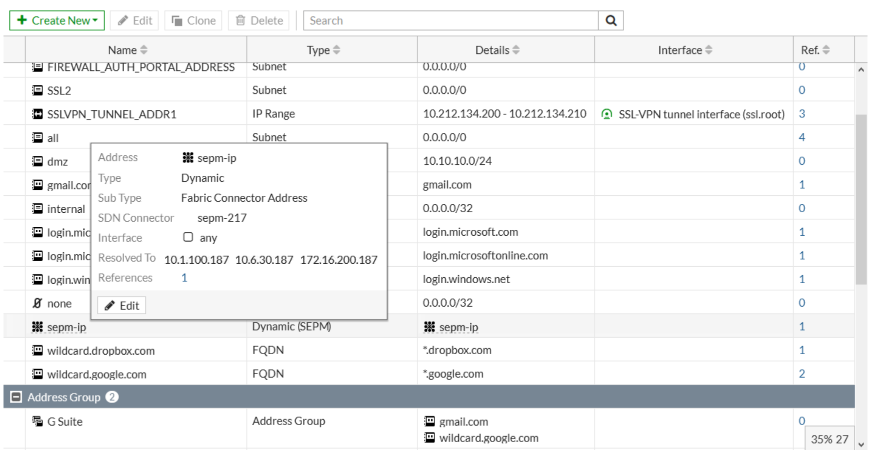
f. Click *OK*.

Filter options are only available for active computers that are configured and registered in SEPM. Free-form filters can be created manually by clicking *Create* and entering the filter, in the format: `filter_type=value`.

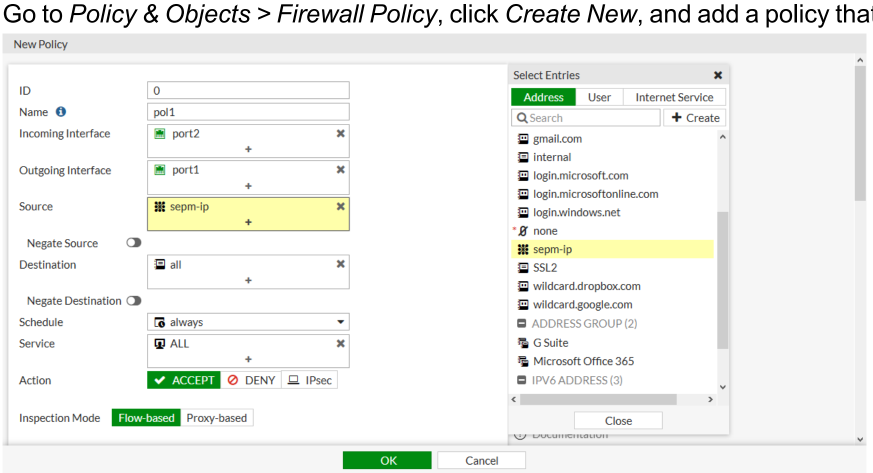
Possible manual filter types are: `GroupName`, `GroupID`, `ComputerName`, `ComputerUUID`, and `OSName`. For example: `GroupName=MyGroup`.

---

3. Go to *Policy & Objects > Addresses* and hover the cursor over the name of the new address to see the resolved IP addresses of the host.

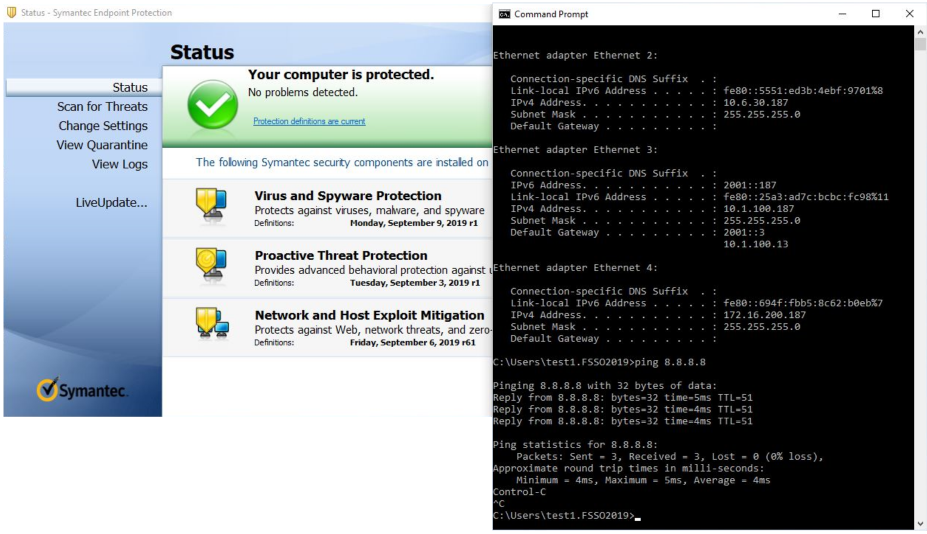


4. Go to *Policy & Objects > Firewall Policy*, click *Create New*, and add a policy that uses the dynamic IP address.

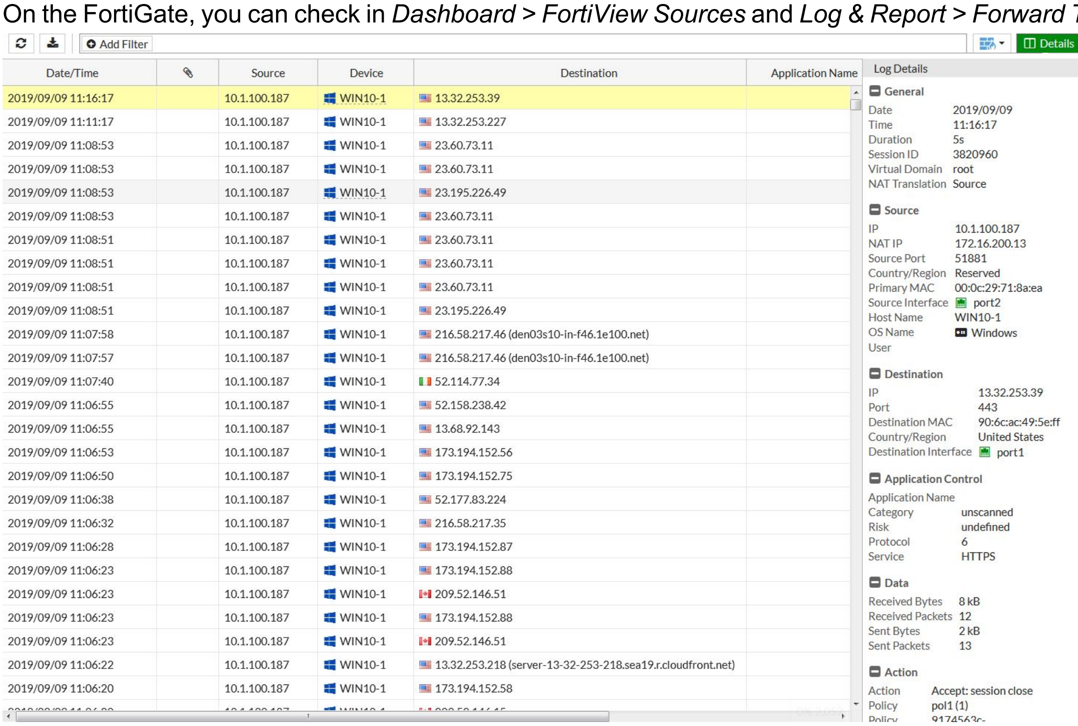


To verify the configuration:

1. On the client PC, check that it is managed by SEPM to access the Internet.



2. On the FortiGate, you can check in *Dashboard > FortiView Sources* and *Log & Report > Forward Traffic*.



Date/Time		Source	Device	Destination	Application Name
2019/09/09 11:16:17		10.1.100.187	WIN10-1	13.32.253.39	
2019/09/09 11:11:17		10.1.100.187	WIN10-1	13.32.253.227	
2019/09/09 11:08:53		10.1.100.187	WIN10-1	23.60.73.11	
2019/09/09 11:08:53		10.1.100.187	WIN10-1	23.60.73.11	
2019/09/09 11:08:53		10.1.100.187	WIN10-1	23.195.226.49	
2019/09/09 11:08:53		10.1.100.187	WIN10-1	23.60.73.11	
2019/09/09 11:08:51		10.1.100.187	WIN10-1	23.60.73.11	
2019/09/09 11:08:51		10.1.100.187	WIN10-1	23.60.73.11	
2019/09/09 11:08:51		10.1.100.187	WIN10-1	23.60.73.11	
2019/09/09 11:08:51		10.1.100.187	WIN10-1	23.195.226.49	
2019/09/09 11:07:58		10.1.100.187	WIN10-1	216.58.217.46 (den03s10-in-f46.1e100.net)	
2019/09/09 11:07:57		10.1.100.187	WIN10-1	216.58.217.46 (den03s10-in-f46.1e100.net)	
2019/09/09 11:07:40		10.1.100.187	WIN10-1	52.114.77.34	
2019/09/09 11:06:55		10.1.100.187	WIN10-1	52.158.238.42	
2019/09/09 11:06:55		10.1.100.187	WIN10-1	13.68.92.143	
2019/09/09 11:06:53		10.1.100.187	WIN10-1	173.194.152.56	
2019/09/09 11:06:50		10.1.100.187	WIN10-1	173.194.152.75	
2019/09/09 11:06:38		10.1.100.187	WIN10-1	52.177.83.224	
2019/09/09 11:06:32		10.1.100.187	WIN10-1	216.58.217.35	
2019/09/09 11:06:28		10.1.100.187	WIN10-1	173.194.152.87	
2019/09/09 11:06:23		10.1.100.187	WIN10-1	173.194.152.88	
2019/09/09 11:06:23		10.1.100.187	WIN10-1	209.52.146.51	
2019/09/09 11:06:23		10.1.100.187	WIN10-1	173.194.152.88	
2019/09/09 11:06:23		10.1.100.187	WIN10-1	209.52.146.51	
2019/09/09 11:06:22		10.1.100.187	WIN10-1	13.32.253.218 (server-13-32-253-218.sea19.r.cloudfront.net)	
2019/09/09 11:06:20		10.1.100.187	WIN10-1	173.194.152.58	

Because this traffic is not authenticated traffic but is based on source IP address only, it is not shown in the GUI firewall monitor or in the `diagnose firewall auth list` CLI command.

To configure Symantec endpoint connector on FortiGate in the CLI:

1. Create the fabric connector:

```
config system sdn-connector
    edit "sepm-217"
        set type sepm
        set server "172.18.60.217"
        set username "admin"
        set password *********
        set status enable
    next
end
```

2. Create the dynamic IP address:

```
config firewall address
    edit "sepm-ip"
        set type dynamic
        set sdn "sepm-217"
        set filter "ComputerName=win10-1"
        config list
            edit "10.1.100.187"
            next
            edit "10.6.30.187"
            next
            edit "172.16.200.187"
            next
        end
    next
end
```

3. Add the dynamic IP address to the firewall policy:

```
config firewall policy
    edit 1
        set name "pol1"
        set srcintf "port2"
        set dstintf "port1"
        set srcaddr "sepm-ip"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set ssl-ssh-profile "certificate-inspection"
        set av-profile "default"
        set logtraffic all
        set fsso disable
        set nat enable
    next
end
```

To troubleshoot Symantec SD connector in the CLI:

```
# diagnose debug application sepmd -1

Output is sent every minute (default). All IPv4 learned from SEPM. IPv6 also sent but not
```

```
yet supported.

2019-09-09 12:01:09 sepmd sdn connector sepm-217 start updating IP addresses
2019-09-09 12:01:09 sepmd checking firewall address object sepm-ip, vd 0
2019-09-09 12:01:09 sepmd sdn connector sepm-217 finish updating IP addresses
2019-09-09 12:01:09 sepmd reap child pid: 18079
2019-09-09 12:02:09 sepmd sdn connector sepm-217 prepare to update
2019-09-09 12:02:09 sepmd sdn connector sepm-217 start updating
2019-09-09 12:02:09 sepm-217 sdn connector will retrieve token after 9526 secs
2019-09-09 12:02:09 sym_new_ip_addr ComputerName win10-1
    ComputerUuid AC894D56-BD86-A786-7DDB-7FD98B718AE0, OsName Windows 10
    IP 172.16.200.187
    GroupName My Company\Windows-desktops, GroupId E61FDEA2AC10C80E46D0B31BB58D7CB3
    DomainName Default, DomainId 6C507580AC10C80E5F3CAED5B1711A8E
2019-09-09 12:02:09 sym_new_ip_addr ComputerName win10-1
    ComputerUuid AC894D56-BD86-A786-7DDB-7FD98B718AE0, OsName Windows 10
    IP 10.6.30.187
    GroupName My Company\Windows-desktops, GroupId E61FDEA2AC10C80E46D0B31BB58D7CB3
    DomainName Default, DomainId 6C507580AC10C80E5F3CAED5B1711A8E
2019-09-09 12:02:09 sym_new_ip_addr ComputerName win10-1
    ComputerUuid AC894D56-BD86-A786-7DDB-7FD98B718AE0, OsName Windows 10
    IP 10.1.100.187
    GroupName My Company\Windows-desktops, GroupId E61FDEA2AC10C80E46D0B31BB58D7CB3
    DomainName Default, DomainId 6C507580AC10C80E5F3CAED5B1711A8E
2019-09-09 12:02:09 2001:0000:0000:0000:0000:0000:0000:0187 is not in IPv4 presentation
format


2019-09-09 12:02:09 sepmd sdn connector sepm-217 start updating IP addresses
2019-09-09 12:02:09 sepmd checking firewall address object sepm-ip, vd 0
2019-09-09 12:02:09 sepmd sdn connector sepm-217 finish updating IP addresses
2019-09-09 12:02:09 sepmd reap child pid: 18089
2019-09-09 12:03:09 sepmd sdn connector sepm-217 prepare to update
2019-09-09 12:03:09 sepmd sdn connector sepm-217 start updating
2019-09-09 12:03:09 sepm-217 sdn connector will retrieve token after 9466 secs
2019-09-09 12:03:09 sym_new_ip_addr ComputerName win10-1
    ComputerUuid AC894D56-BD86-A786-7DDB-7FD98B718AE0, OsName Windows 10
    IP 172.16.200.187
    GroupName My Company\Windows-desktops, GroupId E61FDEA2AC10C80E46D0B31BB58D7CB3
    DomainName Default, DomainId 6C507580AC10C80E5F3CAED5B1711A8E
2019-09-09 12:03:09 sym_new_ip_addr ComputerName win10-1
    ComputerUuid AC894D56-BD86-A786-7DDB-7FD98B718AE0, OsName Windows 10
    IP 10.6.30.187
    GroupName My Company\Windows-desktops, GroupId E61FDEA2AC10C80E46D0B31BB58D7CB3
    DomainName Default, DomainId 6C507580AC10C80E5F3CAED5B1711A8E
2019-09-09 12:03:09 sym_new_ip_addr ComputerName win10-1
    ComputerUuid AC894D56-BD86-A786-7DDB-7FD98B718AE0, OsName Windows 10
    IP 10.1.100.187
    GroupName My Company\Windows-desktops, GroupId E61FDEA2AC10C80E46D0B31BB58D7CB3
    DomainName Default, DomainId 6C507580AC10C80E5F3CAED5B1711A8E
2019-09-09 12:03:09 2001:0000:0000:0000:0000:0000:0000:0187 is not in IPv4 presentation
format
```

To list the SEPM daemon SDN connectors:

```
# diagnose test application sepmd 1
sepm SDN connector list:
  name: sepm-217, status: enabled, updater_interval: 60
```

To list the SEPM daemon SDN filters:

```
# diagnose test application sepmd 2
sepm SDN connector sepm-217 filter list:
  name: sepm-ip, vd 0, filter 'ComputerName=win10-1'
```

RADIUS single sign-on agent

With RADIUS single sign-on (RSSO), a FortiGate can authenticate users who have authenticated on a remote RADIUS server. Based on which user group the user belongs to, the security policy applies the appropriate UTM profiles.

The FortiGate does not interact with the remote RADIUS server; it only monitors RADIUS accounting records that the server forwards (originating from the RADIUS client). These records include the user IP address and user group. The remote RADIUS server sends the following accounting messages to the FortiGate:

Message	Action
Start	If the information in the start message matches the RSSO configuration on the FortiGate, the user is added to the local list of authenticated firewall users.
Stop	The user is removed from the local list of authenticated firewall users because the user session no longer exists on the RADIUS server.

You can configure an RSSO agent connector using the FortiOS GUI; however, in most cases, you will need to use the CLI. There are some default options you may need to modify that can only be done in the CLI.

```
config user radius
    edit <name>
        set rsso enable
        set rsso-radius-response enable
        set rsso-secret <password>
        set rsso-endpoint-attribute <attribute>
        set sso-attribute <attribute>
        set delimiter {plus | comma}
    next
end
```

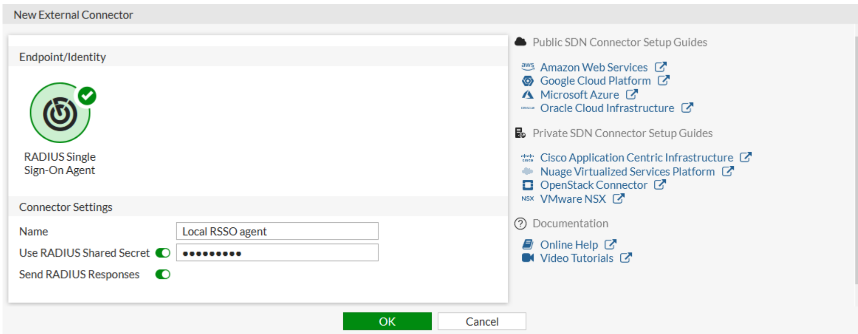
To configure an RSSO agent connector:

1. Create the new connector:
   a. Go to *Security Fabric > External Connectors*.
   b. Click *Create New*.
   c. In the *Endpoint/Identity* section, click *RADIUS Single Sign-On Agent*. The *New Fabric Connector* pane opens.
   d. Enter the connector name.

e. Enable *Use RADIUS Shared Secret*.

> The value entered in *Use RADIUS Shared Secret* must be identical to what the remote RADIUS server uses to authenticate when it sends RADIUS accounting messages to the FortiGate.

f. Enable *Send RADIUS Responses*.

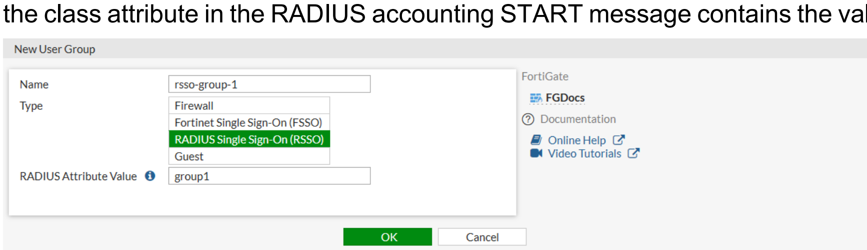


> You should enable *Send RADIUS Responses* because some RADIUS servers continue to send the same RADIUS accounting message several times if there is no response.

g. Click *OK*.

2. Edit the network interface:
   a. Go to *Network > Interfaces*.
   b. Double-click the interface that will receive the RADIUS accounting messages. The *Edit Interface* pane opens.
   c. In the *Administrative Access* section, select the *RADIUS Accounting* checkbox. This will open listening for port 1813 on this interface. The FortiGate will then be ready to receive RADIUS accounting messages.
   d. Click *OK*.
3. Create a local RSSO user group:
   a. Go to *User & Authentication > User Groups*.
   b. Click *Create New*.
   c. Enter the group name.
   d. For the *Type* field, click *RADIUS Single-Sign-ON (RSSO)*.
   e. Enter a value for *RADIUS Attribute Value*.
      This value by default is the class attribute. The FortiGate uses the content of this attribute in RADIUS accounting start messages to map a user to a FortiGate group, which then can be used in firewall policies.

      In this example configuration, the FortiGate will only add a remote RADIUS user to the local firewall user list if the class attribute in the RADIUS accounting START message contains the value group1.



If your users are in multiple groups, you will need to add multiple local RSSO user group.

If the RADIUS attribute value used to map users to a local RSSO group is different than the RADIUS attribute in the RADIUS accounting messages forwarded by the server, you must change it in the CLI.

   f. Click *OK*.
4. Edit the local RSSO agent to modify default options using the CLI.
   For example, the default value for `rsso-endpoint-attribute` might work in common remote access scenarios where users are identified by their unique `Calling-Station-Id`, but in other scenarios the user name might be in a different attribute.

```
config user radius
    edit "Local RSSO Agent"
        set rsso-endpoint-attribute <attribute>
        set sso-attribute <attribute>
    next
end
```

5. Add the local RSSO user group to a firewall policy.

Verifying the RSSO configuration

Verification requires a working remote RADIUS server configured for RADIUS accounting forwarding and wireless or wired clients that use RADIUS for user authentication.

For a quick test, you can use one of the publicly available RADIUS test tools to send RADIUS accounting start and stop messages to the FortiGate. You can also use radclient.

To verify the RSSO configuration:

1. In radclient, enter the RADIUS attributes. These attributes are then executed with the FortiGate IP parameters (sends accounting messages to port 1813) and shared password you configured. `-x` is used for verbose output:

```
root@ControlPC:~# echo "Acct-Status-Type =Start,Framed-Ip-Address=10.1.100.185,User-
Name=test2,Acct-Session-Id=0211a4ef,Class=group1,Calling-Station-Id=00-0c-29-44-BE-B8" |
 radclient -x 10.1.100.1 acct 123456
Sending Accounting-Request of id 180 to 10.1.100.1 port 1813
        Acct-Status-Type = Start
        Framed-IP-Address = 10.1.100.185
        User-Name = "test2"
        Acct-Session-Id = "0211a4ef"
        Class = 0x67726f757031
        Calling-Station-Id = "00-0c-29-44-BE-B8"
rad_recv: Accounting-Response packet from host 10.1.100.1 port 1813, id=180, length=20
root@ControlPC:~#
```

2. Verify that the user is in the local firewall user list with the correct type (`rsso`) and local firewall group (`rsso-group1`):

```
# diagnose firewall auth l

10.1.100.185, test2
        type: rsso, id: 0, duration: 5, idled: 5
        flag(10): radius
        server: vdom1
        packets: in 0 out 0, bytes: in 0 out 0
        group_id: 3
        group_name: rsso-group-1

----- 1 listed, 0 filtered ------
```

Exchange Server connector

FortiGate can collect additional information about authenticated users from corporate Microsoft Exchange Servers. After a user logs in, the additional information can be viewed in various parts of the GUI.

The Exchange connector must be mapped to the LDAP server that is used for authentication.

The following attributes are retrieved:

USER_INFO_FULL_NAME	USER_INFO_COMPANY	USER_INFO_CITY
USER_INFO_FIRST_NAME	USER_INFO_DEPARTMENT	USER_INFO_STATE
USER_INFO_LAST_NAME	USER_INFO_GROUP	USER_INFO_POSTAL_CODE
USER_INFO_LOGON_NAME	USER_INFO_TITLE	USER_INFO_COUNTRY
USER_INFO_TELEPHONE	USER_INFO_MANAGER	USER_INFO_ACCOUNT_EXPIRES
USER_INFO_EMAIL	USER_INFO_STREET	
USER_INFO_USER_PHOTO	USER_INFO_POST_OFFICE_BOX	

Kerberos Key Distribution Center (KDC) automatic discovery is enabled by default. The FortiGate must be able to use DNS to resolve the KDC IP addresses, otherwise the FortiGate will be unable to retrieve additional user information from the Exchange Server.

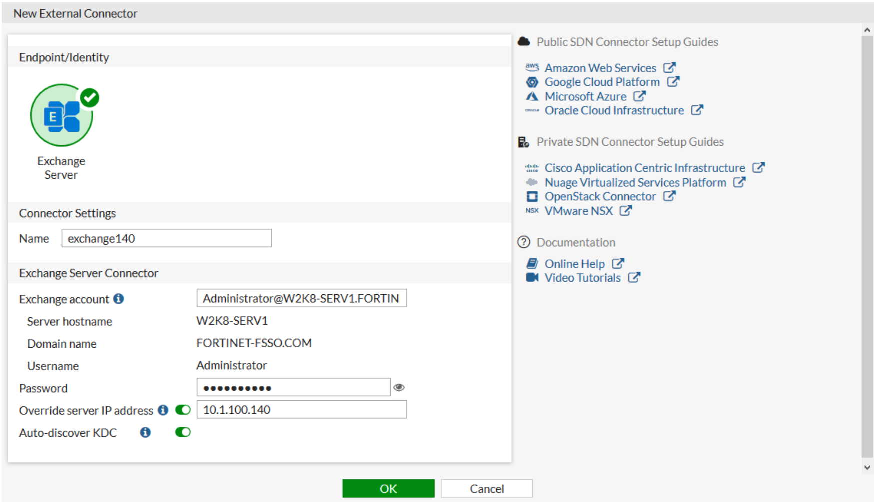
KDC automatic discovery can be disabled, and one or more internal IP addresses that the FortiGate can reach can be configured for KDC.

The Override server IP address is enabled when the IP address of the Exchange server cannot be resolved by DNS and must be entered manually.

To configure an Exchange connector in the GUI:

1. Go to *Security Fabric > External Connectors* and click *Create New*.
2. In the *Endpoint/Identity* section, click *Exchange Server*.
3. Set *Name* to *exchange140*.
4. Set *Exchange account* to *Administrator@W2K8-SERV1.FORTINET-FSSO.COM*.
   Administrator is the username, *W2K8-SERV1* is the exchange server name, and *FORTINET-FSSO.COM* is the domain name.
5. Set *Password* to the password.
6. Enable *Override server IP address* and set it to *10.1.100.140*.

7. Ensure that *Auto-discover KDC* is enabled.

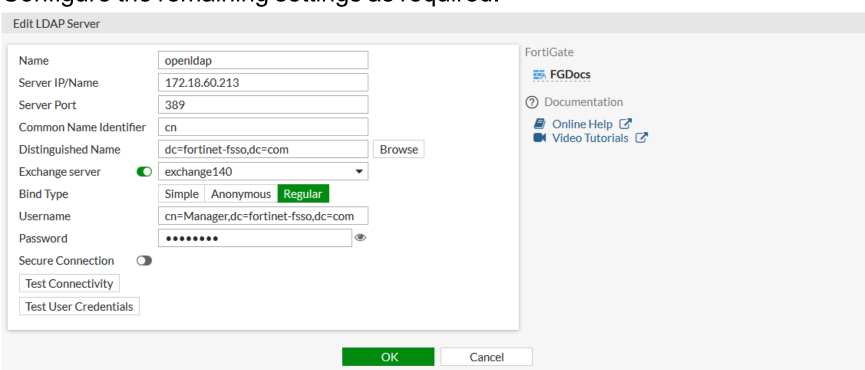


If *Auto-discover KDC* is disabled, one or more KDC IP addresses can be manually entered.

8. Click *OK*.

To link the connector to the LDAP server in the GUI:

1. Go to *User & Authentication > LDAP Servers*.
2. Edit an existing LDAP server, or click *Create New* to create a new one.
3. Enable *Exchange server*, and select the connector from the list.
4. Configure the remaining settings as required.



5. Click *OK*.

To configure an Exchange connector with automatic KDC discovery in the CLI:

```
config user exchange
    edit "exchange140"
        set server-name "W2K8-SERV1"
        set domain-name "FORTINET-FSSO.COM"
        set username "Administrator"
        set password **********
        set ip 10.1.100.140
        set auto-discover-kdc enable
    next
end
```

To link the connector to the LDAP server in the CLI:

```
config user ldap
    edit "openldap"
        set server "172.18.60.213"
        set cnid "cn"
        set dn "dc=fortinet-fsso,dc=com"
        set type regular
        set username "cn=Manager,dc=fortinet-fsso,dc=com"
        set password **********
        set group-member-check group-object
        set group-object-filter "(&(objectclass=groupofnames)(member=*))"
        set member-attr "member"
        set user-info-exchange-server "exchange140"
    next
end
```

Verification

To verify that KDC auto-discovery is working:

```
# diagnose wad debug enable category all
# diagnose wad debug enable level verbose
# diagnose debug enable
# diagnose wad user exchange test-auto-discover
```

```
wad_diag_session_acceptor(3115): diag socket 20 accepted.
__wad_fmem_open(557): fmem=0x12490bd8, fmem_name='cmem 9188 bucket', elm_sz=9188, block_
sz=73728, overhead=0, type=advanced
Starting auto-discover test for all configured user-exchanges.
[NOTE]: If any errors are returned, try manually configuring IPs for the reported errors.

wad_rpc_nspi_test_autodiscover_kdc(1835): Starting DNS SRV request for srv(0x7f938e052050)
query(_kerberos._udp.FORTINET-FSSO.COM)
wad_dns_send_srv_query(705): 1:0: sending DNS SRV request for remote peer _kerberos._
udp.FORTINET-FSSO.COM id=0
1: DNS response received for remote host _kerberos._udp.FORTINET-FSSO.COM req-id=0
wad_dns_parse_srv_resp(409): _kerberos._udp.FORTINET-FSSO.COM: resp_type(SUCCESS)
  srv[0]: name(w2k12-serv1.fortinet-fsso.com) port(88) priority(0) weight(100)
    addr[0]: 10.1.100.131
    addr[1]: 10.6.30.131
    addr[2]: 172.16.200.131
    addr[3]: 2003::131
    addr[4]: 2001::131
  srv[1]: name(fsso-core-DC.Fortinet-FSSO.COM) port(88) priority(0) weight(100)
    addr[0]: 10.6.30.16
    addr[1]: 172.16.200.16
  srv[2]: name(w2k12-serv1.Fortinet-FSSO.COM) port(88) priority(0) weight(100)
    addr[0]: 10.1.100.131
    addr[1]: 172.16.200.131
    addr[2]: 10.6.30.131
    addr[3]: 2001::131
    addr[4]: 2003::131
wad_rpc_nspi_dns_on_discover_kdc_done(1787): Received response for DNS autodiscover req
(0x7f938dfe8050) query(_kerberos._udp.FORTINET-FSSO.COM) n_rsp(3)
```

```
Completed auto-discover test for all configured user-exchanges.
```

To check the collected information after the user has been authenticated:

1. In the GUI, go to *Dashboard* > *Users & Devices*, expand the *Firewall Users* widget, and hover over the user name.
2. In the CLI, run the following diagnose command:

```
# diagnose wad user info 20 test1
'username' = 'test1'
'sourceip' = '10.1.100.185'
'vdom' = 'root'
'cn' = 'test1'
'givenName' = 'test1'
'sn' = 'test101'
'userPrincipalName' = 'test1@Fortinet-FSSO.COM'
'telephoneNumber' = '604-123456'
'mail' = 'test1@fortinet-fsso.com'
'thumbnailPhoto' = '/tmp/wad/user_info/76665fff62ffffffffffffffffff75ff68fffffffffa'
'company' = 'Fortinet'
'department' = 'Release QA'
'memberOf' = 'CN=group321,OU=Testing,DC=Fortinet-FSSO,DC=COM'
'memberOf' = 'CN=g1,OU=Testing,DC=Fortinet-FSSO,DC=COM'
'memberOf' = 'CN=group21,OU=Testing,DC=Fortinet-FSSO,DC=COM'
'memberOf' = 'CN=group1,OU=Testing,DC=Fortinet-FSSO,DC=COM'
'manager' = 'CN=test6,OU=Testing,DC=Fortinet-FSSO,DC=COM'
'streetAddress' = 'One Backend Street 1901'
'l' = 'Burnaby'
'st' = 'BC'
'postalCode' = '4711'
'co' = 'Canada'
'accountExpires' = '9223372036854
```

If the results are not as expected, verify what information FortiGate can collect from the Exchanger Server:

```
# diagnose test application wad 2500
# diagnose test application wad 162
```

Threat feeds

Threat feeds dynamically import an external block list from an HTTP server in the form of a plain text file, or from a STIX/TAXII server. Block lists can be used to enforce special security requirements, such as long term policies to always block access to certain websites, or short term requirements to block access to known compromised locations. The lists are dynamically imported, so that any changes are immediately imported by FortiOS.

There are four types of threat feeds:

FortiGuard Category	The file contains one URL per line. It is available as a *Remote Category* in Web Filter profiles, SSL inspection exemptions, and proxy addresses. See Web rating override on page 1318 for more information. Example:

	`http://example/com.url`
	`https://example.com/url`
	`http://example.com:8080/url`

IP Address	The file contains one IP/IP range/subnet per line. It is available as an *External IP Block List* in DNS Filter profiles, and as a *Source/Destination* in IPv4, IPv6, and proxy policies.
	Example:
	`192.168.2.100`
	`172.200.1.4/16`
	`172.16.1.2/24`
	`172.16.8.1-172.16.8.100`
	`2001:0db8::eade:27ff:fe04:9a01/120`
	`2001:0db8::eade:27ff:fe04:aa01-2001:0db8::eade:27ff:fe04:ab01`

Domain Name	The file contains one domain per line. Simple wildcards are supported. It is available as a *Remote Category* in DNS Filter profiles. See External resources for DNS filter on page 2511 for more information.
	Example:
	`mail.*.example.com`
	`*-special.example.com`
	`www.*example.com`
	`example.com`

Malware Hash	The file contains one hash per line in the format `<hex hash> [optional hash description]`. Each line supports MD5, SHA1, and SHA256 hex hashes. It is automatically used for virus outbreak prevention on antivirus profiles with `external-blocklist` enabled.
	Note: For optimal performance, do not mix different hashes in the list. Only use one of MD5, SHA1, or SHA256.
	Example:
	`292b2e6bb027cd4ff4d24e338f5c48de`
	`dda37961870ce079defbf185eeeef905 Trojan-Ransom.Win32.Locky.abfl`
	`3fa86717650a17d075d856a41b3874265f8e9eab Trojan-Ransom.Win32.Locky.abfl`
	`c35f705df9e475305c0984b05991d444450809c35dd1d96106bb8e7128b9082f`
	`Trojan-Ransom.Win32.Locky.abfl`
	See External malware block list on page 1098 for an example.

External resources file format

File format requirements for a HTTP/HTTPS external resources file:

- The file is in plain text format with each URL list, IP address, domain name, or malware hash occupying one line.
- The file is limited to 10 MB or 128 × 1024 (131072) entries, whichever limit is hit first.
- The entry limit also follows the table size limitation defined by CMDB per model.
- The external resources update period can be set to 1 minute, hourly, daily, weekly, or monthly (43200 min, 30 days).
- The external resources type as category (URL list) and domain (domain name list) share the category number range 192 to 221 (total of 30 categories).
- There is no duplicated entry validation for the external resources file (entry inside each file or inside different files).
- If the number of entries exceed the limit, a warning is displayed. Additional entries beyond the threshold will not be loaded.

For domain name list (type = domain):

- Simple wildcards are allowed in the domain name list, for example: *.test.com.
- IDN (international domain name) is supported.

For IP address list (type = address):

- The IP address can be a single IP address, subnet address, or address range. For example, 192.168.1.1, 192.168.10.0/24, or 192.168.100.1-192.168.100.254.
- The address can be an IPv4 or IPv6 address. An IPv6 address does not need to be in [ ] format.

For URL list (type=category):

- The scheme is optional, and will be truncated if found; https:// and http:// are not required.
- Wildcards are allowed at the beginning or end or the URL, for example: *.domain.com or domain.com.*.
- IDN and UTF encoding URL are supported .
- The URL can be an IPv4 or IPv6 address. An IPv6 URL must be in [ ] format.

To determine the external resource table size limit for your device:

```
# print tablesize
...
system.external-resource: 0 256 512
...
```

For this device, a FortiGate 60E, the global limit is 512 and the limit per VDOM is 256.

Create a threat feed

To create a threat feed in the GUI:

1. Go to *Security Fabric* > *External Connectors*.
2. Click *Create New*.
3. In the *Thread Feeds* section, click on the required feed type.
4. Configure the connector settings:

Name	Enter a name for the threat feed connector.
URI of external resource	Enter the link to the external resource file. HTTP, HTTPS, and STIX protocols are supported.
HTTP basic authentication	Enable/disable basic HTTP authentication. When enabled, enter the username and password in the requisite fields.
Refresh Rate	The time interval to refresh the external resource, in minutes (1 - 43200, default = 5). The applicable threat feed will be triggered to refresh between 0 minutes and the configured value. When the refresh is triggered, if another task is being processed be the schedule worker, the refresh task will be added to the queue.
Comments	Optionally, enter a description of the connector.
Status	Enable/disable the connector.

5. Click *OK*.

To create a threat feed in the CLI:

```
config system external-resource
    edit <name>
        set status {enable | disable}
        set type {category | address | domain | malware}
        set category <integer>
        set username <string>
        set password <string>
        set comments <string>
        *set resource <resource-uri>
        set user-agent <string>
        *set refresh-rate <integer>
        set source-ip <ip address>
        set interface-select-method {auto | sdwan | specify}
    next
end
```

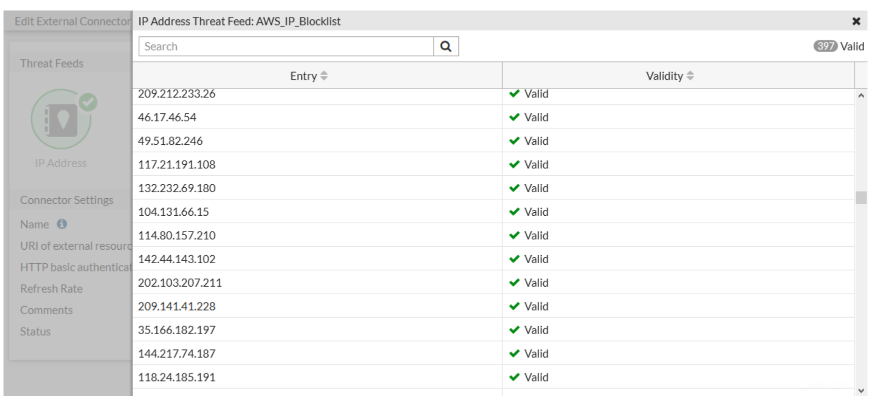
Parameters marked with an asterisk (*) are mandatory and must be filled in. Other parameters either have default values or are optional.

When multi VDOM mode is enabled, threat feed external connectors can be defined in the global VDOM or within a VDOM. See Threat feed connectors per VDOM on page 2515 for example configurations.

Update history

To review the update history of a threat feed, go to *Security Fabric > External Connectors*, select a feed, and click *Edit*. The *Last Update* field shows the date and time that the feed was last updated.

Click *View Entries* to view the current entries in the list.



EMS threat feed

A FortiGate can pull malware threat feeds from FortiClient EMS, which in turn receives malware hashes detected by FortiClients. The malware hash can be used in an antivirus profile when AV scanning is enabled with block or monitor actions. See Malware threat feed from EMS on page 1101 for an example.

External blocklist policy

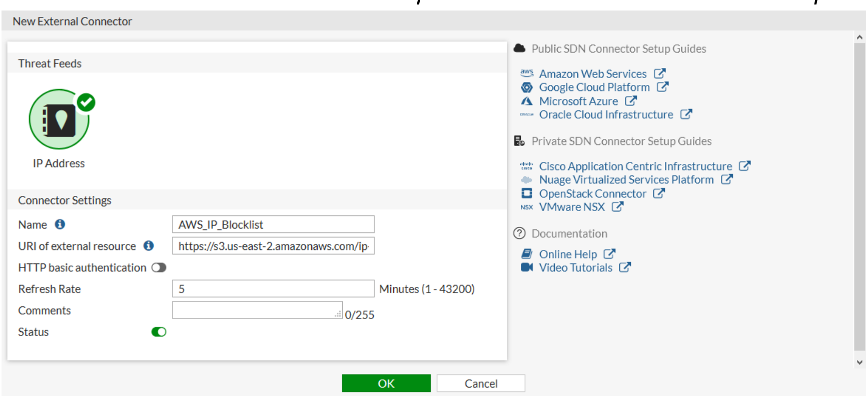
You can use the external blocklist (threat feed) for web filtering, DNS, and in firewall policies.

Sample configuration

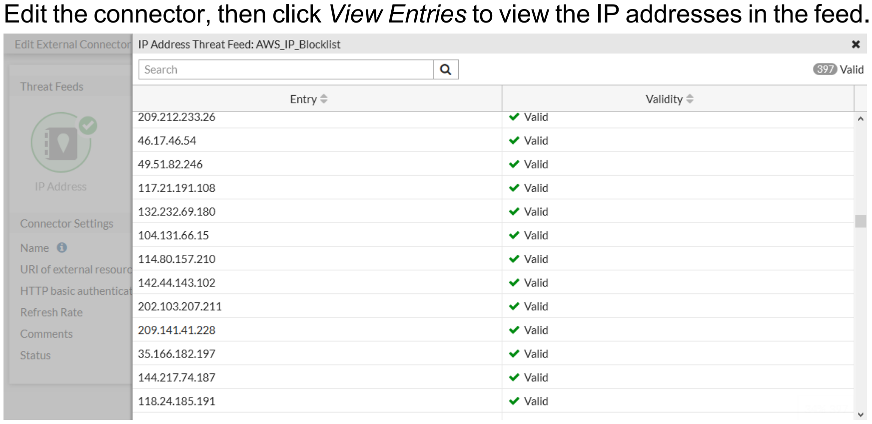
In this example, an IP address blocklist connector is created so that it can be used in a firewall policy.

To configure an external block list connector in the GUI:

1. Go to *Security Fabric > External Connectors* and click *Create New*.
2. In the *Threat Feeds* section, click *IP Address*.
3. Set *Name* to *AWS_IP_Blocklist*.
4. Set the *URI of external resource* to *https://s3.us-east-2.amazonaws.com/ip-blocklist/ip.txt*.



5. Configure the remaining settings as required, then click *OK*.
6. Edit the connector, then click *View Entries* to view the IP addresses in the feed.



The blocklist can now be used in web filter and DNS profiles, and in firewall policies.

To configure an external block list connector in the CLI:

```
config system external-resource
   edit "AWS_IP_Blocklist"
      set status enable
      set type address
      set username ' '
      set password *********
```

```
        set comments ''
        set resource "https://s3.us-east-2.amazonaws.com/ip-blocklist/ip.txt"
        set refresh-rate 15
    next
end
```

To apply an external block list to a firewall policy in the CLI:

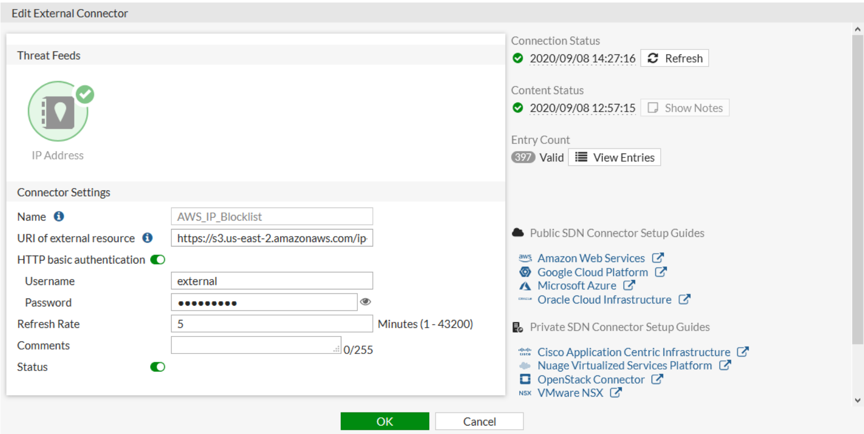
```
config firewall policy
    edit 1
        set name "policyid-1"
        set srcintf "wan2"
        set dstintf "wan1"
        set srcaddr "all"
        set dstaddr "AWS_IP_Blocklist"
        set action accept
        set schedule "always"
        set service "ALL"
        set logtraffic all
        set auto-asic-offload disable
        set nat enable
    next
end
```

External blocklist authentication

Thread feed external connectors support username and password authentication.

To enable username and password authentication in a thread feed connector:

1. Go to *Security Fabric > External Connectors*.
2. Edit an existing *Threat Feed* or create a new one by selecting *Create New*.
3. Enable *HTTP basic authentication*
4. Enter the *Username* and *Password*.



5. Click *OK*.

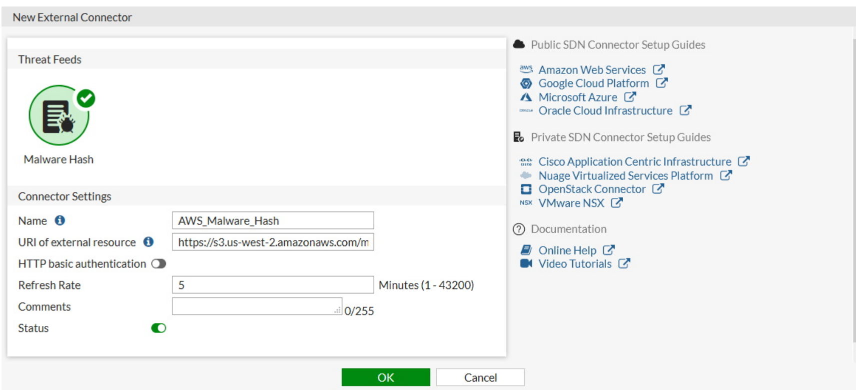
External blocklist file hashes

The malware hash threat feed connector supports a list of file hashes that can be used as part of virus outbreak prevention.

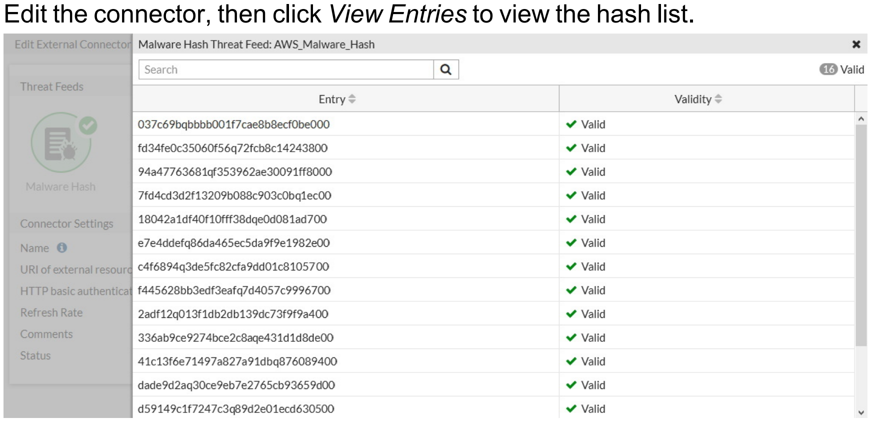
This example retrieves a malware hash from an Amazon S3 bucket, and then enables malware block lists in a antivirus profile.

To configure a malware hash connector in the GUI:

1. Go to *Security Fabric > External Connectors* and click *Create New*.
2. In the *Threat Feeds* section, click *Malware Hash*.
3. Set *Name* to *AWS_Malware_Hash*.
4. Set the *URI of external resource* to *https://s3.us-west-2.amazonaws.com/malware-hash-feeds/fortinet-malware-hash-list*.



5. Click *OK*.
6. Edit the connector, then click *View Entries* to view the hash list.



7. Go to *Security Profiles > AntiVirus* and create a new profile, or edit an existing one.
8. Enable *Use external malware block list*.
9. Click the + and select *AWS_Malware_Hash* from the list.
10. Click *OK*.

To configure a malware hash connector in the CLI:

```
config system external-resource
    edit "AWS_Malware_Hash"
```

```
        set type malware
        set resource "https://s3.us-west-2.amazonaws.com/malware-hash-feeds/fortinet-
malware-hash-list"
    next
end
```

```
config antivirus profile
    edit "av-profile"
            set external-blocklist-enable-all disable
            set external-blocklist "AWS_Malware_Hash"
        end
    next
end
```

Logs

The `filehash` and `filehashsrc` are included in outbreak prevention detection event logs.

This example shows the log generated when a file is detected by external malware hash list outbreak prevention:

```
1: date=2018-07-30 time=13:59:41 logid="0207008212" type="utm" subtype="virus"
eventtype="malware-list" level="warning" vd="root" eventtime=1532984381 msg="Blocked by
local malware list." action="blocked" service="HTTP" sessionid=174963 srcip=192.168.101.20
dstip=172.16.67.148 srcport=37045 dstport=80 srcintf="lan" srcintfrole="lan" dstintf="wan1"
dstintfrole="wan" policyid=1 proto=6 direction="incoming" filename="mhash_block.com"
checksum="90f0cb57" quarskip="No-skip" virus="mhash_block.com" dtype="File Hash"
filehash="93bdd30bd381b018b9d1b89e8e6d8753" filehashsrc="test_list"
url="http://172.16.67.148/mhash_block.com" profile="mhash_test" agent="Firefox/43.0"
analyticssubmit="false"
```

External resources for DNS filter

External resources provides the ability to dynamically import an external block list into an HTTP server. This feature enables the FortiGate to retrieve a dynamic URL, domain name, IP address, or malware hash list from an external HTTP server periodically. The FortiGate uses these external resources as the web filter's remote categories, DNS filter's remote categories, policy address objects, or antivirus profile's malware definitions. If external resources are updated, FortiGate objects are also updated dynamically.

External resource is divided into four types:

- URL list (type = category)
- Domain name list (type = domain)
- IP address list (type = address)
- Malware hash list (type = malware)

Remote categories and external IP block list

The DNS filter profile can use two types of external resources: *domain type* (domain name list) and *address type* (IP address list).

When a *domain type* external resource is configured, it is treated as a remote category in the DNS filter profile. If the domain name in DNS query matches the entry in this external resource file, it is treated as the remote category and follows the action configured for this category in DNS filter profile.

When an *address type* external resource is configured, it can be enabled as *external-ip-blocklist* in DNS filter profile. If a DNS resolved IP address in DNS response matches the entry in the *external-ip-blocklist*, this DNS query is blocked by the DNS filter.

For external resources file format and limits, see External resources file format on page 2505.

Configuring external resources in the CLI

In the CLI, you can configure external resources files in an external HTTP server. Under global, configure the external resources file location and specify the resource type.

To configure external resources:

```
config system external-resource
   edit "Ext-Resource-Type-as-Domain-1"
      set type domain
      set category 194
      set resource "http://172.16.200.66/external-resources/Ext-Resource-Type-as-Domain-
1.txt"
      set refresh-rate 1
   next
   edit "Ext-Resource-Type-as-Address-1"
      set status enable
      set type address
      set username ' '
      set password *********
      set comments ''
      set resource "http://172.16.200.66/external-resources/Ext-Resource-Type-as-Address-
1.txt"
      set refresh-rate 1
   next
end
```

In each VDOM, the domain type external resource can be used in the DNS filter as remote category. In this example, the domain name list in the Ext-Resource-Type-as-Domain-1.txt file is treated as a remote category (category ID 194). The IP address list in the Ext-Resource-Type-as-Address-1.txt file can be applied in the DNS filter as an `external-ip-blocklist`. If the DNS resolved IP address matches any entry in the list in that file, the DNS query is blocked.

To configure the external IP block list and apply it to a policy:

```
config dnsfilter profile
   edit "default"
      set comment "Default dns filtering."
      config ftgd-dns
         config filters
            edit 1
               set category 194
               set action block
            next
            edit 2
               set category 12
            next
            edit 3
            next
```

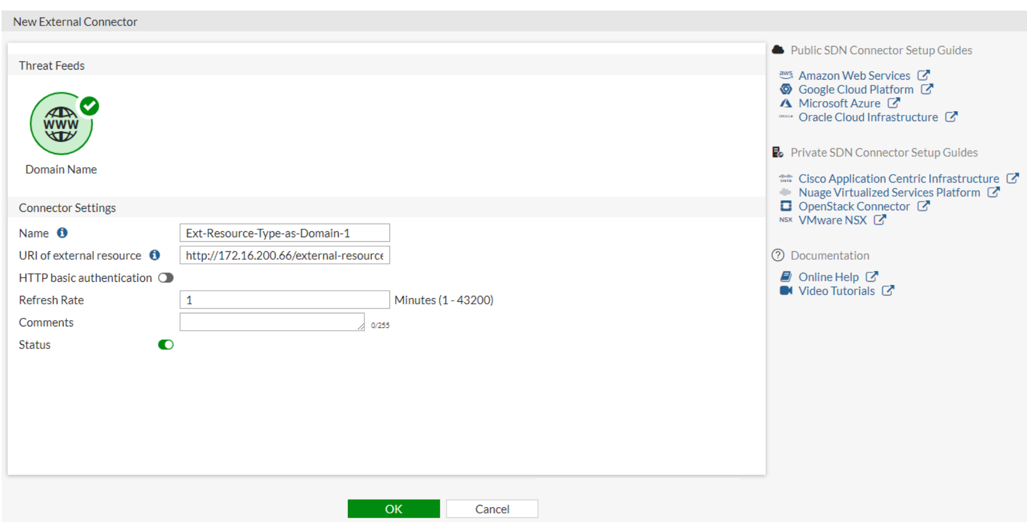
```
            end
        end
        set block-botnet enable
        set external-ip-blocklist "Ext-Resource-Type-as-Address-1"
    next
end
```

```
config firewall policy
    edit 1
        set name "DNSFilter"
        set srcintf "port10"
        set dstintf "port9"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set logtraffic all
        set dnsfilter-profile "default"
        set profile-protocol-options "protocol"
        set ssl-ssh-profile "protocols"
        set nat enable
    next
end
```

Configuring external resources in the GUI

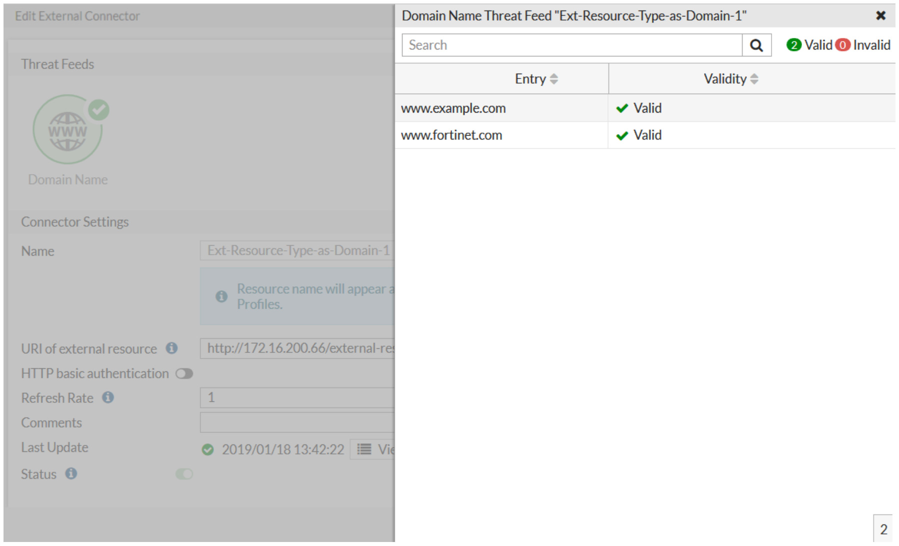
To configure, edit, or view the entries for external resources in the GUI:

1. Go to *Global > Security Fabric > External Connectors*.
2. Click *Create New*.
3. In the *Threat Feeds* section, select *Domain Name* or *IP Address*.
4. Enter the Resource *Name*, URL, location of the resource file, resource authentication credentials, and *Refresh Rate*.

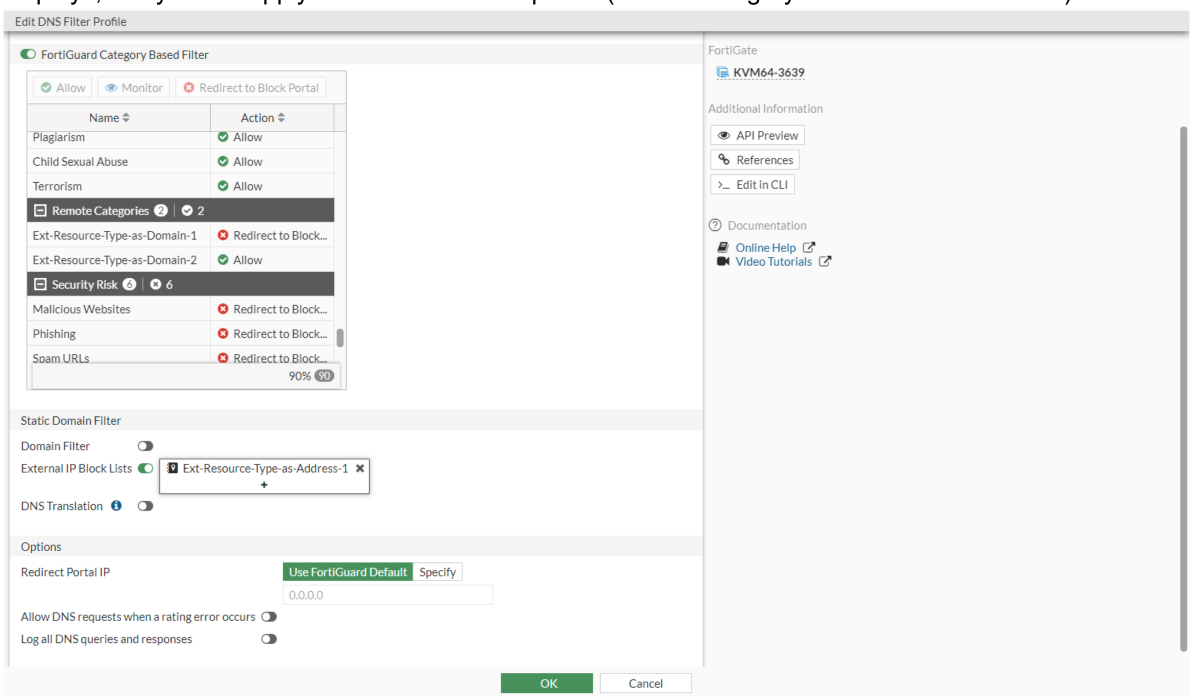


5. Click *OK*.
6. Double-click the *Threat Feeds Object* you just configured to open the *Edit* page.

7. Click *View Entries* to view the entry list in the external resources file.



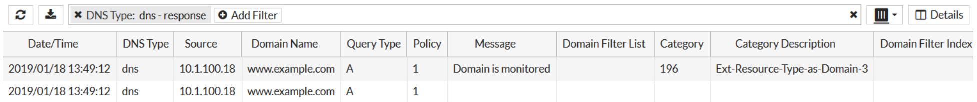
8. Go to *VDOM* > *Security Profiles* > *DNS Filter* and open a DNS filter profile. The configured external resources displays, and you can apply it in each DNS filter profile (remote category or external IP block lists).



Log sample

Remote categories

Go to *VDOM* > *Log & Report* > *Security Events* and select the *DNS Query* card. Some domains that match the remote category list are rated as remote category, overriding their original domain rating.



DNS Type: dns - response | Add Filter | Details

Date/Time	DNS Type	Source	Domain Name	Query Type	Policy	Message	Domain Filter List	Category	Category Description	Domain Filter Index
2019/01/18 13:49:12	dns	10.1.100.18	www.example.com	A	1	Domain is monitored		196	Ext-Resource-Type-as-Domain-3	
2019/01/18 13:49:12	dns	10.1.100.18	www.example.com	A	1					

Log example:

```
1: date=2019-01-18 time=13:49:12 logid="1501054802" type="utm" subtype="dns" eventtype="dns-
response" level="notice" vd="vdom1" eventtime=1547848151 policyid=1 sessionid=82998
srcip=10.1.100.18 srcport=42985 srcintf="port10" srcintfrole="undefined" dstip=172.16.95.16
dstport=53 dstintf="port9" dstintfrole="undefined" proto=17 profile="default" xid=38234
qname="www.example.com" qtype="A" qtypeval=1 qclass="IN" ipaddr="93.184.216.34" msg="Domain
is monitored" action="pass" cat=196 catdesc="Ext-Resource-Type-as-Domain-3"

2: date=2019-01-18 time=13:49:12 logid="1500054000" type="utm" subtype="dns" eventtype="dns-
query" level="information" vd="vdom1" eventtime=1547848151 policyid=1 sessionid=82998
srcip=10.1.100.18 srcport=42985 srcintf="port10" srcintfrole="undefined" dstip=172.16.95.16
dstport=53 dstintf="port9" dstintfrole="undefined" proto=17 profile="default" xid=38234
qname="www.example.com" qtype="A" qtypeval=1 qclass="IN"
```

External IP block lists

Go to *VDOM* > *Log & Report* > *Security Events* and select the *DNS Query* card. If the DNS query resolved IP address matches the entry in the `external-ip-blocklist`, the DNS query is blocked.

DNS Type: dns - response | Add Filter | Details

Date/Time	DNS Type	Source	Domain Name	Query Type	Policy	Message	Domain Filter List	Category	Categoı
2019/01/18 13:48:08	dns	10.1.100.18	www.example.com	A	1	Domain was blocked because it is in the domain-filter list	Ext-Resource-Type-as-Address-1		
2019/01/18 13:48:08	dns	10.1.100.18	www.example.com	A	1				

Log example:

```
1: date=2019-01-18 time=13:50:53 logid="1501054400" type="utm" subtype="dns" eventtype="dns-
response" level="warning" vd="vdom1" eventtime=1547848253 policyid=1 sessionid=83206
srcip=10.1.100.18 srcport=47281 srcintf="port10" srcintfrole="undefined" dstip=172.16.95.16
dstport=53 dstintf="port9" dstintfrole="undefined" proto=17 profile="default" xid=7501
qname="www.example.com" qtype="A" qtypeval=1 qclass="IN" msg="Domain was blocked because it
is in the domain-filter list" action="redirect" domainfilteridx=0 domainfilterlist="Ext-
Resource-Type-as-Address-1"

2: date=2019-01-18 time=13:50:53 logid="1500054000" type="utm" subtype="dns" eventtype="dns-
query" level="information" vd="vdom1" eventtime=1547848253 policyid=1 sessionid=83206
srcip=10.1.100.18 srcport=47281 srcintf="port10" srcintfrole="undefined" dstip=172.16.95.16
dstport=53 dstintf="port9" dstintfrole="undefined" proto=17 profile="default" xid=7501
qname="www.example.com" qtype="A" qtypeval=1 qclass="IN"
```

Threat feed connectors per VDOM

When multi-VDOM mode is enabled, a threat feed external connector can be defined in global or within a VDOM. Global threat feeds can be used in any VDOM, but cannot be edited within the VDOM. FortiGuard category and domain name-based external feeds have an added category number field to identify the threat feed. The threat feed name in global must start with `g-`. Threat feed names in VDOMs cannot start with `g-`.

FortiGuard category and domain name-based external feed entries must have a number assigned to them that ranges from 192 to 221. This number can be assigned to both external feed types. However, when a category number is used under a global entry, such as 192 with the name `g-cat-192`, this category number cannot be used in any other global or VDOM entries. If a category is used under a VDOM entry, such as 192 under VDOM1 with the name `cat-192`, the category 192 can be used in another VDOM or root with the name `cat-192`.

A thread feed connector can only be used in profiles in the VDOM that it was created in. Global connectors can be used in all VDOMs.

Each VDOM can have a maximum of 256 thread feed entries. But in total, a FortiGate can only have 511 thread feed entries.

To configure an external threat feed connector under global in the GUI:

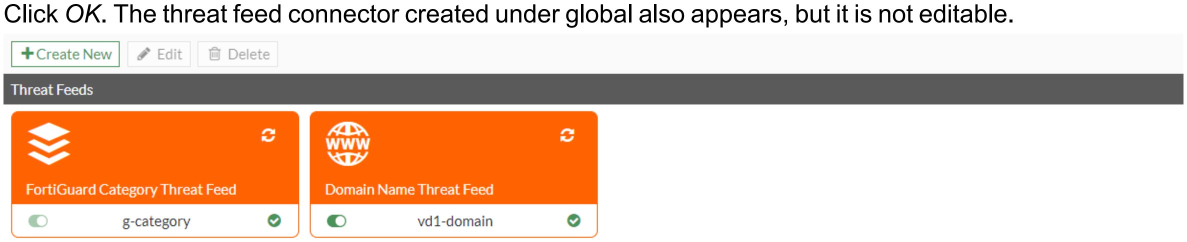
1. Go to *Security Fabric > External Connectors* and click *Create New*.
2. In the *Threat Feeds* section, click *FortiGuard Category*.
3. Enter a name that begins with `g-`.
4. Configure the other settings as needed.
5. Click *OK*.

To configure an external threat feed connector under global in the CLI:

```
config global
    config system external-resource
        edit "g-category"
            set status enable
            set type category
            set category 192
            set comments ''
            set resource "http://172.16.200.55/external-resource-test/513-FDGCategory.txt"
            set refresh-rate 5
        next
    end
end
```

To configure an external threat feed connector under a VDOM in the GUI:

1. Go to *Security Fabric > External Connectors* and click *Create New*.
2. In the *Threat Feeds* section, click *Domain Name*.
3. Enter a name that does not begin with `g-`.
4. Configure the other settings as needed.
5. Click *OK*. The threat feed connector created under global also appears, but it is not editable.

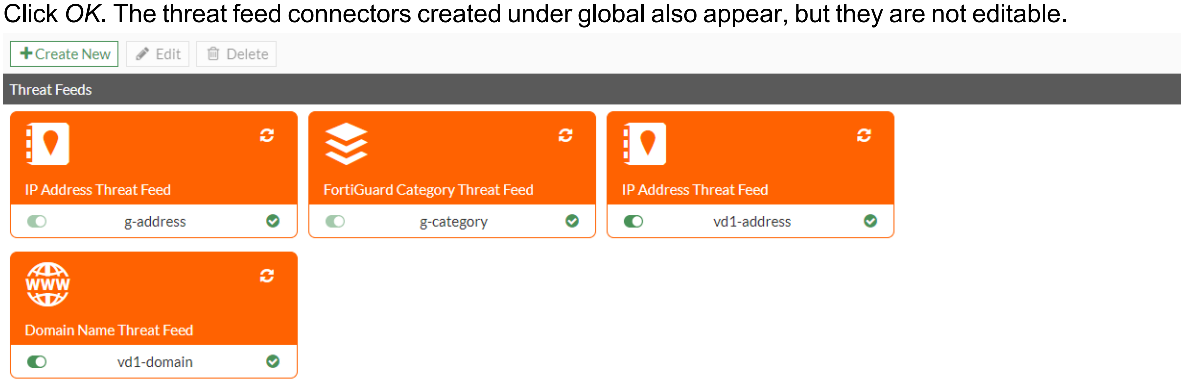


To configure an external threat feed connector under a VDOM in the CLI:

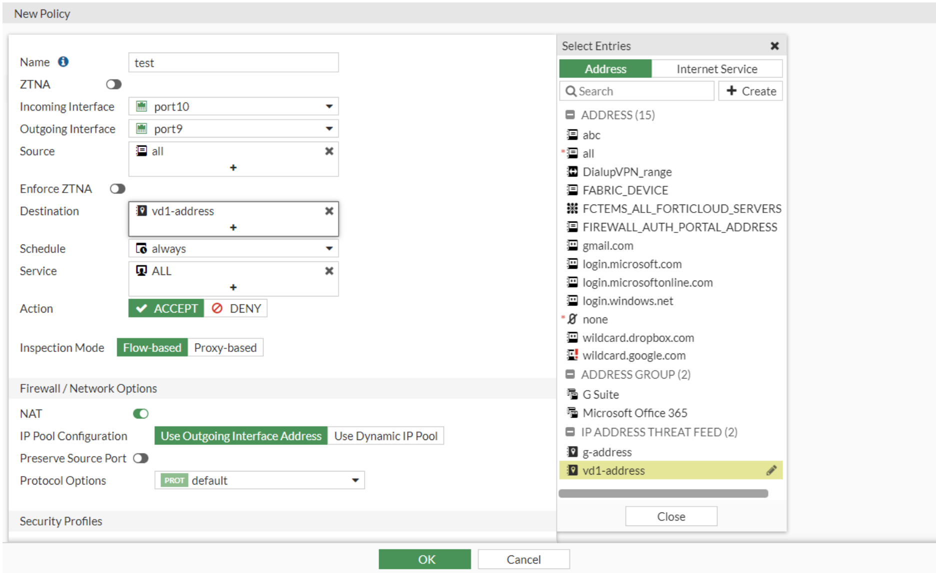
```
config vdom
    edit vd1
        config system external-resource
            edit "vd1-domain"
                set status enable
                set type domain
                set category 193
                set comments ''
                set resource "http://172.16.200.55/external-resource-test/513-Domain.txt"
                set refresh-rate 5
            next
        end
    next
end
```

To use an IP address threat feed in a policy in the GUI:

1. Configure an IP address connector in global:
   a. Go to *Security Fabric > External Connectors* and click *Create New*.
   b. In the *Threat Feeds* section, click *IP Address*.
   c. Enter a name that begins with `g-`.
   d. Configure the other settings as needed.
   e. Click *OK*.
2. Configure an IP address connector in the VDOM (vd1):
   a. Go to *Security Fabric > External Connectors* and click *Create New*.
   b. In the *Threat Feeds* section, click *IP Address*.
   c. Enter a name that does not begin with `g-`.
   d. Configure the other settings as needed.
   e. Click *OK*. The threat feed connectors created under global also appear, but they are not editable.



3. Configure the firewall policy in the VDOM (vd1):
   a. Go to *Policy & Objects > Firewall Policy* and click *Create New*.
   b. For *Destination*, select *vd1-address*. Since this policy is configured under vd1, *g-address* can also be set as the destination.



c. Configure the other settings as needed.

d. Click *OK*.

To use an IP address threat feed in a policy in the CLI:

1. Configure the IP address connectors:

```
config global
    config system external-resource
        edit "g-address"
            set status enable
            set type address
            set username ''
            set comments ''
            set resource "http://172.16.200.55/external-resource-test/513-IP.txt"
            set refresh-rate 5
        next
    end
end

config vdom
    edit vd1
        config system external-resource
            edit "vd1-address"
                set status enable
                set type address
                set comments ''
                set resource "http://172.16.200.55/external-resource-test/513-IP.txt"
                set user-agent "curl/7.58.0"
                set refresh-rate 5
            next
        end
    next
end
```

2. In the VDOM, configure a firewall policy with the external address as the destination address:

```
config vdom
    edit vd1
        config firewall policy
            edit 1
                set name "test"
                set srcintf "port10"
                set dstintf "port9"
                set srcaddr "all"
                set dstaddr "vd1-address"
                set action accept
                set schedule "always"
                set service "ALL"
                set profile-protocol-options "protocol"
                set nat enable
            next
        end
    next
end
```

Since this firewall policy is configured under `vd1`, `g-address` can also be set as the `dstaddr`.

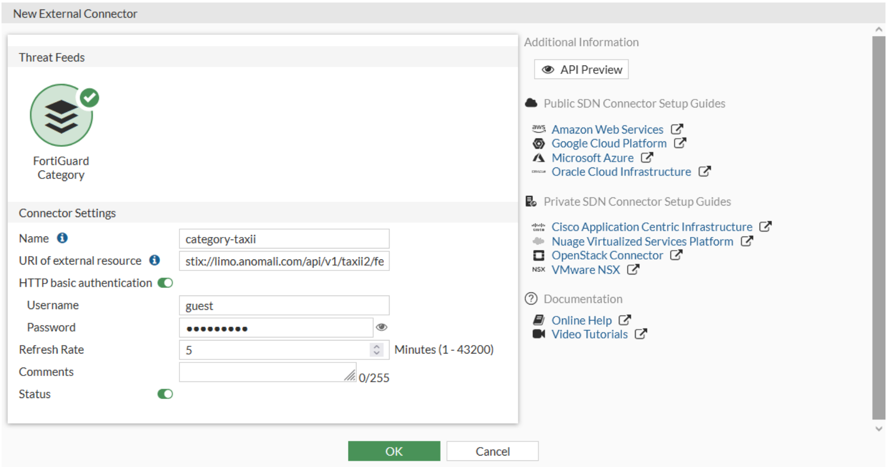
STIX format for external threat feeds

The FortiGate's external threat feeds support feeds that are in the STIX/TAXII format. Use the `stix://` prefix in the URI to denote the protocol.

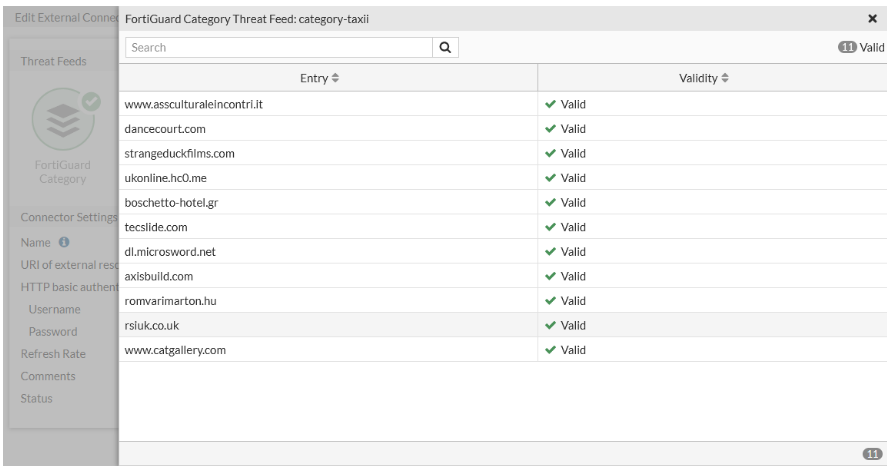
All external threat feeds support the STIX format. In this example, a FortiGuard Category threat feed in the STIX format is configured.

To configure a FortiGuard Category threat feed in the STIX format in the GUI:

1. Go to *Security Fabric* > *External Connectors* and click *Create New*.
2. Select *FortiGuard Category* from the *Threat Feeds* section.
3. Configure the connector:
   - *Name*: *category-taxii*
   - *URI of external resource*: *stix://limo.anomali.com/api/v1/taxii2/feeds/collections/200/objects/*
   - *HTTP basic authentication*: Enable and enter the username and password, such as *guest* and *guest*.



4. Click *OK*.
5. Edit the connector, and click *View Entries* in the right side bar to view the retrieved entries.



To configure a FortiGuard Category threat feed in the STIX format in the CLI:

```
config system external-resource
    edit "category-taxii"
        set category 194
        set username "guest"
        set password guest
        set resource "stix://limo.anomali.com/api/v1/taxii2/feeds/collections/200/objects/"
    next
end
```

If the connector is used in webfilter that blocks category 194, the traffic that matches the retrieved URLs, such as *rsiuk.co.uk*, is blocked:

```
1: date=2021-10-06 time=18:07:46 eventtime=1633568867163763708 tz="-0700" logid="0316013056"
type="utm" subtype="webfilter" eventtype="ftgd_blk" level="warning" vd="vd1" policyid=1
sessionid=174974 srcip=10.1.100.12 srcport=48284 srcintf="port2" srcintfrole="undefined"
srcuuid="c6753ba2-231b-51ec-1675-090f2b5f1384" dstip=78.129.255.151 dstport=443
dstintf="port1" dstintfrole="undefined" dstuuid="c6753ba2-231b-51ec-1675-090f2b5f1384"
proto=6 service="HTTPS" hostname="rsiuk.co.uk" profile="test" action="blocked"
reqtype="direct" url="https://rsiuk.co.uk/" sentbyte=75 rcvdbyte=0 direction="outgoing"
```

```
msg="URL belongs to a denied category in policy" method="domain" cat=194 catdesc="category-taxii"
```

Monitoring the Security Fabric using FortiExplorer for Apple TV

FortiExplorer for Apple TV allows you to use a TV screen to monitor your entire Security Fabric.

FortiExplorer for Apple TV is an analysis tool that provides easy to use NOC and SOC monitoring capabilities. The app features real-time data traffic, visual alerts, as well as a general overview of hardware devices, operating systems, and interfaces. The monitor also provides a wireless health summary of your entire network across multiple buildings. If an access point goes offline, you will be notified about the network's health. After the issues are resolved, you will immediately see the health update on your screen.

Getting started with FortiExplorer for Apple TV

Download FortiExplorer for Apple TV from the app store on Apple TV. After the app is installed, add devices using the Apple TV remote or by sharing a login profile with FortiExplorer. Once the devices are added, you can use FortiExplorer for Apple TV to view real-time data in the Network Operations Center, Security Operations Center, and Software-Defined Branch.

To get started with FortiExplorer for Apple TV:

1. Download the app and add devices to FortiExplorer for Apple TV.
   You can add devices by sharing a login profile with FortiExplorer or logging into the device directly on FortiExplorer for Apple TV.
2. View the physical topology of the Fabric to identify risks
3. View the Fabric components as seen on the root FortiGate.

4. View an executive summary of the three largest areas of security focus in the Security Fabric.
5. View data collected by FortiAnalyzer on the endpoints on your network.
6. View vulnerability data collected by FortiClient EMS.
7. Use the Software-Defined Branch module to monitor interface SD-WAN usage and associated service level agreements.

NOC and SOC example

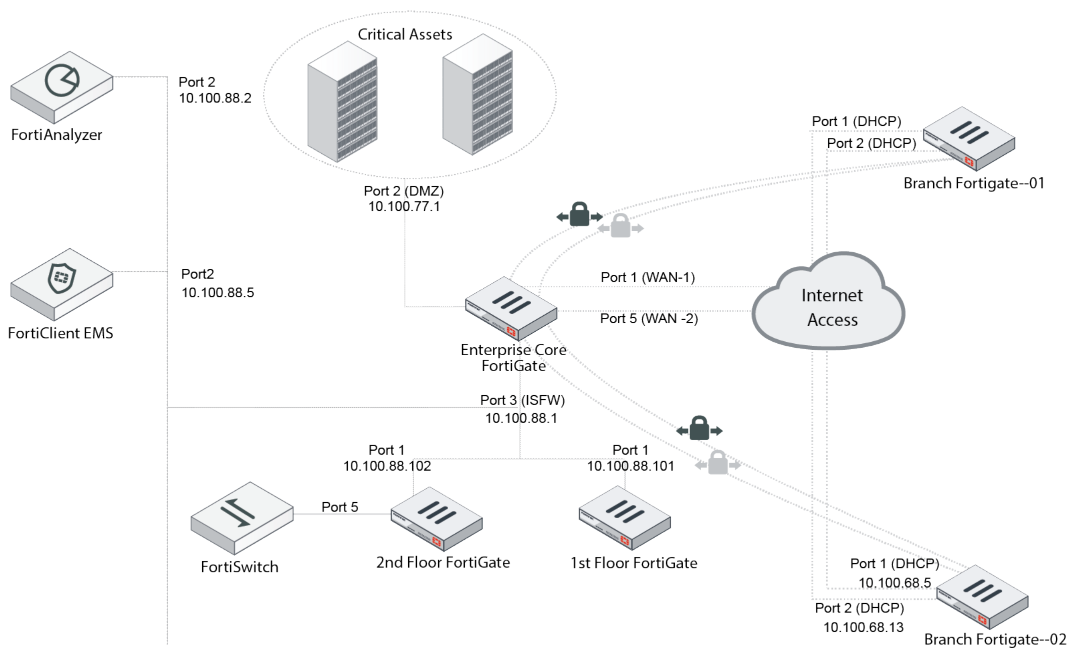
In this example, you have configured your FortiGates, FortiAnalyzer and other devices in your Security Fabric. Now you want to use FortiExplorer for Apple TV to display the status of the devices on a TV in your Network Operation Center or Security Operation Center.

Topology

This topology has a Headquarter and two Branches. Within the Headquarter is the Enterprise Core and two FortiGates acting as ISFWs. In addition, an on-premise FortiAnalyzer collects all logging information from the fabric devices. The FortiClient EMS manages all the endpoints within the topology.

The two branches are configured with SD-WAN with VPN overlays to the Enterprise Core. Traffic is steered towards the overlays and underlays based on SD-WAN Rules.

Using FortiExplorer for Apple TV, you will be able to monitor the different components in this topology.



To take advantage of the views in the FortiExplorer for Apple TV, you should configure:

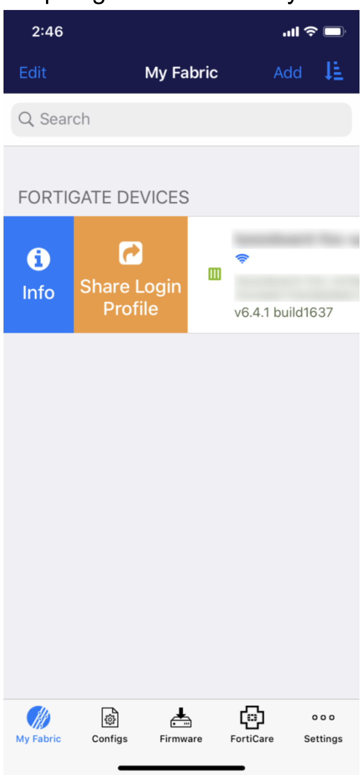
- Security Fabric on all FortiGates. See Configuring the root FortiGate and downstream FortiGates on page 2208.
- FortiAnalyzer Logging. See Configuring FortiAnalyzer on page 2215.
- FortiClient EMS. See Configuring FortiClient EMS on page 2234

Adding the root FortiGate to FortiExplorer for Apple TV

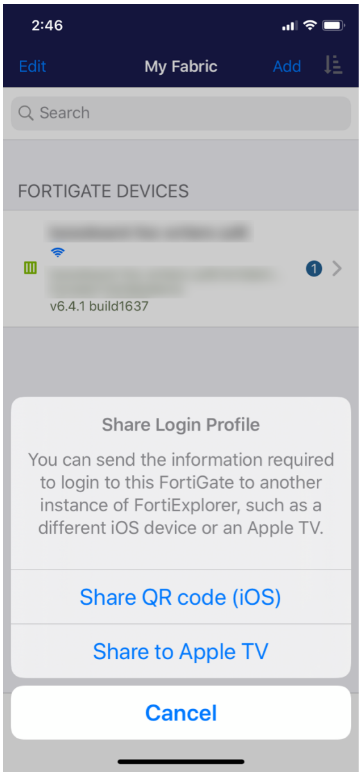
By adding the root FortiGate, you can view the entire topology and navigate to branch FortiGates in the SD-WAN view. If you are already using FortiExplorer on a mobile device, you can connect the same FortiGate device to Apple TV by sharing the login credentials on both devices. Alternatively, you can manually connect to your root FortiGate directly from the app.

To share login credentials between FortiExplorer and FortiExplorer for Apple TV:

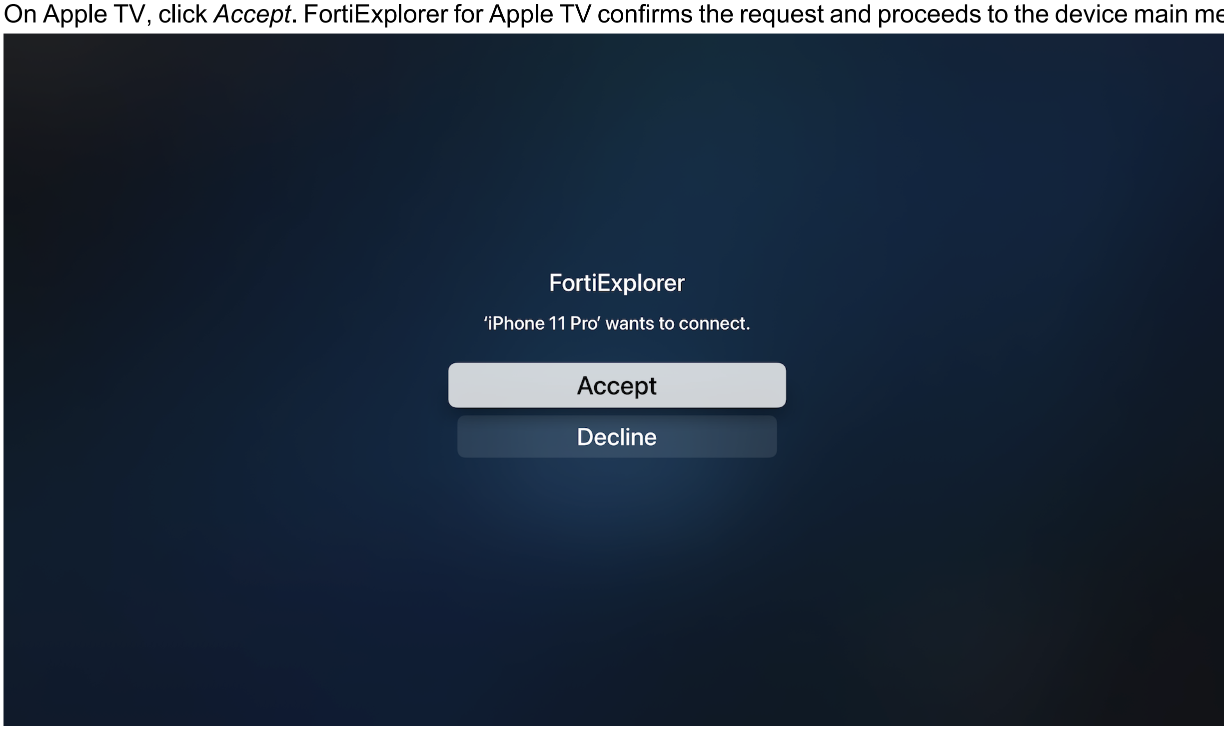
1. Connect the FortiExplorer and FortiExplorer for Apple TV devices to the same network.
2. On FortiExplorer for Apple TV, click *New FortiGate*.
3. In FortiExplorer, go to *My Fabric*.
4. Swipe right on the device you want to share, and tap *Share Login Profile*.



5. Tap *Share to Apple TV*.

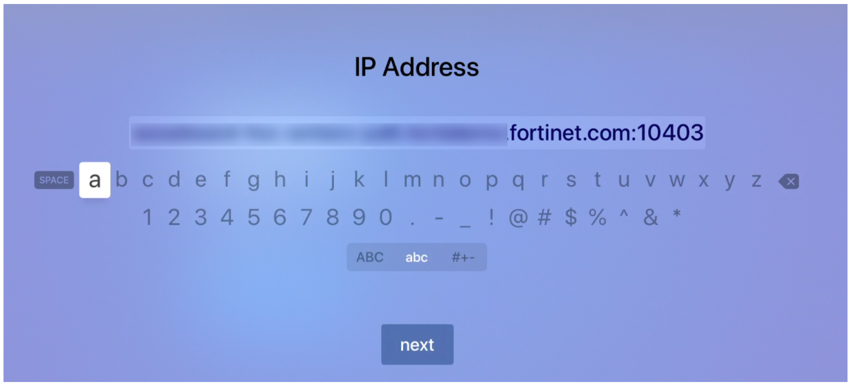


6. On Apple TV, click *Accept*. FortiExplorer for Apple TV confirms the request and proceeds to the device main menu.



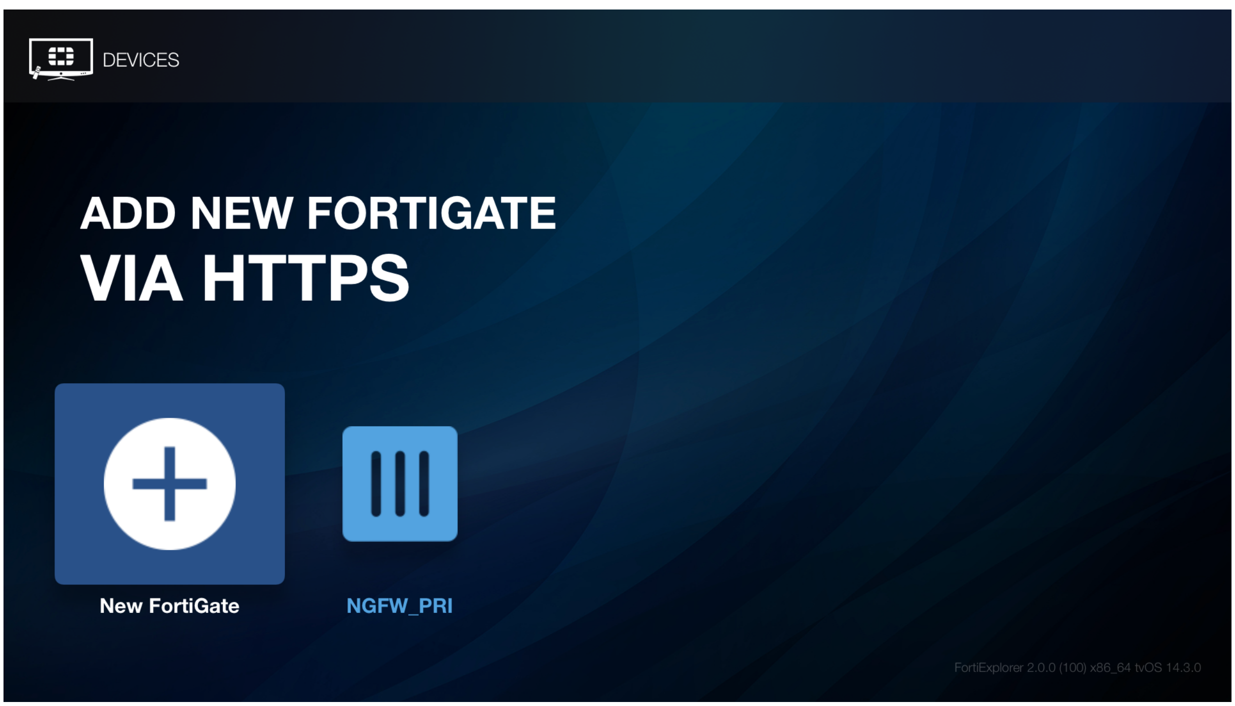
To add devices to FortiExplorer for Apple TV:

1. In the *Devices* menu, click *New FortiGate*. The *Login to FortiGate* dialog box is displayed.
2. In the *IP Address/Host Name* field, take one of the following actions:
   - Enter the device IP address and port, if not using the default admin port 443
   - Enter the full host name including the domain. Enter port if not using the default admin port 443.



3. Enter the *Username* and *Password* for the FortiGate device.
4. Click *Remember* to save time entering the login credentials later.
5. Click *Login*. The device is added to FortiExplorer for Apple TV.

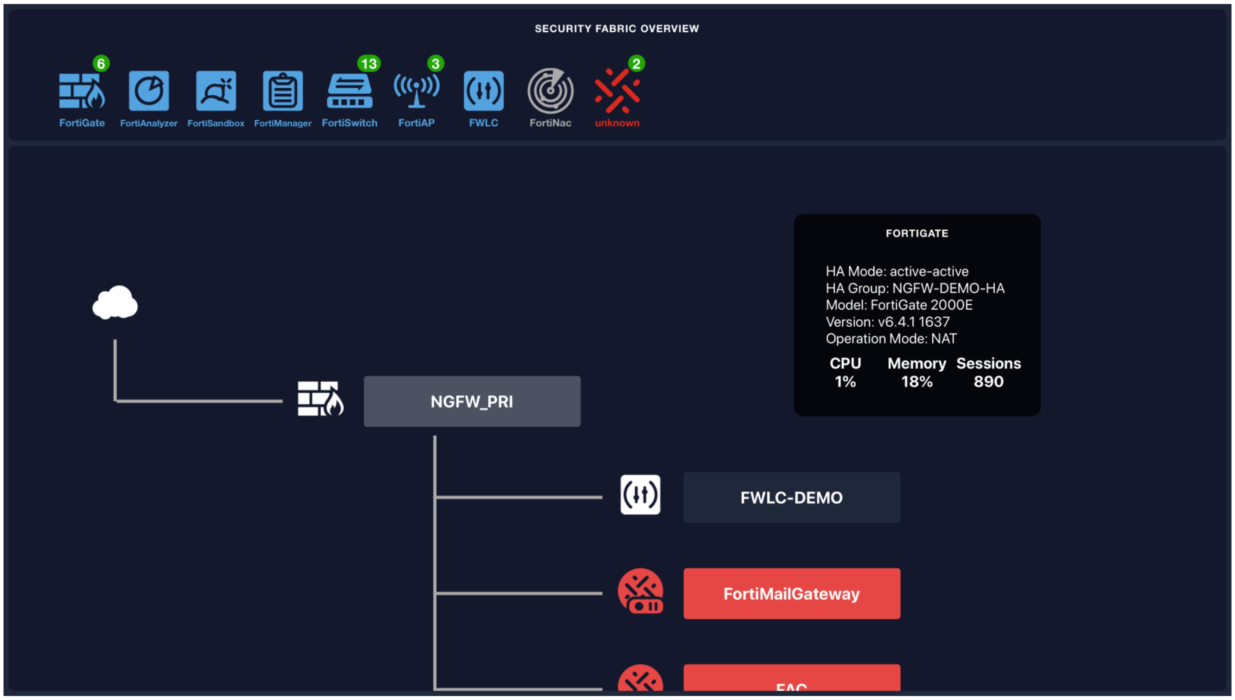
If the IP or hostname is not defined in the CN or SAN field of your certificate, you will receive a prompt that "Your connection is not private". You may choose to continue with your connection.



Viewing the Fabric Topology monitor

Use the Fabric Topology monitor to view the physical topology of the Fabric to identify risks. FortiGate devices with version 6.4. and above can drilldown further to see additional information for devices such as FortiGates, FortiAPs, and FortiSwitches.

To view the Fabric Topology monitor, go to *Network Operations Center > Fabric Topology*. This monitor displays the same information as the *Physical Topology* on the FortiGate



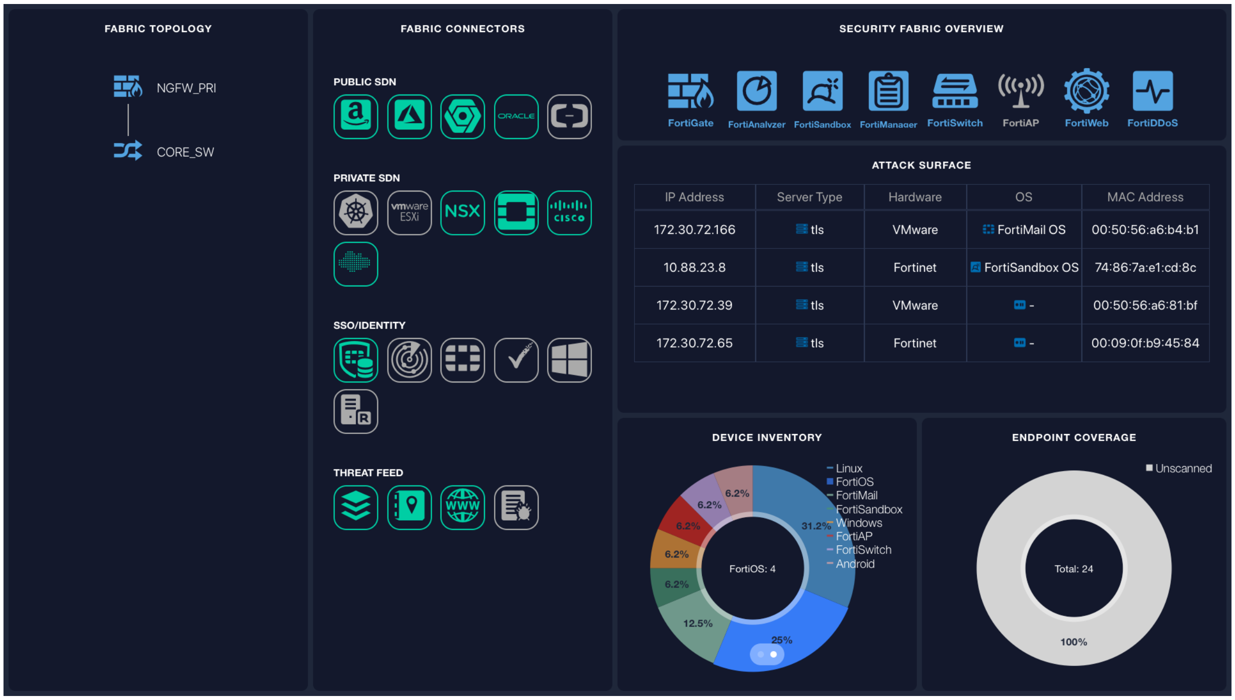
Use your remote to navigate through the devices in the Fabric topology. Click a device to view the drilldown information. To return to the default view, click the *Menu* button.

Viewing the Fabric Overview monitor

Use the Fabric Overview monitor to view the Fabric components as seen on the Dashboard of the Fabric Root FortiGate in the example topology. Each device must be authorized and be part of the Fabric.

For information about configuring the Security Fabric, see Fortinet Security Fabric on page 2204

To view the Fabric Overview monitor, go to *Network Operations Center > Fabric Overview*.



The Security Fabric monitor has multiple panes. To see data populated on the panes, ensure that proper configurations are applied on the Fabric devices:

Pane	Description	Configuration
Fabric Connectors	Displays external SDN connectors that are enabled.	Configure *Security Fabric* > *External Connectors*.
Security Fabric Overview	Displays the number of devices in the topology.	Configure *Security Fabric* > *Fabric Connectors*.
Attack Surface	Displays devices detected by the FortiGate with a server tag.	Ensure Device Detection is configured on the interfaces(s). Go to *Network* > *Interfaces*.
Device Inventory	Displays devices based on Hardware Vendor and detected OS	Ensure Device Detection is configured on the interface(s). Go to *Network* > *Interfaces*.
Endpoint Coverage	Displays the number of online devices and the percentage of Unscanned, Vulnerable, and Secured devices.	Ensure Device Detection is configured on the interface(s). Vulnerability scan results come from FortiClient EMS. Go to *Network* > *Interfaces*.

Device related information only corresponds to devices local to the FortiGate. Device information from downstream FortiGates do not propagate to the Upstream FortiGate.

Viewing the Security Rating monitor

The Security Rating monitor is separated into three major scorecards: *Security Posture*, *Fabric Coverage*, and *Optimization*, which provide an executive summary of the three largest areas of security focus in the Security Fabric.

To see the Security Rating summary, the root FortiGate and all FortiGates within the Fabric should have the proper FortiGuard Security Rating license. Security rating is performed on the root FortiGate. Its reports are generated periodically.

To view the Security Rating monitor, go to *Network Operations Center* > *Security Rating*.

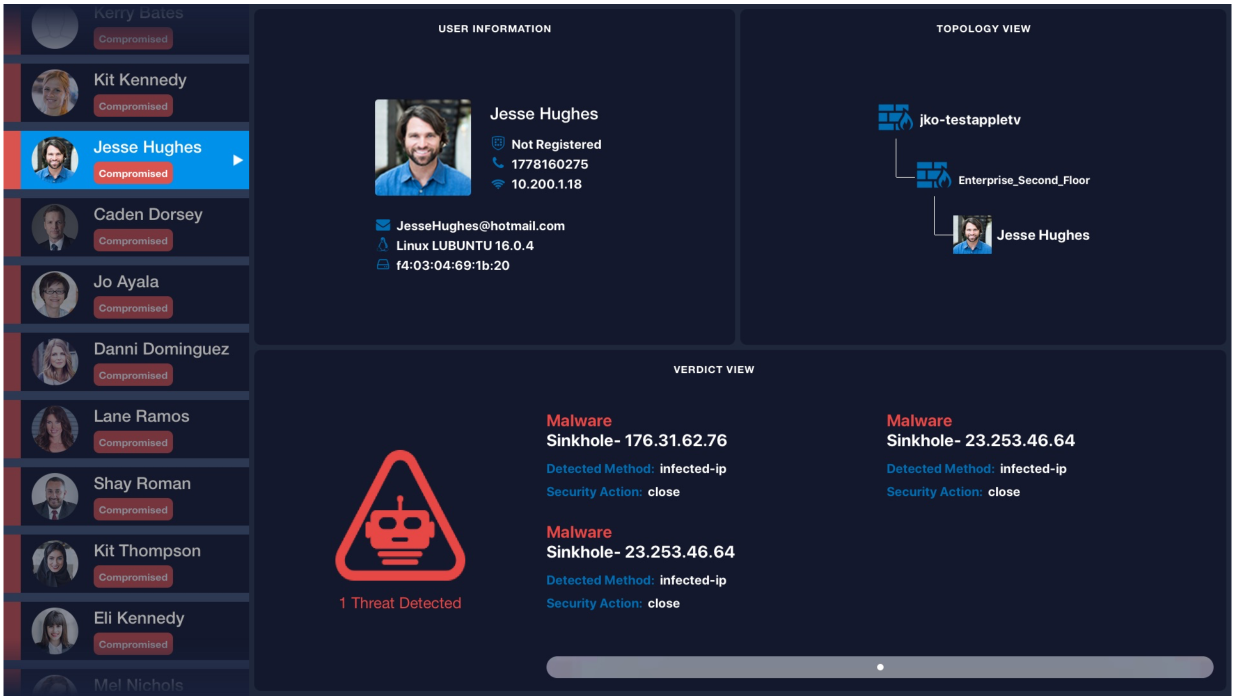
The scorecards show an overall score of the performance and sub-categories. The point score represents the net score for all passed and failed items in that area.

For more information about the Security Rating score, see Security Fabric score on page 2341.

Viewing the Compromised Hosts monitor

The Compromised Hosts monitor leverages the data collected by FortiAnalyzer on the endpoints on your network. To see compromised hosts, the FortiAnalyzer must have a FortiGuard Indicators of Compromise license. The IOC service helps identify compromised hosts based on infected websites that it may have visited.

This monitor captures the same information as seen on the *Compromised Hosts* monitor on the FortiGate.



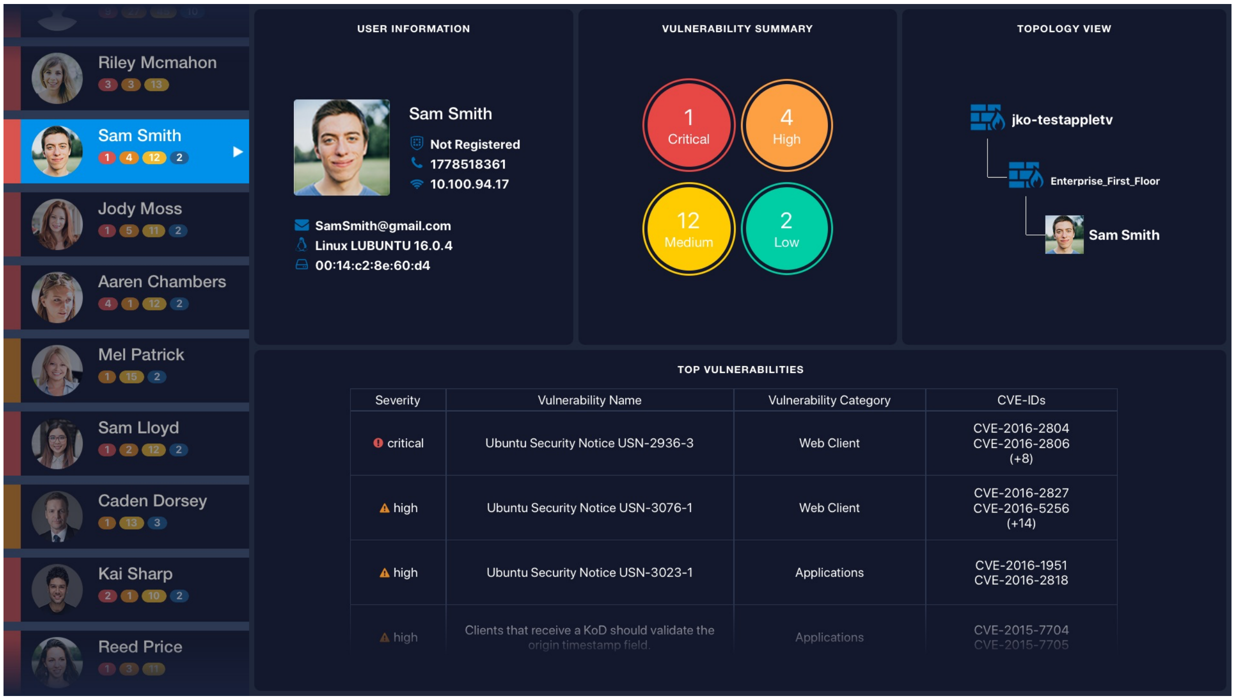
To view the Compromised Hosts monitor:

1. Go to *Security Operations Center* > *Compromised Hosts*.
2. In the left-hand pane, scroll through the user list. The monitor displays three panes:
   - The *User Information* pane displays the user's contact information and IP address.
   - The *Topology View* pane displays the user's location in the topology.
   - The *Verdict View* pane displays the *Malware*, *Detected Method*, and *Security Action*.

Viewing the Vulnerability Monitor

The Vulnerability Monitor obtains data from FortiClient EMS. It displays vulnerabilities detected by the FortiClient endpoint, categorized into Critical, High, Medium and Low risk. In this example, an on-premise FortiClient EMS is connected on the root FortiGate’s Fabric Connector.

This monitor captures the same information as seen on the *Top Vulnerable Endpoint Devices* monitor on the FortiGate.



To view the Vulnerability Monitor:

1. Go to *Security Operations Center* > *Vulnerability Monitor*. The monitor displays a user list and their vulnerabilities.
2. Use your remote to scroll through the user list. The vulnerability details are displayed on the right side of the monitor.

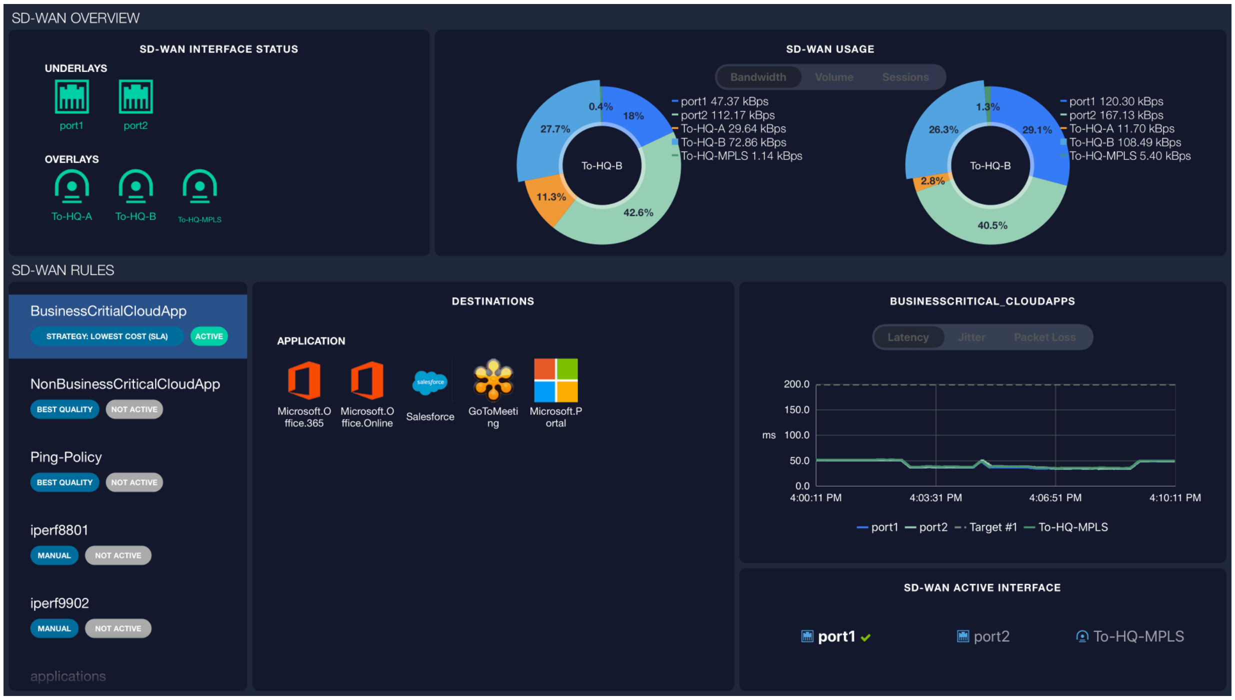
   - The *User Information* pane displays the user's contact details and IP address.
   - The *Vulnerability Summary* pane displays the number of vulnerabilities categorized into *Critical*, *High*, *Medium* and *Low* risk.
   - The *Topology View pane* displays the user's location in the topology.
   - The *Top Vulnerabilities* pane displays the top vulnerabilities by severity.

Using the SD-WAN monitor

In the example topology, the branches are configured to use SD-WAN. You can use the top-right navigation menu in the SD-WAN monitor to navigate to the Branch FortiGate to display information about the SD-WAN.

To view the SD-WAN monitor, go to *Software-Defined Branch* > *SD-WAN Monitor*.

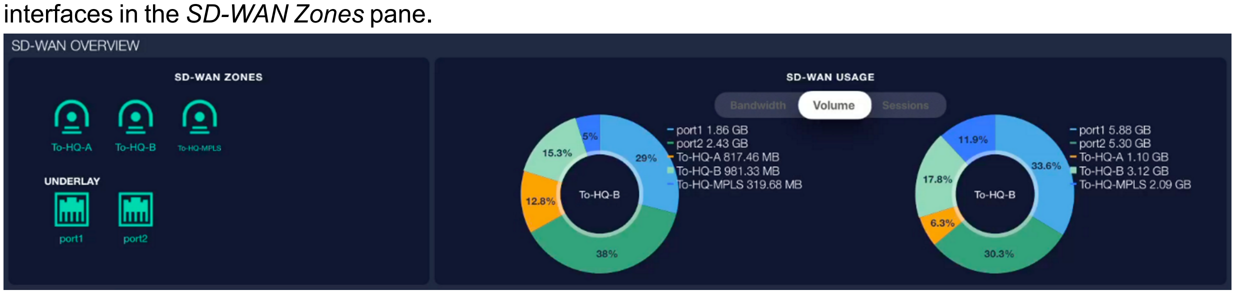
The SD-WAN monitor summarizes the SD-WAN members, Zones, SD-WAN Rules and health checks deployed on the FortiGate. It shows the interface member's SD-WAN usage and its associated service level agreements. The monitor contains a chart that shows if the ports are meeting the SLA target for bandwidth, jitter and latency per the health check in use in each SD-WAN Rule.



Some of the SD-WAN statistics are only available in FOS 6.4.1 and higher.

To view SD-WAN usage charts:

1. In the *SD-WAN Overview* area, Use your remote to select the *SD-WAN Usage* pane.
2. Scroll left and right to view *Bandwidth*, *Volume* and *Sessions* charts for the *VIRTUAL-WAN-LINK* and *Underlay* interfaces in the *SD-WAN Zones* pane.



To view SLA targets:

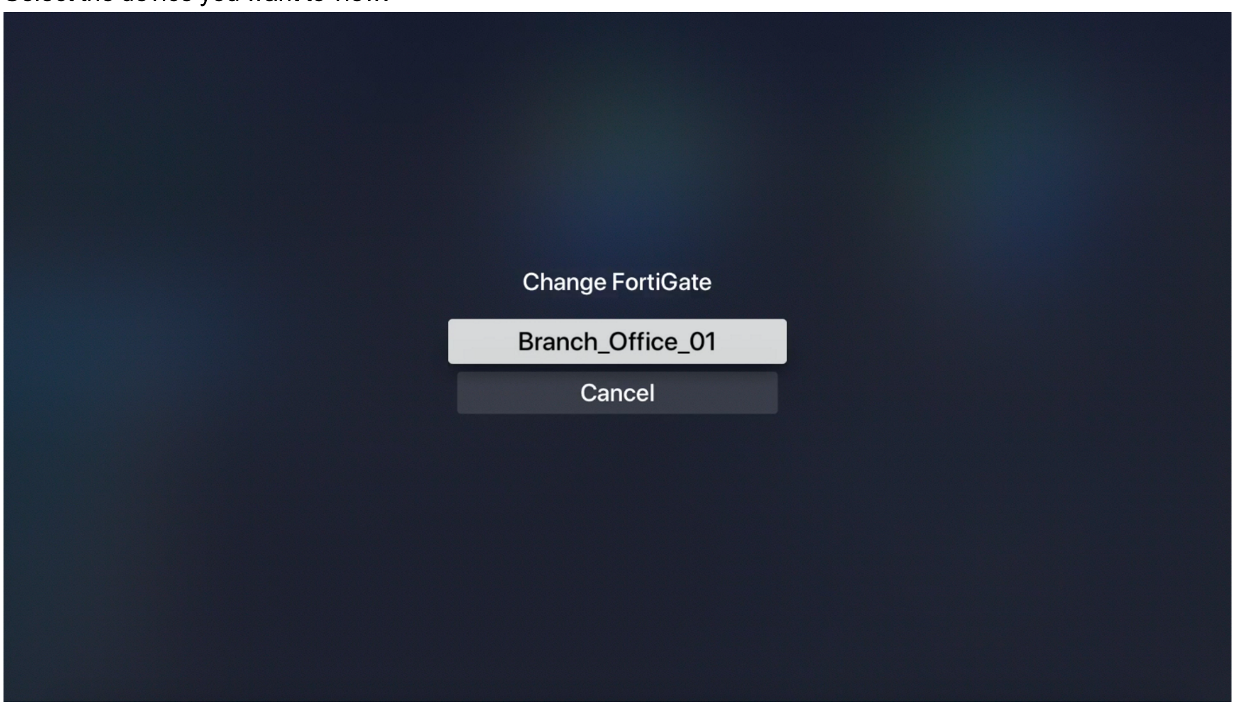
1. In the *SD-WAN Rules* area, use your remote to scroll the rules pane at the left-side of the monitor.
   - The *Destinations* pane displays the destination details.
   - The *Performance SLA* pane displays the SLA targets for the rule.
   - The *SD-WAN Active Interface* pane displays a checkmark next to the active interface.

2. Use your remote to navigate between the *Latency*, *Jitter*, and *Packet Loss* charts.

To view a branch in the topology:

1. Use your remote to swipe to the top navigation in the monitor. Wait for the topology to load.
2. At the top-right of the monitor, select the current device.

3. Select the device you want to view.



Troubleshooting

The following topics provide troubleshooting information for the Fortinet Security Fabric:

- Viewing a summary of all connected FortiGates in a Security Fabric on page 2533
- Diagnosing automation stitches on page 2535

Viewing a summary of all connected FortiGates in a Security Fabric

In downstream FortiGates, the `diagnose sys csf global` command shows a summary of all of the connected FortiGates in the Security Fabric.

To view a Security Fabric summary on a downstream FortiGate:

```
# diagnose sys csf global
Current vision:
[
  {
    "path":"FGVM01TM19000001",
    "mgmt_ip_str":"104.196.102.183",
    "mgmt_port":10403,
    "sync_mode":1,
    "saml_role":"identity-provider",
    "admin_port":443,
    "serial":"FGVM01TM19000001",
    "host_name":"admin-root",
    "firmware_version_major":6,
    "firmware_version_minor":2,
    "firmware_version_patch":0,
    "firmware_version_build":1010,
    "subtree_members":[
      {
        "serial":"FGVM01TM19000002"
      },
      {
        "serial":"FGVM01TM19000003"
      },
      {
        "serial":"FGVM01TM19000004"
      },
      {
        "serial":"FGVM01TM19000005"
      }
    ]
  },
  {
    "path":"FGVM01TM19000001:FGVM01TM19000002",
    "mgmt_ip_str":"104.196.102.183",
    "mgmt_port":10423,
    "sync_mode":1,
    "saml_role":"service-provider",
    "admin_port":443,
    "serial":"FGVM01TM19000002",
    "host_name":"Branch_Office_01",
    "firmware_version_major":6,
    "firmware_version_minor":2,
    "firmware_version_patch":0,
    "firmware_version_build":1010,
```

```
    "upstream_intf":"Branch-HQ-A",
    "upstream_serial":"FGVM01TM19000001",
    "parent_serial":"FGVM01TM19000001",
    "parent_hostname":"admin-root",
    "upstream_status":"Authorized",
    "upstream_ip":22569994,
    "upstream_ip_str":"10.100.88.1",
    "subtree_members":[
    ],
    "is_discovered":true,
    "ip_str":"10.0.10.2",
    "downstream_intf":"To-HQ-A",
    "idx":1
  },
  {
    "path":"FGVM01TM19000001:FGVM01TM19000003",
    "mgmt_ip_str":"104.196.102.183",
    "mgmt_port":10407,
    "sync_mode":1,
    "saml_role":"service-provider",
    "admin_port":443,
    "serial":"FGVM01TM19000003",
    "host_name":"Enterprise_Second_Floor",
    "firmware_version_major":6,
    "firmware_version_minor":2,
    "firmware_version_patch":0,
    "firmware_version_build":1010,
    "upstream_intf":"port3",
    "upstream_serial":"FGVM01TM19000001",
    "parent_serial":"FGVM01TM19000001",
    "parent_hostname":"admin-root",
    "upstream_status":"Authorized",
    "upstream_ip":22569994,
    "upstream_ip_str":"10.100.88.1",
    "subtree_members":[
    ],
    "is_discovered":true,
    "ip_str":"10.100.88.102",
    "downstream_intf":"port1",
    "idx":2
  },
  {
    "path":"FGVM01TM19000001:FGVM01TM19000004",
    "mgmt_ip_str":"104.196.102.183",
    "mgmt_port":10424,
    "sync_mode":1,
    "saml_role":"service-provider",
    "admin_port":443,
    "serial":"FGVM01TM19000004",
    "host_name":"Branch_Office_02",
    "firmware_version_major":6,
    "firmware_version_minor":2,
    "firmware_version_patch":0,
    "firmware_version_build":1010,
    "upstream_intf":"HQ-MPLS",
    "upstream_serial":"FGVM01TM19000001",
```

```
      "parent_serial":"FGVM01TM19000001",
      "parent_hostname":"admin-root",
      "upstream_status":"Authorized",
      "upstream_ip":22569994,
      "upstream_ip_str":"10.100.88.1",
      "subtree_members":[
      ],
      "is_discovered":true,
      "ip_str":"10.0.12.3",
      "downstream_intf":"To-HQ-MPLS",
      "idx":3
    },
    {
      "path":"FGVM01TM19000001:FGVM01TM19000005",
      "mgmt_ip_str":"104.196.102.183",
      "mgmt_port":10404,
      "sync_mode":1,
      "saml_role":"service-provider",
      "admin_port":443,
      "serial":"FGVM01TM19000005",
      "host_name":"Enterprise_First_Floor",
      "firmware_version_major":6,
      "firmware_version_minor":2,
      "firmware_version_patch":0,
      "firmware_version_build":1010,
      "upstream_intf":"port3",
      "upstream_serial":"FGVM01TM19000001",
      "parent_serial":"FGVM01TM19000001",
      "parent_hostname":"admin-root",
      "upstream_status":"Authorized",
      "upstream_ip":22569994,
      "upstream_ip_str":"10.100.88.1",
      "subtree_members":[
      ],
      "is_discovered":true,
      "ip_str":"10.100.88.101",
      "downstream_intf":"port1",
      "idx":4
    }
  ]
```


Diagnosing automation stitches

Diagnose commands are available to:

- Test an automation stitch
- Enable or disable log dumping for automation stitches
- Display the settings of every automation stitch
- Display statistics on every automation stitch

To test an automation stitch:

```
diagnose automation test <automation-stitch-name>
```

Example:

```
# diagnose automation test HA-failover
automation test is done. stitch:HA-failover
```

To toggle log dumping:

```
diagnose test application autod 1
```

Examples:

```
# diagnose test application autod 1
autod log dumping is enabled
```

```
# diagnose test application autod 1
autod log dumping is disabled

autod logs dumping summary:
autod dumped total:7 logs, num of logids:4
```

To display the settings for all of the automation stitches:

```
diagnose test application autod 2
```

Example:

```
# diagnose test application autod 2
csf: enabled root:yes
total stitches activated: 3

stitch: Compromised-IP-Banned
        destinations: all
        trigger: Compromised-IP-Banned

       local hit: 0 relayed to: 0 relayed from: 0
        actions:
               Compromised-IP-Banned_ban-ip type:ban-ip interval:0

stitch: HA-failover
        destinations: HA-failover_ha-cluster_25;
        trigger: HA-failover

       local hit: 0 relayed to: 0 relayed from: 0
        actions:
               HA-failover_email type:email interval:0
               subject: HA Failover
               mailto:admin@example.com;

stitch: rebooot
       destinations: all
       trigger: reboot

       local hit: 0 relayed to: 0 relayed from: 0
       actions:
               action1 type:alicloud-function interval:0
                       delay:1 required:yes
                       Account ID: id
                       Region: region
                       Function domain: fc.aliyuncs.com
                       Version: versoin
```

```
                        Service name: serv
                        Function name: funcy
                        headers:
```

To display statistic on all of the automation stitches:

```
diagnose test application autod 3
```

Example:

```
stitch: Compromised-IP-Banned
       local hit: 0 relayed to: 0 relayed from: 0
       last trigger:Wed Dec 31 20:00:00 1969
       last relay:Wed Dec 31 20:00:00 1969
       actions:
               Compromised-IP-Banned_ban-ip:
                       done: 1 relayed to: 0 relayed from: 0
                       last trigger:Wed Dec 31 20:00:00 1969
                       last relay:

stitch: HA-failover
       local hit: 0 relayed to: 0 relayed from: 0
       last trigger:Thu May 24 11:35:22 2018
       last relay:Thu May 24 11:35:22 2018
       actions:
               HA-failover_email:
                       done: 1 relayed to: 1 relayed from: 1
                       last trigger:Thu May 24 11:35:22 2018
                       last relay:Thu May 24 11:35:22 2018

stitch: rebooot
       local hit: 2 relayed to: 1 relayed from: 1
       last trigger:Fri May  3 13:30:56 2019
       last relay:Fri May  3 13:30:23 2019
       actions:
               action1
                       done: 1 relayed to: 0 relayed from: 0
                       last trigger:Fri May  3 13:30:56 2019
                       last relay:

logid2stitch mapping:
id:20103  local hit: 0 relayed to: 0 relayed from: 0
       License Expiry
       lambada

id:32138  local hit: 2 relayed to: 1 relayed from: 1
       Compromised-IP-Banned
       HA-failover
       rebooot

action run cfg&stats:
total:2 cur:0 done:1 drop:1
       email:
               flags:10
               stats: total:1 cur:0 done:1 drop:0
       fortiexplorer-notification:
               flags:1
```

```
        stats: total:0 cur:0 done:0 drop:0
alert:
        flags:0
        stats: total:0 cur:0 done:0 drop:0
disable-ssid:
        flags:7
        stats: total:0 cur:0 done:0 drop:0
quarantine:
        flags:7
        stats: total:0 cur:0 done:0 drop:0
quarantine-forticlient:
        flags:4
        stats: total:0 cur:0 done:0 drop:0
quarantine-nsx:
        flags:4
        stats: total:0 cur:0 done:0 drop:0
ban-ip:
        flags:7
        stats: total:0 cur:0 done:0 drop:0
aws-lambda:
        flags:11
        stats: total:0 cur:0 done:0 drop:0
webhook:
        flags:11
        stats: total:0 cur:0 done:0 drop:0
cli-script:
        flags:10
        stats: total:0 cur:0 done:0 drop:0
azure-function:
        flags:11
        stats: total:1 cur:0 done:0 drop:1
google-cloud-function:
        flags:11
        stats: total:0 cur:0 done:0 drop:0
alicloud-function:
        flags:11
        stats: total:0 cur:0 done:0 drop:0
```

Log and Report

Logging and reporting are useful components to help you understand what is happening on your network, and to inform you about certain network activities, such as the detection of a virus, a visit to an invalid website, an intrusion, a failed log in attempt, and myriad others.

Logging records the traffic that passes through, starts from, or ends on the FortiGate, and records the actions the FortiGate took during the traffic scanning process. After this information is recorded in a log message, it is stored in a log file that is stored on a log device (a central storage location for log messages). FortiGate supports sending all log types to several log devices, including FortiAnalyzer, FortiAnalyzer Cloud, FortiGate Cloud, and syslog servers. Approximately 5% of memory is used for buffering logs sent to FortiAnalyzer. The FortiGate system memory and local disk can also be configured to store logs, so it is also considered a log device. See Log settings and targets on page 2547 for more information.

Reports show the recorded activity in a more readable format. A report gathers all the log information that it needs, then presents it in a graphical format with a customizable design and automatically generated charts showing what is happening on the network. Reports can be generated on FortiGate devices with disk logging and on FortiAnalyzer devices.

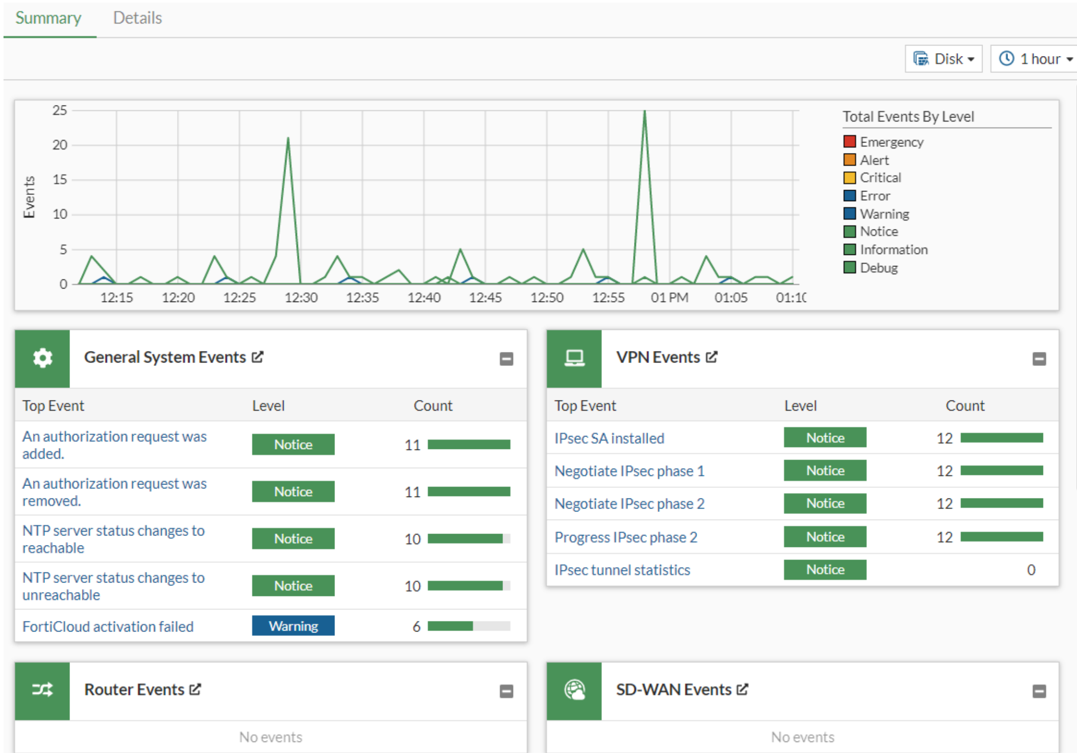
FortiView is a more comprehensive network reporting and monitoring tool. It integrates real-time and historical data into a single view in FortiOS. For more information, see FortiView monitors and widgets on page 107.

Performance statistics are not logged to disk. Performance statistics can be received by a syslog server or by FortiAnalyzer.

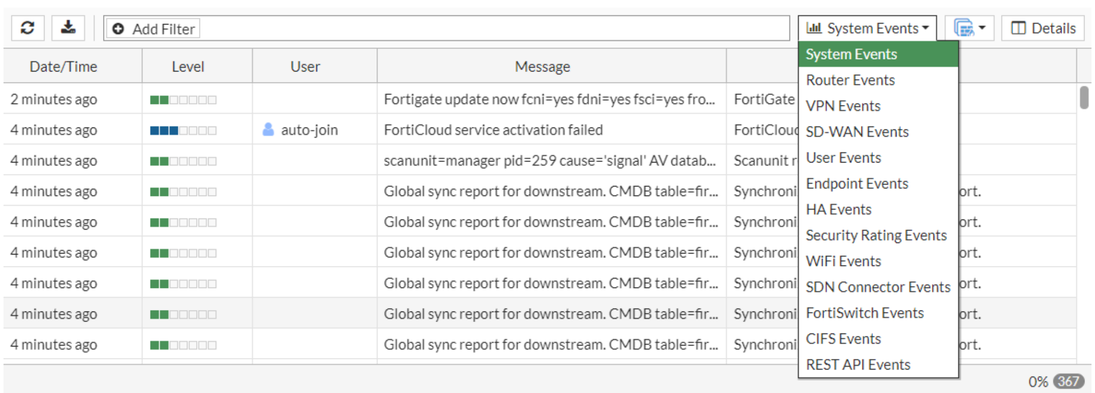
The following topics provide information about logging and reporting:

Viewing event logs

Event log subtypes are available on the *Log & Report > System Events* page. Not all of the event log subtypes are available by default. See System Events log page on page 2541 for more information.



When viewing event logs in the *Details* tab, use the event log subtype dropdown list on the to navigate between event log types.



System Events	Always available.
Router Events	Always available.
VPN Events	Available when *VPN* is enabled in *System* > *Feature Visibility*.
SD-WAN Events	Always available.
User Events	Always available.
Endpoint Events	Available when *Endpoint Control* is enabled in *System* > *Feature Visibility*.
HA Events	Always available.

Security Rating Events	Always available, but logs are only generated when a Security Rating License is registered.
WAN Opt. & Cache Events	Available on devices with two hard disks by default. On devices with one hard disk, the disk usage must be set to `wanopt` and then *WAN Opt. & Cache* must be enabled in *System* > *Feature Visibility*.
WiFi Events	Available on hardware devices when *WiFi Controller* is enabled in *System* > *Feature Visibility*.
FortiExtender Events	Available when *FortiExtender* is enabled in *System* > *Feature Visibility*.
SDN Connector Events	Always available.
FortiSwitch Events	Always available.
CIFS Events	Always available.
REST API Events	Always available.

System Events log page

The *Log & Report* > *Events* page is now renamed *System Events*. The *System Events* page includes:

- A *Summary* tab that displays the top five most frequent events in each type of event log and a line chart to show aggregated events by each severity level. Clicking on a peak in the line chart will display the specific event count for the selected severity level.
- A *Details* tab that displays individual, detailed log views for event type.

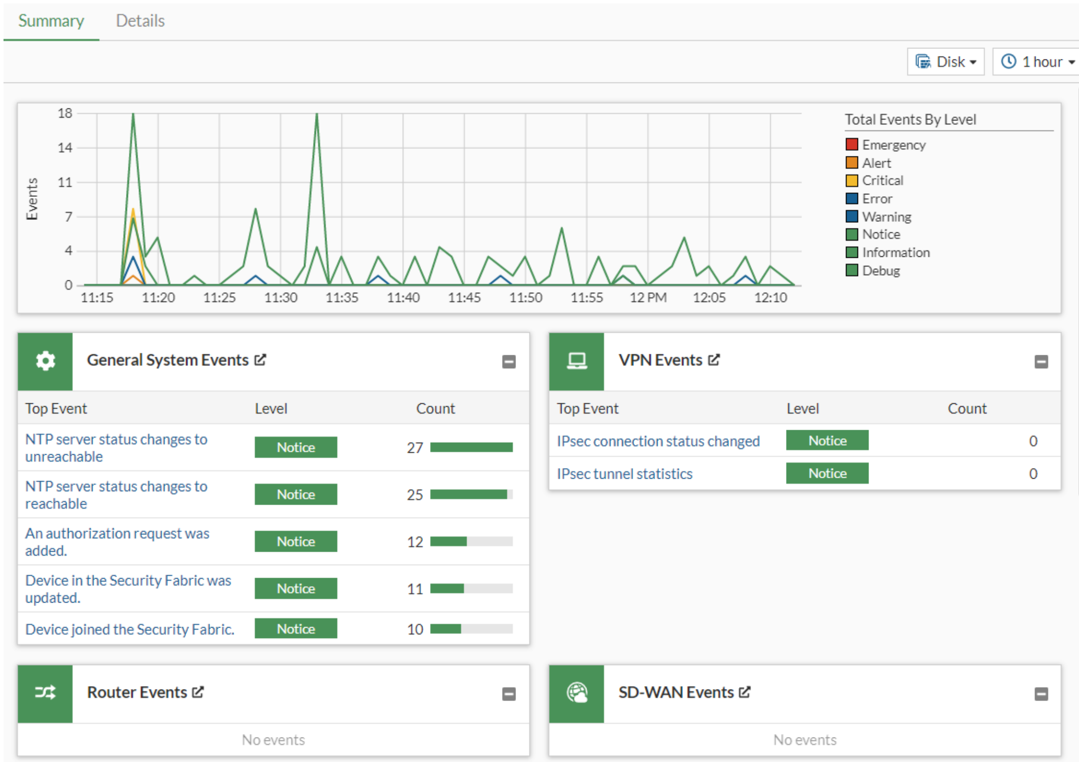
Clicking on an event in the *Summary* tab will automatically bring users to the *Details* tab with the appropriate filters applied.

Disk logging and historical FortiView must be enabled for the *Summary* tab to display valid data. See Log settings and targets on page 2547 for more information.

To review system events in the GUI:

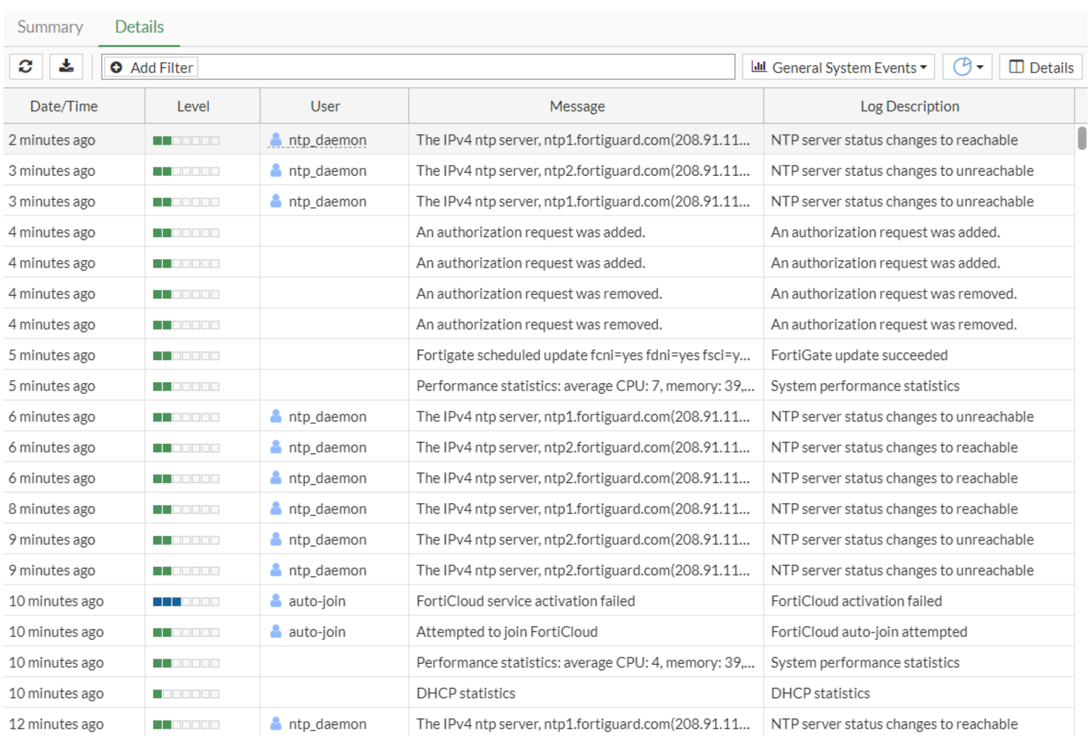
1. Go to *Log & Report* > *System Events*. The *Summary* tab opens.
2. On the right-side of the screen, select the time range from the dropdown list.
   The line chart will display all of the system events, and the non-empty event cards will list up to five *Top Event* entries within the time range set.

   Data is retrieved from FortiView with the *5 minutes* range updated first. When selecting either the *1 hour* or *24 hours* time range, there may be a delay to update *Top Event* entries.



3. Review the details of system events:
   - Click the event card name.

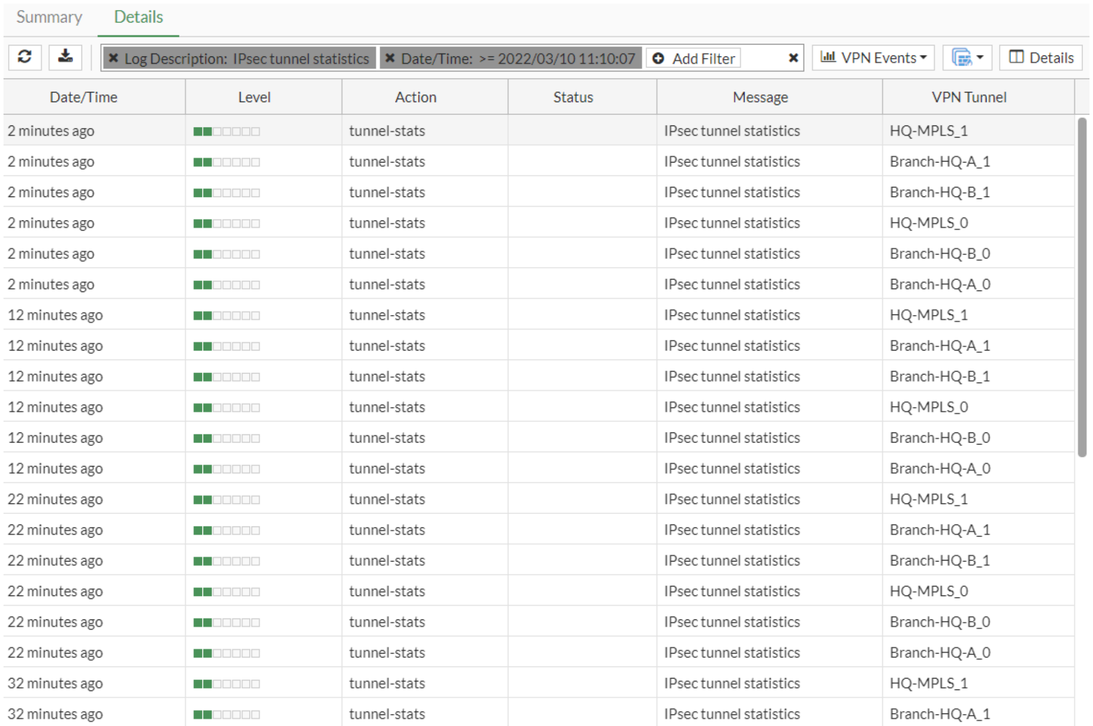
     The *Details* tab displays all event entries for the selected type of event log. The type of event log can be changed in the top-right, dropdown list.



Date/Time	Level	User	Message	Log Description
2 minutes ago		ntp_daemon	The IPv4 ntp server, ntp1.fortiguard.com(208.91.11...	NTP server status changes to reachable
3 minutes ago		ntp_daemon	The IPv4 ntp server, ntp2.fortiguard.com(208.91.11...	NTP server status changes to unreachable
3 minutes ago		ntp_daemon	The IPv4 ntp server, ntp1.fortiguard.com(208.91.11...	NTP server status changes to unreachable
4 minutes ago			An authorization request was added.	An authorization request was added.
4 minutes ago			An authorization request was added.	An authorization request was added.
4 minutes ago			An authorization request was removed.	An authorization request was removed.
4 minutes ago			An authorization request was removed.	An authorization request was removed.
5 minutes ago			Fortigate scheduled update fcni=yes fdni=yes fsci=y...	FortiGate update succeeded
5 minutes ago			Performance statistics: average CPU: 7, memory: 39,...	System performance statistics
6 minutes ago		ntp_daemon	The IPv4 ntp server, ntp1.fortiguard.com(208.91.11...	NTP server status changes to unreachable
6 minutes ago		ntp_daemon	The IPv4 ntp server, ntp2.fortiguard.com(208.91.11...	NTP server status changes to reachable
6 minutes ago		ntp_daemon	The IPv4 ntp server, ntp2.fortiguard.com(208.91.11...	NTP server status changes to reachable
8 minutes ago		ntp_daemon	The IPv4 ntp server, ntp1.fortiguard.com(208.91.11...	NTP server status changes to reachable
9 minutes ago		ntp_daemon	The IPv4 ntp server, ntp2.fortiguard.com(208.91.11...	NTP server status changes to unreachable
9 minutes ago		ntp_daemon	The IPv4 ntp server, ntp2.fortiguard.com(208.91.11...	NTP server status changes to unreachable
10 minutes ago		auto-join	FortiCloud service activation failed	FortiCloud activation failed
10 minutes ago		auto-join	Attempted to join FortiCloud	FortiCloud auto-join attempted
10 minutes ago			Performance statistics: average CPU: 4, memory: 39,...	System performance statistics
10 minutes ago			DHCP statistics	DHCP statistics
12 minutes ago		ntp_daemon	The IPv4 ntp server, ntp1.fortiguard.com(208.91.11...	NTP server status changes to reachable

- Click a *Top Event* entry in an event card.

  The *Details* tab displays system events with filters for the selected event entry and time range. The type of event log can be changed in the top-right, dropdown list.



Up to 100 *Top Event* entries can be listed in the CLI using the `diagnose fortiview result event-log` command.

To list system events in the CLI:

```
# diagnose fortiview result event-log

    data(1646760000-1646846401):
    0). subtype-ha | eventname-HA device interface failed | level-warning | count-1 |
    1). subtype-system | eventname-DHCP statistics | level-information | count-40 |
    2). subtype-system | eventname-Super admin left VDOM | level-information | count-13 |
    3). subtype-system | eventname-Admin performed an action from GUI | level-warning |
count-5 |
    4). subtype-system | eventname-Super admin entered VDOM | level-information | count-4 |
    5). subtype-system | eventname-Global setting changed | level-notice | count-3 |
    6). subtype-system | eventname-Attribute configured | level-information | count-2 |
    7). subtype-system | eventname-Clear active sessions | level-warning | count-2 |
    8). subtype-system | eventname-Disk log rolled | level-notice | count-2 |
    9). subtype-system | eventname-Log rotation requested by FortiCron | level-notice |
count-1 |
    10). subtype-system | eventname-Report generated successfully | level-notice | count-1 |
    11). subtype-system | eventname-Test | level-warning | count-1 |
    12). subtype-system | eventname-VDOM added | level-notice | count-1 |
    13). subtype-user | eventname-Authentication failed | level-notice | count-1 |
    14). subtype-user | eventname-Authentication lockout | level-warning | count-1 |
    15). subtype-user | eventname-FortiGuard override failed | level-warning | count-1 |
```

The data is collected from FortiView for the last 24 hours by default. To specify a specific time range, customize the time filter using the `diagnose fortiview time` command.

To filter the time range of system events in the CLI:

```
# diagnose fortiview time <arg1> <arg2>
```

Where `<arg1>` is the start time in YYYY-MM-DD HH:MM:SS and `<arg2>` is the end time in YYYY-MM-DD HH:MM:SS.

Security Events log page

The *Log & Report* UTM log subtypes have been combined into the *Security Events* log page. The *Security Events* log page includes:

- A *Summary* tab that displays the five most frequent events for all of the enabled UTM security events.
- A *Details* tab that displays individual, detailed logs for each UTM type.

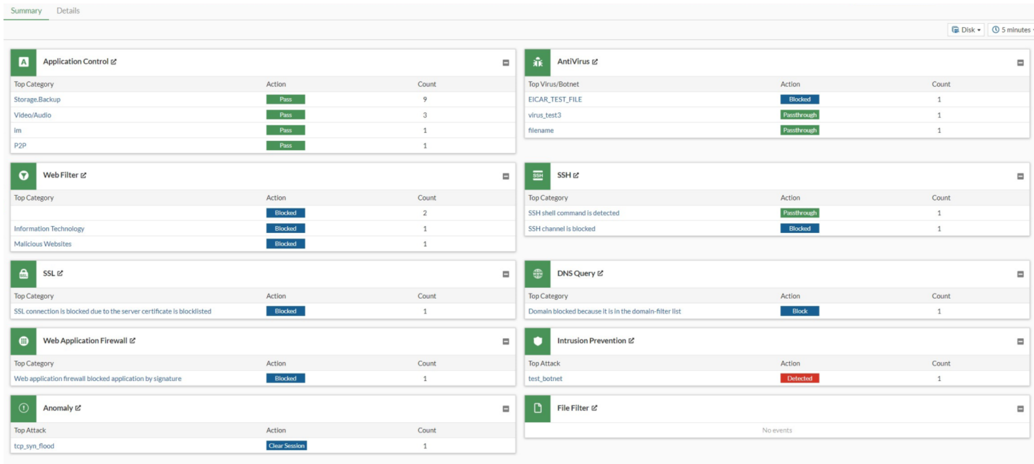
Clicking on an event in the *Summary* tab will bring users to the *Details* tab with the appropriate filters automatically applied.

Disk logging and historical FortiView must be enabled for the *Summary* tab to display valid data. See Log settings and targets on page 2547 for more information.

To review security events in the GUI:

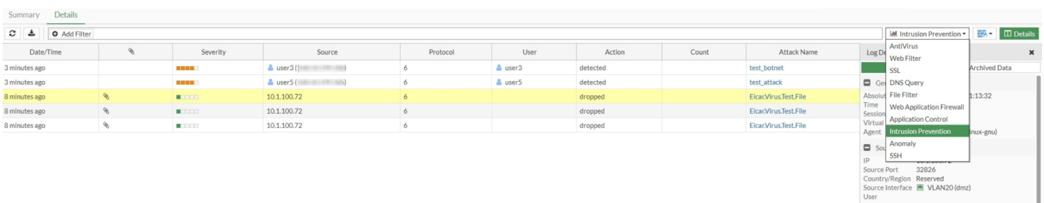
1. Go to *Log & Report > Security Events*.
   The *Summary* tab displays up to five top events for each enabled, non-empty security event cards.
2. On the right-side of the screen, select the time range from the dropdown list.
   The non-empty security event cards will list up to five top entries within the time range set.

   Data is retrieved from FortiView with the *5 minutes* range updated first. When selecting either the *1 hour* or *24 hours* time range, there may be a delay to update top security event entries.



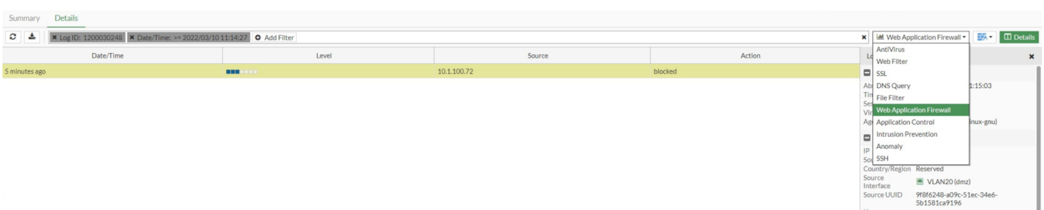
3. Review the details of security events:
    - Click the security event card name.

      The *Details* tab displays all event entries for the selected type of security event. The security event type can be changed in the top-right dropdown list.

    - Click a top event entry in a security event card.

      The *Details* tab displays security events with filters for the selected event entry and time filter. The security event type can be changed in the top-right dropdown list.

Up to 100 top security event entries can be listed in the CLI using the `diagnose fortiview result security-log` command.

To list security events in the CLI:

```
# diagnose fortiview result security-log [<filters>]
```

To list security events in the CLI with no filters applied:

```
# diagnose fortiview result security-log

    data(1646862300-1646948701):
    0). logcat-2 | logcatname-virus | logid-0211008192 | eventname-EICAR_TEST_FILE |
eventname_field-virus | action-blocked | count-1 |
    1). logcat-2 | logcatname-virus | logid-0211008192 | eventname-virus_test3 | eventname_
field-virus | action-passthrough | count-1 |
```

```
    2). logcat-2 | logcatname-virus | logid-0212008448 | eventname-filename | eventname_
field-virus | action-passthrough | count-1 |
    3). logcat-3 | logcatname-webfilter | logid-0318012800 | eventname- | eventname_field-
catdesc | action-blocked | count-2 |
    4). logcat-3 | logcatname-webfilter | logid-0316013056 | eventname-Information
Technology | eventname_field-catdesc | action-blocked | count-1 |
    5). logcat-3 | logcatname-webfilter | logid-0316013056 | eventname-Malicious Websites |
eventname_field-catdesc | action-blocked | count-1 |
    6). logcat-4 | logcatname-ips | logid-0419016384 | eventname-Eicar.Virus.Test.File |
eventname_field-attack | action-dropped | count-3 |
    7). logcat-4 | logcatname-ips | logid-0422016400 | eventname-test_botnet | eventname_
field-attack | action-detected | count-1 |
    8). logcat-7 | logcatname-anomaly | logid-0720018432 | eventname-tcp_syn_flood |
eventname_field-attack | action-clear_session | count-1 |
    9). logcat-10 | logcatname-app-ctrl | logid-1059028704 | eventname-Storage.Backup |
eventname_field-appcat | action-pass | count-9 |
    10). logcat-10 | logcatname-app-ctrl | logid-1059028704 | eventname-Video/Audio |
eventname_field-appcat | action-pass | count-3 |
    11). logcat-10 | logcatname-app-ctrl | logid-1059028672 | eventname-im | eventname_
field-appcat | action-pass | count-1 |
    12). logcat-10 | logcatname-app-ctrl | logid-1059028704 | eventname-P2P | eventname_
field-appcat | action-pass | count-1 |
    13). logcat-15 | logcatname-dns | logid-1501054400 | eventname-Domain blocked because it
is in the domain-filter list | eventname_field-logid | action-block | count-1 |
    14). logcat-17 | logcatname-ssl | logid-1700062300 | eventname-SSL connection is blocked
due to the server certificate is blocklisted | eventname_field-logid | action-blocked |
count-1 |
    15). logcat-16 | logcatname-ssh | logid-1600061002 | eventname-SSH shell command is
detected | eventname_field-logid | action-passthrough | count-1 |
    16). logcat-16 | logcatname-ssh | logid-1601061010 | eventname-SSH channel is blocked |
eventname_field-logid | action-blocked | count-1 |
    17). logcat-12 | logcatname-waf | logid-1200030248 | eventname-Web application firewall
blocked application by signature | eventname_field-logid | action-blocked | count-1 |
    18). logcat-8 | logcatname-voip | logid-0814044032 | eventname-Logid_44032 | eventname_
field-logid | action-permit | count-1 |
    19). logcat-5 | logcatname-emailfilter | logid-0513020480 | eventname-SPAM notification
| eventname_field-logid | action-blocked | count-1 |
```

To list blocked security events in the CLI:

```
# diagnose fortiview result security-log action=blocked

    data(1646862600-1646949001):
    0). logcat-2 | logcatname-virus | logid-0211008192 | eventname-EICAR_TEST_FILE |
eventname_field-virus | action-blocked | count-1 |
    1). logcat-3 | logcatname-webfilter | logid-0318012800 | eventname- | eventname_field-
catdesc | action-blocked | count-2 |
    2). logcat-3 | logcatname-webfilter | logid-0316013056 | eventname-Information
Technology | eventname_field-catdesc | action-blocked | count-1 |
    3). logcat-3 | logcatname-webfilter | logid-0316013056 | eventname-Malicious Websites |
eventname_field-catdesc | action-blocked | count-1 |
    4). logcat-17 | logcatname-ssl | logid-1700062300 | eventname-SSL connection is blocked
due to the server certificate is blocklisted | eventname_field-logid | action-blocked |
count-1 |
    5). logcat-16 | logcatname-ssh | logid-1601061010 | eventname-SSH channel is blocked |
eventname_field-logid | action-blocked | count-1 |
```

```
    6). logcat-12 | logcatname-waf | logid-1200030248 | eventname-Web application firewall
blocked application by signature | eventname_field-logid | action-blocked | count-1 |
    7). logcat-5 | logcatname-emailfilter | logid-0513020480 | eventname-SPAM notification |
eventname_field-logid | action-blocked | count-1 |
```

Log settings and targets

Log settings determine what information is recorded in logs, where the logs are stored, and how often storage occurs. Log settings can be configured in the GUI and CLI. In the GUI, *Log & Report > Log Settings* provides the settings for local and remote logging.

Local Log		
	Disk	Define local log storage on the FortiGate: • *Enable*: Logs will be stored on a local disk. Local disk logging is not available in the GUI if the Security Fabric is enabled. When the Security Fabric is enabled, disk logging can still be configured on the root FortiGate in the CLI but is not available for downstream FortiGates. • *Disable*: Logs will be stored remotely to FortiAnalyzer/FortiManager or to a Cloud logging device.
	Enable Local Reports	Define log reporting on the FortiGate: • *Enable*: Local reports will be available on the FortiGate. Reports can be reviewed in *Log & Report > Local Reports*. If the Security Fabric is enabled, *Local Reports* can be enabled in *System > Feature Visibility*. • *Disable*: Local reports will not be available on the FortiGate.
	Enable Historical FortiView	Define the presentation of log information on FortiView: • *Enable*: Historical log data will be available on a FortiView monitor. By default, logs older than seven days are deleted. Disk logging must be enabled. • *Disable*: Historical log data will not be available on FortiView.
Remote Logging and Archiving		
	Send logs to FortiAnalyzer/FortiManager	Define the status of remote logging to FortiAnalyzer and FortiManager: • *Enable*: Logs are sent to FortiAnalyzer or FortiManager for remote logging. HTTP transaction logs are also sent to a FortiAnalyzer unit to generate additional details in reports. • *Disable*: Logs are stored to system memory, a local disk, or a Cloud logging device.
	Server	Set the server IP address for the FortiAnalyzer or FortiManager. Use *Test Connectivity* to test the connection status to the server.
	Connection status	Displays authorization status on FortiAnalyzer: • *Successful*: The FortiGate is connected to the FortiAnalyzer. Remote logging to the FortiAnalyzer can be configured.

	• *Unauthorized*: The FortiGate is not connected to the FortiAnalyzer. Click Authorize to review the approval status on FortiAnalyzer or see Configuring FortiAnalyzer on page 2215 for more information.
Storage usage	Presents the storage used and the total storage available on the remote logging device.
Analytics usage	Presents the analytics space used and the total analytics space available on the remote logging device.
Archives usage	Presents the archive space used and the total archive space available on the remote logging device.
Upload option	Select the frequency of log uploads to the remote device: • *Real Time*: Logs are sent to the remote device in real time. • *Every Minute*: Logs are sent to the remote device once every minute. This option is unavailable if the Security Fabric Connections is enabled. • *Every 5 Minutes*: Logs are sent to the remote device once every five minutes. This is the default option. This option is unavailable if the Security Fabric Connection is enabled. • `store-and-upload`: Store logs to a local disk before uploading to FortiAnalyzer or FortiManager at a scheduled time. This option is only available for CLI configuration.
Allow access to FortiGate REST API	Define access to FortiGate REST API: • *Enable*: The REST API accesses the FortiGate topology and shares data and results. • *Disable*: The REST API foes not share data and results.
Verify FortiAnalyzer certificate	Define the FortiAnalyzer certificate verification process: • *Enable*: The FortiGate will verify the FortiAnalyzer serial number against the FortiAnalyzer certificate. When verified, the serial number is stored in the FortiGate configuration. • *Disable*: The FortiGate will not verify the FortiAnalyzer certificate against the serial number.
Cloud Logging Settings	
Type	Specify remote logging to the FortiGate Cloud or FortiAnalyzer Cloud device. If multiple devices are enabled, the default preference is FortiAnalyzer Cloud.
Connection status	Displays the current connection status to the selected *Type*. Use *Test Connectivity* to test the connection status to the Cloud logging device.
Upload option	Select the frequency of log uploads to the Cloud device: • *Real Time*: Logs are sent to the Cloud device in real time. • *Every Minute*: Logs are sent to the Cloud device once every minute. This option is unavailable if the Security Fabric Connection is enabled.

	• *Every 5 Minutes*: Logs are sent to the Cloud device once every five minutes. This is the default option. This option is unavailable if the Security Fabric Connection is enabled.
Allow access to FortiGate REST API	Define access to FortiGate REST API: • *Enable*: The REST API accesses the FortiGate topology and shares data and results. • *Disable*: The REST API foes not share data and results.
Verify FortiAnalyzer Cloud certificate	Define the FortiAnalyzer Cloud certificate verification process: • *Enable*: The FortiGate will verify the FortiAnalyzer serial number against the FortiAnalyzer certificate. When verified, the serial number is stored in the FortiGate configuration. • *Disable*: The FortiGate will not verify the FortiAnalyzer certificate against the serial number.
UUIDs in Traffic Log	
Policy	Define the use of policy UUIDs in traffic logs: • *Enable*: Policy UUIDs are stored in traffic logs. UUIDs can be matched for each source and destination that match a policy in the traffic log. See Source and destination UUID logging on page 2569 for more information. • *Disable*: Policy UUIDs are excluded from the traffic logs.
Address	Define the use of address UUIDs in traffic logs: • *Enable*: Address UUIDs are stored in traffic logs. When viewing *Forward Traffic* logs, a filter is automatically set based on UUID. • *Disable*: Address UUIDs are excluded from traffic logs.
Log Settings	
Event Logging	Define the allowed set of event logs to be recorded: • *All*: All event logs will be recorded. • *Customize*: Select specific event log types to be recorded. Deselect all options to disable event logging.
Local Traffic Log	Define the allowed set of traffic logs to be recorded: • *All*: All traffic logs to and from the FortiGate will be recorded. • *Customize*: Select specific traffic logs to be recorded. Deselect all options to disable traffic logging. Local traffic logging is disabled by default due to the high volume of logs generated.
GUI Preferences	
Resolve Hostnames	Define the translation of IP addresses to host names: • *Enable*: IP addresses are translated to host names using reverse DNS lookup. If the DNS server is not available or is slow to reply, requests may time out. • *Disable*: IP addresses are not translated to host names.

Resolve Unknown Applications	Define the resolution of unknown applications: • *Enable*: Unknown applications are resolved using the Internet Service Database. • *Disable*: Unknown applications are not resolved.

Configuring logs in the CLI

The FortiGate can store logs locally to its system memory or a local disk. Logs can also be stored externally on a storage device, such as FortiAnalyzer, FortiAnalyzer Cloud, FortiGate Cloud, or a syslog server. See Sending traffic logs to FortiAnalyzer Cloud on page 2558 for more information on Premium (AFAC) and Standard (FAZC) subscriptions.

Disk logging

Disk logging must be enabled for logs to be stored locally on the FortiGate. By default, logs older than seven days are deleted from the disk. Log age can be configured in the CLI. Approximately 75% of disk space is available for log storage. Log storage space can be determined using the `diagnose sys logdisk usage` command.

To configure local disk logging:

```
config log disk setting
    set status enable
    set maximum-log-age <integer>
    set max-log-file-size <integer>
end
```

Remote logging

The process to configure FortiGate to send logs to FortiAnalyzer or FortiManager is identical. Remote logging to FortiAnalyzer and FortiManager can be configured using both the GUI and CLI. When using the CLI, use the `config log fortianalyzer setting` command for both FortiAnalyzer and FortiManager.

If VDOMs are configured on the FortiGate, multiple FortiAnalyzers and syslog servers can be added globally. SeeConfiguring multiple FortiAnalyzers (or syslog servers) per VDOM on page 2560 and Configuring multiple FortiAnalyzers on a FortiGate in multi-VDOM mode on page 2562 for more information.

To configure remote logging to FortiAnalyzer:

```
config log fortianalyzer setting
    set status enable
    set server <server_IP>
    set upload option {store-and-upload | realtime | 1-minute | 5-minute}
end
```

Remote logging can also be configured to FortiCloud, FortiSIEM, and syslog servers. Up to four syslog servers or FortiSIEM devices can be configured using the `config log syslogd` command and can send logs to syslog in CSV and CEF formats.

To configure remote logging to FortiCloud:

```
config log fortiguard setting
    set status enable
    set source-ip <source IP used to connect FortiCloud>
end
```

To configure remote logging to a syslog server:

```
config log syslogd setting
    set status enable
    set server <syslog_IP>
    set format {default | cev | cef}
end
```

Log filters

Log filter settings can be configured to determine which logs are recorded to the FortiAnalyzer, FortiManager, and syslog servers. This allows certain logging levels and types of logs to be directed to specific log devices.

To configure log filters for FortiAnalyzer:

```
config log fortianalyzer filter
    set severity <level>
    set forward-traffic {enable | disable}
    set local-traffic {enable | disable}
    set multicast-traffic {enable | disable}
    set sniffer-traffic {enable | disable}
end
```

To configure log filters for a syslog server:

```
config log syslogd filter
    set severity <level>
    set forward-traffic {enable | disable}
    set local-traffic {enable | disable}
    set multicast-traffic {enable | disable}
    set sniffer-traffic {enable | disable}
end
```

Email alerts

FortiGate events can be monitored at all times using email alerts. Email alerts send notifications to up to three recipients and can be triggered based on log event and severity level. Email alerts will be sent every five minutes by default but this can be configured in the CLI.

To configure email alerts:

```
config alertemail setting
    set username <name>
    set mailto1 <email>
    set filter-mode {category | threshold}
```

```
    set email-interval <integer>
    set IPS-logs {enable | disable}
    set HA-logs {enable | disable}
    set antivirus-logs {enable | disable}
    set webfilter-logs {enable | disable}
    set log-disk-usage-warning {enable | disable}
end
```

Threat weight

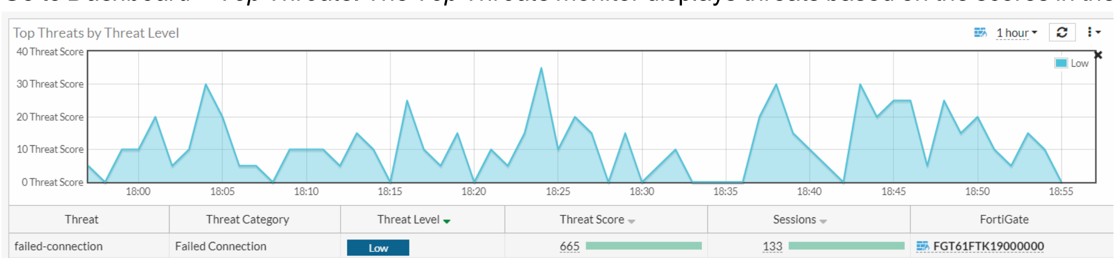
Threat weight helps aggregate and score threats based on user-defined severity levels. It adds several fields such as threat level (`crlevel`), threat score (`crscore`), and threat type (`craction`) to traffic logs. Threat weight logging is enabled by default and the settings can be customized. Threats can be viewed from the *Top Threats* FortiView dashboard.

To configure threat weight settings:

1. Go to *Log & Report* > *Threat Weight*.
2. Adjust the settings as needed, such as individual weights per threat type and risk level values.
3. Click *Apply*.

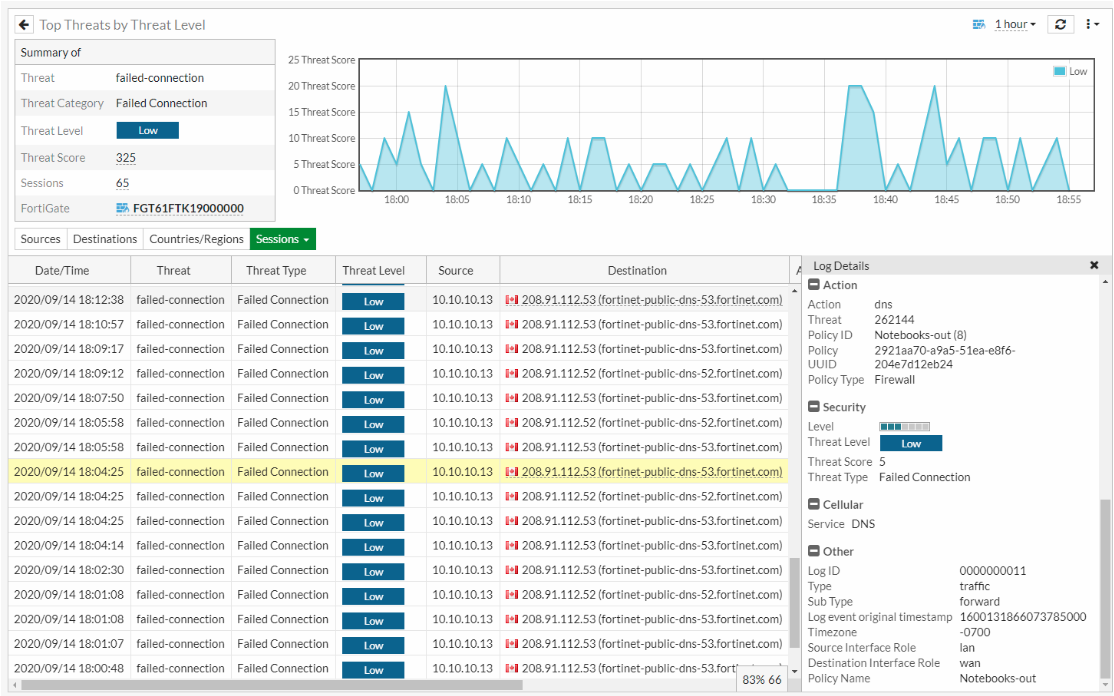
To add the Top Threats monitor to the dashboard:

1. In the tree menu, click *Dashboard* and in the FortiView section, click the + sign (*Add Monitor*).
2. In the *Security* section, enable *Show More* and click *Top Threats*.
3. Configure the settings as needed.
4. Click *Add Monitor*.
5. Go to *Dashboard* > *Top Threats*. The *Top Threats* monitor displays threats based on the scores in the traffic logs.



6. Double-click a threat to view the summary.
7. Click *Sources*, *Destinations*, *Countries/Regions*, or *Sessions* to view more information. Double-click an entry to

view the log details.



Logging to FortiAnalyzer

The following topics provide instructions on logging to FortiAnalyzer:

- FortiAnalyzer Reports page in the GUI on page 2553
- FortiAnalyzer log caching on page 2555
- Sending traffic logs to FortiAnalyzer Cloud on page 2558
- Configuring multiple FortiAnalyzers (or syslog servers) per VDOM on page 2560
- Configuring multiple FortiAnalyzers on a FortiGate in multi-VDOM mode on page 2562

FortiAnalyzer Reports page in the GUI

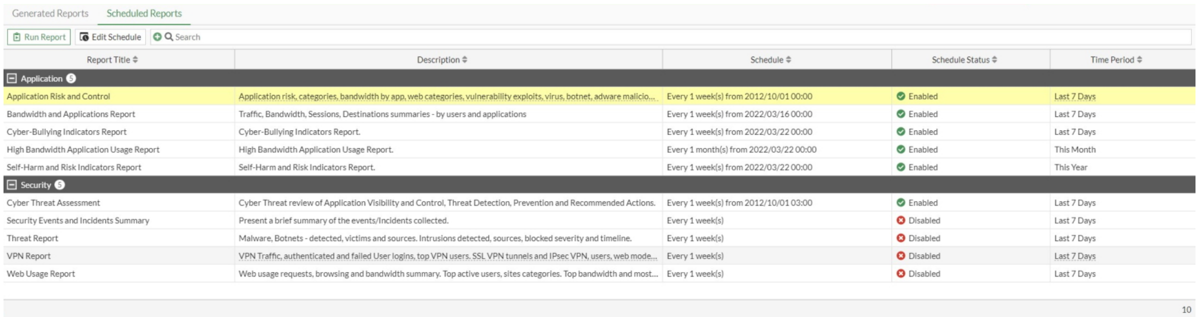
FortiAnalyzer reports can be viewed in the GUI on the *Log & Report > FortiAnalyzer Reports* page. Administrators can generate, delete, and edit report schedules, and view and download generated reports.

FortiAnalyzer must be configured in FortiOS. If the FortiGate is unauthorized on FortiAnalyzer, or the connection to FortiAnalyzer is down, the *FortiAnalyzer Reports* page loads with *No results*.

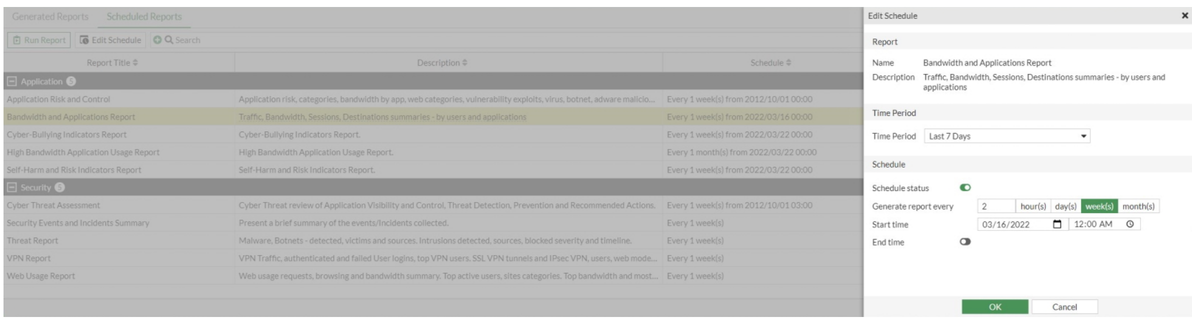
When the Security Fabric is enabled, only the root FortiGate can run, edit, and delete FortiAnalyzer reports. Downstream FortiGates can only view the generated reports.

To edit a report schedule:

1. Go to *Log & Report > FortiAnalyzer Reports* and select the *Scheduled Reports* tab.

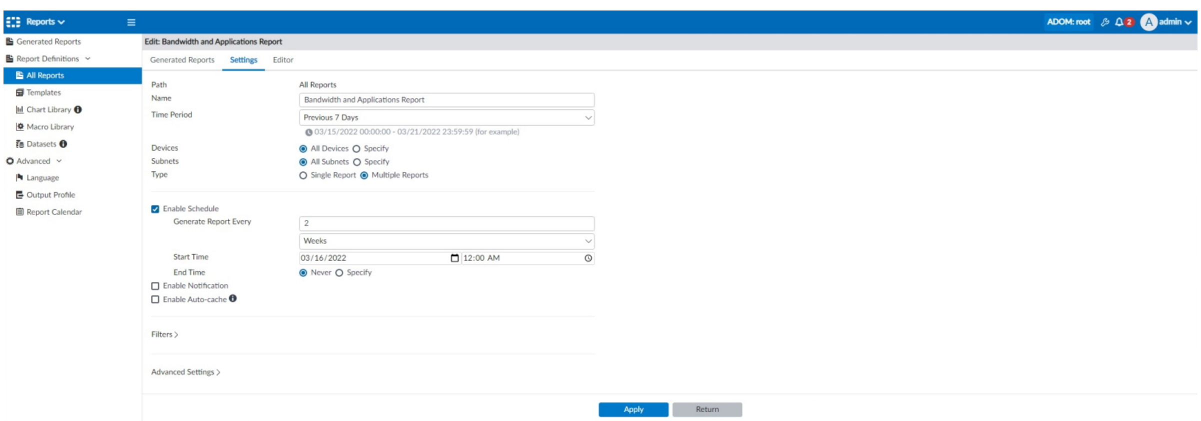


2. Select a report and click *Edit Schedule*. The *Edit Schedule* pane opens. In this example, the schedule for the Bandwidth and Applications report is changed to run from every week to every two weeks.
3. In the *Schedule* section, set the values for *Generate report every* to *2 week(s)*.



4. Click *OK*.

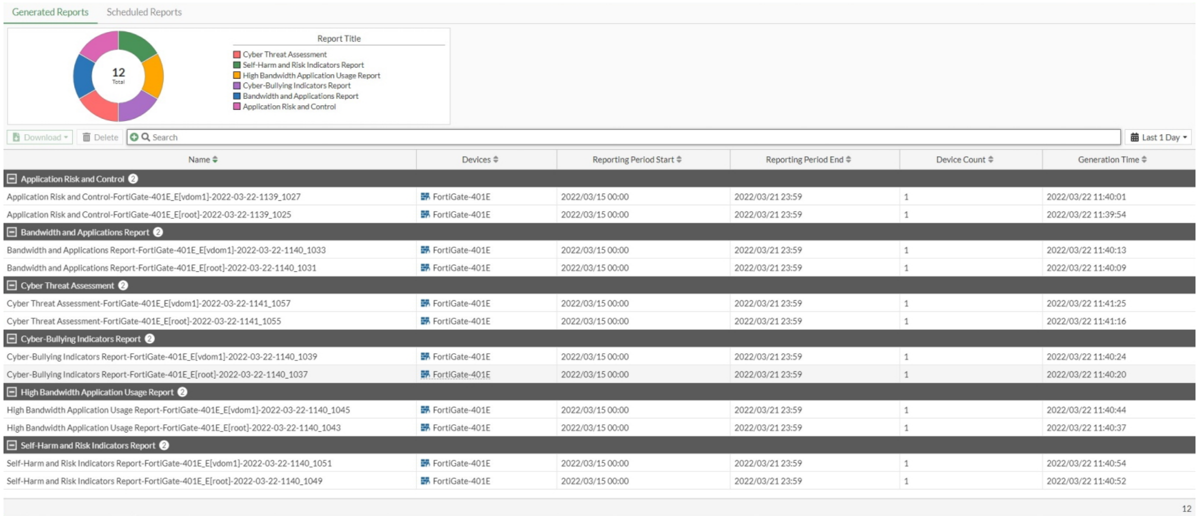
   The schedule is also updated automatically in FortiAnalyzer for the same report (go to *Reports > Report Definitions > All Reports* and edit the report to view the settings).



To view and download reports:

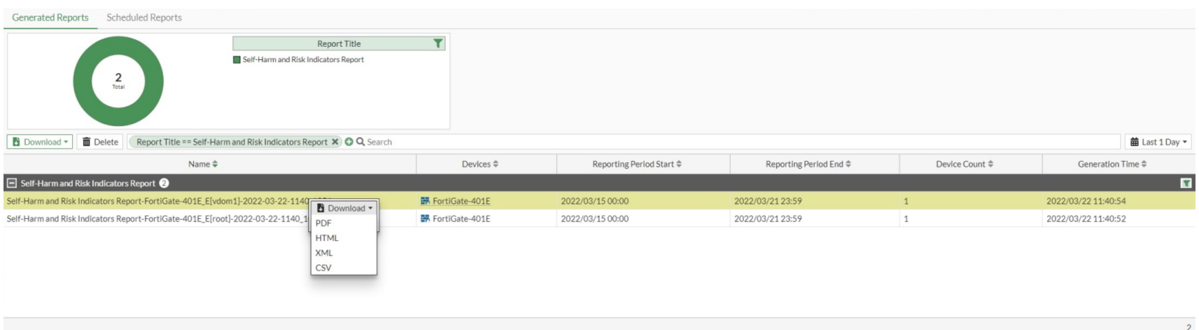
1. Go to *Log & Report > FortiAnalyzer Reports* and select the *Generated Reports* tab.

   A pie chart displays the total count of FortiAnalyzer reports, categorized by report title. Generated reports are listed below and arranged by title, which includes reports from all VDOMs.



In this example, the Self-Harm and Risk Indicators reports are filtered, and the report for vdom1 is downloaded.

2. In the pie chart, click the green segment to filter the Self-Harm and Risk Indicators reports
3. In the filtered results, select the report for vdom1. Right-click and select *Download*.



4. Select a file format. The report is saved to the default download location.

FortiAnalyzer log caching

Reliable logging to FortiAnalyzer prevents lost logs when the connection between FortiOS and FortiAnalyzer is disrupted. When reliable mode is enabled:

1. Logs are cached in a FortiOS memory queue.
2. FortiOS sends logs to FortiAnalyzer, and FortiAnalyzer uses `seq_no` to track received logs.
3. After FortiOS sends logs to FortiAnalyzer, logs are moved to a confirm queue in FortiOS.
4. FortiOS periodically queries FortiAnalyzer for the latest `seq_no` of the last log received, and clears logs from the confirm queue up to the `seq_no`.
5. If the connection between FortiOS and FortiAnalyzer is disrupted, FortiOS resends the logs in the confirm queue to FortiAnalyzer when the connection is reestablished.

FortiAnalyzer 7.2.0 and later is required.

To enable reliable mode:

```
config log fortianalyzer setting
    set reliable enable
end
```

To view the memory and confirm queues:

1. Verify that log synchronization is enabled for FortiAnalyzer:

```
# diagnose test application fgtlogd 1
vdom-admin=0
mgmt=root

fortilog:
faz: global , enabled
    server=172.16.200.251, realtime=1, ssl=1, state=connected
    server_log_status=Log is allowed.,
    src=, mgmt_name=FGh_Log_root_172.16.200.251, reliable=1, sni_prefix_type=none,
    required_entitlement=none, region=ca-west-1,,
    logsync_enabled:1, logsync_conn_id:65535, seq_no:790
...
```

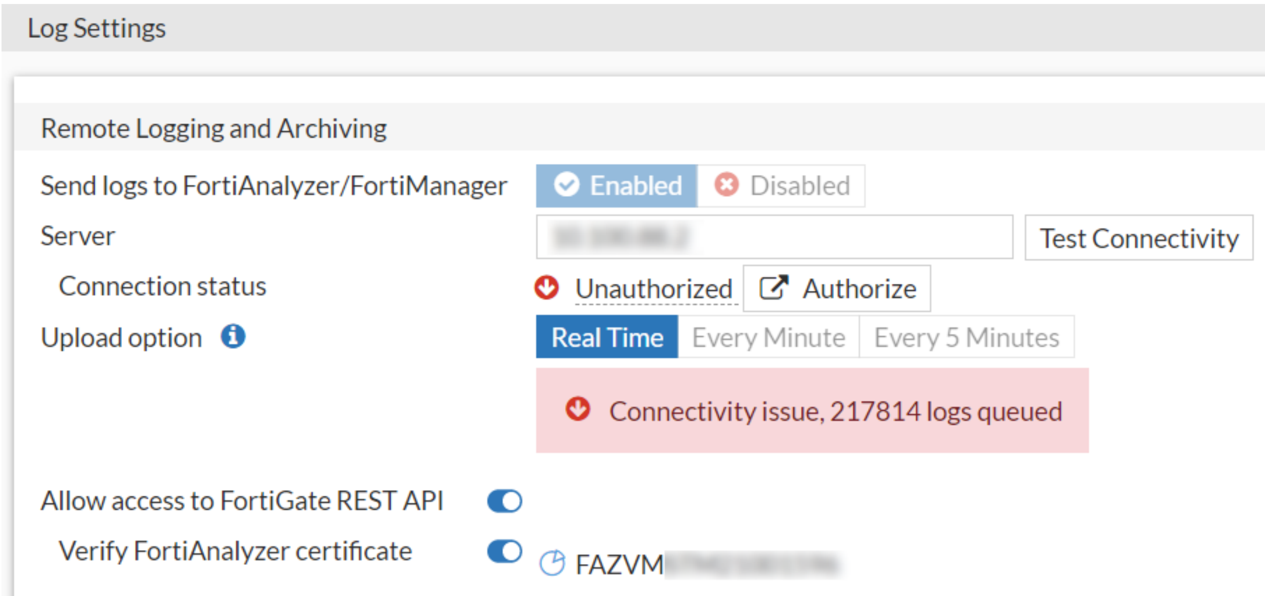
2. When a network disruption disconnects FortiOS from FortiAnalyzer and FortiOS continues to generate logs, the logs are cached in the memory queue.
   - View the number of logs in the cache and queue:

```
# diagnose test application fgtlogd 41

cache maximum: 189516595(180MB) objects: 40 used: 27051(0MB) allocated: 29568(0MB)

VDOM:root
Memory queue for: global-faz
    queue:
        num:9 size:6976(0MB) total size:26068(0MB) max:189516595(180MB) logs:28
Confirm queue for: global-faz
    queue:
        num:29 size:19092(0MB) total size:27051(0MB) max:189516595(180MB) logs:7

# diagnose test application fgtlogd 30
VDOM:root
Memory queue for: global-faz
        queue:
                num:9 size:6976(0MB) total size:26068(0MB) max:189516595(180MB)
                        type:3, cat=1, log_count=1, seq_no=0, data len=359 size:435
                        type:3, cat=1, log_count=1, seq_no=0, data len=307 size:383
                        ......
                        type:3, cat=0, log_count=4, seq_no=0, data len=1347 size:1423
                        type:3, cat=4, log_count=1, seq_no=0, data len=653 size:729
                'total log count':28,  'total data len':6292

Confirm queue for: global-faz
        queue:
                num:29 size:19092(0MB) total size:26068(0MB) max:189516595(180MB)
                        type:3, cat=1, log_count=1, seq_no=1, data len=290 size:366
                        type:3, cat=1, log_count=1, seq_no=2, data len=233 size:309
                        ......
```

```
                    type:3, cat=0, log_count=1, seq_no=28, data len=524 size:600
                    type:3, cat=1, log_count=1, seq_no=29, data len=307 size:383
            'total log count':76,  'total data len':16888
```

There are nine OFTP items cached to the memory queue, and 29 OFTP items to send from FortiOS to FortiAnalyzer that are waiting for confirmation from FortiAnalyzer.

- Go to *Log & Report* > *Log Settings* to view the queue in the GUI:



3. Re-establish the connection between FortiOS and FortiAnalyzer and confirm that the queue has cleared by checking the `seq_no`, which indicates the latest confirmation log from FortiAnalyzer:

```
# diagnose test application fgtlogd 30
VDOM:root
Memory queue for: global-faz
    queue:
        num:0 size:0(0MB) total size:0(0MB) max:189516595(180MB)
        'total log count':0,  'total data len':0

Confirm queue for: global-faz
    queue:
        num:0 size:0(0MB) total size:0(0MB) max:189516595(180MB)
        'total log count':0,  'total data len':0
```

The queue has been cleared, meaning that FortiOS received confirmation from FortiAnalyzer and cleared the confirm queue.

```
# diagnose test application fgtlogd 1
vdom-admin=0
mgmt=root

fortilog:
faz: global , enabled
        server=172.16.200.251, realtime=1, ssl=1, state=connected
        server_log_status=Log is allowed.,
        src=, mgmt_name=FGh_Log_root_172.16.200.251, reliable=1, sni_prefix_type=none,
        required_entitlement=none, region=ca-west-1,
```

```
logsync_enabled:1, logsync_conn_id:65535, seq_no:67
    status: ver=6, used_disk=0, total_disk=0, global=0, vfid=0 conn_verified=Y
    SNs: last sn update:38 seconds ago.
        Sn list:
       (FAZ-VMTM21000000,age=38s)
    queue: qlen=0.
```

OFTP items with a `seq_no` lower than 67 have been sent to FortiAnalyzer and were confirmed.

Sending traffic logs to FortiAnalyzer Cloud

FortiGates with a FortiCloud Premium subscription (AFAC) for Cloud-based Central Logging & Analytics, can send traffic logs to FortiAnalyzer Cloud in addition to UTM logs and event logs. After the Premium subscription is registered through FortiCare, FortiGuard will verify the purchase and authorize the AFAC contract. Once the contract is verified, FortiGuard will deliver the contract to FortiGate.

FortiGates with a Standard FortiAnalyzer Cloud subscription (FAZC) can only send UTM and event logs. FortiGates with a Premium subscription will send the UTM and event logs even if the Standard subscription has expired.

FortiAnalyzer Cloud does not support DLP/IPS archives at this time.

Example

In the following example, you will configure a FortiGate with a valid Premium subscription (AFAC) and expired Standard subscription (FAZC) to send traffic logs to FortiAnalyzer Cloud.

1. Configure the log delivery.
```
config log fortianalyzer-cloud setting
   set status enable
   set ips-archive disable
   set access-config enable
   set enc-algorithm high
   set ssl-min-proto-version default
   set conn-timeout 10
   set monitor-keepalive-period 5
   set monitor-failure-retry-period 5
   set certificate ''
   set source-ip ''
   set interface-select-method auto
   set upload-option realtime
   set priority default
   set max-log-rate 0
end
```
2. Verify the status of the FortiCloud Premium subscription (AFAC) and standard FortiAnalyzer Cloud subscription (FAZC).
The `FAZC` and `AFAC` fields display the subscription expiration date. The `Support contract` field displays the FortiCare account information. The `User ID` field displays the ID for FortiAnalyzer-Cloud instance.
```
# diagnose test update info
...
FAZC,Tue Sep 24 16:00:00 2030
```

```
AFAC,Mon Nov 29 16:00:00 2021
...
Support contract: pending_registration=255 got_contract_info=1
account_id=[****@fortinet.com] company=[Fortinet] industry=[Technology]
User ID: 979090
```

The `FAZC` and `AFAC` subscriptions are valid (date of verification is November 29, 2020).

3. Check the status of FortiAnalyzer Cloud.

```
# execute log fortianalyzer-cloud test-connectivity
FortiAnalyzer Host Name: FAZVM64-VIO-CLOUD
FortiAnalyzer Adom Name: root
FortiGate Device ID: FG101FTK19000000
Registration: registered
Connection: allow
Adom Disk Space (Used/Allocated): 50351453B/53687091200B
Analytics Usage (Used/Allocated): 41368925B/37580963840B
Analytics Usage (Data Policy Days Actual/Configured): 60/60 Days
Archive Usage (Used/Allocated): 8982528B/16106127360B
Archive Usage (Data Policy Days Actual/Configured): 235/365 Days
Log: Tx & Rx (log not received)
IPS Packet Log: Tx & Rx
Content Archive: Tx & Rx
Quarantine: Tx & Rx
Certificate of Fortianalyzer valid and serial number is:FAZVCLTM20000000
```

4. When the FortiCloud Premium (AFAC) and standard FortiAnalyzer Cloud (FAZC) subscriptions are valid, the FortiGate sends the traffic, event, and UTM logs to the remote FortiAnalyzer Cloud.

Traffic:

```
# execute log filter device fortianalyzer-cloud
# execute log filter category traffic
# execute log filter dump
category: traffic
device: fortianalyzer-cloud
start-line: 1
view-lines: 10
max-checklines: 0
HA member:
Oftp search string:
# execute log display
6512 logs found.
10 logs returned.
1: date=2020-11-29 time=13:57:33 id=6900668351836585985 itime="2020-11-29 13:57:34"
      euid=3 epid=1027 dsteuid=3 dstepid=101 logflag=1 logver=604041797 type="traffic"
      subtype="forward" level="notice" action="accept" policyid=1 sessionid=46536
      srcip=10.1.100.72 dstip=172.16.100.55 transip=172.16.200.7 srcport=40797 dstport=53
      transport=40797 trandisp="snat" duration=190 proto=17 sentbyte=268 rcvdbyte=0
      sentpkt=4 rcvdpkt=0 logid=0000000013 service="DNS" app="DNS" appcat="unscanned"
      srcintfrole="undefined" dstintfrole="undefined" srcserver=0 dstserver=0
      policytype="policy" eventtime=1606687054554969021 poluuid="c041939c-2930-51eb-1448-
      34c44a663331" srcmac="00:0c:29:eb:86:d6" mastersrcmac="00:0c:29:eb:86:d6"
      dstmac="e8:1c:ba:c2:86:63" masterdstmac="e8:1c:ba:c2:86:63" srchwvendor="VMware"
      osname="Linux" srccountry="Reserved" dstcountry="Reserved" srcintf="dmz"
      dstintf="wan1" policyname="to_WAN" tz="-0800" devid="FG101FTK19000000" vd="root"
      dtime="2020-11-29 13:57:33" itime_t=1606687054 devname="FortiGate-101F_F"
```

Event:

```
# execute log filter device fortianalyzer-cloud
# execute log filter category event
# execute log filter dump
```

```
category: event
device: fortianalyzer-cloud
start-line: 1
view-lines: 10
max-checklines: 0
HA member:
Oftp search string:
# execute log display
1067 logs found.
10 logs returned.
1: date=2020-11-29 time=14:12:16 id=6900672144292708352 itime="2020-11-29 14:12:17"
     euid=3 epid=3 dsteuid=3 dstepid=3 logver=604041797 logid=0100038404 type="event"
     subtype="system" level="error" msg="unable to resolve FortiGuard hostname"
     logdesc="FortiGuard hostname unresolvable" hostname="service.fortiguard.net"
     eventtime=1606687936888734117 tz="-0800" devid="FG101FTK19000000" vd="root"
     dtime="2020-11-29 14:12:16" itime_t=1606687937 devname="FortiGate-101F_F"
```

UTM:

```
# execute log filter device fortianalyzer-cloud
# execute log filter category utm-virus
# execute log filter dump
category: virus
device: fortianalyzer-cloud
start-line: 1
view-lines: 10
max-checklines: 0
HA member:
Oftp search string:
# execute log display
4 logs found.
4 logs returned.
1: date=2020-11-27 time=15:53:41 id=6899956121704857638 itime="2020-11-27 15:53:45"
     euid=1027 epid=101 dsteuid=3 dstepid=101 logver=604041797 type="utm"
     subtype="virus" level="warning" action="passthrough" sessionid=1957747803
     policyid=1 srcip=168.10.199.186 dstip=172.252.3.20 srcport=22765 dstport=80 proto=6
     vrf=32 logid=0212008448 service="NNTP" user="user3" group="group1"
     eventtime=1606521221884991620 crscore=5 craction=2 crlevel="low"
     srcintfrole="undefined" dstintfrole="undefined" direction="incoming"
     filefilter="file-pattern" filetype="ignored" filename="file_test" checksum="12345"
     eventtype="filename" srcintf="ssl.root" dstintf="x1" msg="File is blocked." tz="-
     0800" devid="FG101FTK19000000" vd="root" dtime="2020-11-27 15:53:41" itime_
     t=1606521225 devname="FortiGate-101F_F"
```

5. When the FortiGate has a valid Premium FortiCloud subscription (AFAC) and an expired Standard FortiCloud subscription (FAZC), the FortiGate still sends the logs to the remote FortiAnalyzer Cloud.

Configuring multiple FortiAnalyzers (or syslog servers) per VDOM

In a VDOM, multiple FortiAnalyzer and syslog servers can be configured as follows:

- Up to three override FortiAnalyzer servers
- Up to four override syslog servers

If the VDOM `faz-override` and/or `syslog-override` setting is enabled or disabled (default) before upgrading, the setting remains the same after upgrading.

If the override setting is disabled, the GUI displays the global FortiAnalyzer1 or syslog1 setting. If the override setting is enabled, the GUI displays the VDOM override FortiAnalyzer1 or syslog1 setting.

You can only use CLI to enable the override to support multiple log servers.

To enable FortiAnalyzer and syslog server override under VDOM:

```
config log setting
    set faz-override enable
    set syslog-override enable
end
```

When `faz-override` and/or `syslog-override` is enabled, the following CLI commands are available for configuring VDOM override:

To configure VDOM override for FortiAnalyzer:

1. Configure the FortiAnalyzer override settings:

```
config log fortianalyzer/fortianalyzer2/fortianalyzer3 override-setting
    set status enable
    set server "123.12.123.123"
    set reliable enable
end
```

2. Configure the override filters:

```
config log fortianalyzer/fortianalyzer2/fortianalyzer3 override-filter
    set severity information
    set forward-traffic enable
    set local-traffic enable
    set multicast-traffic enable
    set sniffer-traffic enable
    set anomaly enable
    set voip enable
    set dlp-archive enable
    set dns enable
    set ssh enable
    set ssl enable
end
```

To configure VDOM override for a syslog server:

1. Configure the syslog override settings:

```
config log syslogd/syslogd2/syslogd3/syslogd4 override-setting
    set status enable
    set server "123.12.123.12"
    set facility local1
end
```

2. Configure the override filters:

```
config log syslogd/syslogd2/syslogd3/syslogd4 override-filter
    set severity information
    set forward-traffic enable
    set local-traffic enable
    set multicast-traffic enable
    set sniffer-traffic enable
    set anomaly enable
```

```
        set voip enable
        set dns enable
        set ssh enable
        set ssl enable
    end
```

Configuring multiple FortiAnalyzers on a FortiGate in multi-VDOM mode

This topic shows a sample configuration of multiple FortiAnalyzers on a FortiGate in multi-VDOM mode.

In this example:

- The FortiGate has three VDOMs:
  - Root (management VDOM)
  - VDOM1
  - VDOM2
- There are four FortiAnalyzers.
  These IP addresses are used as examples in the instructions below.
  - FAZ1: `172.16.200.55`
  - FAZ2: `172.18.60.25`
  - FAZ3: `192.168.1.253`
  - FAZ4: `192.168.1.254`
- Set up FAZ1 and FAZ2 under global.
  - These two collect logs from the root VDOM and VDOM2.
  - FAZ1 and FAZ2 must be accessible from management VDOM root.
- Set up FAZ3 and FAZ4 under VDOM1.
  - These two collect logs from VDOM1.
  - FAZ3 and FAZ4 must be accessible from VDOM1.

To set up FAZ1 as global FortiAnalyzer 1 from the GUI:

Prerequisite: FAZ1 must be reachable from the management root VDOM.

1. Go to *Global > Log & Report > Log Settings*.
2. Enable *Send logs to FortiAnalyzer/FortiManager*.
3. Enter the FortiAnalyzer IP.
   In this example: `172.16.200.55`.
4. For *Upload option*, select *Real Time*.
5. Click *Apply*.

To set up FAZ2 as global FortiAnalyzer 2 from the CLI:

Prerequisite: FAZ2 must be reachable from the management root VDOM.

```
config log fortianalyzer2 setting
    set status enable
    set server "172.18.60.25"
    set upload-option realtime
end
```

To set up FAZ3 and FAZ4 as VDOM1 FortiAnalyzer 1 and FortiAnalyzer 2:

Prerequisite: FAZ3 and FAZ4 must be reachable from VDOM1.

```
config log setting
   set faz-override enable
end

config log fortianalyzer override-setting
   set status enable
   set server "192.168.1.253"
   set upload-option realtime
end

config log fortianalyzer2 override-setting
   set status enable
   set server "192.168.1.254"
   set upload-option realtime
end
```

Checking FortiAnalyzer connectivity

To use the diagnose command to check FortiAnalyzer connectivity:

1. Check the global FortiAnalyzer status:

```
FGTA(global) # diagnose test application miglogd 1
faz: global , enabled
        server=172.16.200.55, realtime=3, ssl=1, state=connected, src=, mgmt_name=FGh_
Log_root_172.16.200.55, reliable=1
                status: ver=6, used_disk=0, total_disk=0, global=0, vfid=0 conn_
verified=N
                SNs: last sn update:1369 seconds ago.
                        Sn list:

                queue: qlen=0.
filter: severity=6, sz_exclude_list=0
         voip dns ssh ssl
subcategory:
        traffic: forward local multicast sniffer
        anomaly: anomaly

        server: global, id=0, fd=90, ready=1, ipv6=0, 172.16.200.55/514
        oftp-state=5
faz2: global , enabled
        server=172.18.60.25, realtime=1, ssl=1, state=connected, src=, mgmt_name=FGh_
Log_root_172.18.60.25, reliable=0
                status: ver=6, used_disk=0, total_disk=0, global=0, vfid=0 conn_
verified=N
                SNs: last sn update:1369 seconds ago.
                        Sn list:

                queue: qlen=0.
filter: severity=6, sz_exclude_list=0
         voip dns ssh ssl
subcategory:
```

```
        traffic: forward local multicast sniffer
        anomaly: anomaly

        server: global, id=1, fd=95, ready=1, ipv6=0, 172.18.60.25/514
        oftp-state=5
```

2. Check the VDOM1 override FortiAnalyzer status:

```
FGTA(global) # diagnose test application miglogd 3101
faz: vdom, enabled, override
        server=192.168.1.253, realtime=1, ssl=1, state=connected, src=, mgmt_name=FGh_
Log_root_192.168.1.253, reliable=1
                status: ver=6, used_disk=0, total_disk=0, global=0, vfid=0 conn_
verified=N
                SNs: last sn update:1369 seconds ago.
                        Sn list:
                        (FAZ-VM0000000001,age=17s)
                queue: qlen=0.
filter: severity=6, sz_exclude_list=0
         voip dns ssh ssl
subcategory:
        traffic: forward local multicast sniffer
        anomaly: anomaly

        server: vdom, id=0, fd=72, ready=1, ipv6=0, 192.168.1.253/514
        oftp-state=5
faz2: vdom, enabled, override
        server=192.168.1.254, realtime=1, ssl=1, state=connected, src=, mgmt_name=FGh_
Log_root_192.168.1.254, reliable=0
                status: ver=6, used_disk=0, total_disk=0, global=0, vfid=0 conn_
verified=N
                SNs: last sn update:1369 seconds ago.
                        Sn list:
                        (FL-1KET318000008,age=17s)
                queue: qlen=0.
filter: severity=6, sz_exclude_list=0
         voip dns ssh ssl
subcategory:
        traffic: forward local multicast sniffer
        anomaly: anomaly

        server: vdom, id=1, fd=97, ready=1, ipv6=0, 192.168.1.254/514
        oftp-state=5
faz3: vdom, disabled, override
```

Advanced and specialized logging

The following topics provide information on advanced and specialized logging:

- Logs for the execution of CLI commands on page 2565
- Log buffer on FortiGates with an SSD disk on page 2566
- Source and destination UUID logging on page 2569
- Configuring and debugging the free-style filter on page 2570

- Logging the signal-to-noise ratio and signal strength per client on page 2572
- RSSO information for authenticated destination users in logs on page 2574
- Destination user information in UTM logs on page 2577

Logs for the execution of CLI commands

The `cli-audit-log` option records the execution of CLI commands in system event logs (log ID 44548). In addition to `execute` and `config` commands, `show`, `get`, and `diagnose` commands are recorded in the system event logs.

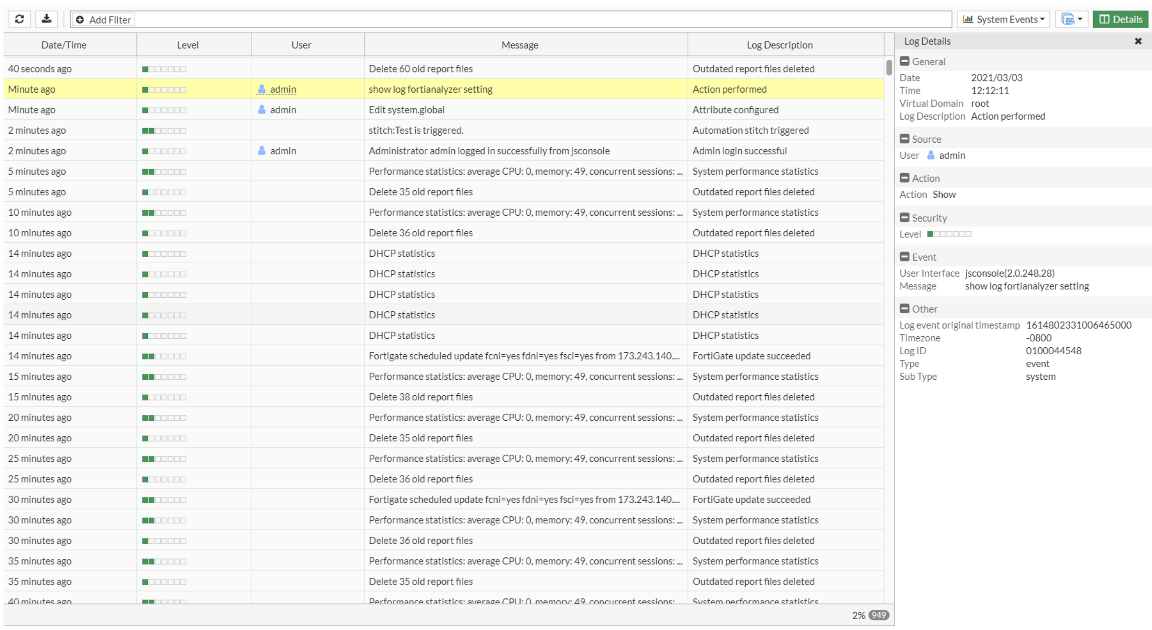
The `cli-audit-log` data can be recorded on memory or disk, and can be uploaded to FortiAnalyzer, FortiGate Cloud, or a syslog server.

To enable the CLI audit log option:

```
config system global
    set cli-audit-log enable
end
```

To view system event logs in the GUI:

1. Run the command in the CLI (`# show log fortianalyzer setting`).
2. Go to *Log & Report > System Events*.
3. Select *General System Events*.
4. Select the log entry and click *Details*.



To display the logs:

```
# execute log filter device disk
# execute log filter category event
# execute log filter field subtype system
```

```
# execute log filter field logid 0100044548
```

```
# execute log display
```

Sample log:

```
1: date=2020-11-16 time=10:43:00 eventtime=1605552179970875703 tz="-0800" logid="0100044548"
type="event" subtype="system" level="information" vd="root" logdesc="Action performed"
user="admin" ui="jsconsole(2.0.225.112)" action="Show" msg="show log fortianalyzer setting"
```

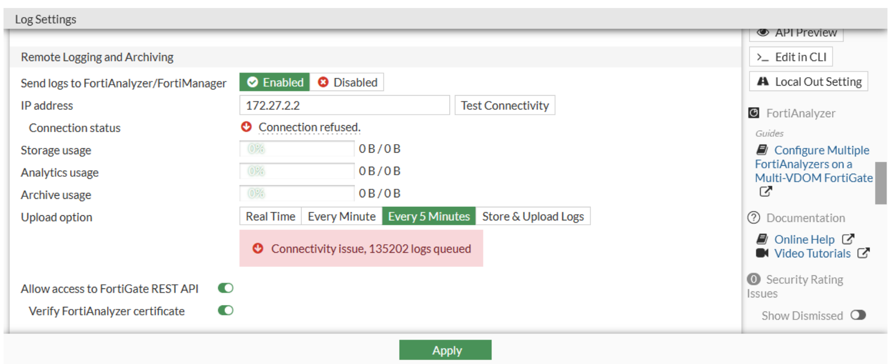
```
2: date=2020-11-16 time=10:42:43 eventtime=1605552163502003054 tz="-0800" logid="0100044548"
type="event" subtype="system" level="information" vd="root" logdesc="Action performed"
user="admin" ui="jsconsole(2.0.225.112)" action="Get" msg="get sys status"
```

```
3: date=2020-11-16 time=09:47:04 eventtime=1605548824762387718 tz="-0800" logid="0100044548"
type="event" subtype="system" level="information" vd="root" logdesc="Action performed"
user="admin" ui="jsconsole(2.0.228.202)" action="Diagnose" msg="diagnose log test"
```

Log buffer on FortiGates with an SSD disk

FortiGates with an SSD disk have a configurable log buffer. When the connection to FortiAnalyzer is unreachable, the FortiGate is able to buffer logs on disk if the memory log buffer is full. The logs queued on the disk buffer can be sent successfully once the connection to FortiAnalyzer is restored.

The number of logs queued on the disk buffer is visible in the *Log & Report* > *Log Settings* page:



The queued logs are buffered to the memory first and then disk. `Main miglogd` handles the disk buffering job, while `miglogd-children` handles the memory buffering. Disk buffer statistics only appear under `Main miglogd`, and memory buffer statistics only appears under `miglogd-children`. If the total buffer is full, new logs will overwrite the old logs.

To configure the log buffer:

1. Allocate disk space (MB) to temporarily store logs to FortiAnalyzer:

```
config system global
    set faz-disk-buffer-size 200
end
```

2. Check the `Main miglogd` and `miglogd-children` statistics. The 200 MB disk buffer has been set, and there are currently no logs buffered in memory or on disk when FortiAnalyzer is reachable:

```
# diagnose test application miglogd 41 0
cache maximum: 106100940(101MB) objects: 0 used: 0(0MB) allocated: 0(0MB)
VDOM:root
Queue for: global-faz

    memory queue:
        num:0 size:0(0MB) max:101906636(97MB) logs:0

    disk max queue size:200MB total:0MB
        totol items:0
        disk queue agents:
            devid:-1-10-0-1
            buffer path:/var/log/qbuf/10.0/1
            saved size:0MB cached size:0
            save roll:0 restore roll:0
            restore id:0 space:0MB

# diagnose test application miglogd 41 1
cache maximum: 106100940(101MB) objects: 0 used: 0(0MB) allocated: 0(0MB)
VDOM:root
Queue for: global-faz

    memory queue:
        num:0 size:0(0MB) max:101906636(97MB) logs:0

    disk queue client:
        devid:-1-10-0-1 status:buffering
        Total in cache:0 size:0(0MB) max:4MB logs:0
```

3. Disable the connection between the FortiGate and FortiAnalyzer. For example, delete the FortiGate from the FortiAnalyzer authorized device list.
   Assuming a massive number of logs (~ 300000) are recorded during this downtime, the logs will be queued in the memory buffer first. If the memory buffer is full, then the remaining logs will be queued on the disk buffer.
4. Check the `Main miglogd` and `miglogd-children` statistics again. All 97 MB of the memory buffer is occupied, and 76 of the 200 MB has been taken from the disk buffer:

```
# diagnose test application miglogd 41 0
cache maximum: 106100940(101MB) objects: 0 used: 0(0MB) allocated: 0(0MB)
VDOM:root
Queue for: global-faz

        memory queue:
                num:0 size:0(0MB) max:101906636(97MB) logs:0

        disk max queue size:200MB total:76MB
                totol items:128917
                disk queue agents:
                        devid:-1-10-0-1
                        buffer path:/var/log/qbuf/10.0/1
                        saved size:76MB cached size:3324984
                        save roll:19 restore roll:0
                        restore id:0 space:0MB

# diagnose test application miglogd 41 1
cache maximum: 106100940(101MB) objects: 165721 used: 101908358(97MB) allocated:
106449280(101MB)
```

```
VDOM:root
Queue for: global-faz

        memory queue:
                num:165718 size:101906500(97MB) max:101906636(97MB) logs:165718

        disk queue client:
                devid:-1-10-0-1 status:restoring
                restore id:1267 space:0MB
                Total in cache:3 size:1858(0MB) max:4MB logs:3
```

The overall `miglogd` statistics shows the total cached logs is the sum of the logs buffered in memory and on disk:

```
# diagnose test application miglogd 6
mem=0, disk=11, alert=0, alarm=0, sys=0, faz=300053, faz-cloud=0, webt=0, fds=0
interface-missed=44
Queues in all miglogds: cur:165718  total-so-far:165718
global log dev statistics:
faz 0: sent=0, failed=0, cached=300053, dropped=0 , relayed=0
Num of REST URLs: 0
```

5. Enable the connection between FortiAnalyzer and the FortiGate.
6. After a while, check the `miglogd` statistics to confirm that all buffered logs are being sent to FortiAnalyzer successfully:

```
# diagnose test application miglogd 6
mem=0, disk=11, alert=0, alarm=0, sys=0, faz=300058, faz-cloud=0, webt=0, fds=0
interface-missed=44
Queues in all miglogds: cur:4294832957  total-so-far:165726
global log dev statistics:
faz 0: sent=300058, failed=0, cached=0, dropped=0 , relayed=0
Num of REST URLs: 15

# diagnose test application miglogd 41 0
cache maximum: 106100940(101MB) objects: 0 used: 0(0MB) allocated: 0(0MB)
VDOM:root
Queue for: global-faz

        memory queue:
                num:0 size:0(0MB) max:101906636(97MB) logs:0

        disk max queue size:200MB total:0MB
                totol items:0
                disk queue agents:
                        devid:-1-10-0-1
                        buffer path:/var/log/qbuf/10.0/1
                        saved size:0MB cached size:0
                        save roll:20 restore roll:20
                        restore id:1267 space:0MB

# diagnose test application miglogd 41 1
cache maximum: 106100940(101MB) objects: 0 used: 0(0MB) allocated: 0(0MB)
VDOM:root
Queue for: global-faz

        memory queue:
                num:0 size:0(0MB) max:101906636(97MB) logs:0
```

```
disk queue client:
        devid:-1-10-0-1 status:buffering
        Total in cache:0 size:0(0MB) max:4MB logs:0
```

Source and destination UUID logging

The `log-uuid` setting in `system global` is split into two settings: `log-uuid-address` and `log-uuid policy`.

The traffic log includes two `internet-service` name fields: *Source Internet Service* (`srcinetsvc`) and *Destination Internet Service* (`dstinetsvc`).

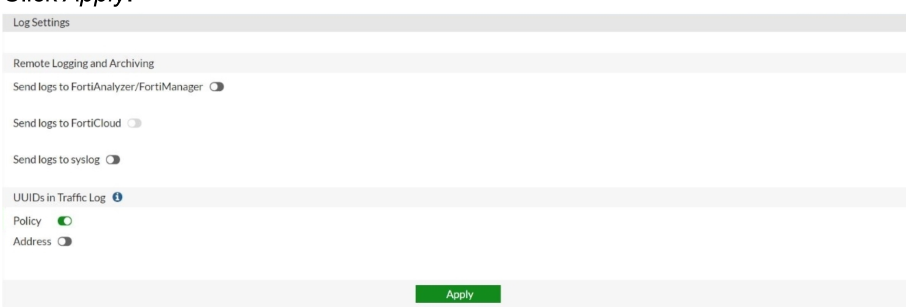
Log UUIDs

UUIDs can be matched for each source and destination that match a policy that is added to the traffic log. This allows the address objects to be referenced in log analysis and reporting.

As this may consume a significant amount of storage space, this feature is optional. By default, policy UUID insertion is enabled and address UUID insertion is disabled.

To enable address and policy UUID insertion in traffic logs using the GUI:

1. Go to *Log & Report > Log Settings*.
2. Under *UUIDs in Traffic Log*, enable *Policy* and/or *Address*.
3. Click *Apply*.



To enable address and policy UUID insertion in traffic logs using the CLI:

```
config system global
   set log-uuid-address enable
   set log-uuid-policy enable
end
```

Sample log

```
date=2019-01-25 time=11:32:55 logid="0000000013" type="traffic" subtype="forward"
level="notice" vd="vdom1" eventtime=1528223575srcip=192.168.1.183 srcname="PC24"
srcport=33709 srcintf="lan" srcintfrole="lan" dstip=192.168.70.184 dstport=80 dstintf="wan1"
dstintfrole="wan" srcuuid="27dd503e-883c-51e7-ade1-7e015d46494f" dstuuid="27dd503e-883c-
51e7-ade1-7e015d46494f" poluuid="9e0fe24c-1808-51e8-1257-68ce4245572c" sessionid=5181
proto=6 action="client-rst" policyid=4 policytype="policy" service="HTTP" trandisp="snat"
transip=192.168.70.228 transport=33709 appid=38783 app="Wget" appcat="General.Interest"
```

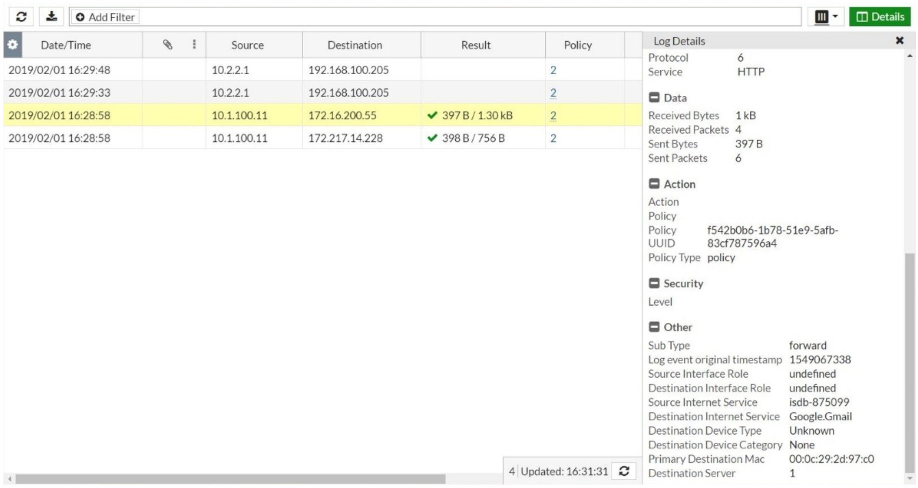
```
apprisk="low" applist="default" duration=5 sentbyte=450 rcvdbyte=2305 sentpkt=6 wanin=368
wanout=130 lanin=130 lanout=130 utmaction="block" countav=2 countapp=1 crscore=50 craction=2
devtype="Linux PC" devcategory="None" osname="Linux" mastersrcmac="00:0c:29:36:5c:c3"
srcmac="00:0c:29:36:5c:c3" srcserver=0 utmref=65523-1018
```

Internet service name fields

Traffic logs for `internet-service` include two fields: *Source Internet Service* and *Destination Internet Service*.

To view the internet service fields using the GUI:

1. Go to *Log & Report* > *Forward Traffic*.
2. Double-click on an entry to view the *Log Details*. The *Source Internet Service* and *Destination Internet Service* fields are visible in the *Log Details* pane.



Sample log

```
date=2019-01-25 time=14:17:04 logid="0000000013" type="traffic" subtype="forward"
level="notice" vd="vdom1" eventtime=1548454622
srcip=10.1.100.11 srcport=51112 srcintf="port3" srcintfrole="undefined" dstip=172.217.14.228
dstport=80 dstintf="port1" dstintfrole="undefined" poluuid="af519380-2094-51e9-391c-
b78e8edbddfc" srcinetsvc="isdb-875099" dstinetsvc="Google.Gmail" sessionid=6930 proto=6
action="close" policyid=2 policytype="policy" service="HTTP" dstcountry="United States"
srccountry="Reserved" trandisp="snat" transip=172.16.200.2 transport=51112 duration=11
sentbyte=398 rcvdbyte=756 sentpkt=6 rcvdpkt=4 appcat="unscanned" devtype="Router/NAT Device"
devcategory="Fortinet Device" mastersrcmac="90:6c:ac:41:7a:24" srcmac="90:6c:ac:41:7a:24"
srcserver=0 dstdevtype="Unknown" dstdevcategory="Fortinet Device"
masterdstmac="08:5b:0e:1f:ed:ed" dstmac="08:5b:0e:1f:ed:ed" dstserver=0
```

Configuring and debugging the free-style filter

Free-style filters allow users to define a filter for logs that are captured to each individual logging device type. Filters can include log categories and specific log fields. The filters can be created as an inclusive list or exclusive list.

Free-style filters can also be used to filter logs that have been captured on logging devices already to narrow down the list of logs to view.

```
config log syslogd filter
    config free-style
        edit <id>
            set category <option>
            set filter <string>
            set filter-type {include | exclude}
        next
    end
end
```

`category <option>`	Set the log category. The following options are available: `traffic`, `event`, `virus`, `webfilter`, `attack`, `spam`, `anomaly`, `voip`, `dlp`, `app-ctrl`, `waf`, `dns`, `ssh`, `ssl`, `file-filter`, `icap`, and `ztna`.
`filter <string>`	Enter the filter criteria. Multiple values can be added, for example: `set filter "logid <id> <id>"`
`filter-type {include | exclude}`	Include/exclude logs that match the filter.

Use the following commands to view the results when multiple fields are used:

```
# execute log filter free-style "logid <id> <id>"
```

```
# execute log filter free-style "srcip <IP_address> <IP_address>"
```

```
# execute log filter free-style "(logid <id>) or (srcip <IP_address> <IP_address>)"
```

```
# execute log filter free-style "(srcip <IP_address>) and (dstip <IP_address>)"
```

In this example, the free-style filter is set to filter log IDs 0102043039 and 0102043040. The source IPs, 192.168.2.5 and 192.168.2.205, are also checked.

To configure the syslogd free-style filter with multiple values:

```
config log syslogd filter
    config free-style
        edit 1
            set category event
            set filter "logid 0102043039 0102043040"
        next
    end
end
```

To view the syslogd free-style filter results:

```
# execute log filter free-style "logid 0102043039 0102043040"
# execute log filter dump
category: event
device: disk
start-line: 1
view-lines: 10
max-checklines: 0
HA member:
log search mode: on-demand
pre-fetch-pages: 2
```

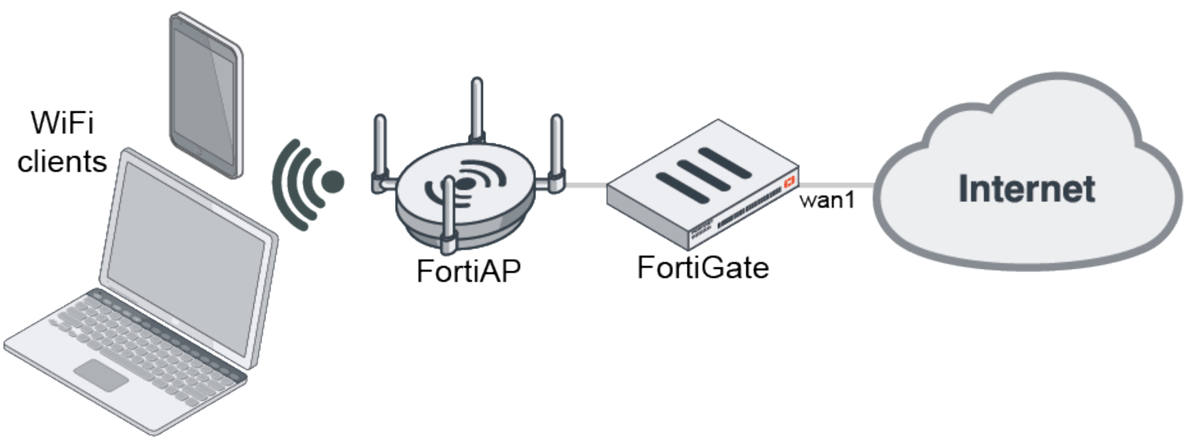
```
Filter: logid 0102043039 0102043040
Oftp search string: (and (or logid=="0102043039" not-exact logid=="0102043040" not-exact))
```

```
# execute log filter free-style "(logid 0102043039) or (srcip 192.168.2.5 192.168.2.205)"
# execute log filter dump
category: event
device: disk
start-line: 1
view-lines: 10
max-checklines: 0
HA member:
log search mode: on-demand
pre-fetch-pages: 2
Filter: (logid 0102043039) or (srcip 192.168.2.5 192.168.2.205)
Oftp search string: (or (or (or srcip==192.168.2.5) (or srcip==192.168.2.205)) (or
logid=="0102043039" not-exact))
```

Logging the signal-to-noise ratio and signal strength per client

The signal-to-noise ratio (`snr`) and signal strength (`signal`) are logged per client in the WiFi event and traffic logs.

When a WiFi client connects to a tunnel or local-bridge mode SSID on an FortiAP that is managed by a FortiGate, signal-to-noise ratio and signal strength details are included in WiFi event logs for local-bridge traffic statistics and authentication, and in forward traffic logs for tunnel traffic. This allows you to store and view clients' historical signal strength and signal-to-noise ratio information.



To verify when a client is connecting to an SSID:

1. Go to *Log & Report > System Events* and select *WiFi Events*.
   The *Signal* and *Signal/Noise* columns show the signal strength and signal-to-noise ratio for each applicable client.

Add Filter | WiFi Events | Details

Date/Time	Level	Action	Message	SSID	Channel	Signal	Signal/Noise
2020/05/29 10:00:16		fake-ap-on-air	Fake AP On-air starr-ssid.fap.02 90:6c:ac:8a:69:41 chan 44 live ...	starr-ssid.fap.02	44	-34	
2020/05/29 10:00:15		DHCP-ACK	DHCP ACK for IP 11.10.80.2 from server 11.10.80.1 for client 4...	FOS_QA_Starr_140E_Guest-11			
2020/05/29 10:00:15		DHCP-REQUEST	DHCP REQUEST for IP 11.10.80.2 offered by server 11.10.80.1 ...	FOS_QA_Starr_140E_Guest-11			
2020/05/29 10:00:15		DHCP-OFFER	DHCP OFFER of IP 11.10.80.2 from server 11.10.80.1 for client ...	FOS_QA_Starr_140E_Guest-11			
2020/05/29 10:00:14		client-ip-detected	Client 48:ee:0c:23:43:d1 had an IP address detected (by DHCP ...	FOS_QA_Starr_140E_Guest-11	6	-45	50
2020/05/29 10:00:14		DHCP-DISCOVER	DHCP DISCOVER from client 48:ee:0c:23:43:d1	FOS_QA_Starr_140E_Guest-11			
2020/05/29 10:00:04		client-authentication	Client 48:ee:0c:23:43:d1 authenticated.	FOS_QA_Starr_140E_Guest-11	6	-45	50
2020/05/29 10:00:04		WPA-4/4-key-msg	AP received 4/4 message of 4-way handshake from client 48:ee:...	FOS_QA_Starr_140E_Guest-11	6		
2020/05/29 10:00:04		WPA-3/4-key-msg	AP sent 3/4 message of 4-way handshake to client 48:ee:0c:23:...	FOS_QA_Starr_140E_Guest-11	6		
2020/05/29 10:00:04		WPA-2/4-key-msg	AP received 2/4 message of 4-way handshake from client 48:ee:...	FOS_QA_Starr_140E_Guest-11	6		
2020/05/29 10:00:04		WPA-1/4-key-msg	AP sent 1/4 message of 4-way handshake to client 48:ee:0c:23:...	FOS_QA_Starr_140E_Guest-11	6		
2020/05/29 10:00:04		assoc-resp	AP sent association response frame to client 48:ee:0c:23:43:d1	FOS_QA_Starr_140E_Guest-11	6		
2020/05/29 10:00:04		assoc-req	AP received association request frame from client 48:ee:0c:23:4...	FOS_QA_Starr_140E_Guest-11	6		
2020/05/29 10:00:04		auth-resp	AP sent authentication response frame to client 48:ee:0c:23:43:...	FOS_QA_Starr_140E_Guest-11	6		
2020/05/29 10:00:04		auth-req	AP received authentication request frame from client 48:ee:0c:...	FOS_QA_Starr_140E_Guest-11	6		
2020/05/29 09:59:30		oper-txpower	AP FP231ETF20000455 radio 1 oper txpower is changed to 26 ...				
2020/05/29 09:59:28		oper-txpower	AP FP231ETF20000455 radio 1 oper txpower is changed to 4 d...				
2020/05/29 09:59:24		config-txpower	AP FP231ETF20000455 radio 1 cfg txpower is changed to 27 d...				
2020/05/29 09:58:46		fake-ap-on-air	Fake AP On-air starr-ssid.fap.02 90:6c:ac:8a:69:41 chan 44 live ...	starr-ssid.fap.02	44	-34	
2020/05/29 09:57:16		fake-ap-on-air	Fake AP On-air starr-ssid.fap.02 90:6c:ac:8a:69:41 chan 44 live ...	starr-ssid.fap.02	44	-34	
2020/05/29 09:55:46		fake-ap-on-air	Fake AP On-air starr-ssid.fap.02 90:6c:ac:8a:69:41 chan 44 live ...	starr-ssid.fap.02	44	-34	

0% 108

2. WiFi event log messages include the `signal` and `snr` values:

```
date=2020-05-27 time=11:26:28 logid="0104043579" type="event" subtype="wireless"
level="notice" vd="vdom1" eventtime=1590603988877156921 tz="-0700" logdesc="Wireless
client IP assigned" sn="FP231ETF20000455" ap="FP231ETF20000455" vap="stability3"
ssid="FOS_QA_Starr_140E_Guest-11" radioid=1 user="N/A" group="N/A"
stamac="1c:87:2c:b6:a8:49" srcip=11.10.80.2 channel=6 radioband="802.11n,g-only"
signal=-45 snr=50 security="WPA2 Personal" encryption="AES" action="client-ip-detected"
reason="Reserved 0" mpsk="N/A" msg="Client 1c:87:2c:b6:a8:49 had an IP address detected
(by DHCP packets)."
```

```
date=2020-05-27 time=11:26:11 logid="0104043573" type="event" subtype="wireless"
level="notice" vd="vdom1" eventtime=1590603970962702892 tz="-0700" logdesc="Wireless
client authenticated" sn="FP231ETF20000455" ap="FP231ETF20000455" vap="stability3"
ssid="FOS_QA_Starr_140E_Guest-11" radioid=1 user="N/A" group="N/A"
stamac="1c:87:2c:b6:a8:49" srcip=0.0.0.0 channel=6 radioband="802.11n,g-only" signal=-45
snr=50 security="WPA2 Personal" encryption="AES" action="client-authentication"
reason="Reserved 0" mpsk="N/A" msg="Client 1c:87:2c:b6:a8:49 authenticated."
```

To verify tunnel traffic when a client is connecting to a tunnel mode SSID:

1. Go to *Log & Report > Forward Traffic*.
   The *Signal* and *Signal/Noise* columns show the signal strength and signal-to-noise ratio for each applicable client.

Add Filter | Details

Date/Time		Source	Device	Destination	Application Name	Result	Policy ID	Signal	Signal/Nois
2020/05/29 10:19:04		11.10.80.3	00:1e:e5:df:b1:63	108.175.12.139 (img2.westca.com)	HTTPS.BROWSER	938 B / 389 B	wmm (13)	-32	62
2020/05/29 10:19:04		11.10.80.3	00:1e:e5:df:b1:63	108.175.12.139 (img2.westca.com)	HTTPS.BROWSER	938 B / 389 B	wmm (13)	-32	62
2020/05/29 10:19:02		11.10.80.6	WIFI23	142.232.230.11 (www.bcit.ca)	SSL_TLSv1.2	3.67 kB / 97.47 kB	wmm (13)	-30	64
2020/05/29 10:18:58		11.10.80.3	00:1e:e5:df:b1:63	108.175.12.139 (img2.westca.com)	HTTPS.BROWSER	938 B / 389 B	wmm (13)	-32	62
2020/05/29 10:18:51		11.10.80.3	00:1e:e5:df:b1:63	149.7.32.209 (widgetdata-backup.tradingview.com)	SSL_TLSv1.2	255.25 kB / 903.92 kB	wmm (13)	-32	62
2020/05/29 10:18:46		11.10.80.3	00:1e:e5:df:b1:63	108.175.12.139 (img2.westca.com)	HTTPS.BROWSER	938 B / 389 B	wmm (13)	-34	60
2020/05/29 10:18:46		11.10.80.6	WIFI23	172.18.56.163	HTTP.BROWSER	397 B / 669 B	wmm (13)	-30	64
2020/05/29 10:18:35		11.10.80.3	00:1e:e5:df:b1:63	172.16.100.100	DNS	59 B / 292 B	wmm (13)	-34	60
2020/05/29 10:18:35		11.10.80.3	00:1e:e5:df:b1:63	172.16.100.100	DNS	63 B / 240 B	wmm (13)	-34	60
2020/05/29 10:18:35		11.10.80.3	00:1e:e5:df:b1:63	172.16.100.100	DNS	59 B / 166 B	wmm (13)	-34	60
2020/05/29 10:18:35		11.10.80.3	00:1e:e5:df:b1:63	172.16.100.100	DNS	59 B / 292 B	wmm (13)	-34	60
2020/05/29 10:18:35		11.10.80.3	00:1e:e5:df:b1:63	172.16.100.100	DNS	59 B / 292 B	wmm (13)	-34	60
2020/05/29 10:18:35		11.10.80.3	00:1e:e5:df:b1:63	65.39.243.196 (www.everforex.ca)	HTTPS.BROWSER	596.72 kB / 2.97 MB	wmm (13)	-34	60
2020/05/29 10:18:34		11.10.80.3	00:1e:e5:df:b1:63	108.175.12.139 (img2.westca.com)	HTTPS.BROWSER	936 B / 429 B	wmm (13)	-34	60
2020/05/29 10:18:32		11.10.80.3	00:1e:e5:df:b1:63	172.16.100.100	DNS	79 B / 243 B	wmm (13)	-34	60
2020/05/29 10:18:32		11.10.80.3	00:1e:e5:df:b1:63	172.16.100.100	DNS	79 B / 243 B	wmm (13)	-34	60
2020/05/29 10:18:32		11.10.80.3	00:1e:e5:df:b1:63	172.16.100.100	DNS	59 B / 267 B	wmm (13)	-34	60
2020/05/29 10:18:32		11.10.80.3	00:1e:e5:df:b1:63	172.16.100.100	DNS	59 B / 157 B	wmm (13)	-34	60
2020/05/29 10:18:31		11.10.80.3	00:1e:e5:df:b1:63	172.16.100.100	DNS	59 B / 267 B	wmm (13)	-34	60
2020/05/29 10:18:31		11.10.80.3	00:1e:e5:df:b1:63	172.16.100.100	DNS	59 B / 267 B	wmm (13)	-34	60

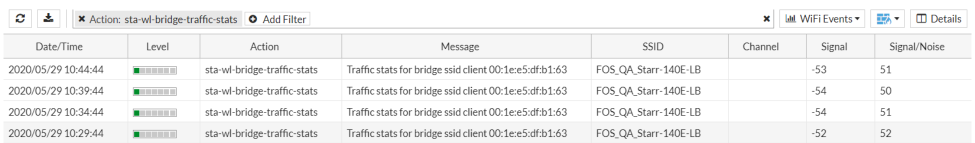
0% 975

2. Forward traffic log messages include the `signal` and `snr` values:

```
date=2020-05-27 time=11:30:26 logid="0000000013" type="traffic" subtype="forward"
level="notice" vd="vdom1" eventtime=1590604226533016978 tz="-0700" srcip=11.10.80.2
srcname="WIFI23" srcport=53926 srcintf="stability3" srcintfrole="lan" srcssid="FOS_QA_
Starr_140E_Guest-11" apsn="FP231ETF20000455" ap="FP231ETF20000455" channel=6
radioband="802.11n,g-only" signal=-31 snr=64 dstip=91.189.91.157 dstport=123
dstintf="wan1" dstintfrole="wan" srccountry="United States" dstcountry="United States"
sessionid=322069 proto=17 action="accept" policyid=13 policytype="policy"
poluuid="7c14770c-1456-51e9-4c57-806e9c499782" policyname="wmm" service="NTP"
trandisp="snat" transip=172.16.200.111 transport=53926 appid=16270 app="NTP"
appcat="Network.Service" apprisk="elevated" applist="g-default" duration=180 sentbyte=76
rcvdbyte=76 sentpkt=1 rcvdpkt=1 utmaction="allow" countapp=1 osname="Linux"
mastersrcmac="1c:87:2c:b6:a8:49" srcmac="1c:87:2c:b6:a8:49" srcserver=0 utmref=65534-66
```

To verify local-bridge traffic statistics when a client is connecting to a local-bridge mode SSID:

1. Go to *Log & Report > System Events* and select *WiFi Events*.
   The *Signal* and *Signal/Noise* columns show the signal strength and signal-to-noise ratio for each applicable client.



Date/Time	Level	Action	Message	SSID	Channel	Signal	Signal/Noise
2020/05/29 10:44:44		sta-wl-bridge-traffic-stats	Traffic stats for bridge ssid client 00:1e:e5:df:b1:63	FOS_QA_Starr-140E-LB		-53	51
2020/05/29 10:39:44		sta-wl-bridge-traffic-stats	Traffic stats for bridge ssid client 00:1e:e5:df:b1:63	FOS_QA_Starr-140E-LB		-54	50
2020/05/29 10:34:44		sta-wl-bridge-traffic-stats	Traffic stats for bridge ssid client 00:1e:e5:df:b1:63	FOS_QA_Starr-140E-LB		-54	51
2020/05/29 10:29:44		sta-wl-bridge-traffic-stats	Traffic stats for bridge ssid client 00:1e:e5:df:b1:63	FOS_QA_Starr-140E-LB		-52	52

2. WiFi event log messages include the `signal` and `snr` values:

```
date=2020-05-26 time=17:48:57 logid="0104043687" type="event" subtype="wireless"
level="information" vd="vdom1" eventtime=1590540537841497433 tz="-0700" logdesc="Traffic
stats for station with bridge wlan" sn="FP231ETF20000455" ap="FP231ETF20000455"
vap="wifi.fap.01" ssid="FOS_QA_Starr-140E-LB-cap-2" srcip=10.128.100.4 user="N/A"
stamac="00:1e:e5:df:b1:63" signal=-53 snr=52 sentbyte=8970016 rcvdbyte=985910
nextstat=300 action="sta-wl-bridge-traffic-stats" msg="Traffic stats for bridge ssid
client 00:1e:e5:df:b1:63"
```

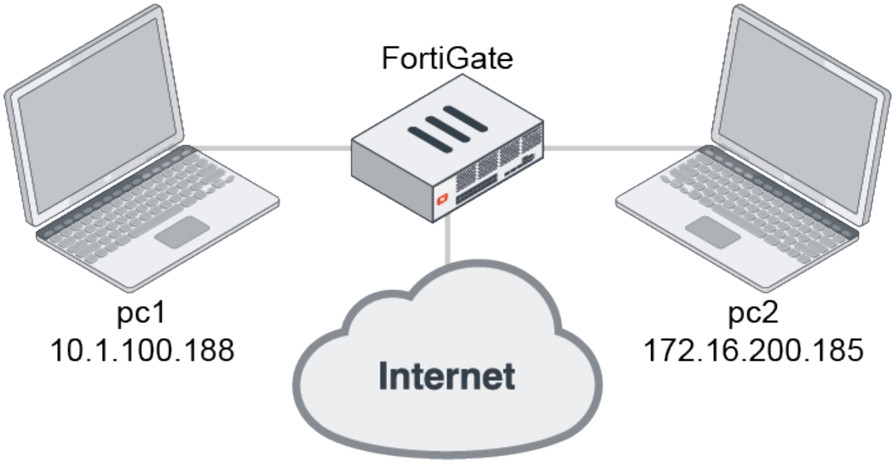
RSSO information for authenticated destination users in logs

FortiGate can use RSSO accounting information from authenticated RSSO users to populate destination users and groups, along with source users and groups.

RSSO user login information can be forwarded by the RADIUS server to the FortiGate that is listening for incoming RADIUS accounting start messages on the RADIUS accounting port. Accounting start messages usually contain the IP address, user name, and user group information. FortiGate uses this information in traffic logs, which include *dstuser* and *dstgroup* fields for user and group destination information .

For instructions on configuring RSSO, see RADIUS single sign-on agent on page 2498.

The three following scenarios show traffic between pc1 and the internet, and pc1 and pc2.



Scenario 1

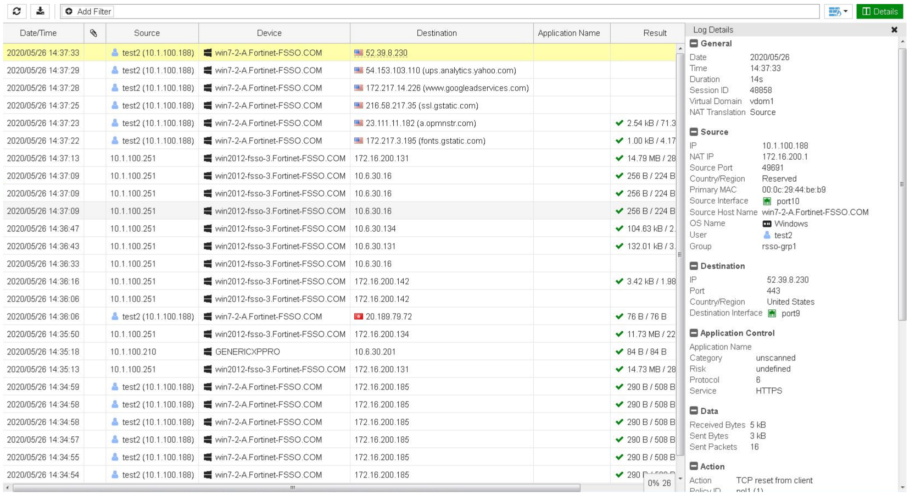
In this scenario, RSSO user *test2* in group *rsso-grp1* is authenticated on pc1. Traffic flows from pc1 to the internet.

Expected result:

In the logs, user *test2* is shown as the source user in the *rsso-grp1* group.

To verify the results:

1. In the GUI, go to *Log & Report > Forward Traffic* and view the details of an entry with test2 as the source.
2. In the *Source* section, *User* is *test2* and *Group* is the *rsso-grp1*.



3. The log message shows the user and group:

```
10: date=2020-05-25 time=15:34:43 logid="0000000013" type="traffic" subtype="forward"
level="notice" vd="vdom1" eventtime=1590446083718007055 tz="-0700" srcip=10.1.100.188
srcname="win7-2-A.Fortinet-FSSO.COM" srcport=56982 srcintf="port10"
srcintfrole="undefined" dstip=172.217.3.195 dstport=443 dstintf="port9"
dstintfrole="undefined" srccountry="Reserved" dstcountry="United States"
sessionid=120651 proto=17 action="accept" policyid=1 policytype="policy"
```

```
poluuid="d130f886-9ec6-51ea-206e-8c561c5244c6" policyname="pol1" user="test2"
group="rsso-grp1" authserver="vdom1" service="udp/443" trandisp="snat"
transip=172.16.200.1 transport=56982 duration=181 sentbyte=2001 rcvdbyte=1820 sentpkt=6
rcvdpkt=4 appcat="unscanned" sentdelta=0 rcvddelta=0 srchwvendor="VMware"
osname="Windows" srcswversion="7" mastersrcmac="00:0c:29:44:be:b9"
srcmac="00:0c:29:44:be:b9" srcserver=0
```

Scenario 2

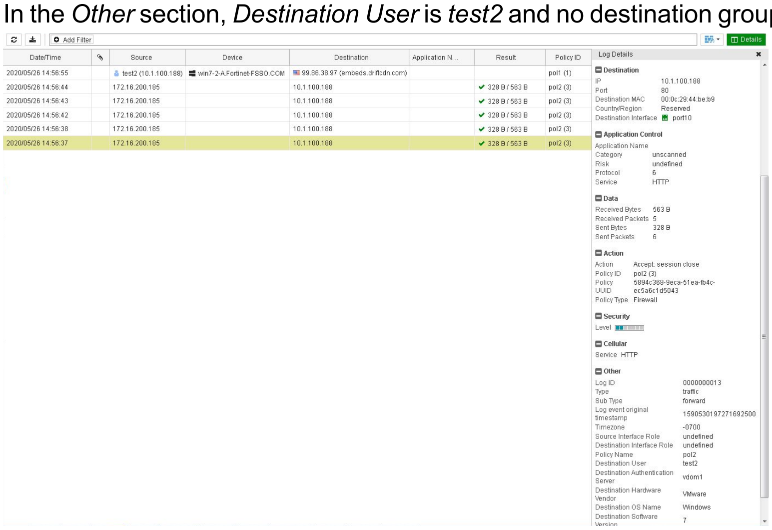
In this scenario, RSSO user *test2* is authenticated on pc1. Traffic is initialized on pc2 (172.16.200.185) going to pc1 (10.1.100.188).

Expected result:

In the logs, user *test2* is shown as the destination user (`dstuser`). No destination group (`dstgroup`) is logged because no RSSO user is logged in on pc2, so the traffic from pc2 is unauthenticated.

To verify the results:

1. In the GUI, go to *Log & Report > Forward Traffic* and view the details of an entry with 172.16.200.185 (pc2) as the source.
2. In the *Other* section, *Destination User* is *test2* and no destination group is shown.



3. The log message shows the destination user:

```
1: date=2020-05-22 time=07:38:06 logid="0000000020" type="traffic" subtype="forward"
level="notice" vd="root" eventtime=1590158286585506922 tz="-0700" srcip=172.16.200.185
identifier=1 srcintf="port9" srcintfrole="undefined" dstip=10.1.100.188 dstintf="port10"
dstintfrole="undefined" srccountry="Reserved" dstcountry="Reserved" sessionid=4395
proto=1 action="accept" policyid=3 policytype="policy" poluuid="d4f18e1e-9c36-51ea-6ec0-
3a354d5910ee" policyname="pol2" dstuser="test2" dstauthserver="root" service="PING"
trandisp="snat" transip=10.1.100.1 transport=0 duration=128 sentbyte=7620 rcvdbyte=5220
sentpkt=127 rcvdpkt=87 appcat="unscanned" sentdelta=7620 rcvddelta=5220
```

Scenario 3

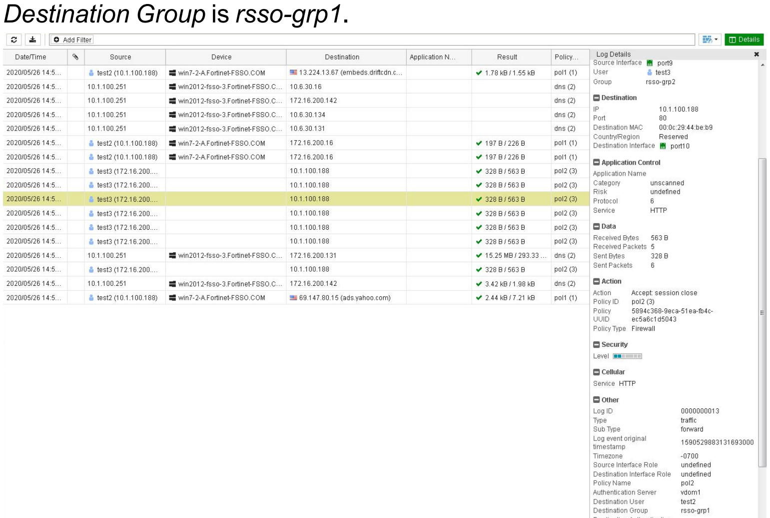
In this scenario, RSSO user *test2* in group *rsso-grp1* is authenticated on pc1, and user *test3* in group *rsso-grp2* is authenticated on pc2. Traffic flows from pc2 to pc1.

Expected result:

In the logs, user *test3* is shown as the source user in the *rsso-grp1* group. User *test2* is shown as destination user (`dstuser`) in the *rsso-grp1* destination group (`dstgroup`). The destination group is logged because an RSSO user is logged in to pc2.

To verify the results:

1. In the GUI, go to *Log & Report > Forward Traffic* and view the details of an entry with 172.16.200.185 (pc2) as the source.
2. In the *Source* section, *User* is *test3* and *Group* is the *rsso-grp2*. In the *Other* section, *Destination User* is *test2* and *Destination Group* is *rsso-grp1*.



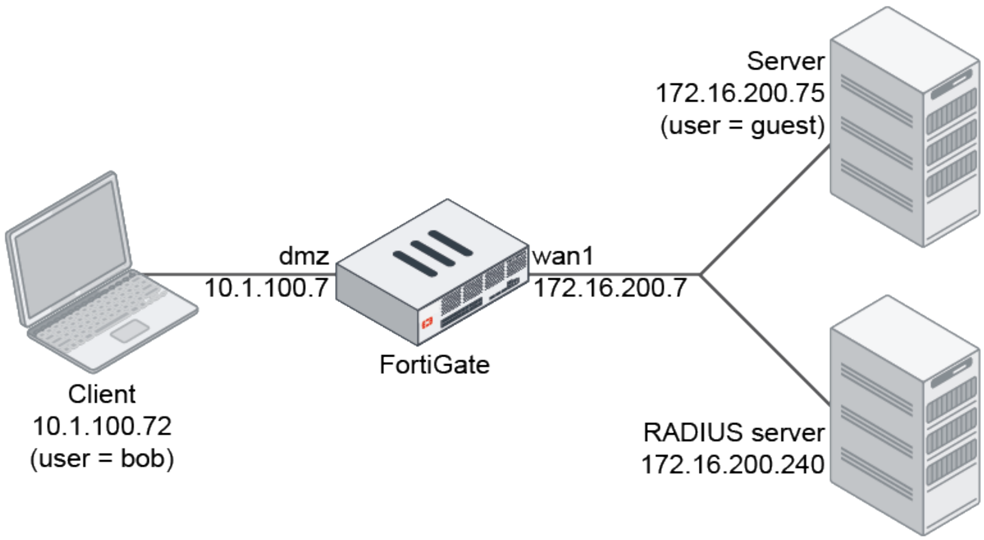
3. The log message shows both the source and the destination users and groups:

```
8: date=2020-05-25 time=14:23:07 logid="0000000013" type="traffic" subtype="forward"
level="notice" vd="vdom1" eventtime=1590441786958007914 tz="-0700" srcip=172.16.200.185
srcport=64096 srcintf="port9" srcintfrole="undefined" dstip=10.1.100.188 dstport=80
dstintf="port10" dstintfrole="undefined" srccountry="Reserved" dstcountry="Reserved"
sessionid=112445 proto=6 action="close" policyid=3 policytype="policy"
poluuid="5894c368-9eca-51ea-fb4c-ec5a6c1d5043" policyname="pol2" user="test3"
group="rsso-grp2" authserver="vdom1" dstuser="test2" dstgroup="rsso-grp1"
dstauthserver="vdom1" service="HTTP" trandisp="snat" transip=10.1.100.1 transport=64096
duration=1 sentbyte=328 rcvdbyte=563 sentpkt=6 rcvdpkt=5 appcat="unscanned"
dsthwvendor="VMware" dstosname="Windows" dstswversion="7"
masterdstmac="00:0c:29:44:be:b9" dstmac="00:0c:29:44:be:b9" dstserver=0
```

Destination user information in UTM logs

The `dstuser` field in UTM logs records the username of a destination device when that user has been authenticated on the FortiGate.

In the following example topology, the user, bob, is authenticated on a client computer. The user, guest, is authenticated on the server. Log are collected for AV and IPS in flow inspection mode. Logs are collected for application control and web filter in proxy mode.



To configure the RADIUS user and user groups:

1. Configure the RADIUS server:

```
config user radius
    edit "Ubuntu_docker"
        set server "172.16.200.240"
        set secret ************
    next
end
```

2. Configure the local user:

```
config user local
    edit "guest"
        set type password
        set passwd ************
    next
end
```

3. Configure the RADIUS user groups:

```
config user group
    edit "RADIUS_User_Group"
        set member "Ubuntu_docker"
    next
    edit "Local_User"
        set member "guest"
    next
end
```

Flow inspection mode

To verify AV and IPS logs in flow mode:

1. Configure the firewall policies:

```
config firewall policy
    edit 1
        set name "WAN_out"
        set srcintf "dmz"
        set dstintf "wan1"
        set action accept
        set srcaddr "all"
        set dstaddr "all"
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set ssl-ssh-profile "deep-inspection"
        set av-profile "g-default"
        set ips-sensor "sensor-11"
        set nat enable
        set groups "RADIUS_User_Group" "Local_User"
    next
    edit 3
        set name "WAN_in"
        set srcintf "wan1"
        set dstintf "dmz"
        set action accept
        set srcaddr "all"
        set dstaddr "all"
        set schedule "always"
        set service "ALL"
        set logtraffic all
        set nat enable
        set groups "RADIUS_User_Group" "Local_User"
    next
end
```

2. Verify the AV log:

```
date=2021-09-14 time=16:37:25 eventtime=1631662646131356720 tz="-0700"
logid="0211008192" type="utm" subtype="virus" eventtype="infected" level="warning"
vd="vdom1" policyid=1 msg="File is infected." action="blocked" service="HTTP"
sessionid=4613 srcip=10.1.100.72 dstip=172.16.200.75 srcport=60086 dstport=80
srcintf="dmz" srcintfrole="undefined" dstintf="wan1" dstintfrole="undefined"
srcuuid="877d43a4-c2f9-51eb-f78f-e09794924d8a" dstuuid="877d43a4-c2f9-51eb-f78f-
e09794924d8a" proto=6 direction="incoming" filename="eicar.com" quarskip="Quarantine-
disabled" virus="EICAR_TEST_FILE" viruscat="Virus" dtype="av-engine"
ref="http://www.fortinet.com/ve?vn=EICAR_TEST_FILE" virusid=2172
url="http://172.16.200.75/eicar.com" profile="g-default" user="bob" group="RADIUS_User_
Group" authserver="Ubuntu_docker" dstuser="guest" agent="Wget/1.17.1"
analyticscksum="275a021bbfb6489e54d471899f7db9d1663fc695ec2fe2a2c4538aabf651fd0f"
analyticssubmit="false" crscore=50 craction=2 crlevel="critical"
```

3. Verify the IPS log:

```
date=2021-09-14 time=16:56:06 eventtime=1631663765992499880 tz="-0700"
logid="0419016384" type="utm" subtype="ips" eventtype="signature" level="alert"
vd="vdom1" severity="info" srcip=10.1.100.72 srccountry="Reserved" dstip=172.16.200.75
srcintf="dmz" srcintfrole="undefined" dstintf="wan1" dstintfrole="undefined"
sessionid=7881 action="dropped" proto=6 service="HTTP" policyid=1
attack="Eicar.Virus.Test.File" srcport=60092 dstport=80 direction="incoming"
attackid=29844 profile="sensor-11" ref="http://www.fortinet.com/ids/VID29844" user="bob"
group="RADIUS_User_Group" authserver="Ubuntu_docker" dstuser="guest"
incidentserialno=17825794 attackcontextid="2/2"
attackcontext="dGVudC1MZW5ndGg6IDY4DQpLZWVwLUFsaXZlOiB0aW1lb3V0PTUsIG1heD0xMDANCkNvbm5lY
3Rpb246IEtlZXAtQWxpdmUNCkNvbnRlbnQtVHlwZTogYXBwbGljYXRpb24veC1tc2Rvcy1wcm9ncmFtDQoNClg1T
yFQJUBBUFs0XFBaWDU0KFBeKTdDQyk3fSRFSUNBUi1TVEFOREFSRC1BTlRJVklSVVMtVEVTVC1GSUxFISRIK0gqP
C9QQUNLRVQ+"
```

Proxy inspection mode

To verify application control and web filter logs in proxy mode:

1. Configure the firewall policies:

```
config firewall policy
    edit 1
        set name "WAN_out"
        set srcintf "dmz"
        set dstintf "wan1"
        set action accept
        set srcaddr "all"
        set dstaddr "all"
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set inspection-mode proxy
        set ssl-ssh-profile "deep-inspection"
        set av-profile "g-default"
        set application-list "g-default"
        set webfilter-profile "1"
        set nat enable
        set groups "RADIUS_User_Group" "Local_User"
    next
    edit 3
        set name "WAN_in"
        set srcintf "wan1"
        set dstintf "dmz"
        set action accept
        set srcaddr "all"
        set dstaddr "all"
        set schedule "always"
        set service "ALL"
        set inspection-mode proxy
        set logtraffic all
        set nat enable
        set groups "RADIUS_User_Group" "Local_User"
    next
end
```

2. Verify the application control log:

```
date=2021-09-14 time=17:05:45 eventtime=1631664345570951500 tz="-0700"
logid="1059028704" type="utm" subtype="app-ctrl" eventtype="signature"
level="information" vd="vdom1" appid=38783 user="bob" group="RADIUS_User_Group"
authserver="Ubuntu_docker" dstuser="guest" srcip=10.1.100.72 dstip=172.16.200.75
srcport=60098 dstport=80 srcintf="dmz" srcintfrole="undefined" dstintf="wan1"
dstintfrole="undefined" proto=6 service="HTTP" direction="outgoing" policyid=1
sessionid=10871 applist="g-default" action="pass" appcat="General.Interest" app="Wget"
hostname="172.16.200.75" incidentserialno=17825796 url="/eicar.com"
msg="General.Interest: Wget," apprisk="low"
```

3. Verify the web filter log:

```
date=2021-09-14 time=17:14:46 eventtime=1631664886585770420 tz="-0700"
logid="0315012546" type="utm" subtype="webfilter" eventtype="urlfilter"
level="information" vd="vdom1" urlfilteridx=1 urlfilterlist="Auto-webfilter-urlfilter_
caex0ojl5" policyid=1 sessionid=15251 user="bob" group="RADIUS_User_Group"
authserver="Ubuntu_docker" dstuser="guest" srcip=10.1.100.72 srcport=60106 srcintf="dmz"
srcintfrole="undefined" srcuuid="877d43a4-c2f9-51eb-f78f-e09794924d8a"
dstip=172.16.200.75 dstport=80 dstintf="wan1" dstintfrole="undefined" dstuuid="877d43a4-
c2f9-51eb-f78f-e09794924d8a" proto=6 service="HTTP" hostname="172.16.200.75" profile="1"
action="passthrough" reqtype="direct" url="http://172.16.200.75/eicar.com" sentbyte=149
rcvdbyte=0 direction="outgoing" msg="URL was allowed because it is in the URL filter
list"
```

Sample logs by log type

This topic provides a sample raw log for each subtype and the configuration requirements.

Traffic Logs > Forward Traffic

Log configuration requirements

```
config firewall policy
    edit 1
        set srcintf "port12"
        set dstintf "port11"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set logtraffic all
        set application-list "g-default"
        set ssl-ssh-profile "certificate-inspection"
        set nat enable
    next
end
```

Sample log

```
date=2019-05-10 time=11:37:47 logid="0000000013" type="traffic" subtype="forward"
level="notice" vd="vdom1" eventtime=1557513467369913239 srcip=10.1.100.11 srcport=58012
srcintf="port12" srcintfrole="undefined" dstip=23.59.154.35 dstport=80 dstintf="port11"
dstintfrole="undefined" srcuuid="ae28f494-5735-51e9-f247-d1d2ce663f4b" dstuuid="ae28f494-
5735-51e9-f247-d1d2ce663f4b" poluuid="ccb269e0-5735-51e9-a218-a397dd08b7eb" sessionid=105048
proto=6 action="close" policyid=1 policytype="policy" service="HTTP" dstcountry="Canada"
srccountry="Reserved" trandisp="snat" transip=172.16.200.2 transport=58012 appid=34050
app="HTTP.BROWSER_Firefox" appcat="Web.Client" apprisk="elevated" applist="g-default"
duration=116 sentbyte=1188 rcvdbyte=1224 sentpkt=17 rcvdpkt=16 utmaction="allow" countapp=1
osname="Ubuntu" mastersrcmac="a2:e9:00:ec:40:01" srcmac="a2:e9:00:ec:40:01" srcserver=0
utmref=65500-742
```

Traffic Logs > Local Traffic

Log configuration requirements

```
config log setting
    set local-in-allow enable
    set local-in-deny-unicast enable
    set local-in-deny-broadcast enable
    set local-out enable
end
```

Sample log

```
date=2019-05-10 time=11:50:48 logid="0001000014" type="traffic" subtype="local"
level="notice" vd="vdom1" eventtime=1557514248379911176 srcip=172.16.200.254 srcport=62024
srcintf="port11" srcintfrole="undefined" dstip=172.16.200.2 dstport=443 dstintf="vdom1"
dstintfrole="undefined" sessionid=107478 proto=6 action="server-rst" policyid=0
policytype="local-in-policy" service="HTTPS" dstcountry="Reserved" srccountry="Reserved"
trandisp="noop" app="Web Management(HTTPS)" duration=5 sentbyte=1247 rcvdbyte=1719 sentpkt=5
rcvdpkt=6 appcat="unscanned"
```

Traffic Logs > Multicast Traffic

Log configuration requirements

```
config firewall multicast-policy
    edit 1
        set dstaddr 230-1-0-0
        set dstintf port3
        set srcaddr 172-16-200-0
        set srcintf port25
        set action accept
        set log enable
    next
end
```

```
config system setting
    set multicast-forward enable
end
```

Sample log

```
date=2019-03-31 time=06:42:54 logid="0002000012" type="traffic" subtype="multicast"
level="notice" vd="vdom1" eventtime=1554039772 srcip=172.16.200.55 srcport=60660
srcintf="port25" srcintfrole="undefined" dstip=230.1.1.2 dstport=7878 dstintf="port3"
dstintfrole="undefined" sessionid=1162 proto=17 action="accept" policyid=1
policytype="multicast-policy" service="udp/7878" dstcountry="Reserved" srccountry="Reserved"
trandisp="noop" duration=22 sentbyte=5940 rcvdbyte=0 sentpkt=11 rcvdpkt=0 appcat="unscanned"
```

Traffic Logs > Sniffer Traffic

Log configuration requirements

```
config firewall sniffer
    edit 3
        set logtraffic all
        set interface "port1"
        set ips-sensor-status enable
        set ips-sensor "sniffer-profile"
    next
end
```

Sample log

```
date=2019-05-10 time=14:18:54 logid="0004000017" type="traffic" subtype="sniffer"
level="notice" vd="root" eventtime=1557523134021045897 srcip=208.91.114.4 srcport=50463
srcintf="port1" srcintfrole="undefined" dstip=104.80.88.154 dstport=443 dstintf="port1"
dstintfrole="undefined" sessionid=2193276 proto=6 action="accept" policyid=3
policytype="sniffer" service="HTTPS" dstcountry="United States" srccountry="Canada"
trandisp="snat" transip=0.0.0.0 transport=0 duration=10 sentbyte=0 rcvdbyte=0 sentpkt=0
rcvdpkt=0 appcat="unscanned" utmaction="allow" countips=1 crscore=5 craction=32768
sentdelta=0 rcvddelta=0 utmref=65162-7772
```

```
config system global
    set log-uuid-address enable
end
```

```
config firewall sniffer
    edit 1
        set logtraffic all
        set ipv6 enable
        set interface "port3"
        set ip-threatfeed-status enable
        set ip-threatfeed "g-source"
    next
end
```

Sample log

```
1: date=2021-01-26 time=15:51:37 eventtime=1611705097880421908 tz="-0800" logid="0004000017"
type="traffic" subtype="sniffer" level="notice" vd="vd1" srcip=10.1.100.12 srcport=34604
srcintf="port3" srcintfrole="undefined" dstip=172.16.200.55 dstport=80 dstintf="port3"
dstintfrole="undefined" srcthreatfeed="g-source" srccountry="Reserved" dstcountry="Reserved"
sessionid=30384 proto=6 action="accept" policyid=1 policytype="sniffer" service="HTTP"
```

```
trandisp="snat" transip=0.0.0.0 transport=0 duration=0 sentbyte=0 rcvdbyte=0 sentpkt=0
rcvdpkt=0 appcat="unscanned"
```

Event Logs > SD-WAN Events

Log configuration requirements

```
config log eventfilter
    set event enable
    set sdwan enable
end
```

Sample log

```
date=2020-03-29 time=16:41:30 logid="0113022923" type="event" subtype="sdwan" level="notice"
vd="root" eventtime=1585525290513555981 tz="-0700" logdesc="Virtual WAN Link status"
eventtype="Health Check" healthcheck="ping1" slatargetid=1 oldvalue="1" newvalue="2"
msg="Number of pass member changed."
```

```
date=2020-03-29 time=16:51:27 logid="0113022925" type="event" subtype="sdwan" level="notice"
vd="root" eventtime=1585525888177637570 tz="-0700" logdesc="Virtual WAN Link SLA
information" eventtype="SLA" healthcheck="ping1" slatargetid=1 interface="R150" status="up"
latency="0.013" jitter="0.001" packetloss="100.000%" inbandwidth="0kbps"
outbandwidth="0kbps" bibandwidth="0kbps" slamap="0x0" metric="packetloss" msg="Health Check
SLA status. SLA failed due to being over the performance metric threshold."
```

Event Logs > System Events

Log configuration requirements

```
config log eventfilter
    set event enable
    set system enable
end
```

Sample log

```
date=2019-05-13 time=11:20:54 logid="0100032001" type="event" subtype="system"
level="information" vd="vdom1" eventtime=1557771654587081441 logdesc="Admin login
successful" sn="1557771654" user="admin" ui="ssh(172.16.200.254)" method="ssh"
srcip=172.16.200.254 dstip=172.16.200.2 action="login" status="success" reason="none"
profile="super_admin" msg="Administrator admin logged in successfully from ssh
(172.16.200.254)"
```

Event Logs > Router Events

Log configuration requirements

```
config log eventfilter
    set event enable
    set router enable
end
```

```
config router bgp
    set log-neighbour-changes enable
end
```

```
config router ospf
    set log-neighbour-changes enable
end
```

Sample log

```
date=2019-05-13 time=14:12:26 logid="0103020301" type="event" subtype="router"
level="warning" vd="root" eventtime=1557781946677737955 logdesc="Routing log" msg="OSPF:
RECV[Hello]: From 31.1.1.1 via port9:172.16.200.1: Invalid Area ID 0.0.0.0"
```

Event Logs > VPN Events

Log configuration requirements

```
config log eventfilter
    set event enable
    set vpn enable
end
```

Sample log

```
date=2019-05-13 time=14:21:42 logid="0101037127" type="event" subtype="vpn" level="notice"
vd="root" eventtime=1557782502722231889 logdesc="Progress IPsec phase 1" msg="progress IPsec
phase 1" action="negotiate" remip=50.1.1.101 locip=50.1.1.100 remport=500 locport=500
outintf="port14" cookies="9091f4d4837ea71c/0000000000000000" user="N/A" group="N/A"
xauthuser="N/A" xauthgroup="N/A" assignip=N/A vpntunnel="test" status="success" init="local"
mode="main" dir="outbound" stage=1 role="initiator" result="OK"
```

Event Logs > User Events

Log configuration requirements

```
config log eventfilter
    set event enable
    set user enable
end
```

Sample log

```
date=2019-05-13 time=15:55:56 logid="0102043008" type="event" subtype="user" level="notice"
vd="root" eventtime=1557788156913809277 logdesc="Authentication success" srcip=10.1.100.11
dstip=172.16.200.55 policyid=1 interface="port10" user="bob" group="local-group1"
authproto="TELNET(10.1.100.11)" action="authentication" status="success" reason="N/A"
msg="User bob succeeded in authentication"
```

Event Logs > Endpoint Events

Log configuration requirements

```
config log eventfilter
    set event enable
    set endpoint enable
end
```

Sample log

```
date=2019-05-14 time=08:32:13 logid="0107045057" type="event" subtype="endpoint"
level="information" vd="root" eventtime=1557847933900764210 logdesc="FortiClient connection
added" action="add" status="success" license_limit="unlimited" used_for_type=4 connection_
type="sslvpn" count=1 user="skubas" ip=172.18.64.250 name="VAN-200957-PC"
fctuid="52C66FE08F724FE0B116DAD5062C96CD" msg="Add a FortiClient Connection."
```

```
date=2019-05-14 time=08:19:38 logid="0107045058" type="event" subtype="endpoint"
level="information" vd="root" eventtime=1557847179037488154 logdesc="FortiClient connection
closed" action="close" status="success" license_limit="unlimited" used_for_type=5
connection_type="sslvpn" count=1 user="skubas" ip=172.18.64.250 name="VAN-200957-PC"
fctuid="52C66FE08F724FE0B116DAD5062C96CD" msg="Close a FortiClient Connection."
```

Event Logs > HA Events

Log configuration requirements

```
config log eventfilter
    set event enable
    set ha enable
end
```

Sample log

```
date=2019-05-10 time=09:53:18 logid="0108037894" type="event" subtype="ha" level="critical"
vd="root" eventtime=1557507199208575235 logdesc="Virtual cluster member joined" msg="Virtual
cluster detected member join" vcluster=1 ha_group=0 sn="FG2K5E3916900286"
```

Event Logs > Security Rating Events

Log configuration requirements

```
config log eventfilter
    set event enable
    set security-rating enable
end
```

Sample log

```
date=2019-05-13 time=14:40:59 logid="0110052000" type="event" subtype="security-rating"
level="notice" vd="root" eventtime=1557783659536252389 logdesc="Security Rating summary"
auditid=1557783648 audittime=1557783659 auditscore="5.0" criticalcount=1 highcount=6
mediumcount=8 lowcount=0 passedcount=38
```

Event Logs > WAN Opt & Cache Events

Log configuration requirements

```
config log eventfilter
    set event enable
    set wan-opt enable
end
```

Sample log

```
date=2019-05-14 time=09:37:46 logid="0105048039" type="event" subtype="wad" level="error"
vd="root" eventtime=1557851867382676560 logdesc="SSL fatal alert sent" session_id=0
policyid=0 srcip=0.0.0.0 srcport=0 dstip=208.91.113.83 dstport=636 action="send" alert="2"
desc="certificate unknown" msg="SSL Alert sent"
```

```
date=2019-05-10 time=15:48:31 logid="0105048038" type="event" subtype="wad" level="error"
vd="root" eventtime=1557528511221374615 logdesc="SSL Fatal Alert received" session_
id=5f88ddd1 policyid=0 srcip=172.18.70.15 srcport=59880 dstip=91.189.89.223 dstport=443
action="receive" alert="2" desc="unknown ca" msg="SSL Alert received"
```

Event Logs > Wireless

Log configuration requirements

```
config log eventfilter
    set event enable
    set wireless-activity enable
end
```

```
config wireless-controller log
    set status enable
end
```

Sample log

```
date=2019-05-13 time=11:30:08 logid="0104043568" type="event" subtype="wireless"
level="warning" vd="vdom1" eventtime=1557772208134721423 logdesc="Fake AP on air"
ssid="fortinet" bssid="90:6c:ac:89:e1:fa" aptype=0 rate=130 radioband="802.11n" channel=6
action="fake-ap-on-air" manuf="Fortinet, Inc." security="WPA2 Personal" encryption="AES"
signal=-93 noise=-95 live=353938 age=505 onwire="no" detectionmethod="N/A" stamac="N/A"
apscan="N/A" sndetected="N/A" radioiddetected=0 stacount=0 snclosest="FP320C3X17001909"
radioidclosest=0 apstatus=0 msg="Fake AP On-air fortinet 90:6c:ac:89:e1:fa chan 6 live
353938 age 505"
```

Event Logs > SDN Connector

Log configuration requirements

```
config log eventfilter
    set event enable
    set connector enable
end
```

Sample log

```
date=2019-05-13 time=16:09:43 logid="0112053200" type="event" subtype="connector"
level="information" vd="root" eventtime=1557788982 logdesc="IP address added" cfgobj="aws1"
action="object-add" addr="54.210.36.196" cldobjid="i-0fe5a1ef16bb94796" netid="vpc-97e81cee"
msg="connector object discovered in addr-obj aws1, 54.210.36.196"
```

```
date=2019-05-13 time=16:09:43 logid="0112053201" type="event" subtype="connector"
level="information" vd="root" eventtime=1557788982 logdesc="IP address removed"
cfgobj="aws1" action="object-remove" addr="172.31.31.101" cldobjid="i-0fe5a1ef16bb94796"
netid="vpc-97e81cee" msg="connector object removed in addr-obj aws1, 172.31.31.101"
```

Event Logs > FortiExtender Events

Log configuration requirements

```
config log eventfilter
    set event enable
    set fortiextender enable
end
```

Sample log

```
date=2019-02-20 time=09:57:22 logid="0111046400" type="event" subtype="fortiextender"
level="notice" vd="root" eventtime=1550685442 logdesc="FortiExtender system activity"
action="FortiExtender Authorized"    msg="ext SN:FX04DN4N16002352 authorized"
```

```
date=2019-02-20 time=09:51:42 logid="0111046401" type="event" subtype="fortiextender"
level="notice" vd="root" eventtime=1550685102 logdesc="FortiExtender controller activity"
sn="FX04DN4N16002352" ip=11.11.11.2 action="ext session-deauthed" msg="ext
SN:FX04DN4N16002352 deauthorized"
```

```
date=2019-02-20 time=10:02:26 logid="0111046409" type="event" subtype="fortiextender"
level="information" vd="root" eventtime=1550685746 logdesc="Remote FortiExtender info
activity" sn="FX04DN4N16002352" ip=11.11.11.2 action="Cellular Connected"
imei="359376060442770" imsi="302720502331361" iccid="89302720403038146410"
phonenumber="+16045067526" carrier="Rogers" plan="Rogers-plan" apn="N/A" service="LTE"
msg="FX04DN4N16002352 STATE: sim with imsi:302720502331361 in slot:2 on carrier:Rogers
connected"
```

```
date=2019-02-20 time=10:33:57 logid="0111046407" type="event" subtype="fortiextender"
level="warning" vd="root" eventtime=1550687636 logdesc="Remote FortiExtender warning
activity" sn="FX04DN4N16002352" ip=11.11.11.2 action="Cellular Disconnected"
imei="359376060442770" imsi="N/A" iccid="N/A" phonenumber="N/A" carrier="N/A" plan="N/A"
apn="N/A" service="LTE" msg="FX04DN4N16002352 STATE: sim with imsi: in slot:2 on carrier:N/A
disconnected"
```

```
date=2019-02-20 time=10:02:24 logid="0111046409" type="event" subtype="fortiextender"
level="information" vd="root" eventtime=1550685744 logdesc="Remote FortiExtender info
activity" sn="FX04DN4N16002352" ip=11.11.11.2 action="Cellular Connecting"
imei="359376060442770" imsi="302720502331361" iccid="89302720403038146410"
phonenumber="+16045067526" carrier="Rogers" plan="Rogers-plan" apn="N/A" service="N/A"
msg="FX04DN4N16002352 STATE: sim with imsi:302720502331361 in slot:2 on carrier:Rogers
connecting
```

```
date=2019-02-20 time=10:47:19 logid="0111046407" type="event" subtype="fortiextender"
level="warning" vd="root" eventtime=1550688438 logdesc="Remote FortiExtender warning
```

```
activity" sn="FX04DN4N16002352" ip=11.11.11.2 action="SIM Change" imei="N/A" slot=2
msg="FX04DN4N16002352 SIM: SIM2 is inserted"
```

```
date=2019-02-20 time=10:57:50 logid="0111046407" type="event" subtype="fortiextender"
level="warning" vd="root" eventtime=1550689069 logdesc="Remote FortiExtender warning
activity" sn="FX04DN4N16002352" ip=11.11.11.2 action="SIM Change" imei="359376060442770"
slot=1 msg="FX04DN4N16002352 SIM: SIM2 is plucked out"
```

```
date=2019-02-20 time=12:02:24 logid="0111046407" type="event" subtype="fortiextender"
level="warning" vd="root" eventtime=1550692942 logdesc="Remote FortiExtender warning
activity" sn="FX04DN4N16002352" ip=11.11.11.2 action="SIM Switch" imei="359376060442770"
reason="sim-switch can't take effect due to unavailability of 2 sim cards"
msg="FX04DN4N16002352 SIM: sim-switch can't take effect due to unavailability of 2 sim
cards"
```

```
date=2019-02-19 time=18:08:46 logid="0111046409" type="event" subtype="fortiextender"
level="information" vd="root" eventtime=1550628524 logdesc="Remote FortiExtender info
activity" sn="FX04DN4N16002352" ip=11.11.11.2 action="Cellular Signal Statistics"
imei="359376060442770" imsi="302720502331361" iccid="89302720403038146410"
phonenumber="+16045067526" carrier="Rogers" plan="Rogers-plan" service="LTE" sinr="7.0 dB"
rsrp="-89 dBm" rsrq="-16 dB" signalstrength="92 dBm" rssi="-54" temperature="40 C" apn="N/A"
msg="FX04DN4N16002352 INFO:  LTE RSSI=-54dBm,RSRP=-89dBm,RSRQ=-
16dB,SINR=7.0dB,BAND=B2,CELLID=061C700F,BW=15MHz,RXCH=1025,TXCH=19025,TAC=8AAC,TEMPERATURE=4
0 C"
```

```
date=2019-02-19 time=18:09:46 logid="0111046409" type="event" subtype="fortiextender"
level="information" vd="root" eventtime=1550628585 logdesc="Remote FortiExtender info
activity" sn="FX04DN4N16002352" ip=11.11.11.2 action="Cellular Data Statistics"
imei="359376060442770" imsi="302720502331361" iccid="89302720403038146410"
phonenumber="+16045067526" carrier="Rogers" plan="Rogers-plan" service="LTE" rcvdbyte=7760
sentbyte=3315 msg="FX04DN4N16002352 INFO: SIM2 LTE, rx=7760, tx=3315, rx_diff=2538, tx_
diff=567"
```

Event Logs > FortiSwitch Events

Log configuration requirements

```
config log eventfilter
    set event enable
    set switch-controller enable
end
```

Sample log

```
date=2020-09-28 time=15:37:02 eventtime=1601332622257714795 tz="-0700" logid="0114032695"
type="event" subtype="switch-controller" level="notice" vd="vdom1" logdesc="FortiSwitch
link" user="Fortilink" sn="S248EPTF18001384" name="S248EPTF18001384" msg="port51 Module re-
initialized to recover from ERROR state."
```

```
date=2020-09-28 time=15:37:02 eventtime=1601332622255619520 tz="-0700" logid="0114032697"
type="event" subtype="switch-controller" level="warning" vd="vdom1" logdesc="FortiSwitch
switch" user="Fortilink" sn="S248EPTF18001384" name="S248EPTF18001384" msg="FortiLink:
internal echo reply timed out"
```

```
date=2020-09-28 time=15:37:01 eventtime=1601332621664809633 tz="-0700" logid="0114032605"
type="event" subtype="switch-controller" level="information" vd="vdom1" logdesc="Switch-
```

```
Controller Tunnel Up" user="Switch-Controller" ui="cu_acd" sn="S248EPTF18001384"
name="S248EPTF18001384" msg="CAPWAP Tunnel Up (169.254.1.3)"
```

```
date=2020-09-28 time=15:36:59 eventtime=1601332619501461995 tz="-0700" logid="0114022904"
type="event" subtype="switch-controller" level="notice" vd="vdom1" logdesc="CAPUTP session
status notification" user="Switch-Controller" ui="cu_acd" sn="S248EPTF18001384"
name="S248EPTF18001384" msg="S248EPTF18001384 Connected via session join" action="session-
join" srcip=169.254.1.3
```

```
date=2020-09-28 time=15:36:26 eventtime=1601332560434649361 tz="-0700" logid="0114032601"
type="event" subtype="switch-controller" level="information" vd="vdom1" logdesc="Switch-
Controller discovered" user="daemon_admin" ui="cmdbsvr" sn="S524DN4K16000116"
name="S524DN4K16000116" msg="S524DN4K16000116 Discovered"
```

```
date=2020-09-28 time=15:36:26 eventtime=1601332560405228924 tz="-0700" logid="0114032601"
type="event" subtype="switch-controller" level="information" vd="vdom1" logdesc="Switch-
Controller discovered" user="daemon_admin" ui="cmdbsvr" sn="S248EPTF18001827"
name="S248EPTF18001827" msg="S248EPTF18001827 Discovered"
```

```
date=2020-09-28 time=15:36:26 eventtime=1601332560336851635 tz="-0700" logid="0114032601"
type="event" subtype="switch-controller" level="information" vd="vdom1" logdesc="Switch-
Controller discovered" user="daemon_admin" ui="cmdbsvr" sn="S248EPTF18001384"
name="S248EPTF18001384" msg="S248EPTF18001384 Discovered"
```

Event Logs > REST API Events

Log configuration requirements

```
config log setting
    set rest-api-set enable
    set rest-api-get enable
end
```

Sample log

```
date=2022-02-02 time=15:52:09 eventtime=1643845930263415066 tz="-0800" logid="0116047301"
type="event" subtype="rest-api" level="information" vd="root" logdesc="REST API request
success" user="admin" ui="GUI(192.168.1.69)" method="GET" path="system.usb-log" status="200"
url="/api/v2/monitor/system/usb-log?vdom=root"
```

```
date=2022-02-02 time=15:52:06 eventtime=1643845926774931021 tz="-0800" logid="0116047301"
type="event" subtype="rest-api" level="information" vd="root" logdesc="REST API request
success" user="admin" ui="GUI(192.168.1.69)" method="GET" path="license.status" status="200"
url="/api/v2/monitor/license/status?vdom=root"
```

```
date=2022-02-02 time=15:52:06 eventtime=1643845926764579729 tz="-0800" logid="0116047301"
type="event" subtype="rest-api" level="information" vd="root" logdesc="REST API request
success" user="admin" ui="GUI(192.168.1.69)" method="GET" path="log.fortianalyzer.setting"
status="200" url="/api/v2/cmdb/log.fortianalyzer/setting?vdom=root"
```

```
date=2022-02-02 time=15:52:06 eventtime=1643845926762372766 tz="-0800" logid="0116047301"
type="event" subtype="rest-api" level="information" vd="root" logdesc="REST API request
success" user="admin" ui="GUI(192.168.1.69)" method="GET" path="system.sandbox"
action="connection" status="200" url="/api/v2/monitor/system/sandbox/connection?vdom=root"
```

```
date=2022-02-02 time=15:52:06 eventtime=1643845926755869998 tz="-0800" logid="0116047301"
type="event" subtype="rest-api" level="information" vd="root" logdesc="REST API request
```

```
success" user="admin" ui="GUI(192.168.1.69)" method="GET" path="system.firmware"
status="200" url="/api/v2/monitor/system/firmware?vdom=root"
```

Event Logs > IOC Detection

Log configuration requirements

```
config log setting
    set local-out enable
    set local-out-ioc-detection enable
end
```

Sample log

```
date=2021-12-20 time=16:43:54 eventtime=1640047434839814226 tz="-0800" logid="0100020214"
type="event" subtype="system" level="warning" vd="root" logdesc="Locally generated traffic
goes to IoC location" srcip=172.16.200.2 srcport=18047 dstip=223.205.1.54 dstport=514
session_id=23563 proto=6
```

```
# Corresponding Traffic Log #
date=2021-12-20 time=16:45:18 eventtime=1640047518959313316 tz="-0800" logid="0001000014"
type="traffic" subtype="local" level="notice" vd="root" srcip=172.16.200.2 srcport=18047
srcintf="unknown-0" srcintfrole="undefined" dstip=223.205.1.54 dstport=514 dstintf="port2"
dstintfrole="undefined" srccountry="Reserved" dstcountry="Thailand" sessionid=23632 proto=6
action="timeout" policyid=0 service="tcp/514" trandisp="noop" app="tcp/514" duration=17
sentbyte=240 rcvdbyte=0 sentpkt=4 rcvdpkt=0 appcat="unscanned" dsthwvendor="Fortinet"
masterdstmac="e8:1c:ba:c2:86:63" dstmac="e8:1c:ba:c2:86:63" dstserver=0
```

Security Logs > Antivirus

Log configuration requirements

```
config antivirus profile
    edit "test-av"
        config http
            set av-scan block
        end
        set av-virus-log enable
        set av-block-log enable
    next
end
```

```
config firewall policy
    edit 1
        set srcintf "port12"
        set dstintf "port11"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set av-profile "test-av"
        set logtraffic utm
```

```
        set nat enable
    next
end
```

Sample log

```
date=2019-05-13 time=11:45:03 logid="0211008192" type="utm" subtype="virus"
eventtype="infected" level="warning" vd="vdom1" eventtime=1557773103767393505 msg="File is
infected." action="blocked" service="HTTP" sessionid=359260 srcip=10.1.100.11
dstip=172.16.200.55 srcport=60446 dstport=80 srcintf="port12" srcintfrole="undefined"
dstintf="port11" dstintfrole="undefined" policyid=4 proto=6 direction="incoming"
filename="eicar.com" quarskip="File-was-not-quarantined." virus="EICAR_TEST_FILE"
dtype="Virus" ref="http://www.fortinet.com/ve?vn=EICAR_TEST_FILE" virusid=2172
url="http://172.16.200.55/virus/eicar.com" profile="g-default" agent="curl/7.47.0"
analyticscksum="275a021bbfb6489e54d471899f7db9d1663fc695ec2fe2a2c4538aabf651fd0f"
analyticssubmit="false" crscore=50 craction=2 crlevel="critical"
```

```
# Corresponding Traffic Log #
date=2019-05-13 time=11:45:04 logid="0000000013" type="traffic" subtype="forward"
level="notice" vd="vdom1" eventtime=1557773104815101919 srcip=10.1.100.11 srcport=60446
srcintf="port12" srcintfrole="undefined" dstip=172.16.200.55 dstport=80 dstintf="port11"
dstintfrole="undefined" srcuuid="48420c8a-5c88-51e9-0424-a37f9e74621e" dstuuid="187d6f46-
5c86-51e9-70a0-fadcfc349c3e" poluuid="3888b41a-5c88-51e9-cb32-1c32c66b4edf" sessionid=359260
proto=6 action="close" policyid=4 policytype="policy" service="HTTP" dstcountry="Reserved"
srccountry="Reserved" trandisp="snat" transip=172.16.200.2 transport=60446 appid=15893
app="HTTP.BROWSER" appcat="Web.Client" apprisk="medium" applist="g-default" duration=1
sentbyte=412 rcvdbyte=2286 sentpkt=6 rcvdpkt=6 wanin=313 wanout=92 lanin=92 lanout=92
utmaction="block" countav=1 countapp=1 crscore=50 craction=2 osname="Ubuntu"
mastersrcmac="a2:e9:00:ec:40:01" srcmac="a2:e9:00:ec:40:01" srcserver=0 utmref=65497-770
```

Security Logs > Web Filter

Log configuration requirements

```
config webfilter profile
    edit "test-webfilter"
        set web-content-log enable
        set web-filter-activex-log enable
        set web-filter-command-block-log enable
        set web-filter-cookie-log enable
        set web-filter-applet-log enable
        set web-filter-jscript-log enable
        set web-filter-js-log enable
        set web-filter-vbs-log enable
        set web-filter-unknown-log enable
        set web-filter-referer-log enable
        set web-filter-cookie-removal-log enable
        set web-url-log enable
        set web-invalid-domain-log enable
        set web-ftgd-err-log enable
        set web-ftgd-quota-usage enable
    next
end
```

```
config firewall policy
    edit 1
```

```
        set name "v4-out"
        set srcintf "port12"
        set dstintf "port11"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set logtraffic utm
        set utm-status enable
        set webfilter-profile "test-webfilter"
        set nat enable
    next
end
```

Sample log

```
date=2019-05-13 time=16:29:45 logid="0316013056" type="utm" subtype="webfilter"
eventtype="ftgd_blk" level="warning" vd="vdom1" eventtime=1557790184975119738 policyid=1
sessionid=381780 srcip=10.1.100.11 srcport=44258 srcintf="port12" srcintfrole="undefined"
dstip=185.244.31.158 dstport=80 dstintf="port11" dstintfrole="undefined" proto=6
service="HTTP" hostname="morrishittu.ddns.net" profile="test-webfilter" action="blocked"
reqtype="direct" url="/" sentbyte=84 rcvdbyte=0 direction="outgoing" msg="URL belongs to a
denied category in policy" method="domain" cat=26 catdesc="Malicious Websites" crscore=30
craction=4194304 crlevel="high"
```

```
# Corresponding traffic log #
date=2019-05-13 time=16:29:50 logid="0000000013" type="traffic" subtype="forward"
level="notice" vd="vdom1" eventtime=1557790190452146185 srcip=10.1.100.11 srcport=44258
srcintf="port12" srcintfrole="undefined" dstip=185.244.31.158 dstport=80 dstintf="port11"
dstintfrole="undefined" srcuuid="ae28f494-5735-51e9-f247-d1d2ce663f4b" dstuuid="ae28f494-
5735-51e9-f247-d1d2ce663f4b" poluuid="ccb269e0-5735-51e9-a218-a397dd08b7eb" sessionid=381780
proto=6 action="close" policyid=1 policytype="policy" service="HTTP" dstcountry="Germany"
srccountry="Reserved" trandisp="snat" transip=172.16.200.2 transport=44258 duration=5
sentbyte=736 rcvdbyte=3138 sentpkt=14 rcvdpkt=5 appcat="unscanned" utmaction="block"
countweb=1 crscore=30 craction=4194304 osname="Ubuntu" mastersrcmac="a2:e9:00:ec:40:01"
srcmac="a2:e9:00:ec:40:01" srcserver=0 utmref=65497-796
```

Security Logs > DNS Query

Log configuration requirements

```
config dnsfilter profile
    edit "dnsfilter_fgd"
        config ftgd-dns
            set options error-allow
        end
        set log-all-domain enable
        set block-botnet enable
    next
end
```

```
config firewall policy
    edit 1
        set srcintf "port12"
        set dstintf "port11"
```

```
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set dnsfilter-profile "dnsfilter_fgd"
        set logtraffic utm
        set nat enable
    next
end
```

Sample log

```
date=2019-05-15 time=15:05:49 logid="1501054802" type="utm" subtype="dns" eventtype="dns-
response" level="notice" vd="vdom1" eventtime=1557957949740931155 policyid=1 sessionid=6887
srcip=10.1.100.22 srcport=50002 srcintf="port12" srcintfrole="undefined"
dstip=172.16.100.100 dstport=53 dstintf="port11" dstintfrole="undefined" proto=17
profile="dnsfilter_fgd" srcmac="a2:e9:00:ec:40:41" xid=57945 qname="changelogs.ubuntu.com"
qtype="AAAA" qtypeval=28 qclass="IN" ipaddr="2001:67c:1560:8008::11" msg="Domain is
monitored" action="pass" cat=52 catdesc="Information Technology"
```

```
date=2019-05-15 time=15:05:49 logid="1500054000" type="utm" subtype="dns" eventtype="dns-
query" level="information" vd="vdom1" eventtime=1557957949653103543 policyid=1
sessionid=6887 srcip=10.1.100.22 srcport=50002 srcintf="port12" srcintfrole="undefined"
dstip=172.16.100.100 dstport=53 dstintf="port11" dstintfrole="undefined" proto=17
profile="dnsfilter_fgd" srcmac="a2:e9:00:ec:40:41" xid=57945 qname="changelogs.ubuntu.com"
qtype="AAAA" qtypeval=28 qclass="IN"
```

```
# Corresponding traffic log #
date=2019-05-15 time=15:08:49 logid="0000000013" type="traffic" subtype="forward"
level="notice" vd="vdom1" eventtime=1557958129950003945 srcip=10.1.100.22 srcport=50002
srcintf="port12" srcintfrole="undefined" dstip=172.16.100.100 dstport=53 dstintf="port11"
dstintfrole="undefined" srcuuid="ae28f494-5735-51e9-f247-d1d2ce663f4b" dstuuid="ae28f494-
5735-51e9-f247-d1d2ce663f4b" poluuid="ccb269e0-5735-51e9-a218-a397dd08b7eb" sessionid=6887
proto=17 action="accept" policyid=1 policytype="policy" service="DNS" dstcountry="Reserved"
srccountry="Reserved" trandisp="snat" transip=172.16.200.2 transport=50002 duration=180
sentbyte=67 rcvdbyte=207 sentpkt=1 rcvdpkt=1 appcat="unscanned" utmaction="allow" countdns=1
osname="Linux" mastersrcmac="a2:e9:00:ec:40:41" srcmac="a2:e9:00:ec:40:41" srcserver=0
utmref=65495-306
```

Security Logs > Application Control

Log configuration requirements

```
# log enabled by default in application profile entry

config application list
    edit "block-social.media"
        set other-application-log enable
        config entries
            edit 1
                set category 2 5 6 23
                set log enable
            next
        end
```

```
    next
end
```

```
config firewall policy
    edit 1
        set name "to_Internet"
        set srcintf "port10"
        set dstintf "port9"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set logtraffic utm
        set application-list "block-social.media"
        set ssl-ssh-profile "deep-inspection"
        set nat enable
    next
end
```

Sample log

```
date=2019-05-15 time=18:03:36 logid="1059028704" type="utm" subtype="app-ctrl"
eventtype="app-ctrl-all" level="information" vd="root" eventtime=1557968615 appid=40568
srcip=10.1.100.22 dstip=195.8.215.136 srcport=50798 dstport=443 srcintf="port10"
srcintfrole="lan" dstintf="port9" dstintfrole="wan" proto=6 service="HTTPS"
direction="outgoing" policyid=1 sessionid=4414 applist="block-social.media"
appcat="Web.Client" app="HTTPS.BROWSER" action="pass" hostname="www.dailymotion.com"
incidentserialno=1962906680 url="/" msg="Web.Client: HTTPS.BROWSER," apprisk="medium"
scertcname="*.dailymotion.com" scertissuer="DigiCert SHA2 High Assurance Server CA"
```

```
date=2019-05-15 time=18:03:35 logid="1059028705" type="utm" subtype="app-ctrl"
eventtype="app-ctrl-all" level="warning" vd="root" eventtime=1557968615 appid=16072
srcip=10.1.100.22 dstip=195.8.215.136 srcport=50798 dstport=443 srcintf="port10"
srcintfrole="lan" dstintf="port9" dstintfrole="wan" proto=6 service="HTTPS"
direction="incoming" policyid=1 sessionid=4414 applist="block-social.media"
appcat="Video/Audio" app="Dailymotion" action="block" hostname="www.dailymotion.com"
incidentserialno=1962906682 url="/" msg="Video/Audio: Dailymotion," apprisk="elevated"
```

```
date=2019-05-15 time=18:03:35 logid="1059028705" type="utm" subtype="app-ctrl"
eventtype="app-ctrl-all" level="warning" vd="root" eventtime=1557968615 appid=16072
srcip=10.1.100.22 dstip=195.8.215.136 srcport=50798 dstport=443 srcintf="port10"
srcintfrole="lan" dstintf="port9" dstintfrole="wan" proto=6 service="HTTPS"
direction="incoming" policyid=1 sessionid=4414 applist="block-social.media"
appcat="Video/Audio" app="Dailymotion" action="block" hostname="www.dailymotion.com"
incidentserialno=1962906681 url="/" msg="Video/Audio: Dailymotion," apprisk="elevated"
```

```
# Corresponding Traffic Log #  date=2019-05-15 time=18:03:41 logid="0000000013"
type="traffic" subtype="forward" level="notice" vd="root" eventtime=1557968619
srcip=10.1.100.22 srcport=50798 srcintf="port10" srcintfrole="lan" dstip=195.8.215.136
dstport=443 dstintf="port9" dstintfrole="wan" poluuid="d8ce7a90-7763-51e9-e2be-741294c96f31"
sessionid=4414 proto=6 action="client-rst" policyid=1 policytype="policy" service="HTTPS"
dstcountry="France" srccountry="Reserved" trandisp="snat" transip=172.16.200.10
transport=50798 appid=16072 app="Dailymotion" appcat="Video/Audio" apprisk="elevated"
applist="block-social.media" appact="drop-session" duration=5 sentbyte=1150 rcvdbyte=7039
sentpkt=13 utmaction="block" countapp=3 devtype="Unknown" devcategory="None"
mastersrcmac="00:0c:29:51:38:5e" srcmac="00:0c:29:51:38:5e" srcserver=0 utmref=0-330
```

Security Logs > Intrusion Prevention

Log configuration requirements

```
# log enabled by default in ips sensor

config ips sensor
    edit "block-critical-ips"
        config entries
            edit 1
                set severity critical
                set status enable
                set action block
                set log enable
            next
        end
    next
end
```

```
config firewall policy
    edit 1
        set name "to_Internet"
        set srcintf "port10"
        set dstintf "port9"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set logtraffic utm
        set ips-sensor "block-critical-ips"
        set nat enable
    next
end
```

Sample log

```
date=2019-05-15 time=17:56:41 logid="0419016384" type="utm" subtype="ips"
eventtype="signature" level="alert" vd="root" eventtime=1557968201 severity="critical"
srcip=10.1.100.22 srccountry="Reserved" dstip=172.16.200.55 srcintf="port10"
srcintfrole="lan" dstintf="port9" dstintfrole="wan" sessionid=4017 action="dropped" proto=6
service="HTTP" policyid=1 attack="Adobe.Flash.newfunction.Handling.Code.Execution"
srcport=46810 dstport=80 hostname="172.16.200.55" url="/ips/sig1.pdf" direction="incoming"
attackid=23305 profile="block-critical-ips" ref="http://www.fortinet.com/ids/VID23305"
incidentserialno=582633933 msg="applications3:
Adobe.Flash.newfunction.Handling.Code.Execution," crscore=50 craction=4096
crlevel="critical"
```

```
# Corresponding Traffic Log #  date=2019-05-15 time=17:58:10 logid="0000000013"
type="traffic" subtype="forward" level="notice" vd="root" eventtime=1557968289
srcip=10.1.100.22 srcport=46810 srcintf="port10" srcintfrole="lan" dstip=172.16.200.55
dstport=80 dstintf="port9" dstintfrole="wan" poluuid="d8ce7a90-7763-51e9-e2be-741294c96f31"
sessionid=4017 proto=6 action="close" policyid=1 policytype="policy" service="HTTP"
dstcountry="Reserved" srccountry="Reserved" trandisp="snat" transip=172.16.200.10
transport=46810 duration=89 sentbyte=565 rcvdbyte=9112 sentpkt=9 rcvdpkt=8
```

```
appcat="unscanned" utmaction="block" countips=1 crscore=50 craction=4096 devtype="Unknown"
devcategory="None" mastersrcmac="00:0c:29:51:38:5e" srcmac="00:0c:29:51:38:5e" srcserver=0
utmref=0-302
```

Security Logs > Anomaly

Log configuration requirements

```
config firewall DoS-policy
    edit 1
        set interface "port12"
        set srcaddr "all"
        set dstaddr "all"
        set service "ALL"
        config anomaly
            edit "icmp_flood"
                set status enable
                set log enable
                set action block
                set threshold 50
            next
        end
    next
end
```

Sample log

```
date=2019-05-13 time=17:05:59 logid="0720018433" type="utm" subtype="anomaly"
eventtype="anomaly" level="alert" vd="vdom1" eventtime=1557792359461869329
severity="critical" srcip=10.1.100.11 srccountry="Reserved" dstip=172.16.200.55
srcintf="port12" srcintfrole="undefined" sessionid=0 action="clear_session" proto=1
service="PING" count=1 attack="icmp_flood" icmpid="0x1474" icmptype="0x08" icmpcode="0x00"
attackid=16777316 policyid=1 policytype="DoS-policy"
ref="http://www.fortinet.com/ids/VID16777316" msg="anomaly: icmp_flood, 51 > threshold 50"
crscore=50 craction=4096 crlevel="critical"
```

Security Logs > Data Leak Prevention

Log configuration requirements

```
config dlp profile
    edit "dlp-file-type-test"
        set comment ''
        set replacemsg-group ''
        config filter
            edit 1
                set name ''
                set severity medium
                set type file
                set proto http-get http-post ftp
                set filter-by file-type
                set file-type 1
                set archive enable
                set action block
```

```
            next
        end
        set dlp-log enable
    next
end
```

```
config firewall policy
    edit 1
        set name "to_Internet"
        set srcintf "port10"
        set dstintf "port9"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set inspection-mode proxy
        set logtraffic utm
        set dlp-profile "dlp-file-type-test"
        set ssl-ssh-profile "deep-inspection"
        set nat enable
    next
end
```

Sample log

```
date=2022-02-03 time=17:45:30 eventtime=1643938572487964255 tz="-0800" logid="0954024576"
type="utm" subtype="dlp" eventtype="dlp" level="warning" vd="root" filteridx=1
dlpextra="dlp-file-size11" filtertype="file-type" filtercat="file" severity="medium"
policyid=1 policytype="policy" sessionid=156237 epoch=300501327 eventid=0 srcip=10.1.100.22
srcport=50354 srccountry="Reserved" srcintf="port10" srcintfrole="lan" dstip=52.216.177.83
dstport=443 dstcountry="United States" dstintf="port9" dstintfrole="wan" proto=6
service="HTTPS" filetype="pdf" direction="incoming" action="block"
hostname="fortinetweb.s3.amazonaws.com" url="https://172.16.200.88/dlp/files/fortiauto.pdf"
forwardedfor="192.168.0.99" agent="curl/7.56.0" httpmethod="GET"
referralurl="https://example.com/referer.html" filename="fortiauto.pdf" filesize=285442
profile="dlp-file-type-test" rawdata="x-forwarded-for=192.168.0.99|Response-Content-
Type=application/pdf"
```

```
# Corresponding Traffic Log #
date=2022-02-03 time=17:45:34 logid="0000000013" type="traffic" subtype="forward"
level="notice" vd="root" eventtime=1557967534 srcip=10.1.100.22 srcport=50354
srcintf="port10" srcintfrole="lan" dstip=52.216.177.83 dstport=443 dstintf="port9"
dstintfrole="wan" poluuid="d8ce7a90-7763-51e9-e2be-741294c96f31" sessionid=3423 proto=6
action="server-rst" policyid=1 policytype="policy" service="HTTPS" dstcountry="United
States" srccountry="Reserved" trandisp="snat" transip=172.16.200.10 transport=50354
duration=5 sentbyte=2314 rcvdbyte=5266 sentpkt=33 rcvdpkt=12 appcat="unscanned" wanin=43936
wanout=710 lanin=753 lanout=753 utmaction="block" countdlp=1 crscore=5 craction=262144
crlevel="low" devtype="Unknown" devcategory="None" mastersrcmac="00:0c:29:51:38:5e"
srcmac="00:0c:29:51:38:5e" srcserver=0 utmref=0-152
```

Security Logs > SSH and Security Logs > SSL

Log configuration requirements

```
config ssh-filter profile
    edit "ssh-deepscan"
        set block shell
        set log shell
        set default-command-log disable
    next
end
```

```
config firewall policy
    edit 1
        set srcintf "port21"
        set dstintf "port23"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set inspection-mode proxy
        set ssh-filter-profile "ssh-deepscan"
        set profile-protocol-options "protocol"
        set ssl-ssh-profile "ssl"
        set nat enable
    next
end
```

For SSL-Traffic-log, enable `logtraffic all`

```
config firewall policy
    edit 1
        set srcintf "dmz"
        set dstintf "wan1"
        set srcaddr "all"
        set dstaddr "all"
        set action accept
        set schedule "always"
        set service "ALL"
        set utm-status enable
        set inspection-mode proxy
        set logtraffic all
        set ssl-ssh-profile "deep-inspection"
        set nat enable
    next
end
```

For SSL-UTM-log

```
#EVENTTYPE="SSL-ANOMALIES"
```

By default, `ssl-anomaly-log` is enabled.

```
config firewall ssl-ssh-profile
    edit "deep-inspection"
        set comment "Read-only deep inspection profile."
        set server-cert-mode re-sign
        set caname "Fortinet_CA_SSL"
        set untrusted-caname "Fortinet_CA_Untrusted"
        set ssl-anomaly-log enable
        set ssl-exemption-log disable
        set ssl-negotiation-log disable
        set rpc-over-https disable
        set mapi-over-https disable
        set use-ssl-server disable
    next
end
```

```
# EVENTTYPE="SSL-EXEMPT"
```

Enable `ssl-exemption-log` to generate ssl-utm-exempt log.

```
config firewall ssl-ssh-profile
    edit "deep-inspection"
        set comment "Read-only deep inspection profile."
        set server-cert-mode re-sign
        set caname "Fortinet_CA_SSL"
        set untrusted-caname "Fortinet_CA_Untrusted"
        set ssl-anomaly-log enable
        set ssl-exemption-log enable
        set ssl-negotiation-log disable
        set rpc-over-https disable
        set mapi-over-https disable
        set use-ssl-server disable
    next
end
```

```
# EVENTTYPE="SSL-negotiation"
```

Enable `ssl-negotiation-log` to log SSL negotiation. Enable `ssl-server-cert-log` to log server certificate information. Enable `ssl-handshake-log` to log TLS handshakes.

```
config firewall ssl-ssh-profile
    edit "deep-inspection"
        set comment "Read-only deep inspection profile."
        set server-cert-mode re-sign
        set caname "Fortinet_CA_SSL"
        set untrusted-caname "Fortinet_CA_Untrusted"
        set ssl-anomaly-log enable
        set ssl-exemption-log enable
        set ssl-negotiation-log enable
        set rpc-over-https disable
        set mapi-over-https disable
        set use-ssl-server disable
        set ssl-server-cert-log enable
        set ssl-handshake-log enable
    next
end
```

Sample log for SSH

```
date=2019-05-15 time=16:18:17 logid="1601061010" type="utm" subtype="ssh" eventtype="ssh-channel" level="warning" vd="vdom1" eventtime=1557962296 policyid=1 sessionid=344 profile="ssh-deepscan" srcip=10.1.100.11 srcport=43580 dstip=172.16.200.44 dstport=22 srcintf="port21" srcintfrole="undefined" dstintf="port23" dstintfrole="undefined" proto=6 action="blocked" direction="outgoing" login="root" channeltype="shell"
```

```
# Corresponding Traffic Log #
date=2019-05-15 time=16:18:18 logid="0000000013" type="traffic" subtype="forward" level="notice" vd="vdom1" eventtime=1557962298 srcip=10.1.100.11 srcport=43580 srcintf="port21" srcintfrole="undefined" dstip=172.16.200.44 dstport=22 dstintf="port23" dstintfrole="undefined" poluuid="49871fae-7371-51e9-17b4-43c7ff119195" sessionid=344 proto=6 action="close" policyid=1 policytype="policy" service="SSH" dstcountry="Reserved" srccountry="Reserved" trandisp="snat" transip=172.16.200.171 transport=43580 duration=8 sentbyte=3093 rcvdbyte=2973 sentpkt=18 rcvdpkt=16 appcat="unscanned" utmaction="block" countssh=1 utmref=65535-0
```

Sample log for SSL

For SSL-Traffic-log

```
date=2019-05-16 time=10:08:26 logid="0000000013" type="traffic" subtype="forward" level="notice" vd="root" eventtime=1558026506763925658 srcip=10.1.100.66 srcport=38572 srcintf="dmz" srcintfrole="dmz" dstip=104.154.89.105 dstport=443 dstintf="wan1" dstintfrole="wan" poluuid="a17c0a38-75c6-51e9-4c0d-d547347b63e5" sessionid=100 proto=6 action="server-rst" policyid=1 policytype="policy" service="HTTPS" dstcountry="United States" srccountry="Reserved" trandisp="snat" transip=172.16.200.11 transport=38572 duration=5 sentbyte=930 rcvdbyte=6832 sentpkt=11 rcvdpkt=19 appcat="unscanned" wanin=1779 wanout=350 lanin=754 lanout=754 utmaction="block"  countssl=1 crscore=5 craction=262144 crlevel="low" utmref=65467-0
```

For SSL-UTM-log

```
#EVENTTYPE="SSL-ANOMALIES"
```

```
date=2019-03-28 time=10:44:53 logid="1700062002" type="utm" subtype="ssl" eventtype="ssl-anomalies" level="warning" vd="vdom1" eventtime=1553795092 policyid=1 sessionid=10796 service="HTTPS" srcip=10.1.100.66 srcport=43602 dstip=104.154.89.105 dstport=443 srcintf="port2" srcintfrole="undefined" dstintf="port3" dstintfrole="undefined" proto=6 action="blocked" msg="Server certificate blocked" reason="block-cert-invalid"
```

```
date=2019-03-28 time=10:51:17 logid="1700062002" type="utm" subtype="ssl" eventtype="ssl-anomalies" level="warning" vd="vdom1" eventtime=1553795476 policyid=1 sessionid=11110 service="HTTPS" srcip=10.1.100.66 srcport=49076 dstip=172.16.200.99 dstport=443 srcintf="port2" srcintfrole="undefined" dstintf="port3" dstintfrole="undefined" proto=6 action="blocked" msg="Server certificate blocked" reason="block-cert-untrusted"
```

```
date=2019-03-28 time=10:55:43 logid="1700062002" type="utm" subtype="ssl" eventtype="ssl-anomalies" level="warning" vd="vdom1" eventtime=1553795742 policyid=1 sessionid=11334 service="HTTPS" srcip=10.1.100.66 srcport=49082 dstip=172.16.200.99 dstport=443 srcintf="port2" srcintfrole="undefined" dstintf="port3" dstintfrole="undefined" proto=6 action="blocked" msg="Server certificate blocked" reason="block-cert-req"
```

```
date=2019-03-28 time=10:57:42 logid="1700062053" type="utm" subtype="ssl" eventtype="ssl-anomalies" level="warning" vd="vdom1" eventtime=1553795861 policyid=1 sessionid=11424 service="SMTPS" profile="block-unsupported-ssl" srcip=10.1.100.66 srcport=41296
```

```
dstip=172.16.200.99 dstport=8080 srcintf="port2" srcintfrole="undefined" dstintf=unknown-0
dstintfrole="undefined" proto=6 action="blocked" msg="Connection is blocked due to
unsupported SSL traffic" reason="malformed input"

date=2019-03-28 time=11:00:17 logid="1700062002" type="utm" subtype="ssl" eventtype="ssl-
anomalies" level="warning" vd="vdom1" eventtime=1553796016 policyid=1 sessionid=11554
service="HTTPS" srcip=10.1.100.66 srcport=49088 dstip=172.16.200.99 dstport=443
srcintf="port2" srcintfrole="undefined" dstintf="port3" dstintfrole="undefined" proto=6
action="blocked" msg="Server certificate blocked" reason="block-cert-sni-mismatch"

# EVENTTYPE="SSL-EXEMPT"

date=2019-03-28 time=11:09:14 logid="1701062003" type="utm" subtype="ssl" eventtype="ssl-
exempt" level="notice" vd="vdom1" eventtime=1553796553 policyid=1 sessionid=12079
service="HTTPS" srcip=10.1.100.66 srcport=49102 dstip=172.16.200.99 dstport=443
srcintf="port2" srcintfrole="undefined" dstintf="port3" dstintfrole="undefined" proto=6
action="exempt" msg="SSL connection exempted" reason="exempt-addr"

date=2019-03-28 time=11:10:55 logid="1701062003" type="utm" subtype="ssl" eventtype="ssl-
exempt" level="notice" vd="vdom1" eventtime=1553796654 policyid=1 sessionid=12171
service="HTTPS" srcip=10.1.100.66 srcport=47390 dstip=50.18.221.132 dstport=443
srcintf="port2" srcintfrole="undefined" dstintf="port3" dstintfrole="undefined" proto=6
action="exempt" msg="SSL connection exempted" reason="exempt-ftgd-cat"

# EVENTYPE="SSL-NEGOTIATION"

date=2020-02-07 time=11:10:58 logid="1702062101" type="utm" subtype="ssl" eventtype="ssl-
negotiation" level="warning" vd="vdom1" eventtime=1581102658589415731 tz="-0800"
action="blocked" policyid=1 sessionid=141224 service="HTTPS" profile="deep-inspection-clone"
srcip=10.1.100.66 srcport=33666 dstip=172.16.200.99 dstport=8080 srcintf="port2"
srcintfrole="undefined" dstintf="port3" dstintfrole="undefined" proto=6
eventsubtype="unexpected-protocol" msg="SSL connection is blocked."

date=2021-06-17 time=16:55:26 eventtime=1623974126384215772 tz="-0700" logid="1702062103"
type="utm" subtype="ssl" eventtype="ssl-negotiation" level="information" vd="vdom1"
action="info" policyid=1 sessionid=6361 service="HTTPS" profile="deep-inspection-clone"
srcip=10.1.100.11 srcport=48892 dstip=18.140.21.233 dstport=443 srcintf="port2"
srcintfrole="undefined" dstintf="port3" dstintfrole="undefined" srcuuid="8666f70e-cfb9-51eb-
4991-9012417d69da" dstuuid="8666f70e-cfb9-51eb-4991-9012417d69da" proto=6
sni="www.fortinet.com" eventsubtype="server-cert-info" hostname="www.fortinet.com"
notbefore="2021-03-13T00:00:00Z" notafter="2022-04-13T23:59:59Z" issuer="DigiCert TLS RSA
SHA256 2020 CA1" cn="*.fortinet.com" san="*.fortinet.com;www.fortinet.com;fortinet.com"
sn="000aa00a00000a00000a00a00aa000a0" ski="df9152b605cc18b346efb34de6907275dbdb2b3c"
certhash="1d55cd34a1ed5d3f69bd825a45e04fbd2efba937" keyalgo="rsa" keysize=2048

date=2021-06-17 time=16:55:26 eventtime=1623974126411127210 tz="-0700" logid="1702062103"
type="utm" subtype="ssl" eventtype="ssl-negotiation" level="information" vd="vdom1"
action="info" policyid=1 sessionid=6361 service="HTTPS" profile="deep-inspection-clone"
srcip=10.1.100.11 srcport=48892 dstip=18.140.21.233 dstport=443 srcintf="port2"
srcintfrole="undefined" dstintf="port3" dstintfrole="undefined" srcuuid="8666f70e-cfb9-51eb-
4991-9012417d69da" dstuuid="8666f70e-cfb9-51eb-4991-9012417d69da" proto=6 tlsver="tls1.3"
sni="www.fortinet.com" cipher="0x1302" authalgo="rsa" kxproto="ecdhe" kxcurve="secp256r1"
eventsubtype="handshake-done" hostname="www.fortinet.com" handshake="full" mitm="yes"
```

Troubleshooting

The following topics provide information about troubleshooting logging and reporting:

- Log-related diagnose commands on page 2603
- Backing up log files or dumping log messages on page 2608
- SNMP OID for logs that failed to send on page 2610

Log-related diagnose commands

This topic shows commonly used examples of log-related diagnose commands.

Use the following diagnose commands to identify log issues:

- The following commands enable debugging log daemon (`miglogd`) at the proper debug level:

  ```
  diagnose debug application miglogd x
  diagnose debug enable
  ```

- The following commands display different status/statistics of `miglogd` at the proper level:

  ```
  diagnose test application miglogd x
  diagnose debug enable
  ```

To get the list of available levels, press `Enter` after `diagnose test/debug application miglogd`. The following are some examples of commonly use levels.

If the debug log display does not return correct entries when log filter is set:

```
diagnose debug application miglogd 0x1000
```

For example, use the following command to display all login system event logs:

```
execute log filter device disk
execute log filter category event
execute log filter field action login

execute log display

Files to be searched:
file_no=65523, start line=0, end_line=237
file_no=65524, start line=0, end_line=429
file_no=65525, start line=0, end_line=411
file_no=65526, start line=0, end_line=381
file_no=65527, start line=0, end_line=395
file_no=65528, start line=0, end_line=458
file_no=65529, start line=0, end_line=604
file_no=65530, start line=0, end_line=389
file_no=65531, start line=0, end_line=384
session ID=1, total logs=3697
back ground search. process ID=26240, session_id=1
 start line=1  view line=10
( action "login" )
ID=1, total=3697, checked=238, found=5
ID=1, total=3697, checked=668, found=13
ID=1, total=3697, checked=1080, found=23
ID=1, total=3697, checked=1462, found=23
ID=1, total=3697, checked=1858, found=23
ID=1, total=3697, checked=2317, found=54
ID=1, total=3697, checked=2922, found=106
ID=1, total=3697, checked=3312, found=111
ID=1, total=3697, checked=3697, found=114
```

You can check and/or debug the FortiGate to FortiAnalyzer connection status.

To show connect status with detailed information:

```
diagnose test application miglogd 1

faz: global , enabled
        server=172.18.64.234, realtime=3, ssl=1, state=connected, src=, mgmt_name=FGh_Log_
vdom1_172.18.64.234, reliable=0, sni_prefix_type=none, required_entitlement=none
                status: ver=6, used_disk=0, total_disk=0, global=0, vfid=0 conn_verified=Y
                SNs: last sn update:107 seconds ago.
                        Sn list:
                        (FL-8HFT718900132,age=107s)
                queue: qlen=0.
filter: severity=6, sz_exclude_list=0
         voip dns ssh ssl cifs
subcategory:
        traffic: forward local multicast sniffer
        anomaly: anomaly

        server: global, id=0, fd=132, ready=1, ipv6=0, 172.18.64.234/514
        oftp-state=5
```

To collect debug information when FortiAnalyzer is enabled:

```
diagnose debug application miglogd 0x100

FGT-B-LOG (global) # <16208> miglog_start_rmt_conn()-1552: setting epoll_hd:0x7fc364e125e0
to _rmt_connect
<16209> miglog_start_rmt_conn()-1552: setting epoll_hd:0x7f72647715e0 to _rmt_connect
<16206> miglog_start_rmt_conn()-1552: setting epoll_hd:0x141f69e0 to _rmt_connect
<16209> _rmt_connect()-1433: oftp is ready.
<16209> _rmt_connect()-1435: xfer_status changed from 2 to 2 for global-faz
<16209> _rmt_connect()-1439: setting epoll_hd:0x7f72647715e0 to _rmt_recv
<16209> _check_oftp_certificate()-248: checking sn:FL-8HFT718900132 vs cert sn:FL-
8HFT718900132
<16209> _check_oftp_certificate()-252: Verified the certificate of peer (172.18.64.234) to
match sn=FL-8HFT718900132
<16209> _faz_post_connection()-292: Certificate verification:enabled, Faz verified:1
<16209> _send_queue_item()-518: xfer_status changed from 2 to 1 for global-faz
<16209> _send_queue_item()-523: type=0, cat=0, logcount=0, len=0
<16209> _oftp_send()-487: dev=global-faz type=17 pkt_len=34

<16209> _oftp_send()-487: opt=253, opt_len=10
<16209> _oftp_send()-487: opt=81, opt_len=12
<16208> _rmt_connect()-1433: oftp is ready.
<16208> _rmt_connect()-1435: xfer_status changed from 2 to 2 for global-faz
<16208> _rmt_connect()-1439: setting epoll_hd:0x7fc364e125e0 to _rmt_recv
<16208> _check_oftp_certificate()-248: checking sn:FL-8HFT718900132 vs cert sn:FL-
8HFT718900132
<16208> _check_oftp_certificate()-252: Verified the certificate of peer (172.18.64.234) to
match sn=FL-8HFT718900132
<16208> _faz_post_connection()-292: Certificate verification:enabled, Faz verified:1
<16208> _send_queue_item()-518: xfer_status changed from 2 to 1 for global-faz
<16208> _send_queue_item()-523: type=0, cat=0, logcount=0, len=0
<16208> _oftp_send()-487: dev=global-faz type=17 pkt_len=34
```

```
<16208> _oftp_send()-487: opt=253, opt_len=10
<16209> _oftp_recv()-1348: opt=252, opt_len=996
<16208> _oftp_send()-487: opt=81, opt_len=12
<16209> _process_response()-960: checking opt code=252
<16209> _faz_process_oftp_resp()-488: ha nmember:1 nvcluster:0 mode:1
<16209> __is_sn_known()-356: MATCHED: idx:0 sn:FL-8HFT718900132
<16209> _faz_process_oftp_resp()-494: Received SN:FL-8HFT718900132 should update:0

<16208> _oftp_recv()-1348: dev=global-faz type=252 pkt_len=1008

<16208> _oftp_recv()-1348: opt=252, opt_len=996
<16208> _process_response()-960: checking opt code=252
<16208> _faz_process_oftp_resp()-488: ha nmember:1 nvcluster:0 mode:1
<16208> __is_sn_known()-356: MATCHED: idx:0 sn:FL-8HFT718900132
<16208> _faz_process_oftp_resp()-494: Received SN:FL-8HFT718900132 should update:0

<16206> _rmt_connect()-1433: oftp is ready.
<16206> _rmt_connect()-1435: xfer_status changed from 2 to 2 for global-faz
<16206> _rmt_connect()-1439: setting epoll_hd:0x141f69e0 to _rmt_recv
<16206> _check_oftp_certificate()-248: checking sn:FL-8HFT718900132 vs cert sn:FL-
8HFT718900132
<16206> _check_oftp_certificate()-252: Verified the certificate of peer (172.18.64.234) to
match sn=FL-8HFT718900132
<16206> _faz_post_connection()-292: Certificate verification:enabled, Faz verified:1
<16206> _send_queue_item()-518: xfer_status changed from 2 to 1 for global-faz
<16206> _send_queue_item()-523: type=0, cat=0, logcount=0, len=0
<16206> _oftp_send()-487: dev=global-faz type=17 pkt_len=34

<16206> _oftp_send()-487: opt=253, opt_len=10
<16206> _oftp_send()-487: opt=81, opt_len=12
<16206> _oftp_recv()-1348: dev=global-faz type=252 pkt_len=1008

<16206> _oftp_recv()-1348: opt=252, opt_len=996
<16206> _process_response()-960: checking opt code=252
<16206> _faz_process_oftp_resp()-488: ha nmember:1 nvcluster:0 mode:1
<16206> __is_sn_known()-356: MATCHED: idx:0 sn:FL-8HFT718900132
<16206> _faz_process_oftp_resp()-494: Received SN:FL-8HFT718900132 should update:0

<16209> _oftp_recv()-1348: dev=global-faz type=1 pkt_len=985

<16209> _oftp_recv()-1348: opt=12, opt_len=16
......
<16209> _build_ack()-784: xfer_status changed from 1 to 2 for global-faz
<16209> _process_response()-960: checking opt code=81
......
<16209> _send_queue_item()-523: type=1, cat=0, logcount=0, len=0
<16209> _oftp_send()-487: dev=global-faz type=1 pkt_len=24

<16209> _oftp_send()-487: opt=1, opt_len=12
<16209> _send_queue_item()-523: type=7, cat=0, logcount=0, len=988
<16209> _oftp_send()-487: dev=global-faz type=252 pkt_len=1008

<16209> _oftp_send()-487: opt=252, opt_len=996
<16208> _oftp_recv()-1348: dev=global-faz type=1 pkt_len=58
```

```
<16208> _oftp_recv()-1348: opt=12, opt_len=16
<16208> _oftp_recv()-1348: opt=51, opt_len=9
<16208> _oftp_recv()-1348: opt=49, opt_len=12
<16208> _oftp_recv()-1348: opt=52, opt_len=9
<16208> _build_ack()-784: xfer_status changed from 1 to 2 for global-faz
<16208> _process_response()-960: checking opt code=52
<16208> _send_queue_item()-523: type=1, cat=0, logcount=0, len=0
<16208> _oftp_send()-487: dev=global-faz type=1 pkt_len=24

<16208> _oftp_send()-487: opt=1, opt_len=12
<16206> _oftp_recv()-1348: dev=global-faz type=1 pkt_len=985

......
<16208> _send_queue_item()-523: type=3, cat=1, logcount=1, len=301
<16206> _oftp_recv()-1348: opt=78, opt_len=55
......
<16206> _build_ack()-784: xfer_status changed from 1 to 2 for global-faz
<16206> _process_response()-960: checking opt code=81
......
<16206> _send_queue_item()-523: type=1, cat=0, logcount=0, len=0
<16206> _oftp_send()-487: dev=global-faz type=1 pkt_len=24

<16206> _oftp_send()-487: opt=1, opt_len=12
<16206> _send_queue_item()-523: type=7, cat=0, logcount=0, len=988
<16206> _oftp_send()-487: dev=global-faz type=252 pkt_len=1008

<16206> _oftp_send()-487: opt=252, opt_len=996
<16206> _add_change_notice_queue_item()-269: Change notice packect added to queue. len=145
......
<16206> _send_queue_item()-523: type=2, cat=0, logcount=0, len=300
<16206> _oftp_send()-487: dev=global-faz type=37 pkt_len=300

......

<16206> _oftp_send()-487: opt=152, opt_len=40
<16206> _oftp_send()-487: opt=74, opt_len=40
<16206> _oftp_send()-487: opt=82, opt_len=93
<16206> _oftp_recv()-1348: dev=global-faz type=1 pkt_len=24

<16206> _oftp_recv()-1348: opt=1, opt_len=12
<16206> _process_response()-960: checking opt code=1
```

To check the FortiGate to FortiGate Cloud log server connection status:

```
diagnose test application miglogd 20

FGT-B-LOG # diagnose test application miglogd 20
Home log server:
    Address: 172.16.95.92:514
Alternative log server:
    Address: 172.16.95.26:514
    oftp status: established
Debug zone info:
    Server IP:      172.16.95.92
    Server port:    514
    Server status:  up
```

```
    Log quota:      102400MB
    Log used:       673MB
    Daily volume:   20480MB
    FDS arch pause: 0
    fams archive pause: 0
```

To check real-time log statistics by log type since the miglogd daemon start:

```
diagnose test application miglogd 4

FGT-B-LOG (global) # diagnose test application miglogd 4
info for vdom: root
disk
event: logs=1238 len=262534, Sun=246 Mon=247 Tue=197 Wed=0 Thu=55 Fri=246 Sat=247
compressed=163038
dns: logs=4 len=1734, Sun=0 Mon=0 Tue=0 Wed=0 Thu=4 Fri=0 Sat=0 compressed=453

report
event: logs=1244 len=225453, Sun=246 Mon=247 Tue=197 Wed=0 Thu=61 Fri=246 Sat=247

faz
event: logs=6 len=1548, Sun=0 Mon=0 Tue=6 Wed=0 Thu=0 Fri=0 Sat=0 compressed=5446

info for vdom: vdom1
memory
traffic: logs=462 len=389648, Sun=93 Mon=88 Tue=77 Wed=0 Thu=13 Fri=116 Sat=75
event: logs=3724 len=1170237, Sun=670 Mon=700 Tue=531 Wed=0 Thu=392 Fri=747 Sat=684
app-ctrl: logs=16 len=9613, Sun=3 Mon=3 Tue=3 Wed=0 Thu=0 Fri=5 Sat=2
dns: logs=71 len=29833, Sun=0 Mon=0 Tue=0 Wed=0 Thu=71 Fri=0 Sat=0

disk
traffic: logs=462 len=389648, Sun=93 Mon=88 Tue=77 Wed=0 Thu=13 Fri=116 Sat=75
compressed=134638
event: logs=2262 len=550957, Sun=382 Mon=412 Tue=307 Wed=0 Thu=306 Fri=459 Sat=396
compressed=244606
app-ctrl: logs=16 len=9613, Sun=3 Mon=3 Tue=3 Wed=0 Thu=0 Fri=5 Sat=2 compressed=3966
dns: logs=71 len=29833, Sun=0 Mon=0 Tue=0 Wed=0 Thu=71 Fri=0 Sat=0 compressed=1499

report
traffic: logs=462 len=375326, Sun=93 Mon=88 Tue=77 Wed=0 Thu=13 Fri=116 Sat=75
event: logs=3733 len=1057123, Sun=670 Mon=700 Tue=531 Wed=0 Thu=401 Fri=747 Sat=684
app-ctrl: logs=16 len=9117, Sun=3 Mon=3 Tue=3 Wed=0 Thu=0 Fri=5 Sat=2

faz
traffic: logs=462 len=411362, Sun=93 Mon=88 Tue=77 Wed=0 Thu=13 Fri=116 Sat=75
compressed=307610
event: logs=3733 len=1348297, Sun=670 Mon=700 Tue=531 Wed=0 Thu=401 Fri=747 Sat=684
compressed=816636
app-ctrl: logs=16 len=10365, Sun=3 Mon=3 Tue=3 Wed=0 Thu=0 Fri=5 Sat=2 compressed=8193
dns: logs=71 len=33170, Sun=0 Mon=0 Tue=0 Wed=0 Thu=71 Fri=0 Sat=0 compressed=0
```

To check log statistics to the local/remote log device since the miglogd daemon start:

```
diagnose test application miglogd 6 1     <<<  1 means the first child daemon
```

```
diagnose test application miglogd 6 2     <<<  2 means the second child daemon
```

```
FGT-B-LOG (global) # diagnose test application miglogd 6 1
mem=4288, disk=4070, alert=0, alarm=0, sys=5513, faz=4307, webt=0, fds=0
interface-missed=208
Queues in all miglogds: cur:0  total-so-far:36974
global log dev statistics:
syslog 0: sent=6585, failed=152, relayed=0
faz 0: sent=13, failed=0, cached=0, dropped=0 , relayed=0
```

To check the miglogd daemon number and increase/decrease miglogd daemon:

```
diagnose test application miglogd 15     <<<  Show miglog ID

diagnose test application miglogd 13     <<<  Increase one miglogd child

diagnose test application miglogd 14     <<<  Decrease one miglogd child

FGT-B-LOG (global) # diagnose test application miglogd 15
Main miglogd: ID=0, children=2, active-children=2
        ID=1, duration=70465.
        ID=2, duration=70465.

FGT-B-LOG (global) # diagnose test application miglogd 13

FGT-B-LOG (global) # diagnose test application miglogd 15
Main miglogd: ID=0, children=3, active-children=3
        ID=1, duration=70486.
        ID=2, duration=70486.
        ID=3, duration=1.

FGT-B-LOG (global) # diagnose test application miglogd 14

FGT-B-LOG (global) # diagnose test application miglogd 15
Main miglogd: ID=0, children=2, active-children=2
        ID=1, duration=70604.
        ID=2, duration=70604.
```

To check the remote queue and see the maximum buffered memory size:

```
diagnose test application miglogd 41

cache maximum: 105405644(100MB) objects: 0 used: 0(0MB) allocated: 0(0MB)

VDOM:root
Queue for: global-faz

       memory queue:
               num:0 size:0(0MB) max:105405644(100MB) logs:0

Queue for: fds

        memory queue:
                num:0 size:0(0MB) max:97852620(93MB) logs:0
```

Backing up log files or dumping log messages

When a log issue is caused by a particular log message, it is very help to get logs from that FortiGate. This topic provides steps for using `execute log backup` or dumping log messages to a USB drive.

Backing up full logs using execute log backup

This command backs up all disk log files and is only available on FortiGates with an SSD disk.

Before running `execute log backup`, we recommend temporarily stopping `miglogd` and `reportd`.

To stop and kill miglogd and reportd:

```
diagnose sys process daemon-auto-restart disable miglogd
diagnose sys process daemon-auto-restart disable reportd
```

Or

1. Determine the process, or thread, ID (PID) of `miglogd` and `reportd`:

   ```
   # diagnose sys top 10 99
   ```

2. Kill each process:

   ```
   # diagnose sys kill 9 <PID>
   ```

To store the log file on a USB drive:

1. Plug in a USB drive into the FortiGate.
2. Run this command:

   ```
   execute log backup /usb/log.tar
   ```

To restart miglogd and reportd:

```
diagnose sys process daemon-auto-restart enable miglogd
diagnose sys process daemon-auto-restart enable reportd
```

Dumping log messages

To dump log messages:

1. Enable log dumping for `miglogd` daemon:

   ```
   (global) # diagnose test application miglogd 26 1
   miglogd(1) log dumping is enabled
   ```

2. Display all `miglogd` dumping status:

   ```
   global) # diagnose test application miglogd 26 0 255
   miglogd(0) log dumping is disabled
   miglogd(1) log dumping is enabled
   miglogd(2) log dumping is disabled
   ```

   ```
   (global) # diagnose test application miglogd 26 2
   miglogd(2) log dumping is enabled

   (global) # diagnose test application miglogd 26 0
   miglogd(0) log dumping is enabled

   (global) # diagnose test application miglogd 26 0 255
   miglogd(0) log dumping is enabled
   ```

```
miglogd(1) log dumping is enabled
miglogd(2) log dumping is enabled
```

3. Let the FortiGate run and collect log messages.
4. List the log dump files:

```
(global) # diagnose test application miglogd 33
2019-04-17 15:50:02          20828      log-1-0.dat
2019-04-17 15:48:31           4892      log-2-0.dat
```

5. Back up log dump files to the USB drive:

```
(global) # diagnose test application miglogd 34

Dumping file miglog1_index0.dat copied to USB disk OK.

Dumping file miglog2_index0.dat copied to USB disk OK.
```

6. Disable log dumping for `miglogd` daemon:

```
(global) # diagnose test application miglogd 26 0
miglogd(0) log dumping is disabled

(global) # diagnose test application miglogd 26 1
miglogd(1) log dumping is disabled

(global) # diagnose test application miglogd 26 2
miglogd(2) log dumping is disabled

(global) # diagnose test application miglogd 26 0 255
miglogd(0) log dumping is disabled
miglogd(1) log dumping is disabled
miglogd(2) log dumping is disabled
```

SNMP OID for logs that failed to send

When a syslog server encounters low-performance conditions and slows down to respond, the buffered syslog messages in the kernel might overflow after a certain number of retransmissions, causing the overflowed messages to be lost. OIDs track the lost messages or failed logs.

SNMP query OIDs include log statistics for global log devices:

- FORTINET-FORTIGATE-MIB:fortinet.fnFortiGateMib.fgLog.fgLogDeviceNumber 1.3.6.1.4.1.12356.101.21.1.1
- FORTINET-FORTIGATE-MIB:fortinet.fnFortiGateMib.fgLog.fgLogDevices.fgLogDeviceTable.fgLogDeviceEntry.fgLogDeviceEntryIndex 1.3.6.1.4.1.12356.101.21.2.1.1.1
- FORTINET-FORTIGATE-MIB:fortinet.fnFortiGateMib.fgLog.fgLogDevices.fgLogDeviceTable.fgLogDeviceEntry.fgLogDeviceEnabled 1.3.6.1.4.1.12356.101.21.2.1.1.2
- FORTINET-FORTIGATE-MIB:fortinet.fnFortiGateMib.fgLog.fgLogDevices.fgLogDeviceTable.fgLogDeviceEntry.fgLogDeviceName 1.3.6.1.4.1.12356.101.21.2.1.1.3
- FORTINET-FORTIGATE-MIB:fortinet.fnFortiGateMib.fgLog.fgLogDevices.fgLogDeviceTable.fgLogDeviceEntry.fgLogDeviceSentCount 1.3.6.1.4.1.12356.101.21.2.1.1.4

- FORTINET-FORTIGATE-MIB:fortinet.fnFortiGateMib.fgLog.fgLogDevices.fgLogDeviceTable.fgLogDeviceEntry.fgLogDeviceRelayedCount 1.3.6.1.4.1.12356.101.21.2.1.1.5
- FORTINET-FORTIGATE-MIB:fortinet.fnFortiGateMib.fgLog.fgLogDevices.fgLogDeviceTable.fgLogDeviceEntry.fgLogDeviceCachedCount 1.3.6.1.4.1.12356.101.21.2.1.1.6
- FORTINET-FORTIGATE-MIB:fortinet.fnFortiGateMib.fgLog.fgLogDevices.fgLogDeviceTable.fgLogDeviceEntry.fgLogDeviceFailedCount 1.3.6.1.4.1.12356.101.21.2.1.1.7
- FORTINET-FORTIGATE-MIB:fortinet.fnFortiGateMib.fgLog.fgLogDevices.fgLogDeviceTable.fgLogDeviceEntry.fgLogDeviceDroppedCount 1.3.6.1.4.1.12356.101.21.2.1.1.8

Where:

- fgLogDeviceNumber is the number of devices in the table.
- fgLogDeviceEnabled is either 1 or 0, indicating whether the device is enabled.
- fgLogDeviceName is the name of the device.

A FortiGate connected to a syslog server or FortiAnalyzer generates statistics that can be seen using the `diagnose test application miglogd` command:

```
(global) # diagnose test application miglogd 6
    mem=404, disk=657, alert=0, alarm=0, sys=920, faz=555, webt=0, fds=0
    interface-missed=460
    Queues in all miglogds: cur:0 total-so-far:526
    global log dev statistics:
    syslog 0: sent=254, failed=139, relayed=0
    syslog 1: sent=220, failed=139, relayed=0
    syslog 2: sent=95, failed=73, relayed=0
    faz 0: sent=282, failed=0, cached=0, dropped=0 , relayed=0
    Num of REST URLs: 3
    /api/v2/monitor/system/csf/ : 0 : 300
    /api/v2/cmdb/system/interface/ : 394.0.673.15877729363538323653.1547149763 : 1200
    /api/v2/monitor/system/ha-checksums/ : 0 : 1200
    faz 1: sent=272, failed=0, cached=0, dropped=0 , relayed=0
    Num of REST URLs: 2
    /api/v2/monitor/system/csf/ : 0 : 300
    /api/v2/cmdb/system/interface/ : 394.0.673.15877729363538323653.1547149763 : 1200
```

The same statistics are also available in `snmpwalk/snmpget` on the OID 1.3.6.1.4.1.12356.101.21.

```
snmpwalk -v2c -c REGR-SYS 172.16.200.1 1.3.6.1.4.1.12356.101.21
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.1.1.0 = INTEGER: 9
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.1.0 = INTEGER: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.1.1 = INTEGER: 1
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.1.2 = INTEGER: 2
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.1.3 = INTEGER: 3
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.1.4 = INTEGER: 4
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.1.5 = INTEGER: 5
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.1.6 = INTEGER: 6
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.1.7 = INTEGER: 7
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.1.8 = INTEGER: 8
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.2.0 = INTEGER: 1
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.2.1 = INTEGER: 1
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.2.2 = INTEGER: 1
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.2.3 = INTEGER: 0
```

```
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.2.4 = INTEGER: 1
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.2.5 = INTEGER: 1
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.2.6 = INTEGER: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.2.7 = INTEGER: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.2.8 = INTEGER: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.3.0 = STRING: "syslog"
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.3.1 = STRING: "syslog2"
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.3.2 = STRING: "syslog3"
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.3.3 = STRING: "syslog4"
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.3.4 = STRING: "faz"
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.3.5 = STRING: "faz2"
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.3.6 = STRING: "faz3"
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.3.7 = STRING: "webtrends"
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.3.8 = STRING: "fds"
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.4.0 = Counter32: 254
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.4.1 = Counter32: 220
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.4.2 = Counter32: 95
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.4.3 = Counter32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.4.4 = Counter32: 282
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.4.5 = Counter32: 272
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.4.6 = Counter32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.4.7 = Counter32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.4.8 = Counter32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.5.0 = Counter32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.5.1 = Counter32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.5.2 = Counter32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.5.3 = Counter32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.5.4 = Counter32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.5.5 = Counter32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.5.6 = Counter32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.5.7 = Counter32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.5.8 = Counter32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.6.0 = Gauge32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.6.1 = Gauge32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.6.2 = Gauge32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.6.3 = Gauge32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.6.4 = Gauge32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.6.5 = Gauge32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.6.6 = Gauge32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.6.7 = Gauge32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.6.8 = Gauge32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.7.0 = Counter32: 139
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.7.1 = Counter32: 139
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.7.2 = Counter32: 73
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.7.3 = Counter32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.7.4 = Counter32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.7.5 = Counter32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.7.6 = Counter32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.7.7 = Counter32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.7.8 = Counter32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.8.0 = Counter32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.8.1 = Counter32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.8.2 = Counter32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.8.3 = Counter32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.8.4 = Counter32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.8.5 = Counter32: 0
```

```
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.8.6 = Counter32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.8.7 = Counter32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.8.8 = Counter32: 0
```

To get the type of logging device that is attached to the FortiGate:

```
root@PC05:/home/tester/autolib/trunk# snmpwalk -v2c -c REGR-SYS 172.16.200.1
1.3.6.1.4.1.12356.101.21.2.1.1.3
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.3.0 = STRING: "syslog"
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.3.1 = STRING: "syslog2"
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.3.2 = STRING: "syslog3"
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.3.3 = STRING: "syslog4"
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.3.4 = STRING: "faz"
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.3.5 = STRING: "faz2"
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.3.6 = STRING: "faz3"
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.3.7 = STRING: "webtrends"
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.3.8 = STRING: "fds"
```

To get the present state of the logging device that is attached to the FortiGate:

```
root@PC05:/home/tester/autolib/trunk# snmpwalk -v2c -c REGR-SYS 172.16.200.1
1.3.6.1.4.1.12356.101.21.2.1.1.2
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.2.0 = INTEGER: 1
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.2.1 = INTEGER: 1
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.2.2 = INTEGER: 1
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.2.3 = INTEGER: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.2.4 = INTEGER: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.2.5 = INTEGER: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.2.6 = INTEGER: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.2.7 = INTEGER: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.2.8 = INTEGER: 0
```

To get the failed log count value:

```
root@PC05:/home/tester/autolib/trunk# snmpwalk -v2c -c REGR-SYS 172.16.200.1
1.3.6.1.4.1.12356.101.21.2.1.1.7
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.7.0 = Counter32: 139
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.7.1 = Counter32: 139
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.7.2 = Counter32: 73
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.7.3 = Counter32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.7.4 = Counter32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.7.5 = Counter32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.7.6 = Counter32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.7.7 = Counter32: 0
FORTINET-FORTIGATE-MIB::fnFortiGateMib.21.2.1.1.7.8 = Counter32: 0
```

VM

This section contains topics on deploying FortiGate-VM:

Amazon Web Services

See the FortiOS 7.2.0 AWS Administration Guide.

Microsoft Azure

See the FortiOS 7.2.0 Azure Administration Guide.

Google Cloud Platform

See the 7.2.0 FortiOS GCP Administration Guide.

Oracle OCI

See the 7.2.0 FortiOS OCI Administration Guide.

AliCloud

See the 7.2.0 FortiOS AliCloud Administration Guide.

Private cloud

See FortiGate Private Cloud in the document library.

VM license

You can access the *FortiGate VM License* page from the *Dashboard* > *Status* page in the *Virtual Machine* widget. Click the device license and select *FortiGate VM License*.

The *FortiGate VM License* page displays the following information:

Field	Description
License status	Displays one of the following statuses: • Valid: VM can connect and validate the license against a FortiManager or FortiGuard server. All features are available. • Warning: VM cannot connect and validate against a FortiManager or FortiGuard server. A check is made against how many days the warning status is continuous. If the number is less than 30 days, the status does not change. • Invalid: VM cannot connect and validate against a FortiManager or FortiGuard server. A check is made against how many days the warning status is continuous. If the number is 30 days or more, the status changes to invalid. GUI access is restricted until a valid license is uploaded. Firewall policies do not work. FortiGuard downloads are unavailable. • Pending: temporary state where the VM attempts to validate its license. Reasons for having a warning or invalid status include: • The network environment does not allow FortiGate-VM to connect to the FortiGuard server. • The license may be expired. Check the expiration date for evaluation or term-based licenses. • Another VM has been already validated with FortiGuard using the same license. See VM license activation for details about duplicated VM instances.
Allocated vCPUs	Number of allocated and total allowable vCPUs
Allocated RAM	Amount of allocated RAM. There are no RAM restrictions.
Expires on	Expiry date (value depends on the type of license)

This information is visible in the CLI by running `get system status` (see CLI troubleshooting).

Uploading a license file

After you submit an order for a FortiGate-VM, Fortinet sends a license registration code to the email address that you entered in the order form. Use this code on the FortiCloud portal to register the FortiGate-VM.

Once the VM is registered, you can download the license file in .LIC format. On the *FortiGate VM License* page, click *Upload*. The system prompts you to reboot and validate the license with the FortiGuard server. Once validated, your FortiGate-VM is fully functional.

The VM license window may also appear immediately after logging in if you are running a VM with an evaluation license that has expired.

In cases where the GUI is not accessible, you can upload the license using secure copy (SCP).

For information about injecting FortiFlex license tokens, see *Injecting tokens into FortiGate-VM* in the *Flex VM Deployment Guide*.

To upload the license using SCP:

1. Enable SCP:

```
config system global
    set admin-scp enable
end
```

2. Enable SSH in the administrative access for the interface where the transfer will take place:

```
config system interface
    edit <interface>
        append allowaccess ssh
    next
end
```

3. On your computer, upload the VM license. This example is for Linux:

```
scp <filename> <admin-user>@<FortiGate_IP>:vmlicense
```

Types of VM licenses

FortiGate-VM offers perpetual licensing (normal series and V-series) and annual subscription licensing (S-series). SKUs are based on the number of vCPUs (1, 2, 4, 8, 16, 32, or unlimited).

The FortiFlex program allows qualified enterprise and MSSP customers to create as many VM entitlements as required. Resource consumption is based upon predefined points that are calculated on a daily basis. For information, see the *FortiFlex Program Guide* in the Fortinet document library.

Feature	Normal series	V-series	S-series	FortiFlex
Licensing and support	The VM base is perpetual. You must separately contract support services on an annual basis. See the price list for details.		Single annually contracted SKU that contains a VM base and a FortiCare service bundle. Four support service bundle types are available: • Only FortiCare • UTM • Enterprise • ATP	An annually contracted program to create multiple sets of a single entitlement per VM. Entitlements contain a VM base and FortiCare bundle. Four support service bundle types are available: • Only FortiCare • UTM • Enterprise • ATP
vCPU number upgrade during contracted term	Not supported.		Supported. You can also upgrade the support service bundle. Contact a Fortinet sales representative to upgrade.	Supported. You can apply different VM entitlement configurations in the FortiFlex portal. API is not supported at this time.
vCPU number downgrade during contracted term	Not supported.			
VDOM support	By default, each CPU level supports up to a certain number of VDOMs. Refer to the FortiGate-VM data sheet for default limits.	By default, all CPU levels do not support adding VDOMs.	By default, all CPU levels do not support adding VDOMs. S-series VM instances support the subscription VDOM license.	

Consuming a new vCPU

In a scenario where you have not allocated all the vCPUs allotted by your VM entitlement, you can add additional vCPUs to your FortiGate VM. The vCPU allocation can be verified in the GUI and CLI.

To confirm the vCPU allocation in the GUI:

1. Go to *Dashboard* > *Status*, and locate the *Virtual Machine* widget.
2. Verify the *Allocate vCPUs* field, which displays the number and percentage of allocated vCPUs.

To confirm the vCPU allocation in the CLI:

```
# get system status | grep "VM Resources"
VM Resources: 1 CPU/4 allowed, 2006 MB RAM
```

You can increase the number of vCPUs on running FortiGate VM models that support hot-adding. Once the hot-adding is complete, perform one of the following for FortiOS to recognize the new CPUs:

- Enter `execute cpu add <number_of_new_vCPUs>`.
- Reboot the FortiGate.

CLI troubleshooting

In some cases, you can view more information from the CLI to diagnose issues with VM licensing. This is also useful when the GUI is inaccessible due to an invalid contract.

Before you begin, ensure your FortiGate has the proper routes to connect to the internet. Run all following debug commands for a full picture of the issue.

To view the license status, expiration date, and VM resources:

```
# get system status
Version: FortiGate-VM64-KVM v6.4.2,build1723,200730 (GA)
...
Serial-Number: FGVM08**********
....
License Status: Valid
License Expiration Date: 2020-12-10
VM Resources: 1 CPU/8 allowed, 2010 MB RAM
...
```

To display license details:

```
# diagnose debug vm-print-license
SerialNumber: FGVM08**********
CreateDate: Tue Dec 10 00:57:32 2019
License expires: Thu Dec 10 00:00:00 2020
Expiry: 366
Key: yes
Cert: yes
Key2: yes
Cert2: yes
Model: 08 (11)
CPU: 8
MEM: 2147483647
```

To display license information from FortiGuard:

```
# diagnose hardware sysinfo vm full
UUID:     abbe****************************
valid:    1
status:   1
code:     200
warn:     0
```

```
copy:     0
received: 4604955037
warning:  4600905081
recv:     202009152207
dup:
```

Field	Value and description
Valid	0 – Invalid 1 – Valid
Status	0 – Startup 1 – Success 2 – Warning 3 – Error 4 – Invalid Copy 5 – Eval Expired 6 - Grace Period. For FortiFlex, there is a two-hour grace period to begin passing traffic upon retrieving the license from FortiCare.
Code	2xx, 3xx – Success 200 – Valid 202 – Accepted (treated as correct response code) 4xx - Error 400 – Expired 401 – Duplicate 5xx – Warning 500 - Warning 502 – Invalid. Cannot connect to FortiGuard Distribution Servers 6xx – Evaluation license expired Other codes - Error

The following are examples of common combinations:

This combination indicates the license is valid and functioning normally:

```
valid: 1
status: 1
code: 200
```

This combination indicates the license itself is valid, but is running on a duplicate instance:

```
valid: 1
status: 4
code: 401
```

This combination indicates the system cannot connect to FortiGuard:

```
valid:    0
status:   2
code:     502
```

This combination indicates the license is expired and invalid:

```
valid: 0
status: 3
code: 400
```

This combination indicates the VM is unlicensed:

```
valid: 0
status: 3
code: 0
```

For FortiFlex licenses, the following command allows you to enter the license token and proxy information:

```
# execute vm-license <token> https://<username>:<password>@<proxy IP address>:<proxy port>
```

FortiOS can receive the following error codes from the FortiCare server:

```
1 - Runtime error (server unhandled error on FortiCare sever)
57 - License Token is invalid
58 - License Token is already used and cannot be used again to retrieve license key
```

The FortiGate can generate the following error code:

```
60 - Failed to request forticare license. Failed to download VM license.
```

Contact Fortinet Support for assistance if your licensing issue persists.

Adding VDOMs with FortiGate v-series

Each FortiGate-VM base license type allows a default number of virtual domains (VDOM). This topic provides sample procedures to add VDOMs beyond the default number using separately purchased VDOM licenses.

This topic consists of the following steps:

1. Activate the FortiGate-VM with the base license.
2. Add more VDOMs to the FortiGate-VM.

To activate the FortiGate-VM with the base license:

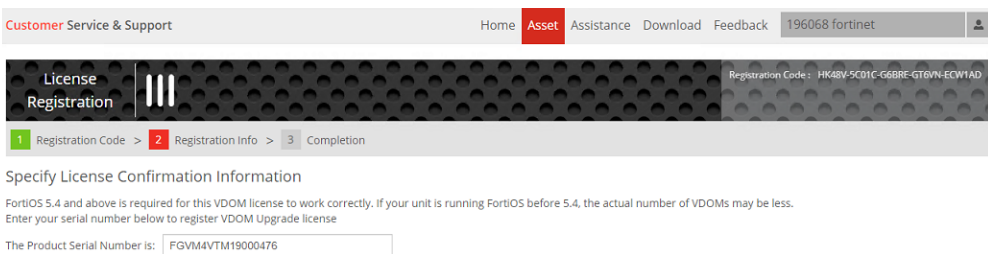
1. Purchase and register the FortiGate-VM base license in FortiCare:
   a. Purchase the FortiGate-VM base license from Fortinet or a Fortinet reseller.
   b. You receive a license certification with a registration code. Open the certification.
   c. Log in to Fortinet Customer Service & Support.
   d. Go to *Register Now* and enter the provided registration code.
   e. Follow the registration process. The serial number generates and displays on the *Registration Completion* page.
   f. Go to *Asset > Manage/View Products*. Click the serial number to download the license file.
2. Upload the FortiGate-VM base license file to FortiOS:
   a. Log in to the FortiGate-VM GUI.
   b. In *Dashboard > Status*, in the *Virtual Machine* widget, click *FortiGate VM License*.
   c. Click the *Upload* button.
   d. Select the FortiGate-VM base license file, then click *OK*. The FortiGate-VM reboots after applying the base license.

3. Verify the FortiGate-VM base license status and VDOM information:
   a. Log in to the FortiGate-VM GUI.
   b. In *Dashboard > Status*, in the *Virtual Machine* widget, ensure that there is a checkmark in front of the FortiGate-VM base license name. The checkmark indicates that the base license is valid.
   c. You can check VDOM information using the CLI. The following output shows that the maximum number of VDOMs is currently one. This is correct since the FortiGate-VM base license only supports the default root VDOM that the system uses.

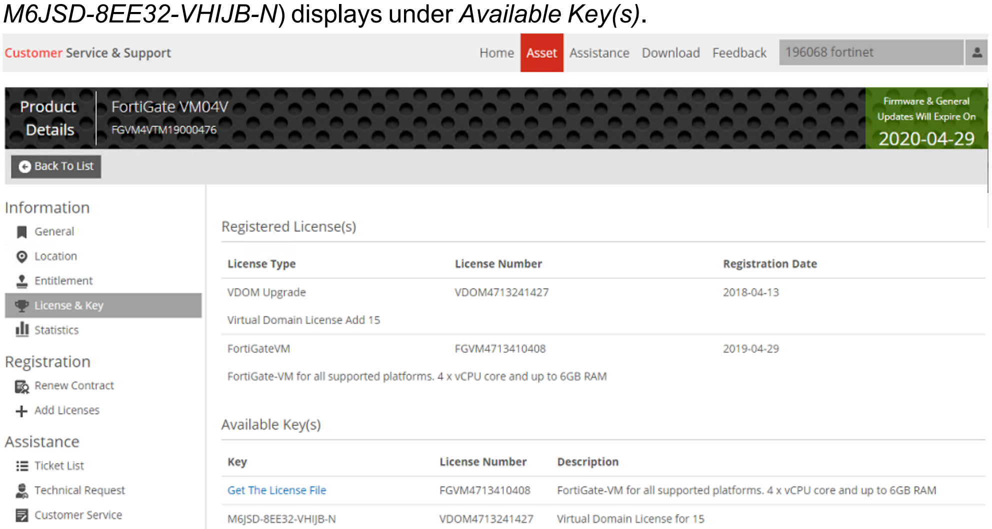
To add more VDOMs to the FortiGate-VM:

You can repeat this procedure multiple times to stack multiple VDOM licenses on the same FortiGate-VM.

1. Purchase and register the FortiGate-VM upgrade license in FortiCare. This example adds 15 VDOMs:
   a. Purchase the FortiGate-VM upgrade license from Fortinet or a Fortinet reseller.
   b. You receive a license certification with a registration code. Open the certification.
   c. Log in to Fortinet Customer Service & Support.
   d. Go to *Asset > Register/Activate* and enter the provided registration code.
   e. On the *Specify License Confirmation Information* screen, enter the FortiGate-VM serial number to apply the VDOM upgrade license to the FortiGate-VM. In this example, the FortiGate-VM serial number is *FGVM4VTM19000476*.



   f. Follow the registration process.
   g. Go to *Asset > Manage/View Products >* . Select the desired product, then click *License & Key*. The VDOM upgrade license displays under *Registered License(s)*, and a key for adding 15 VDOMs (in this example *M6JSD-8EE32-VHIJB-N*) displays under *Available Key(s)*.



2. Apply the FortiGate-VM upgrade license key to FortiOS:
   a. Log in to the FortiGate-VM CLI in the local console or using SSH.
   b. Apply the VDOM upgrade license key:
   ```
   FGVM4VTM19000476 # execute upd-vd-license M6JSD-8EE32-VHIJB-N
   update vdom license succeeded
   ```
3. Verify the FortiGate-VM VDOM information:
   a. Log in to the FortiGate-VM CLI in the local console or using SSH.
   b. Check VDOM information using the CLI. The following output shows that the maximum number of VDOMs is currently 15. When you add VDOMs for the first time on a FortiGate-VM v-series instance, FortiOS does not count the default VDOM, as the default VDOM is the so-called root VDOM that the system uses and FortiOS does not treat it as a countable VDOM in terms of VDOM addition. Therefore, as in this example, if your FortiGate-VM had the default VDOM configuration, then you add 15 VDOMs, FortiOS displays the maximum VDOM number as 15, not 16.

   ```
   # get system status
   Version: FortiGate-VM64-KVM v6.4.4,build1803,201209 (GA)
   Virus-DB: 82.00644(2020-12-18 12:20)
   Extended DB: 82.00644(2020-12-18 12:20)
   Extreme DB: 1.00000(2018-04-09 18:07)
   IPS-DB: 16.00982(2020-12-17 01:04)
   IPS-ETDB: 0.00000(2001-01-01 00:00)
   APP-DB: 16.00982(2020-12-17 01:04)
   INDUSTRIAL-DB: 6.00741(2015-12-01 02:30)
   Serial-Number: FGVM02TM20000000
   IPS Malicious URL Database: 2.00862(2020-12-18 06:12)
   License Status: Invalid Copy
   License Expiration Date: 2021-10-02
   VM Resources: 2 CPU/2 allowed, 2010 MB RAM
   Log hard disk: Available
   Hostname: FGDocs
   Private Encryption: Disable
   Operation Mode: NAT
   Current virtual domain: root
   Max number of virtual domains: 1
   Virtual domains status: 1 in NAT mode, 0 in TP mode
   Virtual domain configuration: disable
   FIPS-CC mode: disable
   Current HA mode: standalone
   Branch point: 1803
   Release Version Information: GA
   FortiOS x86-64: Yes
   System time: Fri Dec 25 13:24:20 2020
   ```

Terraform: FortiOS as a provider

Fortinet's Terraform support provides customers with more ways to efficiently deploy, manage, and automate security across physical FortiGate appliances and virtual environments. You can use Terraform to automate various IT infrastructure needs, thereby diminishing mistakes from repetitive manual configurations.

For example, if Fortinet is releasing a new FortiOS version, your organization may require you to test a new functionality to determine how it may impact the environment before globally deploying the new version. In this case, the ability to rapidly stand up environments and test these functions prior to production environment integration provides a resource-efficient and fault-tolerant approach.

The following example demonstrates how to use the Terraform FortiOS provider to perform simple configuration changes on a FortiGate unit. It requires the following:

- FortiOS 6.0 or later
- FortiOS Provider: This example uses terraform-provider-fortios 1.0.0.
- Terraform: This example uses Terraform 0.11.14.
- REST API administrator created on the FortiGate with the API key

For more information, see the Terraform FortiOS Provider at https://www.terraform.io/docs/providers/fortios/index.html.

To create a REST API administrator:

1. On the FortiGate, go to *System* > *Administrators* and click *Create New* > *REST API Admin*.
2. Enter the *Username* and, optionally, enter *Comments*.
3. Select an *Administrator Profile*.
4. We recommend that you create a new profile with minimal privileges for this terraform script:
   a. In the *Administrator Profile* drop down click *Create New*.
   b. Enter a name for the profile.
   c. Configure the *Access Permissions*:
      - *None*: The REST API is not permitted access to the resource.
      - *Read*: The REST API can send read requests (`HTTP GET`) to the resource.
      - *Read/Write*: The REST API can send read and write requests (`HTTP GET`/`POST`/`PUT`/`DELETE`) to the resource.
   d. Click *OK*.
5. Enter *Trusted Hosts* to specify the devices that are allowed to access this FortiGate.
6. Click *OK*.
   An API key is displayed. This key is only shown once, so you must copy and store it securely.

To configure FortiGate with Terraform Provider module support:

1. Download the terraform-provider-fortios file to a directory on the management computer.
2. Create a new file with the .tf extension for configuring your FortiGate:
   ```
   root@mail:/home/terraform# ls
   terraform-provider-fortios_v1.0.0_x4 test.tf
   ```
3. Edit the test.tf Terraform configuration file:
   In this example, the FortiGate's IP address is 10.6.30.5, and the API user token is 17b*******************63ck. Your provider information must also be changed.
   ```
   # Configure the FortiOS Provider
   provider "fortios" {
   hostname = "10.6.30.5"
   token = "17b*******************63ck"
   }
   ```
4. Create the resources for configuring your DNS object and adding a static route:
   ```
   resource "fortios_system_setting_dns" "test1" {
   primary = "172.16.95.16"
   secondary = "8.8.8.8"
   ```

```
}
resource "fortios_networking_route_static" "test1" {
dst = "110.2.2.122/32"
gateway = "2.2.2.2"
blackhole = "disable"
distance = "22"
weight = "3"
priority = "3"
device = "port2"
comment = "Terraform test"
}
```

5. Save your Terraform configuration file.
6. In the terminal, enter `terraform init` to initialize the working directory.
   It reads the provider if the name follows the convention `terraform-provider-[name]`:

```
root@mail:/home/terraform# terraform init
Initializing the backend...
Terraform has been successfully initialized!
You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.
If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
```

7. Run `terraform -v` to verify the version of loaded provider module:

```
root@mail:/home/terraform# terraform -v
Terraform v0. 11.14
+ provider.fortios v1.0.0
```

8. Enter `terraform plan` to parse the configuration file and read from the FortiGate configuration to see what Terraform changes:
   This example create a static route and updates the DNS address. You can see that Terraform reads the DNS addresses from the FortiGate and then lists them.

```
root@mail:/home/terraform# terraform plan
Refreshing Terraform state in-memory prior to plan...
The refreshed state will be used to calculate this plan, but will not be
persisted to local or remote state storage.
fortios_networking_route_static.test1: Refreshing state... (ID: 2)
fortios_system_setting_dns.test1: Refreshing state... (ID: 96.45.45.45)
------------------------------------------------------------------------
An execution plan has been generated and is shown below.
Resource actions are indicated with the following symbols:
+ create
~ update in-place
Terraform will perform the following actions:
+ fortios_networking_route_static.test1
id: <computed>
blackhole: "disable"
comment: "Terraform test"
device: "port2"
distance: "22"
dst: "110.2.2.122/32"
gateway: "2.2.2.2"
```

```
priority: "3"
weight: "3"
~ fortios_system_setting_dns.test1
primary: "96.45.45.45" => "172.16.95.16"
secondary: "208.91.112.22" => "8.8.8.8"
Plan: 1 to add, 1 to change, 0 to destroy.
------------------------------------------------------------------------
Note: You didn't specify an "-out" parameter to save this plan, so Terraform
can't guarantee that exactly these actions will be performed if
"terraform apply" is subsequently run.
```

If you are running terraform-provider-fortios 1.1.0, you may see the following error:

```
Error: Error getting CA Bundle, CA Bundle should be set when
     insecure is false.
```

In this case, add the following line to the FortiOS provider configuration in the test.tf file:

```
insecure = "true"
```

9. Enter `terraform apply` to continue the configuration:

```
root@mail:/home/terraform# terraform apply
fortios_system_setting_dns.test1: Refreshing state... (ID: 96.45.45.45)
fortios_networking_route_static.test1: Refreshing state... (ID: 2)
An execution plan has been generated and is shown below.
Resource actions are indicated with the following symbols:
+ create
~ update in-place
Terraform will perform the following actions:
+ fortios_networking_route_static.test1
id: <computed>
blackhole: "disable"
comment: "Terraform test"
device: "port2"
distance: "22"
dst: "110.2.2.122/32"
gateway: "2.2.2.2"
priority: "3"
weight: "3"
~ fortios_system_setting_dns.test1
primary: "96.45.45.45" => "172.16.95.16"
secondary: "208.91.112.22" => "8.8.8.8"
Plan: 1 to add, 1 to change, 0 to destroy.
Do you want to perform these actions?
Terraform will perform the actions described above.
Only 'yes' will be accepted to approve.
Enter a value: yes
fortios_networking_route_static.test1: Creating...
blackhole: "" => "disable"
comment: "" => "Terraform test"
device: "" => "port2"
distance: "" => "22"
dst: "" => "110.2.2.122/32"
gateway: "" => "2.2.2.2"
priority: "" => "3"
weight: "" => "3"
fortios_system_setting_dns.test1: Modifying... (ID: 96.45.45.45)
```

```
primary: "96.45.45.45" => "172.16.95.16"
secondary: "208.91.112.22" => "8.8.8.8"
fortios_networking_route_static.test1: Creation complete after 0s (ID: 2)
fortios_system_setting_dns.test1: Modifications complete after 0s (ID: 172.16.95.16)
Apply complete! Resources: 1 added, 1 changed, 0 destroyed.
```

The FortiGate is now configured according to the configuration file.

10. To change or delete something in the future, edit the configuration file and then apply it again. In supported cases, it deletes, adds, or updates new entries as configured. For instance, in this example you can remove the static route and revert the DNS address to its original configuration by changing the .tf file:
    a. Edit the configuration file:

```
# Configure the FortiOS Provider
provider "fortios" {
hostname = "10.6.30.5"
token = "17b*******************63ck"
}
resource "fortios_system_setting_dns" "test1" {
primary = "96.45.45.45"
secondary = "208.91.112.22"
}
#resource "fortios_networking_route_static" "test1" {
# dst = "110.2.2.122/32"
# gateway = "2.2.2.2"
# blackhole = "disable"
# distance = "22"
# weight = "3"
# priority = "3"
# device = "port2"
# comment = "Terraform test"
#}
```

    b. Entering `terraform apply` deletes the static route that is commented out of the configuration file, and reverts the DNS address to the old address:

```
root@mail:/home/terraform# terraform apply
fortios_system_setting_dns.test1: Refreshing state... (ID: 172.16.95.16)
fortios_networking_route_static.test1: Refreshing state... (ID: 2)
An execution plan has been generated and is shown below.
Resource actions are indicated with the following symbols:
~ update in-place
- destroy
Terraform will perform the following actions:
- fortios_networking_route_static.test1
~ fortios_system_setting_dns.test1
primary: "172.16.95.16" => "96.45.45.45"
secondary: "8.8.8.8" => "208.91.112.22"
Plan: 0 to add, 1 to change, 1 to destroy.
Do you want to perform these actions?
Terraform will perform the actions described above.
Only 'yes' will be accepted to approve.
Enter a value: yes
fortios_networking_route_static.test1: Destroying... (ID: 2)
fortios_system_setting_dns.test1: Modifying... (ID: 172.16.95.16)
primary: "172.16.95.16" => "96.45.45.45"
secondary: "8.8.8.8" => "208.91.112.22"
fortios_networking_route_static.test1: Destruction complete after 0s
```

```
fortios_system_setting_dns.test1: Modifications complete after 0s (ID: 96.45.45.45)
Apply complete! Resources: 0 added, 1 changed, 1 destroyed.
```

Troubleshooting

Use the HTTPS daemon debug to begin troubleshooting why a configuration was not accepted:

```
# diagnose debug enable
# diagnose debug application httpsd -1
```

The REST API 403 error means that your administrator profile does not have sufficient permissions.

The REST API 401 error means that you do not have the correct token or trusted host.

PF and VF SR-IOV driver and virtual SPU support

Physical Function (PF) and Virtual Function (VF) PCI Passthrough and SR-IOV drivers in FortiGate guest VM are supported.

PF provides the ability for PCI Passthrough, but requires an entire Network Interface Card (NIC) for a VM. It can usually achieve greater performance than a Virtual Function (VF) based SR-IOV. PF is also expensive. While VF allows one NIC to be shared among multiple guests VMs, PF is allocated to one port on a VM.

The supported driver versions are:

Driver	Version	Hypervisor	PCI-Passthrough/SR-IOV	vSPU (In-guest DPDK)	Note
ixgbe	5.6.5	ESXi, KVM	Yes	Yes	
Ixgbevf	4.6.3	ESXi, KVM	Yes		
i40e	2.10.19.82	ESXi, KVM	Yes	Yes	
i40evf	3.6.15	ESXi, KVM	Yes	Yes	Available in FortiOS 6.4.0 and earlier versions.
Iavf	4.1.1	ESXi, KVM	Yes	Yes	Replaces i40evf in FortiOS 6.4.1 and later versions. Supports Intel E810-C 100G adapters.
Mlx5	4.6-1.0.1	ESXi, KVM	Yes	Yes	Supports Nvidia ConnectX-5 and ConnectX-6 100G adapters.

Driver	Version	Hypervisor	PCI-Passthrough/SR-IOV	vSPU (In-guest DPDK)	Note
Bnxt_en	1.10.1-216.0.416.1	ESXi, KVM	Yes	Yes	Available in FortiOS 6.4.3 and later versions. Supports Broadcom P2100G 100G adapters.
Vmxnet3	1.4.a.0-k-NAPI	ESXi		Yes	The combination of VMware ESXi and NSX-T does not support virtual SPU (vSPU).

Other hypervisors, such as Xen or Microsoft Hyper-V, may work with vSPU, although they are unverified.

All tools and software utilities for UEFI 1.X have been removed from 6.2.0 and later releases. Update to UEFI 2.x to use the UEFI tools or software utilities.

You perform the configuration to use PF or VF on the hypervisor, and do not configure it on the FortiGate.

To check what driver is being used on the FortiGate:

```
# diagnose hardware deviceinfo nic port2
Name:          port2
Driver:        i40e
Version:       2.4.10
Bus:           0000:03:00.0
Hwaddr:        3c:fd:fe:1e:98:02
Permanent Hwaddr:3c:fd:fe:1e:98:02
State:         up
Link:          up
Mtu:           1500
Supported:     auto 1000full 10000full
Advertised:    auto 1000full 10000full
Auto:          disabled
Rx packets:        0
Rx bytes:          0
Rx compressed:         0
...
```

Using OCI IMDSv2

OCI IMDSv2 offers increased security for accessing instance metadata compared to IMDSv1. IMDSv2 is used in OCI SDN connectors and on instance deployments with bootstrap metadata. When upgrading from previous FortiOS builds with legacy IMDSv1 endpoints, the endpoints will be updated to IMDSv2, and the same calls can be made.

The following use cases illustrate IMDSv2 support on the FortiGate-VM.

To configure the Oracle OCI instance to use IMDSv2:

1. In OCI, deploy an instance using IMDSv2 with bootstrap metadata. There are two methods to enable IMDSv2 :
   - Use the OCI command line to deploy an instance using `user-data`. This example uses a MIME file that contains the license and configuration, as well as a JSON file that specifies to disable V1 metadata.

     ```
     oci compute instance launch
     --availability-domain wwwl:US-ASHBURN-AD-1
     --compartment-id
     ocid1.tenancy.oc1..aaaaaaaaaaa3aaaaaaaaaaaaaaaaa7xxxxxxx54aaaaaa4xxxxxxxx55xxxa
     --display-name fos-byol-v6.4.6-b2290-emulated
     --image-id
     ocid1.image.oc1.iad.aaaaaaaa6xxx43xxxxxxxxx7aaaaaaaaaaaaaaaaaaa3xxxxxxxxxxxxxxx
     --subnet-id
     ocid1.subnet.oc1.iad.aaaaaaaaxxxxxxxxx2xxxxxxxxxxxxxxxxxxxx5aaa4xxxxxxxxxxxx42aaa
     --shape VM.Standard1.4
     --assign-public-ip true
     --user-data-file /home/oci/userdata/mime.txt
     --ssh-authorized-keys-file /home/oci/userdata/myfirstkeypair.pub
     --instance-options file://home/oci/scripts/metadatav2.json
     ```

     ```
     root@mail:/home/oci/scripts# cat metadatav2.json
     {
       "areLegacyImdsEndpointsDisabled": true
     }
     ```

   - While the instance is running, edit the instance metadata service version in the GUI ,and change the allowed IMDS version to *VERSION 2 ONLY* (see Getting Instance Metadata in the OCI documentation).

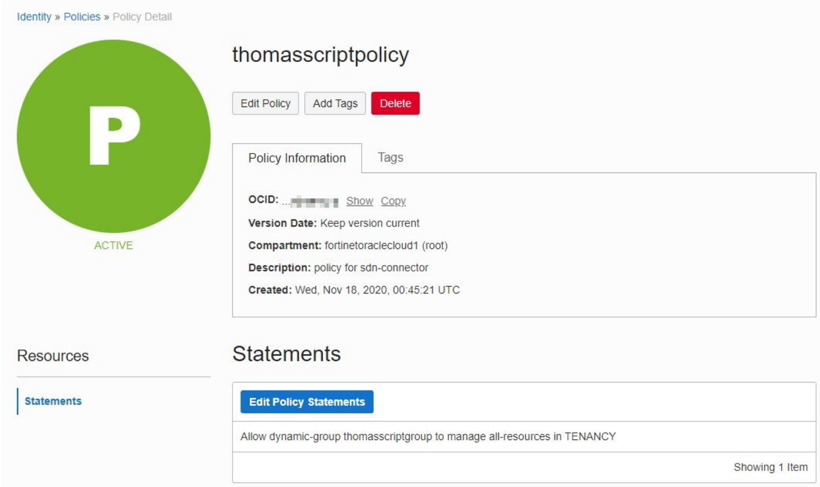
     

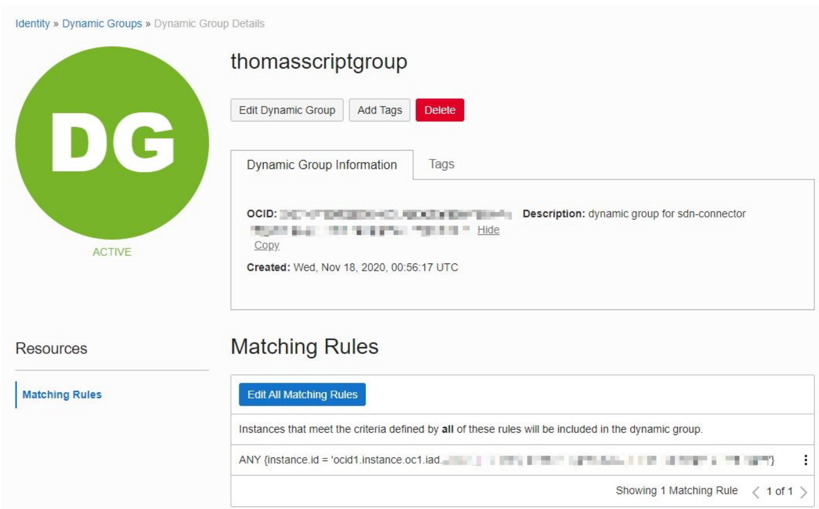
2. The FortiGate will use the metadata v2 endpoints to get the metadata bootstrap information. In FortiOS, verify this by running the following after bootup:

   ```
   # diagnose debug cloudinit show
   ```

To configure an SDN connector with meta-IAM enabled and firewall addresses to obtain dynamic addresses:

1. Configure an IAM policy and dynamic group (see How Policies Work and Managing Dynamic Groups in the OCI documentation).





2. In FortiOS, configure the OCI Fabric connector (see OCI SDN connector using certificates on page 2471 for detailed instructions):
   a. Create the SDN connector.
   b. Verify that the OCI connector comes up (*Security Fabric > External Connectors* page indicates the status is up).
   c. Configure a dynamic firewall address with a filter.
   d. Verify the dynamic firewall address is resolved by the SDN connector.

To manually update the external IP:

```
# execute update-eip
instance: fos-byol-v6.4.6-b2290-emulated
    vnic0: fos-byol-v6.4.6-b2290-emulated
           10.0.0.58 (129.213.138.192)
port1: 10.0.0.58, eip: 129.213.138.192
EIP is updated successfully
```

To verify the OCI daemon debugs related to metadata:

```
# diagnose test application ocid 4
instance: fos-byol-v6.4.6-b2290-emulated
    vnic0: fos-byol-v6.4.6-b2290-emulated
           10.0.0.58

# diagnose test application ocid 5
Compartment
Id:ocid1.tenancy.oc1..aaaaaaaaaaa3aaaaaaaaaaaaaaaaa7xxxxxxx54aaaaaa4xxxxxxxx55xxxa
Instance Id:ocid1.instance.oc1.iad.axxxxxxxxxxxxxxxxxxx4aaaaa5aaaaaaaaa4xxxxxxx2aaaaaaaa
Instance Name:fos-byol-v6.4.6-b2290-emulated
OCI Regarxiehlion:us-ashburn-1

# diagnose test application ocid 6
Instance Principal Token has been refreshed
```

FIPS cipher mode for AWS, Azure, OCI, and GCP FortiGate-VMs

AWS, Azure, OCI, and GCP FortiGate-VMs support FIPS cipher mode. You must remove all VPN configurations before you can enable FIPS CC mode.

FIPS cipher mode only allows a restricted set of ciphers for features that require encryption, such as SSH, IPsec and SSL VPN, and HTTPS. You cannot use insecure protocols such as Telnet, TFTP, and HTTP to access the FortiGate-VM.

You must perform a factory reset to disable `fips-ciphers` mode.

To enable fips-cipher mode:

```
config system fips-cc
    set status fips-ciphers
end
Warning: entering fips-ciphers mode. To exit this mode, factory reset is required.
Do you want to continue? (y/n) y
```

The following behavior occurs when you enable FIPS cipher mode:

- You can restore a license, image, configuration, and so on from an FTP server.
- The following options are available:

SSH algorithms	• aes128-gcm@openssh.com • aes256-gcm@openssh.com • hmac-sha2-256 • hmac-sha2-512
IKE/IPsec phase1 proposals	• aes128-sha256 • aes128-sha256 • aes128-sha384 • aes128-sha384 • aes128-sha512

	• aes128-sha512 • aes128gcm-prfsha256 • aes128gcm-prfsha256 • aes128gcm-prfsha384 • aes128gcm-prfsha384 • aes128gcm-prfsha512 • aes128gcm-prfsha512 • aes256-sha256 • aes256-sha256 • aes256-sha384 • aes256-sha384 • aes256-sha512 • aes256-sha512 • aes256gcm-prfsha256 • aes256gcm-prfsha256 • aes256gcm-prfsha384 • aes256gcm-prfsha384 • aes256gcm-prfsha512 • aes256gcm-prfsha512
IKE/IPsec phase2 proposals	• aes128-sha256 • aes128-sha256 • aes128-sha384 • aes128-sha384 • aes128-sha512 • aes128-sha512 • aes128gcm • aes128gcm • aes256-sha256 • aes256-sha256 • aes256-sha384 • aes256-sha384 • aes256-sha512 • aes256-sha512 • aes256gcm • aes256gcm
IKE/IPsec DH groups	• Default = 19, or any three from 14 - 21, 27 - 32
HTTPS for admin and SSL VPN (with RSA server certificate) TLS suites	PFS: • TLS_AES_256_GCM_SHA384 • ECDHE-RSA-AES256-GCM-SHA384 • DHE-RSA-AES256-GCM-SHA384 • TLS_AES_128_GCM_SHA256 • ECDHE-RSA-AES128-GCM-SHA256 • DHE-RSA-AES128-GCM-SHA256 Elliptic curves:

	• prime256v1 • secp384r1 • secp521r1 DH group: • RFC3526/Oakley group 14 (2048 bits)
HTTPS for admin and SSL VPN (with ECC server certificate) TLS suites	PFS: • TLS_AES_256_GCM_SHA384 • ECDHE-ECDSA-AES256-GCM-SHA384 • TLS_AES_128_GCM_SHA256 • ECDHE-ECDSA-AES128-GCM-SHA256 Elliptic curves: • prime256v1 • secp384r1 • secp521r1

- The FortiCare license is validated.
- FortiGuard databases and engines are updated.
- The DH-RSA-AES128-GCM-SHA256 and DH-RSA-AES256-GCM-SHA384 ciphers are not supported.

Troubleshooting

This section is intended for administrators with super_admin permissions who require assistance with basic and advanced troubleshooting. Administrators with other types of permissions may not be able to perform all of the tasks in this section.

This section contains the following troubleshooting topics:

Troubleshooting methodologies

The sections in this topic provide an overview of how to prepare to troubleshoot problems in FortiGate. They include verifiying your user permissions, establishing a baseline, defining the problem, and creating a plan.

Verify user permissions

Before you begin troubleshooting, verify the following:

- You have administrator privileges for the FortiGate.
- The FortiGate is integrated into your network.
- The operation mode is configured.
- The system time, DNS settings, administrator password, and network interfaces are configured.
- Firmware, FortiGuard AntiVirus, FortiGuard Application Control, and FortiGuard IPS are up to date.

If you are using a FortiGate that has virtual domains (VDOMs) enabled, you can often troubleshoot within your own VDOM. However, you should inform the super_admin for the FortiGate that you will be performing troubleshooting tasks.

You may also need access to other networking equipment, such as switches, routers, and servers to carry out tests. If you do not have access to this equipment, contact your network administrator for assistance.

Establish a baseline

FortiGate operates at all layers of the OSI model. For this reason, troubleshooting can be complex. Establishing baseline parameters for your system before a problem occurs helps to reduce the complexity when you need to troubleshoot.

A best practice is to establish and record the normal operating status. Regular operation data shows trends, and allows you to see where changes occur when problems arise. You can gather this data by using logs and SNMP tools to monitor the system performance or by regularly running information gathering commands and saving the output.

You should back up your FortiOS configuration on a regular basis even when you are not troubleshooting. You can restore the backed up configuration as needed to save time recreating it from the factory default settings.

Use the following CLI commands to obtain normal operating data for a FortiGate:

`get system status`	Displays firmware versions and FortiGuard engine versions, and other system information.
`get system performance status`	Displays CPU and memory states, average network usage, average sessions and session setup rate, viruses caught, IPS attacks blocked, and uptime.
`get hardware memory`	Displays information about memory.

`get system session status`	Displays total number of sessions.
`get router info routing-table all`	Displays all the routes in the routing table, including their type, source, and other useful data.
`get ips session`	Displays memory used and maximum amount available to IPS as well as counts
`get webfilter ftgd-statistics`	Displays a list of FortiGuard related counts of status, errors, and other data.
`diagnose sys session list`	Displays the list of current detailed sessions.
`show sys dns`	Displays the configured DNS servers.
`diagnose sys ntp status`	Displays information about NTP servers.

You can run any commands that apply to your system for information gathering. For example, if you have active VPN connections, use the `get vpn` series of commands to get more information about them.

Use `execute tac report` to get an extensive snapshot of your system. This command runs many diagnostic commands for specific configurations. It also records the current state of each feature regardless of the features deployed on your FortiGate. If you need to troubleshoot later, you can run the same command again and compare the differences to identify any suspicious output.

Define the problem

The following questions are intended to compare the current behavior of the FortiGate with normal operations to help you define the problem. Be specific with your answers. After you define the problem, search for a solution in the troubleshooting scenarios section, and then create a plan to resolve it.

What is the problem?	The problem being observed may not be the actual problem. You should determine where the problem lies before starting to troubleshoot the FortiGate.
Was the device working before?	If the device never worked, it might be defective. For more information, see Troubleshooting your installation on page 74.
Can the problem be reproduced?	If the problem is intermittent, it may be dependent on system load. Intermittent problems are challenging to troubleshoot because they are difficult to reproduce.
What has changed?	Use the FortiGate event log to identify possible configuration changes. There may be changes in the operating environment. For example, there might be a gradual increase in load as more sites are forwarded through the firewall. If something has changed, roll back the change and assess the impact.
What is the scope of the problem?	After you isolate the problem, determine what applications, users, devices, and operating systems the problem affects. The following questions are intended to narrow the scope of the problem and identify what to check during troubleshooting. The more factors you can eliminate, the less you need to check. For this reason, be as specific and accurate as possible when gathering information. • What is not working?

	• Is more than one thing not working? • Is it partly working? If so, what parts are working? • Is it a connectivity issue for the entire device, or is there an application that isn't reaching the Internet? • Where did the problem occur? • When did the problem occur and to which users or groups of users? • What components are involved? • What applications are affected? • Can you use a packet sniffer to trace the problem? • Can you use system debugging or look in the session table to trace the problem? • Do any of the log files indicate a failure has occurred?

Create a troubleshooting plan

After you define the problem and its scope, develop a troubleshooting plan.

Create checklist	Make a list all the possible causes of the problem and how you can test for each cause. Create a checklist to keep track of what has been tried and what is left to test. Checklists are useful when more than one person is performing troubleshooting tasks.
Obtain the required equipment	Testing your solution may require additional networking equipment, computers, or other devices. Network administrators usually have additional networking equipment available to loan you, or a lab where you can bring the FortiGate unit to test. If you do not have access to equipment, check for shareware applications that can perform the same tasks. Often, there are software solutions you can use when hardware is too expensive.
Consult Fortinet troubleshooting resources	After the checklist is created, refer to the troubleshooting scenarios sections to assist with implementing your plan. See Troubleshooting scenarios on page 2638.
Gather information for technical support	If you still require technical assistance after the plan is implemented, be prepared to provide Fortinet technical support with following information: • Firmware build version (use the `get system status` command) • Network topology diagram • Recent configuration file • Recent debug log (optional) • Summary of troubleshooting steps you have taken and the results. Do not provide the output from the `execute tac` report unless the support team requests it. The output from this command is very large and is not required in many cases.

Contact technical support	Before contacting technical support, ensure you have login access (preferably with full read/write privileges) to all networking devices that could be relevant to troubleshooting. If you are using VMs, be prepared to have someone who can log in to the virtual hosting platform in case it is necessary to check and possibly modify resource allocation. For information about contacting technical support, go to FortiCare Support Service page.

Troubleshooting scenarios

The following table is intended to help you diagnose common problems and provides links to the corresponding troubleshooting topics:

Problem	Probable cause	Recommended action
Hardware connections	• Are all of the cables and interfaces connected properly? • Is the LED for the interface green?	Checking the hardware connections on page 2640
FortiOS network settings	• If you are having problems connecting to the management interface, is your protocol enabled on the interface for administrative access? • Does the interface have an IP address?	Checking FortiOS network settings on page 2641
CPU and memory resources	• Is the CPU running at almost 100 percent usage? • Is your FortiGate running low on memory?	Troubleshooting CPU and network resources on page 2644
Modem status	• Is the modem connected? • Are there PPP issues?	Checking the modem status on page 2649
Ping and traceroute	Is the FortiGate experiencing complete packet loss?	Running ping and traceroute on page 2650
Logs	Do you need to identify a problem?	Checking the logs on page 2653
Contents of the routing table (in NAT mode)	• Are there routes in the routing table for default and static routes? • Do all connected subnets have a route in the routing table? • Does a route have a higher priority than it should?	Verifying routing table contents in NAT mode on page 2654
Traffic routes	Is the traffic routed correctly?	Verifying the correct route is being used on page 2655

Problem	Probable cause	Recommended action
Firewall policies	Is the correct firewall policy applied to the expected traffic?	Verifying the correct firewall policy is being used on page 2655
Bridging information in transparent mode	Are you having problems in transparent mode?	Checking the bridging information in transparent mode on page 2656
Firewall session list	• Are there active firewall sessions?	Using a session table on page 2666
Wireless Network	Is the wireless network working properly?	Checking wireless information on page 2657
FortiGuard connectivity	Is the FortiGate communicating properly with FortiGuard?	Verifying connectivity to FortiGuard on page 2678
Sniffer trace	• Is traffic entering the FortiGate? Does the traffic arrive on the expected interface? • Is the ARP resolution correct for the next-hop destination? • Is the traffic exiting the FortiGate to the destination as expected? • Is the FortiGate sending traffic back to the originator?	Performing a sniffer trace or packet capture on page 2658
Packet flow	Is traffic entering or leaving the FortiGate as expected?	Debugging the packet flow on page 2659

Checking the system date and time

The system date and time are important for FortiGuard services, logging events, and sending alerts. The wrong time makes the log entries confusing and difficult to use.

When possible, use Network Time Protocol (NTP) to set the date and time. This is an automatic method that does not require manual intervention. However, you must ensure that the port is allowed through the firewalls on your network. FortiToken synchronization requires NTP in many situations.

For information about setting the system date and time, see Setting the system time on page 1976.

To view and configure the date and time in the GUI:

1. Go to *Dashboard > Status*. The date and time are displayed in the *System Information* widget, next to *System Time*.
2. Go to *System > Settings*.
3. In the *System Time* section, select *NTP*, and then configure the *Time Zone*, and *Set Time* settings as required.

To view the date and time in the CLI:

```
execute date
```

```
execute time
```

To configure the date and time in the CLI:

Use the `set timezone ?` command to display a list of timezones and the integers that represent them.

```
config system global
   set timezone <integer>
end
config system ntp
   set type custom
      config ntpserver
         edit 1
            set server "ntp1.fortinet.net"
         next
         edit 2
            set server "ntp2.fortinet.net"
         next
      end
   set ntpsync enable
   set syncinterval 60
end
```

Checking the hardware connections

If traffic is not flowing from the FortiGate, there may be a problem with the hardware connection.

To check hardware connections:

1. Ensure the network cables are plugged into the interfaces.
2. Verify the LED connection lights for the network cables indicate there is a connection. The lights are typically green when there is a connection.
3. Change the cable when:
   - The cable or its connector are damaged.
   - You are unsure of the type or quality of the cable, such as straight through or crossover.
   - You see exposed wires at the connector.
4. Connect the FortiGate to different hardware.
5. Go to *Network > Interfaces* to ensure the link status for the interface is set to *Up*.
   The link status is based on the physical connection and cannot be set in FortiOS.

To enable an interface in the GUI:

You should still perform basic software connectivity tests to ensure complete connectivity even if there was a problem with the hardware connection. The interface might also be disabled, or its *Status* might be set to *Down*. See Interfaces on page 135.

1. Go to *Network > Interfaces*.
2. Select an interface, such as *Port1*, and click *Edit*.
3. In the *Miscellaneous* area, next to *Status*, click *Enabled*.
4. Click *OK*.

To enable an interface in the CLI:

```
config system interface
```

```
   edit port1
      set status up
   next
end
```

Checking FortiOS network settings

Check the FortiOS network settings if you have problems connecting to the management interface. FortiOS network settings include, interface settings, DNS Settings, and DHCP settings.

Interface settings

If you can access the FortiGate with the management cable only, you can view the interface settings in the GUI.

To view the interface settings in the GUI:

1. Go to *Network > Interfaces*.
2. Select an interface and click *Edit*.
3. Check the following interfaces to ensure they are not blocking traffic.

Setting	Description
Link Status	The status is *Up* when a valid cable is plugged in. The status is *Down* when an invalid cable is plugged in. The Link Status is shown physically by the connection LED for the interface. If the LED is green, the connection is good. If Link Status is *Down*, the interface does not work. Link status also appears in the *Network > Interfaces* page by default.
Addressing mode	Do not use *DHCP* if you do not have a DHCP server. You will not be able to log into an interface in DHCP mode as it will not have an IP address.
IP/Network Mask	An interface requires an IP address to connect to other devices. Ensure there is a valid IP address in this field. The one exception is when *DHCP* is enabled for this interface to get its IP address from an external DHCP server.
IPv6 address	The same protocol must be used by both ends to complete the connection. Ensure this interface and the remote connection are both using IPv4 or both are using IPv6 addresses.
Administrative access	If no protocols are selected, you will have to use the local management cable to connect to the unit. If you are using IPv6, configure the IPv6 administrative access protocols.
Status	Ensure the status is set to *Up* or the interface will not work.

To display the internal interface settings in the CLI:

```
FGT# show system interface <interface_name>
```

To view the list of possible interface settings:

```
config system interface
   edit <interface_name>
      get
end
```

DNS settings

To view DNS settings in the GUI:

Go to *Network > DNS*.

You can trace many networking problems back to DNS issues. Check the following items:

1. Are there values for both the *Primary DNS server* and *Secondary DNS server* fields.
2. Is the *Local Domain Name* correct?
3. Are you using IPv6 addressing? If so, are the IPv6 DNS settings correct?
4. Are you using Dynamic DNS (DDNS)? If so, is it using the correct server, credentials, and interface?
5. Can you contact both DNS servers to verify the servers are operational?
6. If an interface addressing mode is set to DHCP and is set to override the internal DNS, is that interface receiving a valid DNS entry from the DHCP server? Is it a reasonable address and can it be contacted to verify it is operational?
7. Are there any DENY security policies that need to allow DNS?
8. Can any internal device perform a successful traceroute to a location using the FQDN?

DHCP server settings

DHCP servers are common on internal and wireless networks. The DHCP server will cause problems if it is not configured correctly.

To view DHCP server settings in the GUI:

1. Go to *Network > Interfaces*.
2. Select an interface, and click *Edit*.

Check the following items:

1. Is the DHCP server enabled?
2. Is the DHCP server entry set to *Relay*? If so, verify there is another DHCP server to which requests can be relayed. Otherwise, set it to *Server*.
3. Does the DHCP server use a valid IP address range? Are other devices using the addresses? If one or more devices are using IP addresses in this range, you can use the IP reservation feature to ensure the DHCP server does not use these addresses. See DHCP server on page 287
4. Is there a gateway entry? If not, add a gateway entry to ensure that the server's clients have a default route.
5. Is the system DNS setting being used? A best practice is to avoid confusion by using the system DNS whenever possible. However, you can specify up to three custom DNS servers, and you should use all three entries for redundancy.

There are some situations, such as a new wireless interface, or during the initial FortiGate configuration, where interfaces override the system DNS entries. When this happens, it often shows up as intermittent Internet connectivity.

To fix the problem, go to *Network > DNS*, and enable *Use FortiGuard Servers*.

Checking CPU and memory resources

Check the CPU and memory resources when the FortiGate is not working, the network is slow, or there is a reduced firewall session setup rate. All processes share the system resources in FortiOS, including CPU and memory.

To view system resources in the GUI:

Go to *Dashboard > Status*.

The resource information is located in the *CPU* and *Memory* widgets. For information, see Dashboards and Monitors on page 76.

To view system resources in the CLI:

```
get system performance status
```

Sample output:

```
FGT# get system performance status
CPU states: 0% user 0% system 0% nice 100% idle 0% iowait 0% irq 0% softirq
CPU0 states: 0% user 0% system 0% nice 100% idle 0% iowait 0% irq 0% softirq
CPU1 states: 0% user 0% system 0% nice 100% idle 0% iowait 0% irq 0% softirq
CPU2 states: 0% user 0% system 0% nice 100% idle 0% iowait 0% irq 0% softirq
CPU3 states: 0% user 0% system 0% nice 100% idle 0% iowait 0% irq 0% softirq
Memory: 4050332k total, 527148k used (13%), 3381312k free (83%), 141872k freeable (3%)
Average network usage: 41 / 28 kbps in 1 minute, 54 / 44 kbps in 10 minutes, 42 / 34 kbps
     in 30 minutes
Average sessions: 33 sessions in 1 minute, 48 sessions in 10 minutes, 38 sessions in 30
     minutes
Average session setup rate: 0 sessions per second in last 1 minute, 0 sessions per second
     in last 10 minutes, 0 sessions per second in last 30 minutes
Virus caught: 0 total in 1 minute
IPS attacks blocked: 0 total in 1 minute
Uptime: 0 days, 22 hours, 59 minutes
```

The first line of the output shows the CPU usage by category:

```
CPU states: 0% user 0% system 0% nice 100% idle 0% iowait 0% irq 0% softirq
```

The second line of the output shows the memory usage:

```
Memory: 4050332k total, 527148k used (13%), 3381312k free (83%), 141872k freeable (3%)
```

Memory usage should not exceed 90%. Using too much memory prevents some processes from functioning properly. For example, if the system is running low on memory, antivirus scanning enters into *failopen* mode where it drops connections or bypasses the antivirus system.

Other lines of output, such as `average network usage`, `average session setup rate`, `viruses caught`, and `IPS attacks blocked`, help determine why system resource usage is high.

For example:

- A high `average network usage` may indicate high traffic processing on the FortiGate,
- A very low or zero, `average session setup rate` may indicate the proxy is overloaded and unable to do its job.

Troubleshooting CPU and network resources

FortiGate has stopped working

If the FortiGate has stopped working, the first line of the output will look similar to this:

```
CPU states: 0% user 0% system 0% nice 100% idle
```

Network is slow

If your network is running slow, the first line of the output will look similar to this:

```
CPU states: 1% user 98% system 0% nice 1% idle
```

This example shows that all of the CPU is being used by system processes, and the FortiGate is overloaded. When overloading occurs, it is possible a process such as `scanunitid` is using all the resources to scan traffic. In this case you need to reduce the amount of traffic being scanned by blocking unwanted protocols, configuring more security policies to limit scanning to certain protocols, or similar actions.

It is also possible a hacker has accessed your network and is overloading it with malicious activity, such as running a spam server or using zombie PCs to attack other networks on the Internet.

You can use the following command to investigate the problem with the CPU:

```
get system performance top
```

This command shows all of the top processes that are running on the FortiGate and their CPU usage. The process names are on the left. If a process is using most of the CPU cycles, investigate it to determine whether the activity is normal.

Reduced firewall session setup rate

A reduced firewall session setup rate can be caused by a lack of system resources on the FortiGate, or reaching the session count limit for a VDOM.

As a best practice, administrators should record the session setup rate during normal operation to establish a baseline to help define a problem when your are troubleshooting.

The session setup rate appears in the `average sessions` section of the output.

A reduced firewall session setup rate will look similar to this:

```
Average sessions: 80 sessions in 1 minute, 30 sessions in 10 minutes, 42 sessions in 30
      minutes
Average session setup rate: 3 sessions per second in last 1 minute, 0 sessions per second in
      last 10 minutes, 0 sessions per second in last 30 minutes
```

In the example above, there were 80 sessions in 1 minute, or an average of 3 sessions per second.

The values for `10 minutes` and `30 minutes` allow you to take a longer average for a more reliable value if your FortiGate is working at maximum capacity. The smallest FortiGate can have 1,000 sessions established per second across the unit.

The session setup rate is a global command. If you have multiple VDOMs configured with many sessions in each VDOM, the session setup rate per VDOM will be slower than if there are no VDOMs configured.

High memory usage

As with any system, a FortiGate has limited hardware resources, such as memory, and all processes running on the FortiGate share the memory. Each process uses more or less memory, depending on its workload. For example, a process usually uses more memory in high traffic situations. If some processes use all of the available memory, other processes will not be able to run.

When high memory usage occurs, the services may freeze up, connections may be lost, or new connections may be refused.

If you see high memory usage in the *Memory* widget, the FotiGate may be handling high traffic volumes. Alternatively, the FortiGate may have problems with connection pool limits that are affecting a single proxy. If the FortiGate receives large volumes of traffic on a specific proxy, the unit may exceed the connection pool limit. If the number of free connections within a proxy connection pool reaches zero, issues may occur.

To view current memory usage information in the CLI:

```
diagnose hardware sysinfo memory
```

Sample output:

```
total: used: free: shared: buffers: cached: shm:
Mem: 2074185728 756936704 1317249024 0 20701184 194555904 161046528
Swap:       0       0       0
MemTotal:   2025572 kB
MemFree:    1286376 kB
MemShared:        0 kB
Buffers:      20216 kB
Cached:      189996 kB
SwapCached:       0 kB
Active:       56644 kB
Inactive:    153648 kB
HighTotal:        0 kB
HighFree:         0 kB
LowTotal:   2025572 kB
LowFree:    1286376 kB
SwapTotal:        0 kB
SwapFree:         0 kB
```

Troubleshooting high CPU usage

Connection-related problems may occur when FortiGate's CPU resources are over extended. This occurs when you deploy too many FortiOS features at the same time.

Examples of CPU intensive features:

- VPN high-level encryption
- Intensive scanning of all traffic
- Logging all traffic and packets
- Dashboard widgets that frequently perform data updates

For information on customizing the CPU use threshold, see Execute a CLI script based on CPU and memory thresholds on page 2410.

Determining the current level of CPU usage

You can view CPU usage levels in the GUI or CLI. For precise usage values for both overall usage and specific processes, use the CLI.

To view CPU usage in the GUI:

Go to *Dashboard > Status*. Real-time CPU usage information is located in the *CPU* widget.

To view CPU usage in the CLI:

- Show top processes information:
  ```
  diagnose sys top
  ```
- Show top threads information:
  ```
  diagnose sys top-all
  ```

Sample output:

```
Run Time: 86 days, 0 hours and 10 minutes
0U, 0N, 0S, 100I, 0WA, 0HI, 0SI, 0ST; 3040T, 2437F
     bcm.user 93 S < 3.1 0.4
     httpsd 18922 S 1.5 0.5
     httpsd 19150 S 0.3 0.5
     newcli 20195 R 0.1 0.1
     cmdbsvr 115 S 0.0 0.8
     pyfcgid 20107 S 0.0 0.6
     forticron 146 S 0.0 0.5
     httpsd 139 S 0.0 0.5
     cw_acd 166 S 0.0 0.5
     miglogd 136 S 0.0 0.5
     pyfcgid 20110 S 0.0 0.4
     pyfcgid 20111 S 0.0 0.4
     pyfcgid 20109 S 0.0 0.4
     httpsd 20192 S 0.0 0.4
     miglogd 174 S 0.0 0.4
     miglogd 175 S 0.0 0.4
     fgfmd 165 S 0.0 0.3
     newcli 20191 S 0.0 0.3
     initXXXXXXXXXXX 1 S 0.0 0.3
     httpsd 184 s 0.0 0.3
```

The following table explains the codes in the second line of the output:

Code	Description
`U`	Percentage of user space applications that are currently using the CPU
`N`	Percentage of time that the CPU spent on low priority processes since the last shutdown
`S`	Percentage of system processes (or kernel processes) that are using the CPU
`I`	Percentage of idle CPU resources
`WA`	Percentage of time that the CPU spent waiting on IO peripherals since the last shutdown
`HI`	Percentage of time that the CPU spent handling hardware interrupt routines since the last shutdown
`SI`	Percentage of time that the CPU spent handling software interrupt routines since the last shutdown
`ST`	Steal time: Percentage of time a virtual CPU waits for the physical CPU when the hypervisor is servicing another virtual processor
`T`	Total FortiOS system memory in MB
`F`	Free memory in MB

Each additional line of the command output displays information specific to processes or threads that are running on the FortiGate unit. For example, the sixth line of the output is: `newcli 20195 R 0.1 0.1`

The following table describes the data in the sixth line of the output:

Item	Description
`newcli`	The process (or thread) name. Duplicate process or thread names indicate that separate instances of that process or thread are running.
`20195`	The process or thread ID, which can be any number.
`R`	Current state of the process or thread. The process or thread state can be: • `R` - running • `S` - sleep • `Z` - zombie • `D`- disk sleep
`0.1`	The percentage of CPU capacity that the process or thread is using. CPU usage can range from 0.0 for a process or thread that is sleeping to higher values for a process or thread that's taking a lot of CPU time.
`0.1`	The amount of memory that the process or thread is using. Memory usage can range from 0.1 to 5.5 and higher.

You can use the following single-key commands when running `diagnose sys top` or `diagnose sys top-all`:

- `q` to quit and return to the normal CLI prompt.
- `p` to sort the processes by the amount of CPU that the processes are using.
- `m` to sort the processes by the amount of memory that the processes are using.

The output only displays the top processes or threads that are running. For example, if 20 are listed, they are the top 20 currently running, sorted by either CPU or memory usage. You can configure the number of processes or threads displayed, using the following CLI commands:

```
diagnose sys top <integer_seconds> <integer_maximum_lines>
diagnose sys top-all <integer_seconds> <integer_maximum_lines>
```

Where:

- `<integer_seconds>` is the delay in seconds (default is 5)
- `<integer_maximum_lines>` is the maximum number of lines (or processes) to list (default is 20)

Determining which features are using the most CPU resources

You can use the CLI to view the top few processes that are currently running and using the most CPU resources.

To view processes using the most CPU resources:

```
get system performance top
```

The entries at the top are using the most CPU resources. The second column from the right shows CPU usage by percentage. Note which processes are using the most resources and try to reduce their CPU load.

Processes you will see include:

- `ipsengine`: the IPS engine that scans traffic for intrusions
- `scanunitd`: antivirus scanner
- `httpsd`: secure HTTP
- `iked`: internet key exchange (IKE) in use with IPsec VPN tunnels
- `newcli`: active whenever you're accessing the CLI
- `sshd`: there are active secure socket connections
- `cmdbsrv`: the command database server application

Go to the features that are at the top of the list and look for evidence of CPU overuse. Generally, the monitor for a feature is a good place to start.

Checking for unnecessary CPU “wasters”

These are some best practices that will reduce your CPU usage, even if the FortiGate is not experiencing high CPU usage. Note that if the following information instructs you to turn off a feature that you require, disregard that part of the instructions.

- Use hardware acceleration wherever possible to offload tasks from the CPU. Offloading tasks, such as encryption, frees up the CPU for other tasks.
- Avoid the use of GUI widgets that require computing cycles, such as the *Top Sessions* widget. These widgets constantly pol the system for information, which uses CPU and other resources.
- Schedule antivirus, IPS, and firmware updates during off-peak hours. These updates do not usually consume CPU resources but they can disrupt normal operation.
- Check the log levels and which events are being logged. This is the severity of the messages that are recorded. Consider going up one level to reduce the amount of logging. Also, if there are events you do not need to monitor, remove them from the list.

- Log to FortiCloud instead of logging to memory or disk. Logging to memory quickly uses up resources and logging to local disk impacts overall performance and reduces the lifetime of the unit.
  Fortinet recommends logging to FortiCloud to avoid using too much CPU.
- If the disk is almost full, transfer the logs or data off the disk to free up space. When a disk is almost full it consumes a lot of resources to find free space and organize the files.
- If packet logging is enabled on the FortiGate, consider disabling it. When packet logging is enabled, it records every packet that comes through that policy.
- Halt all sniffers and traces.
- Ensure the FortiGate isn't scanning traffic twice. Traffic does not need to be rescanned if it enters the FortiGate on one interface, goes out another, and then comes back in again. Doing so is a waste of resources. However, ensure that traffic truly is being scanned once.
- Reduce the session timers to close unused sessions faster. Enter the following CLI commands, which reduce the default values. Note that, by default, the system adds 10 seconds to `tcp-timewait`.
  ```
  config system global
     set tcp-halfclose-timer 30
     set tcp-halfopen-timer 30
     set tcp-timewait-timer 0
     set udp-idle-timer 60
  end
  ```
- Go to *System* > *Feature Visibility*, and enable only features that you need.

SNMP monitoring

When CPU usage is under control, use SNMP to monitor CPU usage. Alternatively, use logging to record CPU and memory usage every 5 minutes.

Once the system is back to normal, you should set up a warning system that sends alerts when CPU resources are used excessively. A common method to do this is using SNMP. SNMP monitors many values in FortiOS and allows you to set high water marks that generate events. You can run an application on your computer to watch for and record these events.

To enable SNMP:

1. Go to *System* > *SNMP*.
2. Configure an SNMP community.

See SNMP on page 2124.

You can use the *System Resources* widget to record CPU usage if SNMP is too complicated. However, the widget only records problems as they happen and will not send you alerts for problems.

Checking the modem status

You can use the CLI to troubleshoot a modem that is not working properly, or troubleshoot a FortiGate that does not detect the modem.

To diagnose modem issues in the CLI:

```
diagnose sys modem {cmd | com | detect | history | external-modem | query| reset}
```

You should always run the following command after you connect a USB modem to FortiGate:

```
diagnose sys modem detect
```

Use the following command to view the modem's configuration, vendor and custom product identification number:

```
get system modem
```

Use the following commands to resolve connectivity issues:

- `diagnose debug enable`: Activates the debug on the console
- `diagnose debug application modemd`: Dumps communication between the modem and the unit.
- `diagnose debug application ppp`: Dumps the PPP negotiating messages.
- `execute modem dial`: Displays modem debug output.

The modem diagnose output should not contain errors when initializing. You should also verify the number used to dial into your ISP.

Running ping and traceroute

Ping and traceroute are useful tools in network troubleshooting. Alone, either tool can determine network connectivity between two points. However, ping can be used to generate simple network traffic that you can view using `diagnose` commands in FortiGate. This combination can be very powerful when you are trying to locate network problems.

Ping and traceroute can also tell you if your computer or network device has access to a domain name server (DNS). Both tools can use IP addresses or device domain names to determine why particular services, such as email or web browsing, may not work properly.

If ping does not work, it may be disabled on at least one of the interface settings and security policies for that interface.

Both ping and traceroute require particular ports to be open on firewalls to function. Since you typically use these tools to troubleshoot, you can allow them in the security policies and on interfaces only when you need them. Otherwise, keep the ports disabled for added security.

Ping

The ping command sends a very small packet to a destination, and waits for a response. The response has a timer that expires when the destination is unreachable.

Ping is part of layer 3 on the OSI Networking Model. Ping sends Internet Control Message Protocol (ICMP) “echo request” packets to the destination, and listens for “echo response” packets in reply. However, many public networks block ICMP packets because ping can be used in a denial of service (DoS) attack (such as Ping of Death or a smurf attack), or by an attacker to find active locations on the network. By default, FortiGate units have ping enabled while broadcast-forward is disabled on the external interface.

What ping can tell you

Beyond the basic connectivity information, ping can tell you the amount of packet loss (if any), how long it takes the packet to make the round trip, and the variation in that time from packet to packet.

If packet loss is detected, you should investigate the following:

- Possible ECMP, split horizon, or network loops.
- Cabling, to ensure there are no loose connections.
- Verify which security policy was used. To do this:
  Go *to Policy & Objects* > *Firewall Policy* and view the packet count column.

If there is total packet loss, you should investigate the following:

1. Ensure cabling is correct, and all equipment between the two locations is accounted for.
2. Ensure all IP addresses and routing information along the route is configured as expected.
3. Ensure all firewalls, including FortiGate security policies allow PING to pass through.

How to use ping

Ping syntax is the same for nearly every type of system on a network.

To ping from a FortiGate unit:

1. Go to *Dashboad*, and connect to the CLI through either telnet or the CLI widget.
2. Enter `execute ping 10.11.101.101` to send 5 ping packets to the destination IP address. There are no options for this command.
```
Head_Office_620b # execute ping 10.11.101.101
PING 10.11.101.101 (10.11.101.101): 56 data bytes
64 bytes from 10.11.101.101: icmp_seq=0 ttl=255 time=0.3 ms
64 bytes from 10.11.101.101: icmp_seq=1 ttl=255 time=0.2 ms
64 bytes from 10.11.101.101: icmp_seq=2 ttl=255 time=0.2 ms
64 bytes from 10.11.101.101: icmp_seq=3 ttl=255 time=0.2 ms
64 bytes from 10.11.101.101: icmp_seq=4 ttl=255 time=0.2 ms

--- 10.11.101.101 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 0.2/0.2/0.3 ms
```

To ping from a Microsoft Windows PC:

1. Open a command window.
2. Enter `ping 10.11.101.100` to ping the default internal interface of the FortiGate with four packets.
   Other options include:
   - `-t` to send packets until you press `Ctrl+C`
   - `-a` to resolve addresses to domain names where possible
   - `-n X` to send X ping packets and stop
```
C:\>ping 10.11.101.101

Pinging 10.11.101.101 with 32 bytes of data:
Reply from 10.11.101.101: bytes=32 time=10ms TTL=255
Reply from 10.11.101.101: bytes=32 time<1ms TTL=255
Reply from 10.11.101.101: bytes=32 time=1ms TTL=255
Reply from 10.11.101.101: bytes=32 time=1ms TTL=255

Ping statistics for 10.11.101.101:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 10ms, Average = 3ms
```

To ping from a Linux PC:

1. Go to a shell prompt.
2. Enter "`ping 10.11.101.101`".

Traceroute

Where ping will only tell you if it reached its destination and returned successfully, traceroute shows each step of the journey to its destination and how long each step takes. If ping finds an outage between two points, you can use traceroute to locate exactly where the problem is.

Traceroute works by sending ICMP packets to test each hop along the route. It sends three packets, and then increases the time to live (TTL) setting by one each time. This effectively allows the packets to go one hop farther along the route. This is why most traceroute commands display their maximum hop count before they start tracing the route, which is the maximum number of steps it takes before it declares the destination unreachable. Also, the TTL setting may result in steps along the route timing out due to slow responses. There are many possible reasons for this to occur.

By default, traceroute uses UDP datagrams with destination ports numbered from 33434 to 33534. The traceroute utility may also offer the option to select use of ICMP echo request (type 8) instead, which the Windows tracert utility uses. If you must, allow both protocols inbound through the FortiGate security policies (UDP with ports from 33434 to 33534 and ICMP type 8).

To track traceroute packets in the GUI:

Go to *Policy & Objects* > *Firewall Policy* and view the packet count column.

This allows you to verify the connection and confirm which security policy the traceroute packets are using.

What traceroute can tell you

Both ping and traceroute verify connectivity between two points. However, only traceroute shows you each step in the connection path. Also, ping and traceroute use different protocols and ports, so one may succeed where the other fails.

You can verify your DNS connection using traceroute. If you enter an FQDN instead of an IP address for the traceroute, DNS tries to resolve that domain name. If the name isn't resolved, you have DNS issues.

Using traceroute

The traceroute command varies slightly between operating systems. In Microsoft Windows, the command name is shortened to "`tracert`". Also, your output lists different domain names and IP addresses along your route.

To use traceroute on a Microsoft Windows PC:

1. Open a command window.
2. Enter `tracert fortinet.com` to trace the route from the PC to the Fortinet web site.
```
C:\>tracert fortinet.com
Tracing route to fortinet.com [208.70.202.225]
over a maximum of 30 hops:
1 <1 ms <1 ms <1 ms 172.20.120.2
2 66 ms 24 ms 31 ms 209-87-254-xxx.storm.ca [209.87.254.221]
3 52 ms 22 ms 18 ms core-2-g0-0-1104.storm.ca [209.87.239.129]
4 43 ms 36 ms 27 ms core-3-g0-0-1185.storm.ca [209.87.239.222]
5 46 ms 21 ms 16 ms te3-x.1156.mpd01.cogentco.com [38.104.158.69]
```

```
6 25 ms 45 ms 53 ms te8-7.mpd01.cogentco.com [154.54.27.249]
7 89 ms 70 ms 36 ms te3-x.mpd01.cogentco.com [154.54.6.206]
8 55 ms 77 ms 58 ms sl-st30-chi-.sprintlink.net [144.232.9.69]
9 53 ms 58 ms 46 ms sl-0-3-3-x.sprintlink.net [144.232.19.181]
10 82 ms 90 ms 75 ms sl-x-12-0-1.sprintlink.net [144.232.20.61]
11 122 ms 123 ms 132 ms sl-0-x-0-3.sprintlink.net [144.232.18.150]
12 129 ms 119 ms 139 ms 144.232.20.7
13 172 ms 164 ms 243 ms sl-321313-0.sprintlink.net [144.223.243.58]
14 99 ms 94 ms 93 ms 203.78.181.18
15 108 ms 102 ms 89 ms 203.78.176.2
16 98 ms 95 ms 97 ms 208.70.202.225
```

The first column on the left is the hop count, which cannot exceed 30 hops. When that number is reached, the traceroute ends.

The second, third, and fourth columns display how much time each of the three packets takes to reach this stage of the route. These values are in milliseconds and normally vary quite a bit. Typically a value of `<1ms` indicates a local connection.

The fifth column (farthest to the right) shows the domain name of the device and its IP address, or possibly only the IP address.

To perform a traceroute on a Linux PC:

1. Go to a command line prompt.
2. Enter "`traceroute fortinet.com`".
   The Linux traceroute output is very similar to the Windows `tracert` output.

To trace a route from a FortiGate to a destination IP address in the CLI:

```
# execute traceroute www.fortinet.com

traceroute to www.fortinet.com (66.171.121.34), 32 hops max, 84 byte packets
1 172.20.120.2 0.637 ms 0.653 ms 0.279 ms
2 209.87.254.221 <static-209-87-254-221.storm.ca> 2.448 ms 2.519 ms 2.458 ms
3 209.87.239.129 <core-2-g0-2.storm.ca> 2.917 ms 2.828 ms 9.324 ms
4 209.87.239.199 <core-3-bdi1739.storm.ca> 13.248 ms 12.401 ms 13.009 ms
5 216.66.41.113 <v502.core1.tor1.he.net> 17.181 ms 12.422 ms 12.268 ms
6 184.105.80.9 <100ge1-2.core1.nyc4.he.net> 21.355 ms 21.518 ms 21.597 ms
7 198.32.118.41 <ny-paix-gni.twgate.net> 83.297 ms 84.416 ms 83.782 ms
8 203.160.228.217 <217-228-160-203.TWGATE-IP.twgate.net> 82.579 ms 82.187 ms 82.066 ms
9 203.160.228.229 <229-228-160-203.TWGATE-IP.twgate.net> 82.055 ms 82.455 ms 81.808 ms
10 203.78.181.2 82.262 ms 81.572 ms 82.015 ms
11 203.78.186.70 83.283 ms 83.243 ms 83.293 ms
12 66.171.127.177 84.030 ms 84.229 ms 83.550 ms
13 66.171.121.34 <www.fortinet.com> 84.023 ms 83.903 ms 84.032 ms
14 66.171.121.34 <www.fortinet.com> 83.874 ms 84.084 ms 83.810 ms
```

Checking the logs

A log message records the traffic passing through FortiGate to your network and the action FortiGate takes when it scans the traffic. You should log as much information as possible when you first configure FortiOS. If FortiGate logs are too large, you can turn off or scale back the logging for features that are not in use.

It is difficult to troubleshoot logs without a baseline. Before you can determine if the logs indicate a problem, you need to know what logs result from normal operation.

When troubleshooting with log files

- Compare current logs to a recorded baseline of normal operation.
- If you need to, increase the level of logging (such as from Warning to Information) to obtain more information. When increasing logging levels, ensure that you configure email alerts and select both disk usage and log quota. This ensures that you will be notified if the increase in logging causes problems.

To configure the log settings in the GUI:

Go to *Log & Report* > *Log Settings*.

Determine the activities that generate the most log entries:

- Check all logs to ensure important information is not overlooked.
- Filter or order log entries based on different fields, such as level, service, or IP address, to look for patterns that may indicate a specific problem, such as frequent blocked connections on a specific port for all IP addresses.

Logs can help identify and locate any problems, but they do not solve them. The purpose of logs is to speed up your problem solving and save you time and effort.

For more information about logging and log reports, see Log and Report on page 2539.

Verifying routing table contents in NAT mode

Verify the contents of the routing table when a FortiGate has limited or no connectivity.

The routing table stores the routes currently in use for both static and dynamic protocols. Storing a route in the routing table saves time and resources performing a lookup. To ensure the most recently used routes remain in the table, old routes are bumped to make room for new ones. You cannot perform this task when FortiGate is in transparent mode.

If FortiGate is running in NAT mode, verify that all desired routes are in the routing table, including local subnets, default routes, specific static routes, and dynamic routing protocols.

To view the routing table in the CLI:

```
get router info routing-table all
```

Sample output:

```
FGT# get router info routing-table all
Codes:
   K - kernel, C - connected, S - static, R - RIP, B - BGP
   O - OSPF, IA - OSPF inter area
   N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
   E1 - OSPF external type 1, E2 - OSPF external type 2
   i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
   * - candidate default
S* 0.0.0.0/0 [10/0] via 172.20.120.2, wan1
C 10.31.101.0/24 is directly connected, internal
C 172.20.120.0/24 is directly connected, wan1
```

Verifying the correct route is being used

Run a trace route from a machine in the local area network (LAN) to ensure traffic is flowing as expected through the correct route when there is more than one default route.

In the following example output:

- The first hop contains the IP address `10.10.1.99`, which is the internal interface of the FortiGate.
- The second hop contains the IP address `172.20.120.2`, to which the wan1 interface of the FortiGate is connected.

This means the route through the wan1 interface is being used for this traffic.

```
C:\>tracert www.fortinet.com
Tracing route to www.fortinet.com [66.171.121.34]
over a maximum of 30 hops:
1 <1 ms <1 ms <1 ms 10.10.1.99
2 1 ms <1 ms <1 ms 172.20.120.2
3 3 ms 3 ms 3 ms static-209-87-254-221.storm.ca [209.87.254.221]
4 3 ms 3 ms 3 ms core-2-g0-2.storm.ca [209.87.239.129]
5 13 ms 13 ms 13 ms core-3-bdi1739.storm.ca [209.87.239.199]
6 12 ms 19 ms 11 ms v502.core1.tor1.he.net [216.66.41.113]
7 22 ms 22 ms 21 ms 100ge1-2.core1.nyc4.he.net [184.105.80.9]
8 84 ms 84 ms 84 ms ny-paix-gni.twgate.net [198.32.118.41]
9 82 ms 84 ms 82 ms 217-228-160-203.TWGATE-IP.twgate.net [203.160.22
8.217]
10 82 ms 81 ms 82 ms 229-228-160-203.TWGATE-IP.twgate.net [203.160.22
8.229]
11 82 ms 82 ms 82 ms 203.78.181.2
12 84 ms 83 ms 83 ms 203.78.186.70
13 84 ms * 85 ms 66.171.127.177
14 84 ms 84 ms 84 ms fortinet.com [66.171.121.34]
15 84 ms 84 ms 83 ms fortinet.com [66.171.121.34]
```

You can also see the route taken for each session by debugging the packet flow in the CLI. For more information, see Debugging the packet flow on page 2659.

Verifying the correct firewall policy is being used

If you have more than one firewall policy, you can check which policy is being used in the *Policy & Objects* module in the GUI.

To verify the firewall policy in the GUI:

1. Go to *Policy & Objects* > *Firewall Policy*.
2. Look in the *Count* column to see which policy is being used. The count must show traffic increasing.

Debugging the packet flow in the CLI shows the policy ID that's allowing the traffic. For information, see Debugging the packet flow on page 2659.

Checking the bridging information in transparent mode

Checking the bridging information is useful when you are experiencing connectivity problems. When FortiGate is set to transparent mode, it acts like a bridge and sends all incoming traffic out on the other interfaces. Each bridge is a link between interfaces.

When traffic is flowing between the interfaces, you can see the bridges listed in the CLI. If no bridges are listed, this is the likely cause of the connectivity issue. When investigating bridging information, check for the MAC address of the interface or device in question.

How to check the bridging information

To view the list of bridge instances in the CLI:

```
diagnose netlink brctl list
```

Sample output:

```
#diagnose netlink brctl list
list bridge information
1. root.b fdb: size=256 used=6 num=7 depth=2 simple=no
Total 1 bridges
```

How to display forwarding domain information

You can use forwarding domains, or collision domains, in routing to limit where packets are forwarded on the network. Layer 2 broadcasts are limited to the same group. By default, all interfaces are in group 0. For example, if the FortiGate has 12 interfaces, only two may be in the same forwarding domain, which limits packets that are broadcast to those two interfaces. This reduces traffic on the rest of the network.

Collision domains prevent the forwarding of ARP packets to all VLANs on an interface. Without collision domains, duplicate MAC addresses on VLANs may cause ARP packets to be duplicated. Duplicate ARP packets can cause some switches to reset. It's important to know what interfaces are part of which forwarding domains because this determines which interfaces can communicate with each other.

To manually configure forwarding domains in transparent mode in the CLI:

```
config system interface
   edit <interface_name>
      set forward-domain <integer>
   end
```

To display the forward domains information in the CLI:

```
diagnose netlink brctl domain <name> <id>
```

Where `<name>` is the name of the forwarding domain to display and `<id>` is the domain ID.

Sample output:

```
diagnose netlink brctl domain ione 101
show bridge root.b ione forward domain.
```

```
id=101 dev=trunk_1 6
```

To list the existing bridge MAC table in the CLI:

```
diagnose netlink brctl name host <name>
```

Sample output:

```
show bridge control interface root.b host.
fdb: size=256, used=6, num=7, depth=2, simple=no
Bridge root.b host table
```

port no	device	devname	mac addr	ttl	attributes
2	7	wan2	02:09:0f:78:69:00	0	Local Static
5	6	vlan_1	02:09:0f:78:69:01	0	Local Static
3	8	dmz	02:09:0f:78:69:01	0	Local Static
4	9	internal	02:09:0f:78:69:02	0	Local Static
3	8	dmz	00:80:c8:39:87:5a	194	
4	9	internal	02:09:0f:78:67:68	8	
1	3	wan1	00:09:0f:78:69:fe	0	Local Static

To list the existing bridge port list in the CLI:

```
diagnose netlink brctl name port <name>
```

Sample output:

```
show bridge root.b data port.
trunk_1 peer_dev=0
internal peer_dev=0
dmz peer_dev=0
wan2 peer_dev=0
wan1 peer_dev=0
```

Checking wireless information

Check wireless connections, stations, and interfaces when the problem is not caused by a physical interface.

Troubleshooting station connection issues

To check if a station entry is created on access control in the CLI:

```
FG600B3909600253 # diagnose wireless-controller wlac -d sta
* vf=0 wtp=70 rId=2 wlan=open ip=0.0.0.0 mac=00:09:0f:db:c4:03 rssi=0 idle=148 bw=0 use=2
vf=0 wtp=70 rId=2 wlan=open ip=172.30.32.122 mac=00:25:9c:e0:47:88 rssi=-40 idle=0 bw=9
     use=2
```

Enabling diagnostics for a specific station

This example uses the station MAC address to find where it is failing:

```
FG600B3909600253 # diagnose wireless-controller wlac sta_filter 00:25:9c:e0:47:88 1
Set filter sta 00:25:9c:e0:47:88 level 1
FG600B3909600253 # 71419.245 <ih> IEEE 802.11 mgmt::disassoc <== 00:25:9c:e0:47:88 vap open
      rId 1 wId 0 00:09:0f:db:c4:03
71419.246 <dc> STA del 00:25:9c:e0:47:88 vap open rId 1 wId 0
71419.246 <cc> STA_CFG_REQ(34) sta 00:25:9c:e0:47:88 del ==> ws (0-192.168.35.1:5246) rId 1
      wId 0
71419.246 <cc> STA del 00:25:9c:e0:47:88 vap open ws (0-192.168.35.1:5246) rId 1 wId 0
      00:09:0f:db:c4:03 sec open reason I2C_STA_DEL
71419.247 <cc> STA_CFG_RESP(34) 00:25:9c:e0:47:88 <== ws (0-192.168.35.1:5246) rc 0
      (Success).
```

Performing a sniffer trace or packet capture

When you troubleshoot networks and routing in particular, it helps to look inside the headers of packets to determine if they are traveling the route that you expect them to take. Packet sniffing is also known as network tap, packet capture, or logic analyzing.

For FortiGates with NP2, NP4, or NP6 interfaces that are offloading traffic, disable offloading on these interfaces before you perform a trace or it will change the sniffer trace.

Sniffing packets

To perform a sniffer trace in the CLI:

Before you start sniffing packets, you should prepare to capture the output to a file. A large amount of data may scroll by and you will not be able to see it without saving it first. One method is to use a terminal program like puTTY to connect to the FortiGate CLI. Once the packet sniffing count is reached, you can end the session and analyze the output in the file.

The general form of the internal FortiOS packet sniffer command is:

```
# diagnose sniffer packet <interface_name> <'filter'> <verbose> <count> <tsformat>
```

To stop the sniffer, type `CTRL+C`.

<interface_name>	The name of the interface to sniff, such as `port1` or `internal`. This can also be `any` to sniff all interfaces.
<'filter'>	What to look for in the information the sniffer reads. `none` indicates no filtering, and all packets are displayed as the other arguments indicate. The filter must be inside single quotes (‘).
<verbose>	The level of verbosity as one of: • **1 -** print header of packets • **2 -** print header and data from IP of packets • **3 -** print header and data from Ethernet of packets • **4 -** print header of packets with interface name
<count>	The number of packets the sniffer reads before stopping. If you don't put a number here, the sniffer will run until you stop it with <`CTRL+C`>.

<tsformat>	The timestamp format. • `a`: absolute UTC time, yyyy-mm-dd hh:mm:ss.ms • `l`: absolute LOCAL time, yyyy-mm-dd hh:mm:ss.ms • `otherwise`: relative to the start of sniffing, ss.ms

Simple sniffing example:

```
# diagnose sniffer packet port1 none 1 3.
```

This displays the next three packets on the port1 interface using no filtering, and verbose level 1. At this verbosity level, you can see the source IP and port, the destination IP and port, action (such as ack), and sequence numbers.

In the output below, port 443 indicates these are HTTPS packets and that 172.20.120.17 is both sending and receiving traffic.

```
Head_Office_620b # diagnose sniffer packet port1 none 1 3
interfaces=[port1]
filters=[none]
0.545306 172.20.120.17.52989 -> 172.20.120.141.443: psh 3177924955 ack 1854307757
0.545963 172.20.120.141.443 -> 172.20.120.17.52989: psh 1854307757 ack 3177925808
0.562409 172.20.120.17.52988 -> 172.20.120.141.443: psh 4225311614 ack 3314279933
```

Using packet capture

To use packet capture, the FortiGate must have a disk. You can enable `capture-packet` in the firewall policy.

To enable packet capture in the CLI:

```
config firewall policy
    edit <id>
        set capture-packet enable
    next
end
```

For information about using the packet capture tool in the GUI, see Using the packet capture tool on page 474.

Debugging the packet flow

Debug the packet flow when network traffic is not entering and leaving the FortiGate as expected. When debugging the packet flow in the CLI, each command configures a part of the debug action. The final command starts the debug.

For information about using the debug flow tool in the GUI, see Using the debug flow tool on page 478.

To trace the packet flow in the CLI:

```
# diagnose debug flow trace start
```

To follow packet flow by setting a flow filter:

```
# diagnose debug flow {filter | filter6} <option>
```

- Enter `filter` if your network uses IPv4.
- Enter `filter6` if your network uses IPv6.

Replace `<option>` with one of the following variables:

Variable	Description
`addr`	IPv4 or IPv6 address
`clear`	clear filter
`daddr`	destination IPv4 or IPv6 address
`dport`	destination port
`negate`	inverse IPv4 or IPv6 filter
`port`	port
`proto`	protocol number
`saddr`	source address
`sport`	source port
`vd`	index of virtual domain; `-1` matches all

If FortiGate is connected to FortiAnalyzer or FortiCloud, the diagnose debug flow output will be recorded as event log messages and then sent to the devices. Do not run this command longer than necessary, as it generates a significant amount of data.

FortiASIC NP4 or NP6 interface pairs that offload traffic will change the packet flow. Before debugging any NP4 or NP6 interfaces, disable offloading on those interfaces.

To do this, enter `diagnose npu <interface pair> fastpath disable`, where `interface pair` is `np4`, `np6`, `np4lite`, or `np6lite`.

To start flow monitoring with a specific number of packets:

```
# diagnose debug flow trace start <N>
```

To stop flow tracing at any time:

```
# diagnose debug flow trace stop
```

The following example shows the flow trace for a device with an IP address of `203.160.224.97`:

```
# diagnose debug enable
# diagnose debug flow filter addr 203.160.224.97
# diagnose debug flow show function-name enable
# diagnose debug flow trace start 100
```

Sample output: HTTP

To observe the debug flow trace, connect to the website at the following address:

`https://www.fortinet.com`

Comment: SYN packet received:

```
id=20085 trace_id=209 func=resolve_ip_tuple_fast
line=2700 msg="vd-root received a packet(proto=6,
192.168.3.221:1487->203.160.224.97:80) from port5."
```

SYN sent and a new session is allocated:

```
id=20085 trace_id=209 func=resolve_ip_tuple line=2799
msg="allocate a new session-00000e90"
```

Lookup for next-hop gateway address:

```
id=20085 trace_id=209 func=vf_ip4_route_input line=1543
msg="find a route: gw-192.168.11.254 via port6"
```

Source NAT, lookup next available port:

```
id=20085 trace_id=209 func=get_new_addr line=1219
msg="find SNAT: IP-192.168.11.59, port-31925"
direction"
```

Matched security policy. Check to see which policy this session matches:

```
id=20085 trace_id=209 func=fw_forward_handler line=317
msg="Allowed by Policy-3: SNAT"
```

Apply source NAT:

```
id=20085 trace_id=209 func=__ip_session_run_tuple
line=1502 msg="SNAT 192.168.3.221->192.168.11.59:31925"
```

SYN ACK received:

```
id=20085 trace_id=210 func=resolve_ip_tuple_fast line=2700
msg="vd-root received a packet(proto=6, 203.160.224.97:80-
>192.168.11.59:31925) from port6."
```

Found existing session ID. Identified as the reply direction:

```
id=20085 trace_id=210 func=resolve_ip_tuple_fast line=2727
msg="Find an existing session, id-00000e90, reply direction"
```

Apply destination NAT to inverse source NAT action:

```
id=20085 trace_id=210 func=__ip_session_run_tuple
line=1516 msg="DNAT 192.168.11.59:31925-
>192.168.3.221:1487"
```

Lookup for next-hop gateway address for reply traffic:

```
id=20085 trace_id=210 func=vf_ip4_route_input line=1543
msg="find a route: gw-192.168.3.221 via port5"
```

ACK received:

```
id=20085 trace_id=211 func=resolve_ip_tuple_fast line=2700
msg="vd-root received a packet(proto=6,
192.168.3.221:1487->203.160.224.97:80) from port5."
```

Match existing session in the original direction:

```
  id=20085 trace_id=211 func=resolve_ip_tuple_fast line=2727
  msg="Find an existing session, id-00000e90, original
direction"
```

Apply source NAT:

```
id=20085 trace_id=211 func=__ip_session_run_tuple
line=1502 msg="SNAT 192.168.3.221->192.168.11.59:31925"
```

Receive data from client:

```
id=20085 trace_id=212 func=resolve_ip_tuple_fast
line=2700 msg="vd-root received a packet(proto=6,
192.168.3.221:1487->203.160.224.97:80) from port5."
```

Match existing session in the original direction:

```
id=20085 trace_id=212 func=resolve_ip_tuple_fast
line=2727 msg="Find an existing session, id-00000e90,
original direction"
```

Apply source NAT:

```
id=20085 trace_id=212 func=__ip_session_run_tuple
line=1502 msg="SNAT 192.168.3.221->192.168.11.59:31925"
```

Receive data from server:

```
id=20085 trace_id=213 func=resolve_ip_tuple_fast
line=2700 msg="vd-root received a packet(proto=6,
203.160.224.97:80->192.168.11.59:31925) from port6."
```

Match existing session in reply direction:

```
id=20085 trace_id=213 func=resolve_ip_tuple_fast
line=2727 msg="Find an existing session, id-00000e90,
reply direction"
```

Apply destination NAT to inverse source NAT action:

```
id=20085 trace_id=213 func=__ip_session_run_tuple
line=1516 msg="DNAT 192.168.11.59:31925-
>192.168.3.221:1487"
```

Sample output: IPsec (policy-based)

```
id=20085 trace_id=1 msg="vd-root received a packet(proto=1, 10.72.55.240:1->10.71.55.10:8)
     from internal."
id=20085 trace_id=1 msg="allocate a new session-00001cd3"
id=20085 trace_id=1 msg="find a route: gw-66.236.56.230 via wan1"
id=20085 trace_id=1 msg="Allowed by Policy-2: encrypt"
id=20085 trace_id=1 msg="enter IPsec tunnel-RemotePhase1"
id=20085 trace_id=1 msg="encrypted, and send to 15.215.225.22 with source 66.236.56.226"
id=20085 trace_id=1 msg="send to 66.236.56.230 via intf-wan1‟
id=20085 trace_id=2 msg="vd-root received a packet (proto=1, 10.72.55.240:1-1071.55.10:8)
     from internal."
id=20085 trace_id=2 msg="Find an existing session, id-00001cd3, original direction"
id=20085 trace_id=2 msg="enter IPsec ="encrypted, and send to 15.215.225.22 with source
     66.236.56.226‟ tunnel-RemotePhase1"
id=20085 trace_id=2 msgid=20085 trace_id=2 msg="send to 66.236.56.230 via intf-wan1"
```

Testing a proxy operation

To monitor proxy operations in the CLI:

```
diagnose test application <application> <option>
```

To display a list of available application values:

```
diagnose test application ?
```

To display a list of available option values:

```
diagnose test application <application> ?
```

The `<option>` value will depend on the application value used in the command.

For example, if the application is `http`, the CLI command that displays the `<option>` values is:

```
diagnose test application http ?
```

Displaying detail Hardware NIC information

Monitoring the hardware NIC is important because interface errors indicate data link or physical layer issues which may impact the performance of the FortiGate.

To monitor hardware network operations in the CLI:

```
diagnose hardware deviceinfo nic <interface>
```

Sample output:

The following is sample output when the `<interface>` is set to `lan`:

```
System_Device_Name lan
Current_HWaddr 00:09:0f:68:35:60
Permanent_HWaddr 00:09:0f:68:35:60
State up
Link up
Speed 100
Duplex full
[……]
Rx_Packets=5685708
Tx_Packets=4107073
Rx_Bytes=617908014
Tx_Bytes=1269751248
Rx_Errors=0
Tx_Errors=0
Rx_Dropped=0
Tx_Dropped=0
[……]
```

Error descriptions

The `diagnose hardware deviceinfo nic` command displays a list of error names and values that are related to hardware.

The following table describes possible hardware errors:

Field	Description
Rx_Errors = rx error count	Bad frame was marked as error by PHY
Rx_CRC_Errors + Rx_Length_Errors - Rx_Align_Errors	This error is only valid in 10/100M mode
Rx_Dropped or Rx_No_Buffer_ Count	Running out of buffer space
Rx_Missed_Errors	Equals Rx_FIFO_Errors + CEXTERR (Carrier Extension Error Count); only valid in 1000M mode, which is marked by PHY
Tx_Errors = Tx_ Aborted_Errors	ECOL (Excessive Collisions Count); only valid in half-duplex mode
Tx_Window_Errors	Late Collisions (LATECOL) Count Late collisions are collisions that occur after 64-byte time into the transmission of the packet while working in 10 to 100 Mb/s data rate and 512-byte time into the transmission of the packet while working in the 1,000 Mb/s data rate. This register only increments if transmits are enabled and the device is in half-duplex mode.
Rx_Dropped	See Rx_Errors
Tx_Dropped	Not defined
Collisions	Total number of collisions experienced by the transmitter; valid in half-duplex mode
Rx_Length_Errors	Transmission length error
Rx_Over_Errors	Not defined
Rx_CRC_Errors	Frame CRC error
Rx_Frame_Errors	Same as Rx_Align_Errors This error is only valid in 10/100M mode.
Rx_FIFO_Errors	Same as Rx_Missed_Errors - a missed packet count
Tx_Aborted_Errors	See Tx_Errors
Tx_Carrier_Errors	The PHY should assert the internal carrier sense signal during every transmission. Failure to do so may indicate that the link has failed or the PHY has an incorrect link configuration. This register only increments if transmits are enabled. This register isn't valid in internal SerDes 1 mode (TBI mode for the 82544GC/EI) and is valid only when the Ethernet controller is operating at full duplex.
Tx_FIFO_Errors	Not defined
Tx_Heartbeat_Errors	Not defined
Tx_Window_Errors	See LATECOL

Field	Description
Tx_Single_Collision_Frames	Counts the number of times that a successfully transmitted packet encountered a single collision The value increments only if transmits are enabled and the Ethernet controller is in half-duplex mode.
Tx_Multiple_Collision_Frames	A Multiple Collision Count which indicates the number of times that a transmit encountered more than one collision, but less than 16. The value increments only if transmits are enabled and the Ethernet controller is in half-duplex mode.
Tx_Deferred	Counts defer events. A deferred event occurs when the transmitter cannot immediately send a packet due to: • The medium being busy because another device is transmitting • The IPG timer hasn't expired • Half-duplex deferral events are occurring • XOFF frames are being received • he link is not up. This register only increments if transmits are enabled. This counter does not increment for streaming transmits that are deferred due to TX IPG.
Rx_Frame_Too_Longs	The Rx frame is oversized
Rx_Frame_Too_Shorts	The Rx frame is too short
Rx_Align_Errors	This error is only valid in 10/100M mode
Symbol Error Count	Counts the number of symbol errors between reads - SYMERRS. The count increases for every bad symbol that's received, whether or not a packet is currently being received and whether or not the link is up. This register increments only in internal SerDes mode.

Performing a traffic trace

Traffic tracing allows you to follow a specific packet stream. This is useful when you want to confirm that packets are using the route you expect them to take on your network.

To view traffic sessions:

Use this command to view the characteristics of a traffic session though specific security policies.

```
diagnose sys session
```

To trace per-packet operations for flow tracing:

```
diagnose debug flow
```

To trace per-Ethernet frame:

```
diagnose sniffer packet
```

To trace a route from a FortiGate to a destination IP address:

```
# execute traceroute www.fortinet.com
traceroute to www.fortinet.com (66.171.121.34), 32 hops max, 84 byte packets
1 172.20.120.2 0.637 ms 0.653 ms 0.279 ms
2 209.87.254.221 <static-209-87-254-221.storm.ca> 2.448 ms 2.519 ms 2.458 ms
3 209.87.239.129 <core-2-g0-2.storm.ca> 2.917 ms 2.828 ms 9.324 ms
4 209.87.239.199 <core-3-bdi1739.storm.ca> 13.248 ms 12.401 ms 13.009 ms
5 216.66.41.113 <v502.core1.tor1.he.net> 17.181 ms 12.422 ms 12.268 ms
6 184.105.80.9 <100ge1-2.core1.nyc4.he.net> 21.355 ms 21.518 ms 21.597 ms
7 198.32.118.41 <ny-paix-gni.twgate.net> 83.297 ms 84.416 ms 83.782 ms
8 203.160.228.217 <217-228-160-203.TWGATE-IP.twgate.net> 82.579 ms 82.187 ms 82.066 ms
9 203.160.228.229 <229-228-160-203.TWGATE-IP.twgate.net> 82.055 ms 82.455 ms 81.808 ms
10 203.78.181.2 82.262 ms 81.572 ms 82.015 ms
11 203.78.186.70 83.283 ms 83.243 ms 83.293 ms
12 66.171.127.177 84.030 ms 84.229 ms 83.550 ms
13 66.171.121.34 <www.fortinet.com> 84.023 ms 83.903 ms 84.032 ms
14 66.171.121.34 <www.fortinet.com> 83.874 ms 84.084 ms 83.810 ms
```

Using a session table

A session is a communication channel between two devices or applications across the network. Sessions allow FortiOS to inspect and act on a sequential group of packets in a session all at once instead of inspecting each packet individually. Each session has an entry in the session table that includes important information about the session.

You can view FortiGate session tables from the FortiGate GUI or CLI. The most useful troubleshooting data comes from the CLI. The session table in the GUI also provides useful summary information, particularly the current policy number that the session is using.

When to use a session table

Session tables are useful when verifying open connections. For example, if you have a web browser open to browse the Fortinet website, you would expect a session entry from your computer on port 80 to the IP address for the Fortinet website.

You can also use a session table to investigate why there are too many sessions for FortiOS to process.

GUI

To view session information in the GUI:

1. Go to *Security Fabric > Physical Topology*.
2. From the *Metrics* dropdown, select *Sessions*.

Finding the security policy for a specific connection

Every program and device on your network must have an open communication channel or session to pass information. FortiGate manages these sessions with features such as traffic shaping, antivirus scanning, and blocking known bad websites. Each session will have an entry in the session table.

If a secure web browser session is not working properly, you can check the session table to ensure the session is still active and going to the proper address. The session table can also tell you the security policy number it matches, so you can check what is happening in that policy.

1. Get the connection information.

You need to be able to identify the session you want. To do this, you will need:

- The source IP address (usually your computer)
- The destination IP address (if you have it)
- The port number which is determined by the program you are using. Common ports are:
  - Port 80 (HTTP for web browsing)
  - Port 443 (HTTPS for SSL encrypted web browsing)
  - Port 22 (SSH for Secure Shell)
  - Port 25 (SMTP for Mail Transfer)

2. Find the session and policy ID

Go to *Security Fabric > Physical Topology*. From the *Metrics* dropdown, select *Sessions*.

To find your session, search for your source IP address, destination IP address (if you have it), and port number. The policy ID is listed after the destination information.

3. Use filters to find a session

If there are multiple pages of sessions, you can use a filter to hide the sessions you do not need. To filter the sessions in the table, click *Add Filter*, and select an option from the list. You can filter the table by *Destination IP*, *Source IP*, or *Source Port*.

CLI

The session table output in the CLI is very large. The CLI command supports filters to show only the data you need.

To view session data in the CLI:

```
diagnose sys session list
```

An entry is placed in the session table for each traffic session passing through a security policy

To filter session data:

```
diagnose sys session filter <option>
```

The values for `<option>` include the following:

Value	Definition
`clear`	Clear session filter
`dintf`	Destination interface
`dport`	Destination port

Value	Definition
`dst`	Destination IP address
`duration`	Duration of the session
`expire`	Expire
`negate`	Inverse filter
`nport`	NAT'd source port
`nsrc`	NAT'd source ip address
`policy`	Policy ID
`proto`	Protocol number
`proto-state`	Protocol state
`session-state1`	Session state1
`session-state2`	Session state2
`sintf`	Source interface
`sport`	Source port
`src`	Source IP address
`vd`	Index of virtual domain, `-1` matches all

Even though UDP is a sessionless protocol, FortiGate keeps track of the following states:

- When UDP reply does not have a value of 0
- When UDP reply has a value of 1

The following table displays firewall session states from the session table:

State	Description
log	Session is being logged
local	Session is originated from or destined for local stack
ext	Session is created by a firewall session helper
may_dirty	Session is created by a policy For example, the session for `ftp control channel` will have this state but `ftp data channel` won't. This is also seen when NAT is enabled.
ndr	Session will be checked by IPS signature
nds	Session will be checked by IPS anomaly
br	Session is being bridged (TP) mode

Examining the firewall session list

The firewall session list displays all open sessions in FortiGate. Examine the list for strange patterns, such as no sessions apart from the internal network, or all sessions are only to one IP address.

When you examine the firewall session list in the CLI, you can use filters to reduce the output.

To examine the firewall session list in the CLI:

You can use a filter to limit the sessions displayed by source, destination address, port, or NAT'd address. To use more than one filter, enter a separate line for each value.

The following example filters the session list based on a source address of 10.11.101.112:

```
FGT# diagnose sys session filter src 10.11.101.112
FGT# diagnose sys session list
```

The following example filters the session list based on a destination address of 172.20.120.222.

```
FGT# diagnose sys session filter dst 172.20.120.222
FGT# diagnose sys session list
```

To clear all sessions corresponding to a filter:

```
FGT# diagnose sys session filter dst 172.20.120.222
FGT# diagnose sys session clear
```

Checking source NAT information

Checking source NAT is important when you are troubleshooting from the remote end of the connection outside the firewall.

To check the source NAT information in the CLI:

When you display the session list in the CLI, you can match the NAT'd source address (`nsrc`) and port (`nport`). This is useful when multiple internal IP addresses are NAT'd to a common external-facing source IP address.

```
FGT# diagnose sys session filter nsrc 172.20.120.122
FGT# diagnose sys session filter nport 8888
FGT# diagnose sys session list
```

Finding object dependencies

You may be prevented from deleting a configuration object when other configuration objects depend on it. You can use the GUI or CLI to identify objects which depend on, or make reference to the configuration you are trying to delete. Additionally, if you have a virtual interface with dependent objects, you will need to find and remove those dependencies before deleting the interface.

To remove interface object dependencies in the GUI:

1. Go to *Network > Interfaces*. The *Ref.* column displays the number of objects that reference this interface.
2. Select the number in the *Ref .* column for the interface. A window listing the dependencies appears.
3. Use these detailed entries to locate and remove object references to this interface. The trash can icon is enabled

after all the object dependencies are removed.

4. Remove the interface by selecting the check box for the interface, and select *Delete*.

To find object dependencies in the CLI:

When running multiple VDOMs, use the following command in the global configuration only.

```
diagnose sys cmdb refcnt show <path.object.mkey>
```

The command searches for the named object in both the most recently used global and VDOM configurations.

Examples

To verify which objects a security policy with an ID of 1 refers to:

```
diagnose sys cmdb refcnt show firewall.policy.policyid 1
```

To check what is referred to by interface `port1`:

```
diagnose sys cmdb refcnt show system.interface.name port1
```

To show all dependencies for an interface:

```
diagnose sys cmdb refcnt show system.interface.name <interface name>
```

Sample output:

In this example , the interface has dependent objects, including four address objects, one VIP, and three security policies.

```
entry used by table firewall.address:name '10.98.23.23_host'
entry used by table firewall.address:name 'NAS'
entry used by table firewall.address:name 'all'
entry used by table firewall.address:name 'fortinet.com'
entry used by table firewall.vip:name 'TORRENT_10.0.0.70:6883'
entry used by table firewall.policy:policyid '21'
entry used by table firewall.policy:policyid '14'
entry used by table firewall.policy:policyid '19'
```

Diagnosing NPU-based interfaces

Some Fortinet products contain network processors, such as NP4, NP6Lite, or NP6. Offloading requirements will vary depending on the model.

To view the initial session setup for NPU-based interfaces:

```
diagnose debug flow
```

- If the session is programmed into the ASIC (fastpath) correctly, the command will not detect the packets that arrive at the CPU.
- If the NPU functionality is disabled, the CPU detects all the packets. However, you should only disable the NPU functionality for troubleshooting purposes.

To diagnose NPU-based interfaces:

1. Get the NPx or NPU ID and port numbers.
```
diagnose npu <processor> list
```
The output will look like this:
```
ID Model Slot Interface
0 On-board port1 fabric1 fabric3 fabric5
1 On-board fabric2 port2 base2 fabric4
```
2. Disable the NPU functionality.
```
diagnose npu <processor> fastpath disable <dev_id>
```
The dev_id is the NPx ID number.
3. Analyze the packets.
```
diagnose npu <processor> fastpath-sniffer enable port1
```

These commands only apply to NP4 and NP6 interfaces.

The output will look similar to:
```
NP4 Fast Path Sniffer on port1 enabled
```
This causes traffic on port1 of the network processor to be sent to the CPU. This means you can perform a standard sniffer trace and use other diagnostic commands, if it is a standard CPU-driven port.

Identifying the XAUI link used for a specific traffic stream

The diagnose npu np6 xaui-hash command takes a 6-tuple input of the traffic stream to identify the NP6 XAUI link that the traffic passes through.

This command is only available on the 38xxD, 39xxD, 34xxE, 36xxE, and 5001E series devices.

Syntax

```
diagnose npu np6 xaui-hash <interface> <proto> <src_ip> <dst_ip> <src_port> <dst_port>
```

Variable	Description
<interface>	The network interface that the packets are coming from.
<proto>	The proto number, 6 for TCP or 17 for UDP.
<src_ip>	The source IP address.
<dst_ip>	The destination IP address.
<src_port>	The source port.
<dst_port>	The destination port.

Examples

```
# diagnose npu np6 xaui-hash port1 6 1.1.1.1 2.2.2.1 4567 80
 NP6_ID: 0, XAUI_LINK: 2
```

```
# diagnose npu np6 xaui-hash port1 6 1.1.1.1 2.2.2.1 4567 200
NP6_ID: 6, XAUI_LINK: 2

# diagnose npu np6 xaui-hash port1 6 1.1.1.1 2.2.2.1 4567 20
NP6_ID: 1, XAUI_LINK: 2

# diagnose npu np6 xaui-hash port1 6 1.1.1.1 2.2.2.1 4567 23
NP6_ID: 1, XAUI_LINK: 1
```

The `NP6_ID` is the NP index of the model that is being used. It can be found with the `diagnose npu np6 port-list` command.

Date and time settings

Fortinet support may ask you to check the date and time settings for log message timestamp synchronization and for certificates that have a time requirement to check for validity.

To check time settings:

```
execute time
```

To check date settings:

```
execute date
```

If all devices have the same time, it helps to correlate log entries from different devices.

```
execute time
current time is: 12:40:48
last ntp sync:Thu Mar 16 12:00:21 2006
execute date
current date is: 2006-03-16
```

To force synchronization with an NTP server:

```
config system ntp
   set ntpsync {enable | disable}
end
```

If all devices have the same time, it helps to correlate log entries from different devices.

Running the TAC report

The Technical Assistance Center (TAC) report runs an exhaustive series of diagnostic commands. Some of the commands are only needed if you are using features, such as HA, VPN tunnels, or a modem. Fortinet support my ask you to use the report output to provide information about the current state of your FortiGate.

Due the amount of output generated, the report may take a few minutes to run. If you are logging CLI output to a file, you can run this command to familiarize yourself with the diagnostic commands.

To run the TAC report in the CLI:

```
execute tac report
```

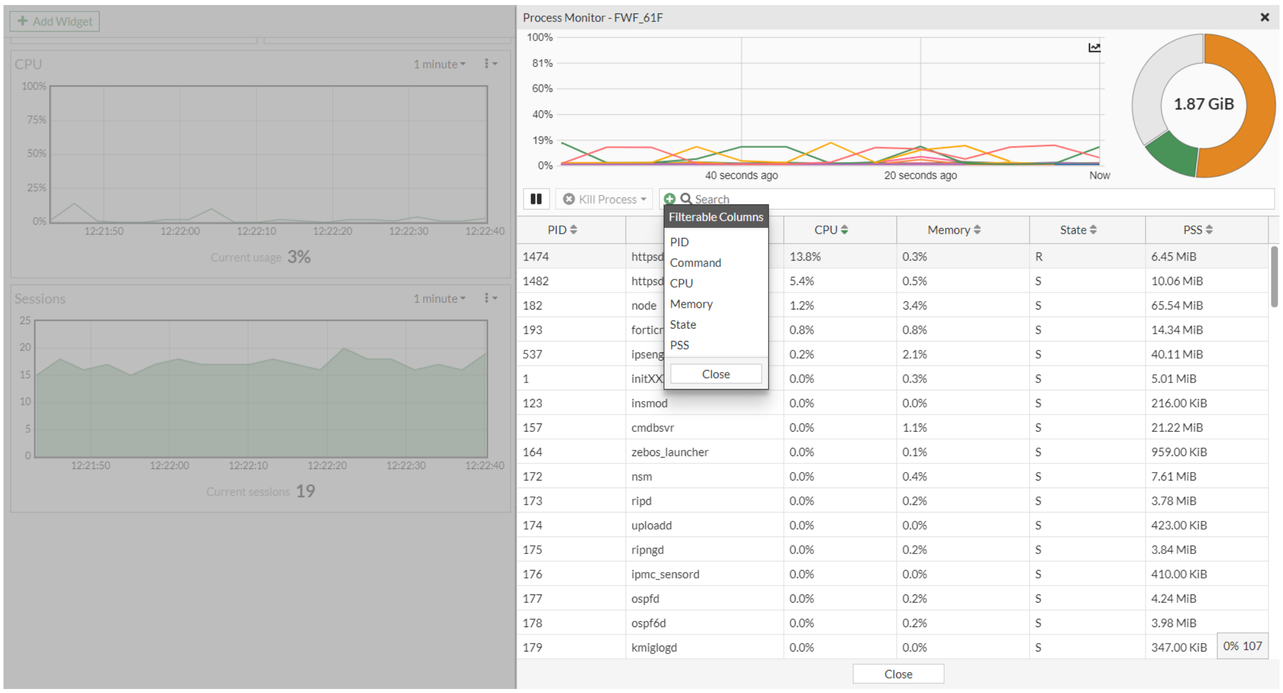
Using the process monitor

The *Process Monitor* displays running processes with their CPU and memory usage levels. Administrators can sort, filter, and terminate processes within the *Process Monitor* pane.

To access the process monitor:

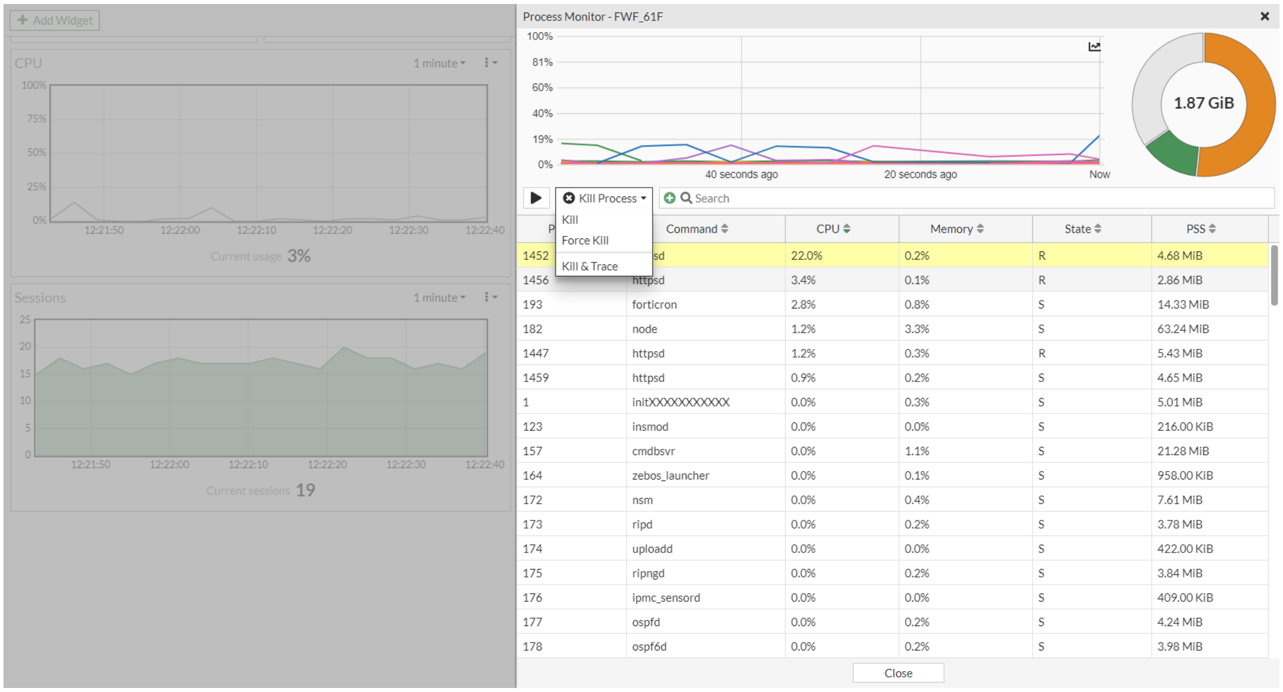
1. Go to *Dashboard* > *Status*:
   - Left-click in the *CPU* or *Memory* widget and select *Process Monitor*.
   - Click the user name in the upper right-hand corner of the screen, then go to *System* > *Process Monitor*.

   The *Process Monitor* appears, which includes a line graph, donut chart, and process list.
2. Click the + beside the search bar to view which columns can be filtered.

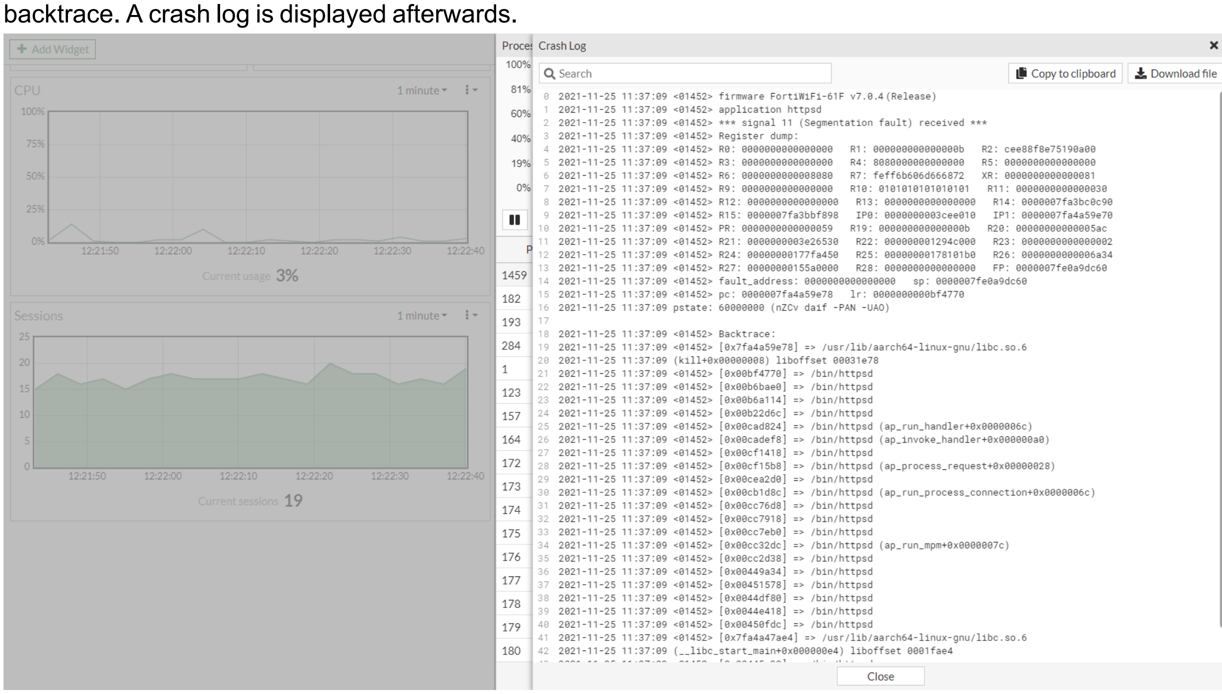


To kill a process within the process monitor:

1. Select a process.
2. Click the *Kill Process* dropdown.



3. Select one of the following options:
   - *Kill*: the standard kill option that produces one line in the crash log (`diagnose debug crashlog read`).
   - *Force Kill*: the equivalent to `diagnose sys kill 9 <pid>`. This can be viewed in the crash log.
   - *Kill & Trace*: the equivalent to `diagnose sys kill 11 <pid>`. This generates a longer crash log and backtrace. A crash log is displayed afterwards.



Other commands

You may be asked to provide the following information when you contact Fortinet support.

- ARP table on page 2675
- IP address on page 2677

ARP table

The ARP table is used to determine the destination MAC addresses of the network nodes, as well as the VLANs and ports from where the nodes are reached.

To view the ARP table:

```
# get system arp

Address           Age(min)   Hardware Addr      Interface
10.10.1.3         1          50:b7:c3:75:ea:dd internal7
192.168.0.190     0          28:f1:0e:03:2a:97 wan1
192.168.0.97      0          f4:f2:6d:37:b0:99 wan1
```

To view the ARP cache in the system:

```
# diagnose ip arp list

index=14 ifname=internal7 10.10.1.3 50:b7:c3:75:ea:dd state=00000004 use=2494 confirm=1995
update=374 ref=3
index=5 ifname=wan1 192.168.0.190 28:f1:0e:03:2a:97 state=00000002 use=88 confirm=86
update=977639 ref=2
index=22 ifname=internal 192.168.1.111 00:0c:29:c6:79:3d state=00000004 use=3724
confirm=9724 update=3724 ref=0
index=5 ifname=wan1 224.0.1.140 01:00:5e:00:01:8c state=00000040 use=924202 confirm=930202
update=924202 ref=1
index=5 ifname=wan1 192.168.0.97 f4:f2:6d:37:b0:99 state=00000002 use=78 confirm=486
update=614 ref=26
index=14 ifname=internal7 10.10.1.11 state=00000020 use=172 confirm=1037790 update=78 ref=2
```

ARP request and cache

The FortiGate must make an ARP request when it tries to reach a new destination. The base ARP reachable value determines how often an ARP request it sent; the default is 30 seconds. The actual ARP reachable time is a random number between half and three halves of the base reachable time, or 15 to 45 seconds. The random number is updated every five minutes.

ARP entries in the ARP cache are updated based on the state of the ARP entry and the objects that are using it, as highlighted in the following output sample:

```
index=5 ifname=wan1 224.0.1.140 01:00:5e:00:01:8c state=00000040 use=924202
confirm=930202 update=924202 ref=1
```

There are multiple possible states for an ARP entry, and the state-transition mechanism can be complex. Common states include the following:

State	Meaning	Description
000000002 or 0x02	REACHABLE	An ARP response was received
000000004 or 0x04	STALE	No ARP response within the expected time
000000008 or 0x08	DELAY	A transition state between STALE and REACHABLE before Probes are sent out
000000020 or 0x20	FAILED	Did not manage to resolve within the maximum configured number of probes
000000040 or 0x40	NOARP	Device does not support ARP, e.g. IPsec interface
000000080 or 0x80	PERMANENT	A statically defined ARP entry

An entry that is in the STALE (0x04) or FAILED (0x20) states with no references to it (ref=0) can be deleted. Many factors affect the state-transmit mechanism and if an entry is used by other subsystems. For example, ARP creation, ARP request/reply, neighbor lookup, routing, and others can cause an ARP entry to be in use or referenced.

The garbage collection mechanism runs every 30 seconds, and checks and removes stale and unreferenced entries if they have been stale for longer than 60 seconds. Garbage collection will also be triggered when the number of ARP entries exceeds the configured threshold. If the threshold is exceeded, no entries can be added to the ARP table.

To set the maximum number of ARP entries threshold:

```
config system global
    set arp-max-entry <integer>
end
```

arp-max-entry <integer>	The maximum number of dynamically learned MAC addresses that can be added to the ARP table (131072 to 2147483647, default = 131072).

To set the ARP reachable time on an interface:

```
config system interface
    edit port1
        set reachable-time <integer>
    next
end
```

reachable-time <integer>	The reachable time (30000 to 3600000, default = 30000).

To clear all of the entries in the ARP table:

```
execute clear system arp table
```

To delete a single ARP entry from the ARP table:

```
diagnose ip arp delete <interface name> <IP address>
```

To add static ARP entries:

```
config system arp-table
    edit 1
        set interface "internal"
        set ip 192.168.50.8
        set mac bc:14:01:e9:77:02
    next
end
```

To view a summary of the ARP table:

```
# diagnose sys device list root

list virtual firewall root info:
ip4 route_cache: table_size=65536 max_depth=2 used=31 total=34
arp: table_size=16 max_depth=2 used=5 total=6
proxy_arp: table_size=256 max_depth=0 used=0 total=0
arp6: table_size=32 max_depth=1 used=3 total=3
proxy_arp6: table_size=256 max_depth=0 used=0 total=0
local table version=00000000 main table version=0000002b
vf=root dev=root vrf=0
vf=root dev=ssl.root vrf=0
...
vf=root dev=internal5 vrf=0
ses=0/0 ses6=0/0 rt=0/0 rt6=0/0
```

IP address

You may want to verify the IP addresses assigned to the FortiGate interfaces are what you expect them to be.

To verify IP addresses:

```
diagnose ip address list
```

The output lists the:

- IP address and mask (if available)
- `index` of the interface (a type of ID number)
- `devname` (the interface name)

While physical interface names are set, virtual interface names can vary. A good way to use this command is to list all of the virtual interface names. For `vsys_ha` and `vsys_fgfm`, the IP addresses are the local host, which are virtual interfaces that are used internally.

Sample output:

```
# diagnose ip address list
IP=10.31.101.100->10.31.101.100/255.255.255.0 index=3 devname=internal
IP=172.20.120.122->172.20.120.122/255.255.255.0 index=5 devname=wan1
IP=127.0.0.1->127.0.0.1/255.0.0.0 index=8 devname=root
IP=127.0.0.1->127.0.0.1/255.0.0.0 index=11 devname=vsys_ha
IP=127.0.0.1->127.0.0.1/255.0.0.0 index=13 devname=vsys_fgfm
```

FortiGuard troubleshooting

The FortiGuard service provides updates to AntiVirus (AV), Antispam (AS), Intrusion Protection Services (IPS), Webfiltering (WF), and more. The FortiGuard Distribution System (FDS) consists of a number of servers across the world that provide updates to your FortiGate unit. Problems can occur with the connection to FDS and its configuration on your local FortiGate unit.

Some of the more common troubleshooting methods are listed here, including:

- Troubleshooting process for FortiGuard updates on page 2679
- FortiGuard server settings on page 2679
- FortiGuard server settings on page 2679

Verifying connectivity to FortiGuard

You can verify FortiGuard connectivity in the GUI and CLI.

To verify FortiGuard connectivity in the GUI:

1. Got to *Dashboard* > *Status*.
2. Check the *Licenses* widget. When FortiGate is connected to FortiGuard, a green check mark appears next to the available FortiGuard services.

To verify FortiGuard connectivity in the CLI:

```
execute ping service.fortiguard.net
```

```
execute ping update.fortiguard.net
```

Sample output:

```
FG100D# execute ping service.fortiguard.net
PING guard.fortinet.net (208.91.112.196): 56 data bytes
64 bytes from 208.91.112.196: icmp_seq=0 ttl=51 time=61.0 ms
64 bytes from 208.91.112.196: icmp_seq=1 ttl=51 time=60.0 ms
64 bytes from 208.91.112.196: icmp_seq=2 ttl=51 time=59.6 ms
64 bytes from 208.91.112.196: icmp_seq=3 ttl=51 time=58.9 ms
64 bytes from 208.91.112.196: icmp_seq=4 ttl=51 time=59.2 ms

--- guard.fortinet.net ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 58.9/59.7/61.0 ms

FG100D# execute ping update.fortiguard.net
PING fds1.fortinet.com (208.91.112.68): 56 data bytes
64 bytes from 208.91.112.68: icmp_seq=0 ttl=53 time=62.0 ms
64 bytes from 208.91.112.68: icmp_seq=1 ttl=53 time=61.8 ms
64 bytes from 208.91.112.68: icmp_seq=2 ttl=53 time=61.3 ms
64 bytes from 208.91.112.68: icmp_seq=3 ttl=53 time=61.9 ms
64 bytes from 208.91.112.68: icmp_seq=4 ttl=53 time=61.8 ms
```

Troubleshooting process for FortiGuard updates

The following process shows the logical steps you should take when troubleshooting problems with FortiGuard updates:

1. **Does the device have a valid license that includes these services?**
   Each device requires a valid FortiGuard license to access updates for some or all of these services. You can verify the status of the support contract for your devices at the Fortinet Support website.
2. **If the device is part of a high availability (HA) cluster, do all members of the cluster have the same level of support?**
   You can verify the status of the support contract for all of the devices in your HA cluster at the Fortinet Support website.
3. **Are services enabled on the device?**
   To see the FortiGuard information and status for a device in the GUI, go to *System > FortiGuard*.
   Use this page to verify the status of each component, and enable each service.
4. **Can the device communicate with FortiGuard servers?**
   Go to *System > FortiGuard* in the GUI, and try to update AntiVirus and IPS, or test the availability of Web Filtering and AS default and alternate ports.
5. **Is there proper routing to reach the FortiGuard servers?**
   Ensure there is a static or dynamic route that allows your FortiGate to reach the FortiGuard servers. Usually a generic default route to the internet is enough, but you may need to verify this if your network is complex.
6. **Are there issues with DNS?**
   An easy way to test this is to attempt a traceroute from behind the FortiGate to an external network using the Fully Qualified Domain Name (FQDN) for a location. If the traceroute FQDN name doesn't resolve, you have general DNS problems.
7. **Is there anything upstream that might be blocking FortiGuard traffic, either on the network or ISP side?**
   Many firewalls block all ports, by default, and ISPs often block ports that are low. There may be a firewall between the FortiGate and the FortiGuard servers that's blocking the traffic. By default, FortiGuard uses port 53. If that port is blocked you need to either open a hole for it or change the port it is using.
8. **Is there an issue with source ports?**
   It is possible that ports that FortiGate uses to contact FortiGuard are being changed before they reach FortiGuard or on the return trip before they reach FortiGate. A possible solution for this is to use a fixed-port at NAT'd firewalls to ensure the port remains the same. You can use packet sniffing to find more information about what's happening with ports.
9. **Are there security policies that include antivirus?**
   If none of the security policies include antivirus, the antivirus database will not be updated. If antivirus is included, only the database type that's used will be updated.

FortiGuard server settings

Your local FortiGate connects to remote FortiGuard servers to get updates to FortiGuard information, such as new viruses that may have been found or other new threats.

This section provides methods to display FortiGuard server information on your FortiGate, and how to use that information and update it to fix potential problems.

Displaying the server list

To get a list of FDS servers FortiGate uses to send web filtering requests:

```
get webfilter status
```

or

```
diagnose debug rating
```

Rating requests are only sent to the server at the top of the list in normal operation. Each server is probed for Round Trip Time (RTT) every two minutes. Rating may not be enabled on your FortiGate.

Optionally, you can add a refresh rate to the end of the command to determine how often the server list is refreshed.

Sample output:

```
Locale        : english
License       : Contract
Expiration    : Thu Oct  9 02:00:00 2011
-=- Server List (Mon Feb 18 12:55:48 2008) -=-
IP             Weight    RTT     Flags    TZ      Packets CurrLost TotalLost
a.b.c.d         0        1       DI       2       1926879 0        11176
10.1.101.1      10       329              1       10263   0        633
10.2.102.2      20       169              0       16105   0        80
10.3.103.3      20       182              0       6741    0        776
10.4.104.4      20       184              0       5249    0        987
10.5.105.5      25       181              0       12072   0        178
```

Output details

The server list includes the IP addresses of alternate servers if the first entry cannot be reached. In this example, the IP addresses are not public addresses.

The following flags in `get webfilter status` indicate the server status:

Flag	Description
D	The server was found through the DNS lookup of the hostname. If the hostname returns more than one IP address, all of them are flagged with D and are used first for INIT requests before falling back to the other servers.
I	The server to which the last INIT request was sent
F	The server hasn't responded to requests and is considered to have failed
T	The server is currently being timed
S	Rating requests can be sent to the server. The flag is set for a server only in two cases: • The server exists in the servers list received from the FortiManager or any other INIT server. • The server list received from the FortiManager is empty so the FortiManager is the only server that the FortiGate knows and it should be used as the rating server.

Sorting the server list

The server list is sorted first by weight. The server with the smallest RTT appears at the top of the list, regardless of weight. When a packet is lost (there has been no response in 2 seconds), it is re-sent to the next server in the list. Therefore, the top position in the list is selected based on RTT, while the other positions are based on weight.

Calculating weight

The weight for each server increases with failed packets and decreases with successful packets. To lower the possibility of using a remote server, the weight isn't allowed to dip below a base weight. The base weight is calculated as the difference in hours between the FortiGate and the server multiplied by 10. The farther away the server is, the higher its base weight is and the lower it appears in the list.

View open and in use ports

Traffic destined for the FortiGate itself, and not being passed through or dropped, is called local-in traffic. It can be from a variety of services, such as HTTPS for administrative access, or BGP for inter-router communication.

Local-in traffic is controlled by local-in policies. To enable viewing local-in policies in the GUI, go to *System > Feature Visibility* and enable *Local In Policy*.

The *Policy & Objects > Local In Policy* page shows a read-only list of the local policies, populated with default values, and values that are automatically enabled when the related service is enabled, for example, enabling BGP opens TCP port 179. For more information, see Local-in policy on page 788.

To view ports that are being listened on, and active connections and the services or processes using them:

```
# diagnose sys tcpsock | grep 0.0.0.0
0.0.0.0:10400->0.0.0.0:0->state=listen err=0 socktype=4 rma=0 wma=0 fma=0 tma=0 inode=10621
process=142/authd
...
0.0.0.0:53->0.0.0.0:0->state=listen err=0 socktype=1 rma=0 wma=0 fma=0 tma=0 inode=8067
process=177/dnsproxy
0.0.0.0:22->0.0.0.0:0->state=listen err=0 socktype=1 rma=0 wma=0 fma=0 tma=0 inode=13390
process=225/sshd
0.0.0.0:541->0.0.0.0:0->state=listen err=0 socktype=1 rma=0 wma=0 fma=0 tma=0 inode=13155
process=215/fgfmd
...
0.0.0.0:9980->0.0.0.0:0->state=listen err=0 socktype=1 rma=0 wma=0 fma=0 tma=0 inode=5063
process=129/httpsd
0.0.0.0:179->0.0.0.0:0->state=listen err=0 socktype=1 rma=0 wma=0 fma=0 tma=0 inode=10583
process=148/bgpd
...
```

For more information on incoming and outgoing ports, see the FortiOS Ports guide.

Additional resources

To learn more about FortiGate and FortiOS, as well information about technical issues, please refer to the following resources:

Technical documentation

Installation, Administration, and Quick Start Guides, as well as other technical documents, are available online at the Fortinet Document Library

Fortinet video library

The Fortinet Video Library hosts a collection of video which provide valuable information about Fortinet products.

Release notes

Issues that arise after the technical documentation has been published will often be listed in the Release Notes. To find these, go to the Fortinet Document Library.

Fortinet Community

The Fortinet Community provides a place to collaborate, share insights and experiences, and get answers to questions. It incorporates the Fortinet Knowledge Base and technical discussion forums. You can access the community at: https://community.fortinet.com/.

Knowledge base

The Fortinet Knowledge Base provides access to a variety of articles, white papers, and other documentation that provides technical insight into a range of Fortinet products. The Knowledge Base is available online at: https://community.fortinet.com/t5/Knowledge-Base/ct-p/knowledgebase.

Fortinet technical discussion forums

An online technical forum allows administrators to contribute to discussions about issues that relate to their Fortinet products. Searching the forum can help an administrator identify if an issue has been experienced by another user. You can access the support forums at: https://forum.fortinet.com/

Fortinet training services online campus

The Fortinet Training Services Online Campus hosts a collection of tutorials and training materials which you can use to increase your knowledge of Fortinet products. https://www.fortinet.com/training.html

Fortinet Support

You defined your problem, researched a solution, put together a plan to find the solution, and executed that plan. At this point, if the problem hasn't been solved, it's time to contact Fortinet Support for assistance.

www.fortinet.com