Release Notes

FortiADC 7.4.3

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

March 29, 2024
FortiADC 7.4.3 Release Notes

TABLE OF CONTENTS

Change Log

Date	Change Description
March 29, 2024	FortiADC 7.4.3 Release Notes initial release.

Introduction

This *Release Notes* covers the new features, enhancements, known issues, and resolved issues of FortiADC™ version 7.4.3, Build 0336.

To upgrade to FortiADC 7.4.3, see Upgrade notes.

FortiADC provides load balancing, both locally and globally, and application delivery control. For more information, visit: https://docs.fortinet.com/product/fortiadc.

What's new

FortiADC 7.4.3 offers the following new features and enhancements:

Hardware

New models: 320F and 420F

FortiADC 7.4.3 introduces the new FortiADC 320F and 420F models. For more information, please refer to the latest FortiADC datasheet.

Server Load Balancing

Auto Populate Real Servers

Introduced in FortiADC version 7.2.3, the Auto Populate Real Servers functionality is now supported in FortiADC 7.4.3. This allows you to auto-populate real servers using the real server's FQDN. When the real server has more than one IP address for DNS server responses, the auto-populate feature allows FortiADC to automatically generate additional real servers with these IP addresses.

Scripting

New load balancing command

The new **LB:set_real_server()** command allows you to specify a real server from the configured pool directly. It can be called in the following events: HTTP_REQUEST, HTTP_DATA_REQUEST, BEFORE_AUTH, AUTH_RESULT, PERSISTENCE, and POST_PERSIST.

Example for events with manual routing selection:

```
when HTTP_REQUEST {
       debug("===>>============begin HTTP scripting============\n")
       -- routing to set
       sr = "test07"
       s = LB:get_current_routing()
       if (s == nil or s == "") then
           debug("===>>set routing to be '%s'\n", sr)
               LB:routing(sr)
               s = sr
       end
       debug("===>>current routing: %s\n", s)

    -- Real Server name
       rs = "rs05"
       ret = LB:set_real_server(rs)
       if ret then
               debug("LB:set_real_server(%s) Success.\n", rs);
       else
        debug("LB:set_real_server(%s) Failed.\n", rs);
       end
       debug("===>>============end  HTTP scripting===========\n")
}
```

Example for events without manual routing selection:

```
when PERSISTENCE {
        debug("===>>============begin scripting===========\n")
        -- Real Server name
        rs = "rs06"
        ret = LB:set_real_server(rs)
        if ret then
                debug("set_real_server(%s) Success.\n", rs);
        else
         debug("set_real_server(%s) Failed.\n", rs);
        end
        debug("===>>============end scripting===========\n")
}
```

New SSL command

The new **SSL:disable()** command allows you to disable SSL dynamically. This is useful when a virtual server supports both SSL and non-SSL services. Disabling SSL only takes effect before the SSL handshake starts. On the client side, the SSL will be disabled right after the TCP connection is established within the TCP_ACCEPTED event. On the server side, SSL will be disabled before a real server connection has established, including the following events: HTTP_REQUEST, BEFORE_AUTH, AUTH_RESULT, PERSISTENCE, POST_PERSIST, and SERVER_BEFORE_CONNECT.

```
--Client side must be TCP ACCEPTED
when TCP_ACCEPTED {
        debug("------> TCP accepted begin:\n");
        srcIP = IP:client_addr();
        srcPort = IP:client_port();
        debug("------> Client ip:port %s:%s\n", srcIP, srcPort);

        destIP = IP:local_addr();
        destPort = IP:local_port();
        debug("------> Local ip:port %s:%s\n", destIP, destPort);

        if tonumber(destPort) == 80 then
                ret = SSL:disable("clientside");
                if ret then
                        debug("------> SSL disable clientside successfully.\n");
                else
                        debug("------> SSL disable clientside failed.\n");
                end
        else
            debug("------> SSL disable clientside skipped.\n");
        end

        debug("------> TCP accepted end.\n");
}
```

```
--Server side can be called within many events
when HTTP_REQUEST {
    debug("------> HTTP Request begin:\n");
        srcIP = IP:client_addr();
        srcPort = IP:client_port();
        debug("------> Client ip:port %s:%s\n", srcIP, srcPort);

        destIP = IP:local_addr();
        destPort = IP:local_port();
        debug("------> Local ip:port %s:%s\n", destIP, destPort);

        if tonumber(destPort) == 80 then
                ret = SSL:disable("serverside");
                if ret then
                        debug("------> SSL disable serverside successfully.\n");
                else
                        debug("------> SSL disable serverside failed.\n");
                end
        else
            debug("------> SSL disable serverside skipped.\n");
        end

        debug("------> HTTP Request end.\n");
}
```

Enhanced TCP command

The **TCP:sockopt()** command has been enhanced to read the Client Address from the TCP socket option by adding the "type" option that can get the information. This is useful for passing the original client IP to the server side after the client IP has already gone through NAT.
Note: Currently only the GET operation is supported for "type".

```
when HTTP_REQUEST {
        debug("============begin scripting.\n")

        clientip = nil
        t = {}
        t["op"] = "get"
        -- Set the custom TCP option type
        t["type"] = 28
        ret = TCP:sockopt(t)
        if ret and (string.len(ret)>=4) then
                debug("------> TCP get sockopt(%d): (returned %d bytes) successfully.\n", t
["type"], string.len(ret));
                print_byte_array(ret)
                clientip = binStrToIpAddress(ret)
                debug("clientip = %s\n", clientip)
        else
                debug("------> TCP get sockopt(%d) failed.\n", t["type"])
        end

        if clientip then
                res = HTTP:header_insert("X-Forwarded-For", clientip)
                if res then
                        debug("------> Header inserted successfully.\n")
                else
                        debug("------> Header failed to insert.\n")
                end
        end
```

```
       debug("============end scripting.\n")
}

function binStrToIpAddress(binStr)
    return  tostring(string.byte(binStr,1)) .. "." .. tostring(string.byte(binStr,2))
..
               "." .. tostring(string.byte(binStr,3)) .. "." .. tostring(string.byte
(binStr,4))
end

function print_byte_array(s)
  for i=1, string.len(s) do
     debug("0x%x.", string.byte(s,i))
  end
  debug("\n")
end
```

Troubleshooting

Client URL support in CLI

New CLI commands have been added to support curl (Client URL) troubleshooting:

- `execute curl` allows you to set the Client URL.
- `execute curl-option` allows you to set the Client URL options listed in the following table.

Option	Description
`header`	Specify the curl header(s) or set as "auto". Multiple headers can be separated by three colons,"`:::`".
`http-version`	Specify the HTTP version: 0.9, 1.0, 1.1, or 2. The default version is 1.1.
`interface`	Specify the interface or IP, or set as "auto".
`raw`	Enable this option to disable all internal HTTP decoding of content or transfer encodings to allow them to pass unaltered in its "raw" form.
`raw-data`	Specify the raw body data or set as "auto". The allowable content length is 1-8192 characters.
`reset`	Reset settings.
`timeout`	Specify the connection timeout in seconds. Range is 1-120 seconds, and the default is 10 seconds.
`view-settings`	View the current options of the curl.

64-bit SNMP MIB ifXTable support for higher speed interfaces

FortiADC 7.4.3 introduces the new SNMP MIB ifXTable with 64-bit counters to allow users to query 10 G statistics.

Hardware, VM, cloud platform, and browser support

This section lists the hardware models, hypervisor versions, cloud platforms, and web browsers supported by FortiADC 7.4.3. All supported platforms are 64-bit version of the system.

Supported Hardware:

- FortiADC 300D
- FortiADC 400D
- FortiADC 100F
- FortiADC 120F
- FortiADC 200F
- FortiADC 220F
- FortiADC 300F
- FortiADC 320F
- FortiADC 420F
- FortiADC 400F
- FortiADC 1000F
- FortiADC 1200F
- FortiADC 2000F
- FortiADC 2200F
- FortiADC 4000F
- FortiADC 4200F
- FortiADC 5000F

For more information on the supported hardware models, see FortiADC's Hardware Documents.

Supported hypervisor versions:

VM environment	Tested Versions
VMware	ESXi 3.5, 4.x, 5.0, 5.1, 5.5, 6.0, 6.5, 6.7, 7.0, 8.0
Microsoft Hyper-V	Windows Server 2012 R2, 2016 and 2019
KVM	Linux version 3.19.0 qemu-img v2.0.0, qemu-img v2.2
Citrix Xen	XenServer 6.5.0
Xen Project Hypervisor	4.4.2, 4.5
OpenStack	Pike
Nutanix	AHV
Proxmox VE	6.4

Supported cloud platforms:

- AWS (Amazon Web Services)
- Microsoft Azure
- GCP (Google Cloud Platform)
- OCI (Oracle Cloud Infrastructure)
- Alibaba Cloud
- IBM Cloud

For more information on the supported cloud platforms, see the FortiADC Private Cloud and Public Cloud documents.

Supported web browsers:

- Mozilla Firefox version 109
- Google Chrome version 110

We strongly recommend you set either of the Web browsers as your default Web browser when working with FortiADC. You may also use other (versions of the) browsers, but you may encounter certain issues with FortiADC's Web GUI.

Resolved issues

The following issues have been resolved in FortiADC 7.4.3 release. For inquiries about particular bugs, please contact Fortinet Customer Service & Support.

Bug ID	Description
1011313	Layer 4 virtual server traffic incorrectly matches when the VM is restarted or in the event of fail-over.
1011002	Request to add restriction to administrator username to forbid "daemon_admin" to be used as an admin name.
1009229	VMware clone of FortiADC image retains the MAC addresses of the original image when new MAC addresses should be assigned.
1007133	Request to hide hardware license information for VMs.
1007062	Httproxy crash caused by hidden field length limit in WAF input validation function.
1005985/1000626	Server health check scripts fail to work after upgrade to FortiADC 7.4.1.
1005261	Request to allow HTTP:persist() script function to be used in the HTTP_RESPONSE event.
1002301	DLP dictionaries incorrectly includes PK dictionaries.
1001137	Httproxy-ssl crash caused by connection release delay.
1000632	Memory leak in fcnacd daemon.
0999197	License upload page is outdated, still using GUI from version 5.x.
0997331	Virtual server port becomes intermittently unresponsive when no local certificate is selected in the real server SSL profile.
0997325	Timezone delay due to outdated zonefile.
0997194	Virtual server down due to thereal server pool server SSL profile using the same local certificate as the virtual server client SSL profile.
0996826	Hidden Field Input Validation is not working due to the HTML form action "#" being appended to the POST URL.
0982605	Configuring L7 Content Routing affects L4 Virtual Server with Content Routing enabled.
0973378	SLBL7 FTPS fails sometimes.
0961404	Request to show actual Web Filter license status.

Known issues

This section lists known issues in version FortiADC 7.4.3, but may not be a complete list. For inquiries about particular bugs, please contact Fortinet Customer Service & Support.

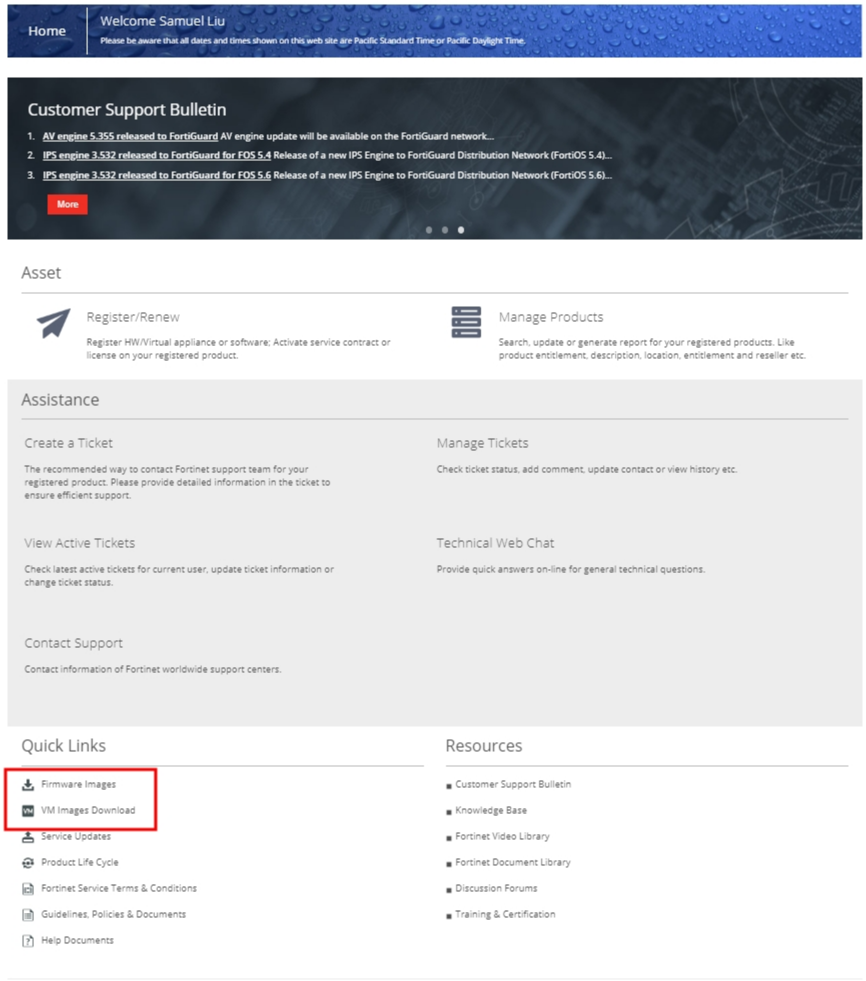
Bug ID	Description
1022505	After reloading the device, the GSLB server gateway status does not synchronize when the health check is enabled in the LLB gateway.

Image checksums

To verify the integrity of the firmware file, use a checksum tool and compute the firmware file's MD5 checksum. Compare it with the checksum indicated by Fortinet. If the checksums match, the file is intact.

MD5 checksums for Fortinet software and firmware releases are available from Fortinet Customer Service & Support. After logging in to the web site, near the bottom of the page, click the Firmware Image Checksums button. (The button appears only if one or more of your devices has a current support contract.) In the File Name field, enter the firmware image file name including its extension, then click Get Checksum Code.

Customer Service & Support image checksum tool



Welcome Samuel Liu

Please be aware that all dates and times shown on this web site are Pacific Standard Time or Pacific Daylight Time.

Customer Support Bulletin

1. AV engine 5.355 released to FortiGuard AV engine update will be available on the FortiGuard network...
2. IPS engine 3.532 released to FortiGuard for FOS 5.4 Release of a new IPS Engine to FortiGuard Distribution Network (FortiOS 5.4)...
3. IPS engine 3.532 released to FortiGuard for FOS 5.6 Release of a new IPS Engine to FortiGuard Distribution Network (FortiOS 5.6)...

Asset

Register/Renew

Register HW/Virtual appliance or software; Activate service contract or license on your registered product.

Manage Products

Search, update or generate report for your registered products. Like product entitlement, description, location, entitlement and reseller etc.

Assistance

Create a Ticket

The recommended way to contact Fortinet support team for your registered product. Please provide detailed information in the ticket to ensure efficient support.

Manage Tickets

Check ticket status, add comment, update contact or view history etc.

View Active Tickets

Check latest active tickets for current user, update ticket information or change ticket status.

Technical Web Chat

Provide quick answers on-line for general technical questions.

Contact Support

Contact information of Fortinet worldwide support centers.

Quick Links

Resources

Upgrade notes

This section includes upgrade information about FortiADC 7.4.3.

Supported upgrade paths

This section discusses the general paths to upgrade FortiADC from previous releases.

If you are upgrading to a version that is in a higher version level, you will need to upgrade to the nearest branch of the major level incrementally until you reach the desired version. For example, to upgrade from 7.1.1 to 7.4.0, you will follow the upgrade path below:

7.1.1 → 7.1.x → 7.2.x → 7.4.0

(wherein "x" refers to the latest version of the branch)

7.2.x to 7.4.x

Direct upgrade via the web GUI or the Console.

7.1.x to 7.2.x

Direct upgrade via the web GUI or the Console.

7.0.x to 7.1.x

Direct upgrade via the web GUI or the Console.

6.2.x to 7.0.x

Direct upgrade via the web GUI or the Console.

6.1.x to 6.2.x

Direct upgrade via the web GUI or the Console.

6.0.x to 6.1.x

Direct upgrade via the web GUI or the Console.

5.4.x to 6.0.x

Direct upgrade via the web GUI or the Console.

5.3.x to 5.4.x

Direct upgrade via the web GUI or the Console.

5.2.x to 5.3.x

Direct upgrade via the web GUI or the Console.

For more information on upgrading from versions earlier than 5.2.x, please see the Upgrade Instructions document for that version.

Upgrading a stand-alone appliance

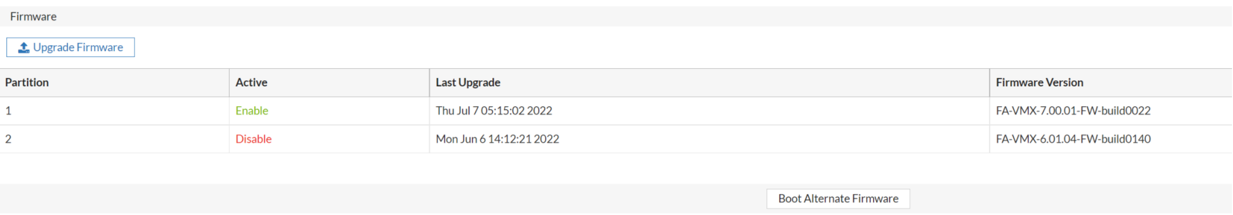
The following figure shows the user interface for managing firmware (either upgrades or downgrades). Firmware can be loaded on two disk partitions: the active partition and the alternate partition. The upgrade procedure:

- Updates the firmware on the inactive partition and then makes it the active partition.
- Copies the firmware on the active partition, upgrades it, and installs it in place of the configuration on the inactive partition.

For example, if partition 1 is active, and you perform the upgrade procedure:

- Partition 2 is upgraded and becomes the active partition; partition 1 becomes the alternate partition.
- The configuration on partition 1 remains in place; it is copied, upgraded, and installed in place of the configuration on partition 2.

This is designed to preserve the working system state in the event the upgrade fails or is aborted.



Partition	Active	Last Upgrade	Firmware Version
1	Enable	Thu Jul 7 05:15:02 2022	FA-VMX-7.00.01-FW-build0022
2	Disable	Mon Jun 6 14:12:21 2022	FA-VMX-6.01.04-FW-build0140

Before you begin:

- You must have super user permission (user admin) to upgrade firmware.
- Download the firmware file from the Fortinet Customer Service & Support website: https://support.fortinet.com/
- Back up your configuration before beginning this procedure. Reverting to an earlier firmware version could reset settings that are not compatible with the new firmware.
- You upgrade the alternate partition. Decide which partition you want to upgrade. If necessary, click **Boot Alternate Firmware** to change the active/alternate partitions.

To update the firmware:

1. Go to **System > Settings**.
2. Click the **Maintenance** tab.

3. Scroll to the **Firmware** section.
4. Click **Upgrade Firmware** to locate and select the firmware file.
5. Click ⬆ to upload the firmware and reboot.
   The system replaces the firmware on the alternate partition and reboots. The alternate (upgraded) partition becomes the active, and the active becomes the alternate.
6. Clear the cache of your web browser and restart it to ensure that it reloads the web UI and correctly displays all interface changes.

Upgrading an HA cluster

The upgrade page includes an option to upgrade the firmware on all nodes in an HA cluster from the primary node.

The following chain of events occur when you use this option:

1. The primary node pushes the firmware image to the member nodes.
2. The primary node notifies the member nodes of the upgrade, and takes on their user traffic during the upgrade.
3. The upgrade command is run on the member nodes, the systems are rebooted, and the member nodes send the primary node an acknowledgment that the upgrade has been completed.
4. The upgrade command is run on the primary node, and it reboots. While the primary node is rebooting, a member node assumes the primary node status, and traffic fails over from the former primary node to the new primary node.

After the upgrade process is completed, the system determines whether the original node becomes the primary node, according to the HA Override settings:

- If Override is enabled, the cluster considers the Device Priority setting. Both nodes usually make a second failover in order to resume their original roles.
- If Override is disabled, the cluster considers the uptime first. The original primary node will have a smaller uptime due to the order of reboots during the firmware upgrade. Therefore, it will not resume its active role. Instead, the node with the greatest uptime will remain the new primary node. A second failover will not occur.

Before you begin, do the following:

1. Make sure that you have super user permission (user admin) on the appliance whose firmware you want to upgrade.
2. Download the firmware file from the Fortinet Customer Service & Support website:
   https://support.fortinet.com/
3. Back up your configuration before beginning this procedure. Reverting to an earlier version of the firmware could reset the settings that are not compatible with the new firmware.
4. Verify that the cluster node members are powered on and available on all of the network interfaces that you have configured. (Note: If required ports are not available, HA port monitoring could inadvertently trigger an additional failover, resulting in traffic interruption during the firmware update.)
5. You upgrade the alternate partition. Decide which partition you want to upgrade. If necessary, click **Boot Alternate Firmware** to change the active/alternate partitions.

To update the firmware for an HA cluster:

1. Log into the web UI of the *primary* node as the `admin` administrator.
2. Go to **System > Settings**.
3. Click the **Maintenance** tab.
4. Scroll to the **Upgrade Firmware** button.
5. Click **Choose File** to locate and select the file.
6. Enable the **HA Cluster Upgrade**.
7. Click ⬆ to upload the firmware and start the upgrade process.

After the new firmware has been installed, the system reboots.

When you update software, you are also updating the web UI. To ensure the web UI displays the updated pages correctly:

- Clear your browser cache.
- Refresh the page.

In most environments, press Ctrl+F5 to force the browser to get a new copy of the content from the web application. See the Wikipedia article on browser caching issues for a summary of tips for many environments: https://en.wikipedia.org/wiki/Wikipedia:Bypass_your_cache.

Special notes and suggestions

7.2.3

- The real server auto-populate feature is currently supported only in FortiADC version 7.2.3. Upgrading from version 7.2.3 to 7.4.0/7.4.1 will cause auto-populated real server related configuration loss, and may cause other unexpected behavior.
  Support for real server auto-population will be extended to later versions in the next release.

7.0.2/7.1.x

- After upgrading to 7.0.2/7.1.x, in Virtual Machine HA environments where both nodes have been installed with certificate embedded licenses you must reinstall those licenses. As some backend certificate files would have been synchronized and overwritten by the HA Peer (due to an existing bug), the certificate file would not be recoverable. Reinstalling the certificate embedded licenses is required to ensure they would work properly where they are needed, such as in ZTNA or FortiSandbox Cloud.

7.0.0

- When deploying the new GSLB based on FortiADC 7.0.0, the verify-CA function will be enabled by default.

6.2.2

- To use the SRIOV feature, users must deploy a new VM.

6.2.0

- In version 6.2.0, the default mode of QAT SSL has been changed to polling.

6.1.4

- Before downgrading from 6.1.4, ensure the new L7 TCP or L7 UDP application profiles are deleted or changed to a profile type that is supported in the downgrade version. Otherwise, this will cause the cmdb to crash.

5.2.0-5.2.4/5.3.0-5.3.1

- The backup configuration file in versions 5.2.0-5.2.4/5.3.0-5.3.1 containing the certificate configuration might not be restored properly (causing the configuration to be lost). After upgrading, please discard the old 5.2.x/5.3.x configuration file and back up the configuration file in the upgraded version again.

www.fortinet.com

Copyright© 2024 Fortinet, Inc. All rights reserved. Fortinet®, FortiGate®, FortiCare® and FortiGuard®, and certain other marks are registered trademarks of Fortinet, Inc., and other Fortinet names herein may also be registered and/or common law trademarks of Fortinet. All other product or company names may be trademarks of their respective owners. Performance and other metrics contained herein were attained in internal lab tests under ideal conditions, and actual performance and other results may vary. Network variables, different network environments and other conditions may affect performance results. Nothing herein represents any binding commitment by Fortinet, and Fortinet disclaims all warranties, whether express or implied, except to the extent Fortinet enters a binding written contract, signed by Fortinet's General Counsel, with a purchaser that expressly warrants that the identified product will perform according to certain expressly-identified performance metrics and, in such event, only the specific performance metrics expressly identified in such binding written contract shall be binding on Fortinet. For absolute clarity, any such warranty will be limited to performance in the same ideal conditions as in Fortinet's internal lab tests. Fortinet disclaims in full any covenants, representations, and guarantees pursuant hereto, whether express or implied. Fortinet reserves the right to change, modify, transfer, or otherwise revise this publication without notice, and the most current version of the publication shall be applicable.