FortiSIEM - Configuring CA Certificates

Version 5.3.1

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

FORTINET VIDEO GUIDE
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/support-and-training/training.html

NSE INSTITUTE
https://training.fortinet.com

FORTIGUARD CENTER
https://fortiguard.com/

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

FEEDBACK
Email: techdoc@fortinet.com

TABLE OF CONTENTS

Change Log

Date	Change Description
09/26/2019	Initial release of Configuring CA Certificates.
11/21/2019	Release of Configuring CA Certificates for 5.2.6.
03/30/2020	Release of Configuring CA Certificates for 5.3.0.

Configuring FortiSIEM for HTTPS Communication Using Public CA Certificates

This document describes how to configure various FortiSIEM nodes for HTTP(S) communication using public CA certificates.

Internal HTTPS Communication Using CA Certificates

This section addresses HTTP(S) communication within various FortiSIEM nodes using public CA certificates.

Prerequisites

The instructions in this document assume that you have completed the following tasks:

1. Setup FQDNs for Supervisor and Worker nodes.
2. Setup FQDNs for Collectors if you plan on using Linux and/or Windows Agents.
3. Configure Collector hostname to be FQDN and then register them using FQDN.
4. Obtain Certificates issued and signed by a well-known Certifying Authority (CA)
   a. If using wildcard certificates, then the same certificate can be used in Super, Workers, and Collectors as long as their FQDN is a direct subdomain of the wildcard domain.
   b. If using per-node certificates, then the certificate's subject name should match the FQDN of the node for Supervisor, Workers, and Collectors.
5. Made sure that collectors can reach Supervisor and Worker nodes using their respective FQDNs.
6. If you have Linux and/or Windows Agents, then also make sure that they can reach the Collectors using their respective FQDN.
7. In the FortiSIEM GUI, **Admin > Settings > Worker Upload** lists the worker addresses using worker FQDNs.

Collector to Supervisor HTTPS Communication

1. On the Supervisor, complete these steps:
   a. Copy your CA certificates to the `/etc/httpd/conf.d` directory.
   b. Modify the `/etc/httpd/conf.d/ssl.conf` file by changing the following settings to point to these certificates:
      - SSLCertificateFile <ca-certificate-file>
      - SSLCertificateKeyFile <ca-certificate-key-file>
      - SSLCertificateChainFile <ca-certificate-chain-file>
2. Before registering the collectors, change the following setting in the `/opt/phoenix/config/collector_config_template.txt` file on the Supervisor:
   `http_client_verify_peer=yes`
3. On each Collector, before you register it, change the following setting in the `/opt/phoenix/config/phoenix_config.txt` file:
   `http_client_verify_peer=yes`
4. Log in to the Collector and verify the Supervisor's certificate using the `curl` command. For example:

```
curl -vv https://<Supervisor-FQDN>
* Rebuilt URL to: https://<Supervisor-FQDN>/
* Trying <IP>...
* TCP_NODELAY set
* Connected to <Supervisor-FQDN> (<IP>) port 443 (#0)
* ALPN, offering h2
* ALPN, offering http/1.1
* Cipher selection: ALL:!EXPORT:!EXPORT40:!EXPORT56:!aNULL:!LOW:!RC4:@STRENGTH
* successfully set certificate verify locations:
* CAfile: /etc/ssl/cert.pem
CApath: none
* TLSv1.2 (OUT), TLS handshake, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Server hello (2):
* TLSv1.2 (IN), TLS handshake, Certificate (11):
* TLSv1.2 (IN), TLS handshake, Server key exchange (12):
* TLSv1.2 (IN), TLS handshake, Server finished (14):
* TLSv1.2 (OUT), TLS handshake, Client key exchange (16):
* TLSv1.2 (OUT), TLS change cipher, Client hello (1):
* TLSv1.2 (OUT), TLS handshake, Finished (20):
* TLSv1.2 (IN), TLS change cipher, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Finished (20):
* SSL connection using TLSv1.2 / ECDHE-RSA-AES256-GCM-SHA384
* ALPN, server did not agree to a protocol
* Server certificate:
* subject: C=<Country>; ST=<State>; L=<Location>; O=<Organization>; OU=<OU>; CN=*.<Domain>
* start date: Jul 26 00:00:00 2019 GMT
* expire date: Jul 30 12:00:00 2021 GMT
* subjectAltName: host "<Supervisor-FQDN>" matched cert's "*.<Domain>"
* issuer: C=<Country>; O=<CA>; OU=<CA-Domain>; CN=<CA name>
* SSL certificate verify ok.
> GET / HTTP/1.1
> Host: <Super-FQDN>
```

If `curl` reports that the verification of the SSL certificate fails, then check your certificate for a mismatch between the `<Supervisor-FQDN>` and the subject name.

5. Register the Collector with the Supervisor using the `phProvisionCollector` command.
   Example usage: `phProvisionCollector --add <Organization-user-name> <Organization-user-password> <Supervisor-FQDN> <Organization-name> <Collector-name>`

Make sure to register the collector using the Supervisor's FQDN, otherwise registration will fail.

Collector to Worker HTTPS Communication

1. On each Worker node, perform the following steps:
   a. Copy your CA certificates to `/etc/httpd/conf.d` directory.
   b. Modify `/etc/httpd/conf.d/ssl.conf` by changing the following settings to point to these certificates:
      - SSLCertificateFile <ca-certificate-file>
      - SSLCertificateKeyFile <ca-certificate-key-file>
      - SSLCertificateChainFile <ca-certificate-chain-file>
2. On Supervisor GUI, go to **Admin > Settings > Worker Upload** and list the FQDNs for each worker.
3. Use `curl` to test connectivity to workers and check that `curl` verifies the certificate to be OK. For example:

```
curl -vv https://<Worker-FQDN>
* Rebuilt URL to: https://<Worker-FQDN>/
* Trying <IP>...
* TCP_NODELAY set
* Connected to <Worker-FQDN> (<IP>) port 443 (#0)
* ALPN, offering h2
* ALPN, offering http/1.1
* Cipher selection: ALL:!EXPORT:!EXPORT40:!EXPORT56:!aNULL:!LOW:!RC4:@STRENGTH
* successfully set certificate verify locations:
* CAfile: /etc/ssl/cert.pem
CApath: none
* TLSv1.2 (OUT), TLS handshake, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Server hello (2):
* TLSv1.2 (IN), TLS handshake, Certificate (11):
* TLSv1.2 (IN), TLS handshake, Server key exchange (12):
* TLSv1.2 (IN), TLS handshake, Server finished (14):
* TLSv1.2 (OUT), TLS handshake, Client key exchange (16):
* TLSv1.2 (OUT), TLS change cipher, Client hello (1):
* TLSv1.2 (OUT), TLS handshake, Finished (20):
* TLSv1.2 (IN), TLS change cipher, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Finished (20):
* SSL connection using TLSv1.2 / ECDHE-RSA-AES256-GCM-SHA384
* ALPN, server did not agree to a protocol
* Server certificate:
* subject: C=<Country>; ST=<State>; L=<Location>; O=<Organization>; OU=<OU>; CN=*.<Domain>
* start date: Jul 26 00:00:00 2019 GMT
* expire date: Jul 30 12:00:00 2021 GMT
* subjectAltName: host "<Worker-FQDN>" matched cert's "*.<Domain>"
* issuer: C=<Country>; O=<CA>; OU=<CA-Domain>; CN=<CA name>
* SSL certificate verify ok.
> GET / HTTP/1.1
> Host: <Worker-FQDN>
```

Linux Agent to Supervisor and Collector HTTPS Communication

1. Set up CA certificates on Supervisor as described in Collector to Supervisor HTTPS Communication.
2. On each Collector node, perform the following steps if you have not done this already for Windows Agent:
   a. Copy your CA certificates to the `/etc/httpd/conf.d` directory.
   b. Modify the `/etc/httpd/conf.d/ssl.conf` file by changing the following settings to point to these certificates:
      - SSLCertificateFile <ca-certificate-file>
      - SSLCertificateKeyFile <ca-certificate-key-file>
      - SSLCertificateChainFile <ca-certificate-chain-file>
3. Configure the Collector FQDN as the `hostname` using `vami_config_net`. Similarly, configure the Collector name in the GUI to be FQDN.
4. Register the Collector, using FQDN as the Collector name.
5. Use `curl` to test connectivity to Collectors via FQDN and check that `curl` verifies the certificate to be OK. For example:

```
curl -vv https://<Collector-FQDN>
* Rebuilt URL to: https://<Collector-FQDN>/
* Trying <IP>...
* TCP_NODELAY set
* Connected to <Collector-FQDN> (<IP>) port 443 (#0)
* ALPN, offering h2
* ALPN, offering http/1.1
* Cipher selection: ALL:!EXPORT:!EXPORT40:!EXPORT56:!aNULL:!LOW:!RC4:@STRENGTH
* successfully set certificate verify locations:
* CAfile: /etc/ssl/cert.pem
CApath: none
* TLSv1.2 (OUT), TLS handshake, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Server hello (2):
* TLSv1.2 (IN), TLS handshake, Certificate (11):
* TLSv1.2 (IN), TLS handshake, Server key exchange (12):
* TLSv1.2 (IN), TLS handshake, Server finished (14):
* TLSv1.2 (OUT), TLS handshake, Client key exchange (16):
* TLSv1.2 (OUT), TLS change cipher, Client hello (1):
* TLSv1.2 (OUT), TLS handshake, Finished (20):
* TLSv1.2 (IN), TLS change cipher, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Finished (20):
* SSL connection using TLSv1.2 / ECDHE-RSA-AES256-GCM-SHA384
* ALPN, server did not agree to a protocol
* Server certificate:
* subject: C=<Country>; ST=<State>; L=<Location>; O=<Organization>; OU=<OU>; CN=*.<Domain>
* start date: Jul 26 00:00:00 2019 GMT
* expire date: Jul 30 12:00:00 2021 GMT
* subjectAltName: host "<Collector-FQDN>" matched cert's "*.<Domain>"
* issuer: C=<Country>; O=<CA>; OU=<CA-Domain>; CN=<CA name>
* SSL certificate verify ok.
> GET / HTTP/1.1
> Host: <Collector-FQDN>
```

6. Register the Linux Agent with the Supervisor using the Supervisor's FQDN.
   Linux Agents:
   When installing Linux Agent, you must add a `-v` option to verify the Supervisor's certificates.

```
./fortisiem-linux-agent-installer-5.3.1..sh -s <Supervisor-FQDN> -i
<Organization-Id> -o <Organization-Name> -u <Agent-User> -p <Agent-Password> -v
```

Linux agent installer options:

- `-c` - CA Certificate bundle file (Optional)
- `-h` - Show this message
- `-i` - Organization Id
- `-n` - Hostname where agent is installed (Optional)
- `-o` - Organization
- `-p` - Agent Registration Password
- `-s` - Super IP/HostName
- `-u` - Agent Registration User
- `-v` - Verify Super and Collector SSL Certificate during TLS handshake (Optional)

Windows Agent to Supervisor and Collector HTTPS Communication

1. Set up CA certificates on Supervisor as described in Collector to Supervisor HTTPS Communication.
2. On each Collector node, perform the following steps if you have not done this already for Linux Agent:
   a. Copy your CA certificates to the `/etc/httpd/conf.d` directory.
   b. Modify the `/etc/httpd/conf.d/ssl.conf` file by changing the following settings to point to these certificates:
      - SSLCertificateFile <ca-certificate-file>
      - SSLCertificateKeyFile <ca-certificate-key-file>
      - SSLCertificateChainFile <ca-certificate-chain-file>
3. Configure the Collector FQDN as the `hostname` using `vami_config_net`. Similarly, configure the Collector name in the GUI to be FQDN.
4. Register the Collector, using FQDN as the Collector name.

To install Windows Agent, follow the instructions in the Windows Agent Installation Guide and modify the `InstallSettings.xml` file with `<SSLCertificate>check</SSLCertificate>` instead of `ignore`.

External HTTPS Communication Using CA Certificates

This section addresses HTTP(S) communication from FortiSIEM to external systems or external systems to FortiSIEM

- Java-based HTTPS Communication From FortiSIEM to External Websites
- Event Forwarding from FortiSIEM to External System Using Syslog/TLS

Java-based HTTPS Communication From FortiSIEM to External Websites

This section addresses the following use cases:

- Communication with External Threat Intelligence websites
- Communication with Ticketing systems e.g. ServiceNow

1. Download the certificate from the desired third-party website and save it to a file.
   For example, on the Unix platform, use `openssl` to download certificate as shown below. The `FQDN` is the server name of the website to which you would like to download the certificate:
   ```
   openssl s_client -connect <FQDN>:<port> -servername <FQDN> < /dev/null | sed -ne '/-BEGIN CERTIFICATE-/,/-END CERTIFICATE-/p' > <filename>.crt
   ```
   Using google.com as an example:
   ```
   openssl s_client -connect google.com:443 -servername google.com < /dev/null | sed -ne '/-BEGIN CERTIFICATE-/,/-END CERTIFICATE-/p' > public.crt
   ```
2. Use the Java `keytool` command to import the certificate to Glassfish and use `changeit` as the password when prompted. You must replace the *`sample_alias`* with any alias you would like to use for this certification and *`filename`* to the certificate file downloaded:
   ```
   keytool -import -trustcacerts -alias <sample_alias> -keystore /opt/glassfish/domains/domain1/config/cacerts.jks -file <filename>.crt
   keytool -import -trustcacerts -alias <sample_alias> -keystore /opt/glassfish/domains/domain1/config/keystore.jks -file <filename>.crt
   ```
   For example:
   ```
   keytool -import -trustcacerts -alias google -keystore /opt/glassfish/domains/domain1/config/cacerts.jks -file public.crt
   keytool -import -trustcacerts -alias google -keystore /opt/glassfish/domains/domain1/config/keystore.jks -file public.crt
   ```

Event Forwarding from FortiSIEM to an External System Using syslog/TLS

FortiSIEM's SSL library can validate an external system's certificate if it is signed by a public CA.

If the external system wants to verify the FortiSIEM node's certificate, then you need to add the following certificate and key to the `phoenix_config.txt` file of the FortiSIEM nodes forwarding the event.

```
[BEGIN phEventForwarder]
…
tls_certificate_file= #/opt/phoenix/bin/.ssh/my_cert.crt
tls_key_file= #/opt/phoenix/bin/.ssh/my_cert.key
[END]
```