Using TLS Certificates as a Client

Overview

You can set up TLS between the JFrog Platform and external services by trusting external service certificates. JFrog services will not allow a SSL/TLS connection with an external service without a validation of the trusted CA certificate.

For example, you may want to connect to remote repositories, your LDAPS, internal proxy, OAuth server, or other external services over HTTPS. To that end, you may need to trust a certificate (for example, a self signed certificate) that was not signed by a trusted Certificate Authority (CA) and is used by the external service.

Page Contents

Trusting a Self-Signed Certificate or a New CA

To trust a new certificate, add the certificate to the `$JFROG_HOME/<product>/var/etc/security/keys/trusted` directory of every service that needs to trust it.

Alternatively, you an also add the certificate to each application's KeyStore. For example, to add a certificate into the JFrog Artifactory KeyStore, you can add it directly to the host's JVM's trusted KeyStore.

> **HA Setup**
>
> For HA setup, you need to add the certificate to every node's trusted directory or KeyStore. The Certificates are not propagated between HA nodes automatically.

Trusting a Self-Signed Certificate in Xray Instances/Nodes

When an Xray instance/node is configured to go through an **SSL proxy** that uses a **self-signed certificate**, you may encounter the following issue when performing tasks such as an online database sync:

```
2021-07-20T14:47:47.500Z [33m[jfxr ][0m [1m[31m[ERROR][0m [c080f44e606d159 ] [samplers:91
] [main               ] Failed to read response from jxrayUrl. Error: Get "https://jxray.jfrog.io/api/v1
/system/ping": x509: certificate signed by unknown authority
```

1. To overcome this issue, you will need to import the **Proxy certificate** into **each** Xray instance/pod by placing it under the following path within the Xray machine/container/pods:`/etc/ssl/certs/`.
2. Next, you will need to restart Xray.
   The path shown above is the default directory used by Go applications (such as Xray) when importing SSL certificates.

Downloading a Certificate

To download/acquire the certificate(s) of the SSL secured server, use the following command:

```
openssl s_client -connect <secure authentication server IP and port> -showcerts < /dev/null > server.crt
```

Examples

RED HAT CDN

```
openssl s_client -connect cdn.redhat.com:443 -showcerts < /dev/null > server.crt
```

LDAP or Active Directory

```
openssl s_client -connect the.ldap.server.net:636 -showcerts < /dev/null > server.crt
```

OAuth (Use the Authorization URL). For example, with GitHub

```
openssl s_client -connect github.com:443 -showcerts < /dev/null > server.crt
```