Web Services and Applications Deployment Guide

Transport Layer Security (TLS)

5/10/2025

Transport Layer Security (TLS)

Contents

Configuring TLS between Web Services and Configuration Server

Web Services can use a secured Transport Layer Security (TLS) connection mechanism to connect to Configuration Server. When configured, Web Services connects to a secure port on Configuration Server, verifies the server's authority, and encrypts/decrypts network traffic. You can configure secured connections to Configuration Server in the following ways:

- Minimal configuration
- Validate the certificate against the CA

Prerequisites

Before configuring Web Services, make sure the Configuration Server secure port is configured as described in Introduction to Genesys Transport Layer Security in the *Genesys Security Deployment Guide* and that all certificates for server host and the certificate authority are configured and available.

Minimal configuration

Web Services does not check the server's certificate against the Certificate Authority, but all traffic is encrypted. To configure Web Services with minimal configuration, all you need to do is configure a connection to a secured port on Configuration Server. You can do this using **either** of the following methods:

- For the initial connection to Configuration Server, set the **tlsEnabled** option to `true` in the **environment.yaml** file. This creates a secured connection to Configuration Server the first time Web Services starts.
- For an environment that is already configured with Configuration Manager synchronization enabled, you can make changes with Configuration Manager as described in the Genesys Security Deployment Guide.

Validate the certificate against the CA

In order to support the client-side certificate check, Web Services needs the public key for the Certificate Authority (CA). Web Services supports the PEM and JKS key storage formats, but recommends using JKS.

Complete the steps below to validate the certificate against the CA.

> ### Important
> The steps described in this procedure are meant to be an example for developers and should not be used in production. For a production environment, you should follow your own company's security policies for creating and signing certificates.

Start

1. If you plan to use a JKS file, you can generate if from a PEM file by importing the PEM certificate, as shown here:

   ```
   keytool -importcert -file ca_cert.pem -keystore ca_cert.jks
   ```

2. Once you have the **ca_cert.jks** file, place it in a location available from your Web Services host, such as:
   - A local folder on the Web Services host
   - A network share
3. Configure the following options in the serverSettings section of the **application.yaml** file:
   - For a PEM file, set caCertificate to the location of the file. For example:

     ```
     caCertificate: /opt/ca_cert.pem
     ```

   - For a JKS file, set caCertificate to the location of the file and set jksPassword to the password for the key storage. For example:

     ```
     caCertificate: /opt/ca_cert.jks
     jksPassword: pa$$word
     ```

End

For TLS for all other servers, it uses the configuration data from Configuration Server.

Next Step

- Back to Configuring security