Genesys Security Deployment Guide

OpenSSL Certificates

12/15/2025

OpenSSL Certificates

Contents

Use OpenSSL certificates if you intend to run any applications that might require secure connections on UNIX. However, if you intend to run all of your applications on Windows, Genesys strongly recommends that you use Windows Certificate Services to generate certificates.

Supported Certificate and Key File Formats

- X.509
- PKCS#8
- DER (`.cer`)
- PEM (`.pem`, `.cer`)
- PKCS#7
- PKCS#12

Java/PSDK-based Applications

If you are going to be installing certificates for Java/PSDK-based applications on UNIX, such as Universal Contact Server (UCS), you will have to convert the private-key files generated by OpenSSL to a format compatible with those applications. The conversion must be done after they are generated but before they are installed, as given in the procedure below (see step 3).

Pre-requisites

The scripts that are used to generate certificates require the OpenSSL toolkit, which you can obtain from the OpenSSL Project website.

You can also obtain build binaries of OpenSSL tools for the Windows operating system from here.

Generation and Installation

To create and install certificates using OpenSSL, complete the following steps:

1. Set up a Certification Authority (CA). **[+] Show steps**

> **Important**
>
> Genesys recommends that you use only one CA instance for your entire call center environment.

a. Create a CA directory in which CA files—scripts, configuration files, and generated certificates—will be stored.

b. Copy the **create_ca.sh** and **create_cert.sh** scripts from the installation package to the CA directory that you just created. Make sure that these scripts have executable permissions.

c. Run the **create_ca.sh** script from the **bash** shell by specifying the proper parameters (see the following table) in the following command line:

```
create_ca.sh [-keySz KEY_SIZE] [-time VALID_TIME] -CN COMMON_NAME [-E EMAIL] [-OU
ORG_UNIT] [-O ORGANIZATION] [-L LOCALITY] [-S STATE] [-C COUNTRY]
```

For example:

```
create_ca.sh -CN “Basic Certification Authority” -E “youremail@yourdomain.com” -OU
“Department” -O “Genesys Telecommunication Labs” -L “Daly City” -S CA -C US
```

Parameter	Description
KEY_SIZE	(Optional) The size, in bits, of the CA private key. The default value is 2048 bits.
VALID_TIME	(Optional) The amount of time, in days, that the CA is valid. The default value is 365 days.
COMMON_NAME	(Mandatory) The name of the CA.
EMAIL	(Optional) The e-mail address of the person who is responsible for this CA.
ORG_UNIT	(Optional) The name of the organization unit that is responsible for this CA.
ORGANIZATION	(Optional) The name of the organization that is responsible for this CA.
LOCALITY	(Optional) The name of the city.
STATE	(Optional) The name of the state or region.
COUNTRY	(Optional) The two-letter abbreviation for the country.

Certificate Authority Files

After successful script execution, the following data structure is created:

- **ca_conf**—This directory contains the following files:
  - **ca_cert.pem**—The CA self-signed certificate file.

> **Important**
> You must copy this file to each computer that will host Genesys components that might require secure data exchange, even if client certificates are not required.

- **ca_priv_key.pem**—The CA private key.
  This file is used to sign all certificates that this CA issues. This file must be read-only, and it must be readable only by the CA administrator account.
- **ca.db**—The internal CA database used by the OpenSSL toolkit.
- **serial.num**—The internal CA file that contains the serial number of the next generated certificate. The serial number is a unique identifier of the certificate that the CA issues.
- **ca.conf**—The internal CA configuration file.

- **repository**—This directory contains the files that this CA generates.

2. Generate certificates as required. **[+] Show steps**

> **Important**
> Genesys recommends that you use the same CA to generate all certificates for a particular environment.

To generate a certificate for a particular host computer:

a. Go to the CA directory in which the CA files are stored.

b. Run the **create_cert.sh** script from the **bash** shell by specifying the parameters (see the following table) in the following command line:

```
create_cert.sh [-keySz KEY_SIZE] [-time VALID_TIME] -host HOST_NAME -CN COMMON_NAME
 [-OU ORG_UNIT] [-O ORGANIZATION] [-L LOCALITY] [-S STATE] [-C COUNTRY]
```

For example:

```
create_cert.sh -host myHOST.domain1.domain2.com -CN myWorkstation
```

Parameter	Description
KEY_SIZE	(Optional) The size, in bits, of the host private key. The default value is 2048 bits.
VALID_TIME	(Optional) The amount of time, in days, that the certificate is valid. The default value is 100

Parameter	Description
	days.
HOST_NAME	(Mandatory) The full name of the DNS host.
COMMON_NAME	(Mandatory) The name of the host.
ORG_UNIT	(Optional) The name of the organization unit.
ORGANIZATION	(Optional) The name of the organization.
LOCALITY	(Optional) The name of the city.
STATE	(Optional) The name of the state or region.
COUNTRY	(Optional) The two-letter abbreviation for the country.

Host Certificate Files

After successful script execution, the following files are created in the repository directory:

- **<serial_#>_<host_name>_cert.pem**—The host certificate for UNIX.
- **<serial_#>.pem**—The auxiliary file for certificate generation for UNIX.
- **<serial_#>_<host_name>_priv_key.pem**—The host private key for UNIX.
- **<serial_#>_<host_name>_cert.pfx**—The PKCS (Public-Key Cryptography Standards) #12 file format, private key, and certificate for Windows.

  where:

  - **<serial_#>** is the serial number of the generated certificate. This number is unique for all certificates that this CA generates.
  - **<host_name>** is the name of your host computer, which is the first part of the full DNS host name.

3. If you are installing certificates on any Java-based PSDK applications, such as Universal Contact Server, convert the private key file to PKCS #8 format. Use the following command:

```
openssl pkcs8 -topk8 -nocrypt -in <serial_#>_<host_name>_priv_key.pem -out
<serial_#>_<host_name>_priv_key_NEW.pem
```

The converted file **<serial_#>_<host_name>_priv_key_NEW.pem** will be compatible with Java-based PSDK applications.

4. Install the certificates. **[+] Show steps**

> **Important**

- If you are using mutual TLS, you must install the CA self-signed certificate file, **ca_cert.pem**, and at least one certificate issued by this CA on each computer that hosts Genesys applications that might require secure data exchange.
- If you are using simple TLS, you need to install only the CA self-signed certificate file, **ca_cert.pem**, on each computer that hosts Genesys applications that might require secure data exchange. You do not need to install certificates on those hosts that are not running any of the server applications.

On UNIX

a. Copy the **ca_cert.pem** file to the computer.

b. Copy the certificate and private key files to a local directory on the computer, as follows:
   - For Java-based PSDK applications: **<serial_#>_<host_name>_cert.pem** and **<serial_#>_<host_name>_priv_key_NEW.pem**
   - For other applications: **<serial_#>_<host_name>_cert.pem** and **<serial_#>_<host_name>_priv_key.pem**

c. Make sure that these files are readable by all Genesys applications that are running on this host computer.

Warning

The **<serial_#>_<host_name>_priv_key.pem** file contains critical security information. Make sure it can only be accessed by personnel authorized to work with this type of information.

When you configure an application to support secure data exchange on UNIX:

- The full path to the **ca_cert.pem** file is copied to the **Trusted CA** text box of the **Certificate properties**.
- The full path to the **<serial_#>_<host_name>_cert.pem** file is copied to the **Certificate** text box of the **Certificate properties**.
- The full path to the **<serial_#>_<host_name>_priv_key.pem** or **<serial_#>_<host_name>_priv_key_NEW.pem** file is copied to the **Certificate Key** text box of the **Certificate properties**.

For more information, see Genesys TLS Configuration.

On Windows

> **Important**
>
> For server applications, the certificates must be installed under the Local Computer account. For client applications, the certificates must be installed under the Current User account. For more information, see Managing Certificates in MMC..

a. From the Windows Start menu, select **Run**, and then execute the `mmc` command to start the Microsoft Management Console (MMC).
b. On the left pane of MMC, click the **Certificates** folder. (If there is no **Certificates** folder on the left pane, see Managing Certificates in MMC.
c. Right-click the **Trusted Root Certification Authorities** folder, and select **All Tasks > Import** from the shortcut menu. This starts the Certificate Import Wizard.
d. On the first Wizard page, click **Next**.
e. On the **File to Import** page, type the full name of the **ca_cert.pem** file that was created during the CA setup, and then click **Next**.
f. On the **Certificate Store** page, select **Place all certificates in the following store**. Make sure that the **Certificate store** text box is set to **Trusted Root Certification Authorities**. Click **Next**.
g. Click **Finish**.
h. On the left pane, click the **Certificates** folder.
i. On the left pane, right-click the **Personal** folder, and select **All Tasks > Import** from the shortcut menu. This starts the Certificate Import Wizard.
j. On the first Wizard page, click **Next**.
k. On the **File to Import** page, type the full name of the **<serial_#>_<host_name>_cert.pfx** file that was created during certificate generation. Click **Next**.
l. On the **Password** page, click **Next**. The host certificates in PKSC #12 format are generated with an empty password.
m. On the **Certificate Store** page, select **Place all certificates in the following store**. Make sure that the **Certificate store** text box is set to `Personal`. Click **Next**.
n. Click **Finish**.
o. Press **F5** to update the MMC view.
p. On the left pane, select **Certificates > Personal > Certificates**.
q. On the right pane, locate the imported certificate in the list, and double-click it.
r. In the **Certificate** dialog box, click the **Details** tab.
s. To view the certificate thumbprint, select **Thumbprint** from the list. The thumbprint, consisting of a string of hexadecimal digits, appears in the lower part of the dialog box.