Genesys Security Deployment Guide

Configuration of Secure Connections

12/14/2025

Configuration of Secure Connections

Contents

After you generate the certificates and install them on the host computers, you must configure Genesys applications to use them on the connections that need to be secure. (By default, connections between Genesys applications are not secure.)

Important

- The instructions in this chapter assume that you are adding Genesys Transport Layer Security (TLS) to existing connections of your Genesys 8.x environment—that is, that your applications have already been installed, properly configured, and associated with hosts and with each other. For information about configuring new hosts, applications, and associations between them, see the Framework Deployment Guide.
- If you are using Genesys components of previous releases, you must upgrade them to release 8.x before you can configure secure connections between them.
- Some components require additional steps to complete the configuration of secure connections. These steps are provided in the Deployment Guide for your particular product or component.
- An application can be both a server and a client application. In this case, they are configured both as a TLS server and a TLS client application. In this case, the same security certificate is used as both a server certificate and a client certificate.

Standard Configuration

You configure all secure connections in the same way, regardless of the types of participating client and server applications. The only exceptions are:

- 8.x releases of Genesys client applications that run on Windows and support TLS do not require client security certificates.
- Connections with Java-based applications—See Configuring Secure Connections to Java/PSDK-Based Applications.
- Configuration Server connections—See Configuring Secure Connections to Configuration Server.
- Local Control Agent connections—See Configuring Secure Connections Between LCA and Solution Control Server.
- Genesys Deployment Agent connections—See Configuring Secure Connections Between Genesys Deployment Agent and its Clients.
- High Availability synchronization connections-See Configuring High Availability Synchronization Connections.

Certificate Chains

Starting with release 8.1.3, Genesys Security Pack on UNIX supports security certificate chains, sending out the intermediate certificates with the root certificate. To enable this support, specify the multiple certificates in a comma-delimited list in the **Certificate** field when configuring TLS. The

certificates are sent in the order in which they are specified.

Multiple Trusted CAs

Starting with release 8.0.0, Genesys Security Pack on UNIX supports multiple Trusted CA certificates for TLS connections. To enable this support, create a PEM file listing all of the certificates issued by the Trusted CAs. Then specify the full path to this file in the **trusted-ca** field when configuring TLS. As security circumstances and requirements change, you can modify the file by adding and removing certificates, or completely replace it by specifying a single Trusted CA.

Configuration Steps

To configure secure connections, perform the following steps:

1. For all server applications, configure a new or existing server port for secure connections. A port must be secure before you can configure a secure connection to that port. **[+] Show steps**

Server-type applications that support Genesys TLS also support multiple server ports. This enables you to set up secure communications on only those connections for which security is considered necessary, rather than all server connections at the same time.

> **Important**
>
> If you intend to use the secure data exchange capabilities on connections to a specific server, Genesys recommends that you configure a new port for such secure connections, and that you leave the existing unsecured port intact for connections that do not require security.

Secure Port on TLS Server Application

To configure a secure port on a TLS server applications, do the following:

a. In Genesys Administrator, click the **Provisioning** tab and navigate to the folder containing the server application.

b. Select the server application and click the **Configuration** tab.

c. In the **Server Info** section, click **Add** in the **Listening Ports** table. The **Port Info** dialog box appears.

d. In the **Port Info** dialog box, on the **General** tab:

- In the **ID** box, enter the port ID.
- In the **Port** box, enter the number of the new port.
- In the **Connection Protocol** box, select the connection protocol, if necessary.
- In the **Select Listening Mode** box, select **Secured**.
- Click **OK**.

e. Click **Save**, **Save & Close**, or **Save & New** to save the new configuration.

Auto-Detect Port on Configuration Server

To configure an Auto-Detect port in the Configuration Server application, so that clients can connect securely to Configuration Server, do the following:

a. In Genesys Administrator, click the **Provisioning** tab and navigate to the Configuration Server on which you want to configure a secure port.

b. Select **Configuration Server** and click the **Configuration** tab.

c. In the **Server Info** section, click **Add** in the **Listening Ports** table. The **Port Info** dialog box appears.

d. In the **Port Info** dialog box, on the **General** tab:

- In the **ID** box, enter the port ID.
- In the **Port** box, enter the number of the new port.
- In the **Connection Protocol** box, select the connection protocol, if necessary.
- In the **Select Listening Mode** box, select **Secured**.
- Click **OK**.

e. In the **Port Info**> dialog box, on the **Advanced** tab:

- In the **Transport Parameters** box, replace `tls=1` with `upgrade=1`.
- Click **OK**.

f. Click **Save**, **Save & Close**, or **Save & New** to save the new configuration.

If security parameters have been configured for the application, during the connection through the Auto-Detect port, Configuration Server checks the validity of security settings. Depending on the results, the client is connected in secure mode or Configuration Server rejects the client connection.

2. (Optional) Enable mutual TLS on the server applications. In the options of each server application, set the **tls-mutual** configuration option to 1.

3. Assign a certificate to be used by the server applications.

> Important
>
> - If you are configuring simple TLS, certificates are optional for Genesys 8.x client applications.
> - Genesys recommends that, unless you have compelling reasons to have any of your applications and/or ports protected by their own individual certificate, you keep the certificate assignment at either the host or application level, as follows:
>   - If you are installing certificates on a Java-based PSDK application, such as Universal Contact Server, assign the certificates at the application level. Then use these certificates to provide secure data exchange for all ports configured on that application (see step 5).
>   - Otherwise, assign the certificates at the host level. Then use these certificates to provide secure data exchange for all applications residing on your hosts (see step 5).

[+] Show steps

After you create new server ports for secure connections, you must configure certificates in one of the following ways:

- Assign a certificate to a host, and then use this certificate to secure data exchange via any secure port of any server application that is located on this host. Use the procedure in the Assign to a Host tab, below.
- Assign a certificate to a server application, and then use this certificate to secure data exchange via any secure port of this application. A certificate that is assigned to an application takes precedence over the certificate that is assigned to a host. Use the procedure in the Assign to an Application tab, below.
- Assign a certificate directly to a specific port of a server application to secure data exchange via this port. A certificate that is assigned to a port takes precedence over certificates that are assigned to hosts and applications. Use the procedure in the Assign to a Port tab, below.

In Genesys Administrator, security-related properties are contained in the **Network Security** section of the Configuration tab for tenant, host, and server-type application objects.

> ### Important
> Before configuring secure connections, make sure that certificates are installed on the host computers on which specific Genesys components run, and that the certificate information is available to you.

Assign to a Host

To use a host's certificate to secure data exchange via any ports of any server applications (including client applications of server type) on that host, you must first assign a certificate to the host, and then complete the server application configuration.

a. In Genesys Administrator, select the host that accommodates the server applications whose connections you need to secure. To find out the name of the host that accommodates an application, look it up in the properties of that application.

b. Click on the host's **Configuration** tab.

c. In the **Network Security** section:

   i. In the **Certificate** text box, specify the full path to the **<serial_#>_<host_name>_cert.pem** file.

   > ### Important
   > If this client uses a certificate chain, specify a comma-delimited list of certificates with their paths.

   ii. (Optional) In the **Description** text box, enter a description for this certificate.

   iii. In the **Certificate Key** text box, specify the full path to the **<serial_#>_<host_name>_priv_key.pem file**.

   iv. In the **Trusted CA** text box, specify the full path to the **ca_cert.pem** file.

> **Important**
>
> If this client uses multiple Trusted CAs, specify the name of the **PEM** file that contains the list of certificates issued by those Trusted CAs.

For information about the installed files, see step 3 of Installing Certificates.

The certificate information now appears in the appropriate fields of the **Network Security** section of the host's **Configuration** tab.

d. Click **Save**, **Save & Close**, or **Save & New**, as appropriate, to save the new configuration.

When you configure the Host object in Genesys Administrator, the certificate information that you specify in the **Network Security** section of the Host's **Configuration** tab is also displayed in the Host's annex (**Options tab > View = Advanced View (Annex)**) in the **[security]** section. The configuration options in the **[security]** section have the same meaning as those in the **Network Security** section of the **Configuration** tab—namely, the parameters of the certificate assigned to this Host.

Assign to an Application

If you intend to use an application's certificate to secure data exchange via any ports of a specific server application, you must first assign a certificate to this application, and then complete the server application configuration.

a. In Genesys Administrator, click the **Provisioning** tab and navigate to the folder containing the application to which you want to assign a certificate.

b. Select the application and click the **Configuration** tab.

c. In the **Certificate Source** box of the **Network Security** section, specify whether the application is going to use a certificate installed locally (select **Application**) or the certificate installed on the host (select **Host**).

d. If you selected **Application** in step c, specify the certificate parameters as follows:

   i. In the **Certificate** text box, specify the full path to the **<serial_#>_<host_name>_cert.pem** file.

> **Important**
>
> If this client uses a certificate chain, specify a comma-delimited list of certificates with their paths.

   ii. (Optional) In the **Description** text box, enter a description for this certificate.

   iii. In the **Certificate Key** text box, specify the full path to the **<serial_#>_<host_name>_priv_key.pem** file.

   iv. In the **Trusted CA** text box, specify the full path to the **ca_cert.pem** file.

> **Important**
>
> If this client uses multiple Trusted CAs, specify the name of the PEM file that contains the list of certificates issued by those Trusted CAs.

For information about the installed files, see step 3 of Installing Certificates.

The certificate information now appears in the appropriate fields of the **Network Security** section of the host's **Configuration** tab.

e. Click **Save**, **Save & Close**, or **Save & New**, as appropriate, to save the new configuration.

When you configure the Application object in Genesys Administrator, the certificate information that you specify in the **Network Security** section of the host's **Configuration** tab is also displayed in the **[security]** section of the host's annex (**Options tab > View = Advanced View (Annex)**). The configuration options in the **[security]** section have the same meaning as those in the **Network Security** section of the **Configuration** tab—namely, the parameters of the certificate assigned to this Application.

Assign to a Port

a. In Genesys Administrator, click the **Provisioning** tab and navigate to the folder containing the server application for which you want to assign a certificate on its port.

b. Select the application and click the **Configuration** tab.

c. Open the **Server Info** section.

d. In the **Listening Ports** table, select the port whose connections you need to secure, and click **Edit**. The **Port Info** dialog box opens.

e. On the **General** tab, set **Select Listening Mode** to **Secured**. In the **Port Properties** dialog box, click the **Certificate** tab and click **Certificate properties**.

f. On the Network Security**Bold text** tab, specify the security parameters, as follows:

i. In the **Certificate** text box, specify the full path to the **<serial_#>_<host_name>_cert.pem** file.

> **Important**
>
> If this client uses a certificate chain, specify a comma-delimited list of certificates with their paths.

ii. (Optional) In the **Description** text box, enter a brief description of the certificate.

iii. In the **Certificate Key** text box, specify the full path to the **<serial_#>_<host_name>_priv_key.pem** file.

iv. In the Trusted CA text box, specify the full path to the **ca_cert.pem** file.

> **Important**
>
> If this client uses multiple Trusted CAs, specify the name of the **PEM** file contains the list of certificates issued by those Trusted CAs.

g. In the **Port Info** dialog box, click **OK**.

> **Important**
>
> The **Advanced** tab of the **Port Info** dialog box presents certificate parameters in a different form. This tab is reserved for future use.

h. In the **Port Properties** dialog box, click **OK**.

i. In the Application Properties dialog box, click **OK** to save the new configuration.

4. If necessary, remove (unassign) a certificate from a host, application, or port. **[+] Show steps**

From a Host

a. In Genesys Administrator, select the host from which to remove the certificate and open the **Configuration** tab.

b. Delete all certificate parameters in the **Network Security** section.

c. Click **Save & Close**, **Save**, or **Save & New**, as appropriate.

From an Application

> **Important**
>
> When you switch from Application certificate assignment to Host assignment, the Application certificate parameters are deleted.

a. In Genesys Administrator, select the application from which to remove the certificate and open the **Configuration** tab.

b. In the **Network Security** section, select **Host** in the **Certificate Source** dropdown list. This will delete all certificate parameters.

c. Click **Save & Close**, **Save**, or **Save & New**, as appropriate to save your changes.

From a Port

a. In Genesys Administrator, select the server application from which you want to remove a certificate from on its port and open the **Configuration** tab.

b. In the **Listening Ports** table in the **Server Info** section, select the port from which you want to delete the certificate and click **Edit**.

c. In the **Port Info** dialog box, click the **Network Security** tab, delete all certificate parameters, and click **OK**.

d. Click **Save & Close**, **Save**, or **Save & New**, as appropriate to save your changes.

5. (Optional, but recommended) Configure each server application to use the host certificate (for non-Java based applications) or application certificate (for Java-based applications), as applicable. If you are configuring mutual TLS, you must also configure each participating client application to use that host certificate. **[+] Show steps**

Use Host Certificate

After you assign a certificate to a host, you can use it to secure data exchange through any secure port of any server application that resides on that host.

a. In Genesys Administrator, navigate to the application that you want to configure to use the host certificate.

b. Select the application, and open the **Network Security** section of the **Configuration** tab.

c. In the **Select Source** field, select **Host** from the drop-down list.

d. Click **Save & Close**, **Save**, or **Save & New**, as appropriate to save your changes.

Use Application Certificate

After you assign a certificate to an application, you can use it to secure data exchange through any secure port of any server application that resides on that host.

a. In Genesys Administrator, navigate to the application that you want to configure to use the application certificate.

b. Select the application, and open the **Network Security** section of the **Configuration** tab.

c. In the **Listening Ports** table, select the port whose connections you need to secure, and click **Edit**. The **Port Info** dialog box opens.

d. Enter the certificate information as required.

e. Click **Save & Close**, **Save**, or **Save & New** as appropriate to save your changes.

6. Configure secure connections from the client applications. **[+] Show steps**

After you configure your server applications so that they have secure ports, you must change the configuration of your client applications, so that they connect to these ports. Remember that you must do this only for the connections on which extra measures are necessary to protect the data that is transferred between the Genesys applications.

The same configuration procedure is used for client applications of server type and user-interface type.

> **Important**
>
> When configuring simple TLS, certificates are optional for Genesys 8.x client applications.

a. Click the **Configuration** tab of the client application.

b. Select a server to which you need to make a secure connection, and click **Edit**.

c. In the **Connection** table in the **General** section, click **Add**, and enter the properties of the secure port that you created for the server during the previous configuration steps. The read-only **Connection Type** property indicates that this connection is secure.

d. If you are configuring mutual TLS, assign the host certificate to this application. Use the procedure in step 5, above.

e. Click **OK**.

f. Click **Save & Close**, **Save**, or **Save & New**, as appropriate, to save the new connection configuration.

The next time this application starts, it will connect to the server over a secure connection.

Configuring Secure Connections to Java/PSDK-Based Applications

Secure connections to Java/PSDK-based applications (such as Universal Contact Server) are configured in the same way as described in Configuration Steps, with the following exception:

- If you are running Java/PSDK-based applications on the same host as C++-based applications, do not use the host certificate to secure data exchange at the application or port level. Although both types of applications use a **.PEM** file for their private key, the internal format differs—Java/PSDK uses PKCS#8 and C++ uses RSA. Instead, use the application's certificate to enable secure data exchange on all secure ports of that application.

Configuring Secure Connections to Message Server

Secure connections to Message Server are configured in the same way as described in Configuration Steps, with the following exceptions:

- Message Server must configure its default port with the security settings if TLS is to be enabled. TLS configuration on secondary listening ports is not supported.
- The client establishing a connection to Message Server must configure the certificate information at the

Application or connection level.

Message Server supports TLS communication between it and the Solution Control Servers in a distributed configuration. In this situation, Message Server acts as the server and its port is configured as secured. The Solution Control Servers then connect to this port.

Configuring Secure Connections to Configuration Server

To configure a secure connection of a client application to Configuration Server, complete the following procedures.

New Server Clients

1. Specify the Auto-Detect port number of Configuration Server during application installation, by using the Installation Wizard. The Installation Wizard will propagate these parameters to the following locations:
   - To the **Command–Line** text box, in the **Server Info** section of the server's **Configuration** tab.
   - To the server application's **startServer.bat** file (for Windows) or **run.sh** file (for UNIX).
   - To the ImagePath in the Application folder in the Registry Editor.
2. Modify a client application configuration by adding a connection to the Configuration Server Application object to the client, as described in step 6 of Configuring Secure Client Connections, but select the Auto-Detect port in step c.

New User-Interface Clients

Start the applications and enter the Auto-Detect port number of Configuration Server in the **Log In** dialog box that appears.

Existing Clients

1. Verify or create the Auto-Detect port of Configuration Server in the Configuration Server Application object.
2. Modify a client application configuration by adding a connection to a Configuration Server Application object to the client, as described in step 6 of Configuring Secure Client Connections, but select the Auto-Detect port in step c.
3. Depending on the method that you use for starting client applications, for existing client applications of server type, change the port information to correspond to the port ID of the Auto-Detect port that you specified for Configuration Server, as follows:
   - In the **Command Line Arguments** text box in the **Server Info** section of the server's

Configuration tab.

- In the server application's **startServer.bat** file (for Windows) or **run.sh** file (for UNIX).
- In the **ImagePath** in the Application folder in the Registry Editor.

Configuring Secure Connections Between LCA and Solution Control Server

A secure connection between Local Control Agent (LCA) and Solution Control Server (SCS) is optional, and requires that you modify the LCA configuration file and the Host object on which LCA is running.

> **Important**
>
> If TLS is configured between LCA and SCS on a host machine, LCA uses TLS only on the connection with SCS. Other applications running on the host are connected through TCP.

Use the **upgrade** and **lca-upgrade** configuration options to configure secure data exchange using TLS on connections between LCA and SCS. These options are configured on the Host computer on which LCA and SCS are running, and where the certificate information is available to you.

For more information about these two options, refer to the *Framework Configuration Options Reference Manual*.

Before you configure the secure connection, you must:

- Install a certificate on the Host computer on which LCA is running.
- Have the certificate information available to you.

[+] Show steps

1. In the LCA configuration file, **lca.cfg**:
   a. If it does not already exist, add the new section **[security]**.
   b. In this section:
      - Use the **upgrade** option to designate this port as secure.
      - Specify the certificate parameters that will be used to secure the connections. The actual parameters are determined by the type of TLS you are using, as follows:
        - If you are using mutual TLS, set the **certificate**, **certificate-key**, and **trusted-ca fields**.
        - If you are using simple TLS, set only the **certificate** and **certificate-key** fields.
        -

   For more information, see the Sample Configuration Files for LCA.

2. In the annex of the host machine on which LCA is running, set the option **lca-upgrade** to `1 (true)`.
3. Restart the host machine and LCA.

Sample Configuration Files for LCA

The following are sample configuration files for LCA in which the values of the `upgrade` options of the `security` section are set. **[+] Show Files**

Mutual TLS:

```
[log]
verbose=standard
standard=stdout, logfile
[security]
upgrade=1
tls-mutual=1
certificate=/home/tech/sec/aix_cert.pem
certificate-key=/home/tech/sec/aix_priv_key.pem
trusted-ca=/home/tech/sec/techpubs_dco.pem
```

Simple TLS:

```
<tt>[log]
verbose=standard
standard=stdout, logfile
[security]
upgrade=1
certificate=/home/tech/sec/aix_cert.pem
certificate-key=/home/tech/sec/aix_priv_key.pem
```

Configuring Secure Connections Between Genesys Deployment Agent and its Clients

A secure connection between Genesys Deployment Agent and its clients is optional, and requires that you modify the Genesys Deployment Agent configuration file and the Host object on which Genesys Deployment Agent is running.

Use the **tls** and **gda-tls** configuration options to configure secure data exchange using TLS on connections between Genesys Deployment Agent and its clients. Refer to the *Framework Configuration Options Reference Manual* for detailed descriptions about these configuration options.

Before you start to configure the secure connections, you must ensure that:

- Certificates are installed on the Host computers on which Genesys Deployment Agent is running.
- The certificate information is available to you.

[+] Show steps

> **Important**
>
> Refer to the *Framework Configuration Options Reference Manual* for detailed descriptions of the options modified in this procedure.

1. In the Genesys Deployment Agent configuration file, **gda.cfg**:
   a. If it does not already exist, add the new section **[security]**.
   b. In this section:
      - Use the **tls** option to designate this port as secure.
      - Specify the certificate parameters that will be used to secure the connections. The actual parameters are determined by the type of TLS you are using, as follows:
        - For Mutual TLS, set the **certificate**, **certificate-key**, and **trusted-ca** fields.
        - For Simple TLS, set only the **certificate** and **certificate-key** fields.

   For more information, see the Sample Configuration Files for Genesys Deployment Agent.
2. In the annex of the host machine on which Genesys Deployment Agent is running, set the option **gda-tls** to 1.
3. Restart Genesys Deployment Agent.

Sample Configuration Files for Genesys Deployment Agent

The following are sample configuration files for Genesys Deployment Agent,in which the values of the transport options in the **[security]** section are set. The settings are the same as any TLS setup, except that they are set in the configuration file instead of the configuration objects. **[+] Show files**

Mutual TLS:

```
[log]
verbose = standard
standard = stdout, gdalog
[web]
rootdir=./gdaroot
[security]
tls=1
tls-mutual=1
certificate=/home/tech/sec/aix_cert.pem
certificate-key=/home/tech/sec/aix_priv_key.pem
trusted-ca=/home/tech/sec/techpubs_dco.pem
```

Simple TLS:

```
[log]
verbose = standard
standard = stdout, gdalog
[web]
rootdir=./gdaroot
[security]
```

```
tls=1
certificate=/home/tech/sec/aix_cert.pem
certificate-key=/home/tech/sec/aix_priv_key.pem
```

Configuring Secure HA Synchronization connections

This section describes how to configure secure connections between primary and backup servers in a high-availability (HA) configuration.

> **Important**
>
> See “Supporting Components” on page 136 for information about components that support secure connections in HA configurations. For information about setting up a HA environment for these Genesys components, see the corresponding product documentation.

The HA synchronization connection is configured by selecting the **HA sync** check box in the **Port Info** dialog box of a specific port. This indicates that the port will be used by the former primary server to connect to the new primary server after a failover. If the **HA sync** check box is not selected, the former primary server will connect to the default port of the new primary server.

> **Important**
>
> Genesys does not recommend using the ports with the port-level assigned certificates for an HA synchronization connection between redundant servers. The secure connection should be configured on a host or application level instead.

[+] Show steps

1. In the **Server Info** section on the **Configuration** tab of the properties of both the primary and backup servers in a redundant pair, create a new port with the same **ID**, and with **Select Listening Mode** set to `Secured`.

   > **Warning**
   >
   > When multiple ports are configured for a server in a Hot Standby redundancy pair, their **ID**s and the **Select Listening Mode** settings of the primary and backup servers must match respectively.

2. In the **Port Info** dialog box of each server, click **OK** to save the new configuration. Then, in the **Configuration** tab of each, click **Save**.
3. In the **Listening Ports** table of each server, select the port that you just created, and click **Edit**.
4. In the **Port Info** dialog box, select the **HA sync** check box, and click **OK**.

5. Click **Save & Close**, **Save**, or **Save & New**, as appropriate, to save the configuration changes.

Certificate Revocation Lists

TLS uses digital certificates to identify the parties in a conversation and then to negotiate an encryption algorithm to use. If the certificates are revoked or expired, the connection will fail to identify the parties and TLS will not set up the encrypted channel.

A Certificate Revocation List (CRL) is a time-stamped list identifying revoked certificates. This list is signed by a CA or CRL issuer and is made freely available in a public repository. Each revoked certificate is identified in a CRL by its certificate serial number. When a system uses a certificate for verifying a remote user's digital signature, for example, that system not only checks the certificate signature and validity but also acquires a suitably-recent CRL and checks that the certificate serial number is not on that CRL. The meaning of “suitably-recent” may vary with local policy, but it usually means the most recently-issued CRL. A new CRL is issued on a regular periodic basis (such as hourly, daily, or weekly). An entry is added to the CRL as part of the next update following notification of revocation. An entry must not be removed from the CRL until it appears on one regularly-scheduled CRL issued beyond the revoked certificate's validity period.

Configuration

Use the configuration option **tls-crl** (in the **[security]** section) to allow the supporting Genesys component to verify certificates against a CRL, by specifying a filename, in PEM format, that contains one or more certificates defining the Certificate Revocation List. Refer to the *Framework Configuration Options Reference Manual* for a full description of this option.

> **Important**
>
> Configuration of a CRL in SIP Server slightly differs from other Genesys components. Refer to the *Framework SIP Server Deployment Guide* for more information.

Cipher Lists

The **cipher-list** configuration option allows the supporting Genesys component to select a list of cipher suites used in TLS. This option is transferred to a third-party library and describes the set of possible cipher suites.

Cipher Formatting Rules

> **Important**

> Cipher list format for an application using the PSDK library is different from that for an application using the Genesys common library. If you are configuring a cipher list for the PSDK-based appliction, refer to the *Platform SDK Developer's Guide* for the proper format, and more information about cipher lists in PSDK.

For applications using the Genesys common library, the cipher list string is a list of cipher operations. Each operation consists of an optional operator character followed by a name. Cipher list strings must conform to the following formatting rules:

- The name is either a valid cipher name or a cipher alias. Valid names contain the characters `a-z`, `A-Z`, `0-9`, and `-`.
- List separator characters are used to separate the names and aliases in the list. A list separator character must be a colon.
- Multi-part names are joined with +.
- The character `!` appearing immediately after a separator indicates a kill operation. The cipher following the character becomes unavailable.
- The character + appearing immediately after a separator indicates an order operation. This moves the active cipher to the current position in the list of ciphers.
- The character `-` appearing immediately after a separator indicates a delete operation. The cipher following the character becomes inactive. The cipher remains available for further operations.
- A non-operator character appearing immediately after a separator indicates an add operation. If the cipher following the character is not currently active, the cipher is added as an active cipher to the end of the list of available ciphers.

All operations occur in the order in which they appear in the list. If the cipher corresponding to a name (or part of a name, for multi-part names) is not available in the library, it is ignored during loading. In this situation, no error message is logged.

Cipher Aliases

Ciphers also have aliases. The following table details the primary cipher aliases.

Alias	Description
kRSA, kDHr, kDHd, kEDH	Key exchange types
aRSA, aDSS, aNULL, aDH	Authentication
DES, 3DES, RC4, RC2, eNULL	Ciphers
MD5, SHA1	Message digests

Groups of commonly-used ciphers also have aliases. This enables multiple aliases to be specified easily. The following table details the cipher group aliases.

Alias	Description
SSLv2	All SSLv2 ciphers
SSLv3	All SSLv3 ciphers

Alias	Description
EXP	All export ciphers
LOW	All low strength ciphers (no export ciphers, normally single DES)
MEDIUM	128-bit encryption
HIGH	Triple DES

Aliases can be joined in a colon-separated list to specify the ciphers to add, move, or delete.

Ciphers Example

The following string is an example of a cipher string:

```
!ADH:RC4+RSA:HIGH:MEDIUM:LOW:EXP:+SSLv2:+EXP
```

This cipher string is interpreted in the following sequence:

1. Do not consider any ciphers that do not authenticate.
2. Use ciphers that use RC4 and RSA.
3. Include the HIGH, MEDIUM, and LOW security ciphers.
4. Add all export ciphers.
5. Pull all SSLv2 and export ciphers to the end of the list.

Configuration

Use the **cipher-list** option to define the list of ciphers. Refer to the *Framework Configuration Options Reference Manual* for a detailed description of the option.

If you are configuring a cipher list for an application using the Genesys common library, refer to the following:

- Cipher Formatting Rules for valid formats of a cipher list.
- Ciphers Example for an example of a valid cipher list.

If you are configuring a cipher list for an application using the PSDK library, refer to the *Platform SDK Developer's Guide*.

> ### Warning
> If you are going to use cipher lists on a host running both PSDK library-based applications and Genesys common library-based applications, do not configure **cipher-list** at the host-level. Configure the option at the application level or lower.

Check for Certificate-Host Matching

The **tls-target-name-check** option in the **[security]** section) enables a case-insensitive comparison of the TLS host name and the certificate's subject field during the authentication process. This option is transferred to a third-party library and describes whether it is necessary or not to check the names.

> **Important**
>
> Security Pack 8.1 and earlier supported only a case-sensitive check of host names.

Refer to Introduction to Genesys Transport Layer Security for details on authentication of TLS-Server and TLS-Client identity, which includes a step to check for certificate-host matching.

If the supporting Genesys component has a TLS-Client role for outbound connection and **tls-target-name-check=no**, then comparison of TLS-Server host name and the certificate's subject field is not made. This is used in cases when some phone devices or programs have the certificate without the host name in subject field, but have a MAC-address or other information.

By default, a comparison is not made, and the connection is allowed. Refer to the *Framework Configuration Options Reference Manual* for a full description of this option.