

User Guide



XEServer & EAM Post-Installation Steps

Post-Installation Steps Overview

This section contains the post-installation tasks that Edifecs recommends after you install Edifecs Application Manager and XEServer.

Replace default encryption keys

Some XEServer and EAM features that use SSL for secure communication come with default keystores that contain encryption keys and certificates. For these components, Edifecs recommends that you replace the default keystore entries with your own keys and certificates.

EAM

The connection between EAM Client and EAM Server is encrypted using SSL. To establish trustworthy channels between EAM Client and EAM Server, EAM uses a default key pair and a certificate (self-signed) distributed with EAM. To read the instructions on how to replace the default encryption keys for EAM, see [Replace Keys for EAM](#).

XEServer

XEServer features that use default encryption keys and certificates distributed with XEServer listed below. Edifecs recommends that you replace these keys to strengthen security. Follow the links below to read the instructions on how to replace the default encryption keys for a specific feature:

- [Jetty components](#) (RESTful Web Service, Web Service, XEServer agent, and so on).
- [Scheduler Services](#)
- [XEServer SSH Server](#)
- [Agent Auto-discovery](#)
- [FHIR Service](#)
- [XPM Service](#)

Replace Default Encryption Keys

This section has the following information:

Replace the Keys for EAM.....	3
Replace the Keys for the XEServer Features.....	9
Replace the Keys for the Jetty Components.....	11
Replace the Keys for XEServer and XESManager.....	12
Replace the Certificate for the FHIR Service.....	15
Replace the Certificate for the XPM Service.....	16

Replace the Keys for EAM

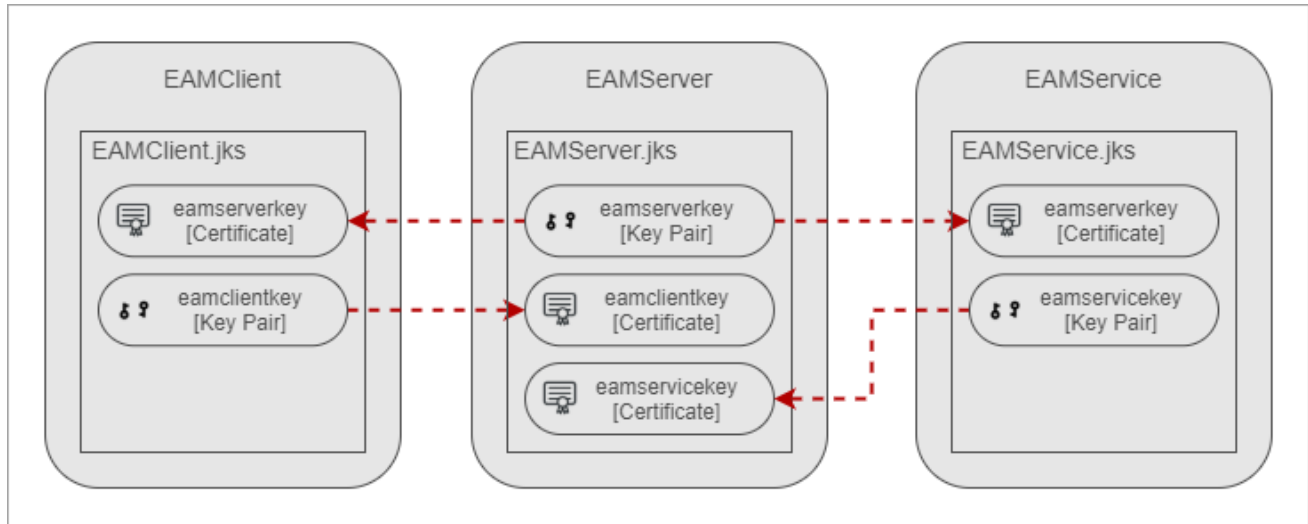
The connection between the EAM Client and EAM Server is encrypted using SSL. To establish trustworthy channels between EAM Client and EAM Server, EAM uses default self-signed certificates distributed with EAM. Edifecs strongly recommends that you replace these default EAM certificates. These certificates are located at the following keystore locations:

EAM Component	Keystore Location	Default Passwords	
		Keystore	Key Pair
EAM Server	\${EAMRoot}\Server\security\keystores\EAMServer.jks	eamserver	eamserver
EAM Client	\${EAMRoot}\Client\configuration\security\EAMClient.jks	eamclient	eamclient
EAM Service	\${EAMRoot}\Server\security\keystores\EAMService.jks	eamservice	eamservice



Tip EAM Service is a process that starts EAM Server if the latter is configured to run as a Windows/Linux service.

In the context of SSL, EAM has three components that use SSL certificates for intercommunication (arrows indicate related keys and certificates):



This section shows you how to generate new keys with self-signed certificates and how to replace the default certificates when you want to and on expiry. However, you can similarly use certificates issued by a certificate Authority (CA) to secure the connection between EAM components.

Generate Keys and Replace Certificates

The main steps to replace certificates for all the three EAM components (EAM Client, EAM Server, and EAM Service) are below. The procedure is rather straightforward and at a high level comprises the following steps:

1. Remove the existing entries in all the three keystores.
2. Generate a new key pair in the EAM Server keystore.
3. Export the EAM Server certificate, and then import this certificate to the keystores used by EAM Client and EAM Service.
4. Generate a new key pair in the EAM Client keystore.
5. Export the EAM Client certificate, and then import this certificate to the EAM Server keystore.
6. Generate a new key pair in the EAM Service keystore.
7. Export the EAM Service certificate, and then import this certificate to the EAM Server keystore.

Below you can view the detailed step-by-step instructions on how to replace the certificates. In this article, the open source GUI tool [KeyStore Explorer](#) is used to manipulate the keystore contents, however you can use any other keystore management utility. All screen shots and instructions refer to KeyStore Explorer 5.4.

EAM Server

First, clear the EAM Server keystore. To do this:

1. Go to `${EAMRoot}\Server\security\keystores\`, and then open *EAMServer.jks* (the default password to the keystore is "eamserver"). The keystore contains three entries:

T	E	Entry Name	Algori...	Key Size	Certificate Expiry	Last Modified
	-	eamclientkey	RSA	1024	1/1/2020 1:00:00 AM ...	3/4/2010 11:22:58 AM ...
		eamserverkey	RSA	1024	1/1/2020 1:00:00 AM ...	3/4/2010 11:22:58 AM ...
	-	eamservicekey	RSA	1024	1/1/2020 1:00:00 AM ...	3/4/2010 11:22:58 AM ...

2. Delete all the three entries.

Then, generate a new key pair in the EAM Server keystore. To do that:

3. Click to create a new key pair.
4. Use the default algorithm **RSA** and 2048-bit key size and then, click **OK**.
5. Set the validity range for the new key pair, and then click to provide DN fields.
6. Enter your organization's details in the DN fields, and then click **OK**.
7. Enter an alias for the new key pair.



Tip Although you can use any alias name, Edifecs recommends that you use **eamserverkey** for consistency.

8. Enter **eamserver** as a password and then click **OK**. This is the default password that EAM Server uses to access the key pair **eamserverkey**. If you want to use a custom password for the key pair, see [Use Custom Keystore Passwords](#).

Now, export the EAM Server certificate from the key pair. To do that:

9. Right-click the key pair entry you have just created, and then click **Export Certificate Chain**.
10. Enter the location where you want to store the EAM Server certificate file (*eamserverkey.cer*). Later, you will have to import this certificate to the truststores of EAM Client and EAM Service.
11. Import the certificates of EAM Client and EAM Service to the EAM Server keystore.



Note You should perform this step later after you create new certificates for EAM Client and EAM Service. Skip this step for now, and you can return to this step later in this scenario.


EAM Client


1. Go to `${EAMRoot}\Client\configuration\security\`, and then open *EAMClient.jks* (the default password to the keystore is "eamclient"). The keystore contains two entries:

EAMClient.jks						
T	E	Entry Name	Algo...	Key...	Certificate Expiry	Last Modified
		eamclientkey	RSA	1024	1/1/2020 1:00:00...	3/4/2010 11:22:5...
	-	eamserverkey	RSA	1024	1/1/2020 1:00:00...	3/4/2010 11:22:5...



2. Delete the two entries.


Then, import the EAM Server certificate (file *eamserverkey.cer* created at [Step 10](#) in EAM Server section). To do this:

3. Click  to import the certificate.
4. Locate the EAM Server certificate file (*eamserverkey.cer*), and then click **Open**.
5. Enter an alias for the new certificate entry.

 **Tip** Although you can use any alias name, Edifecs recommends that you use **eamserverkey** for consistency.

The EAM Server trusted certificate has been imported. Now, generate a new key pair in the EAM Client keystore. To do that:

6. Click  to create a new key pair.
7. Use the default algorithm **RSA** and 2048-bit key size, and then click **OK**.
8. Set the validity range for the new key pair, and then click  to provide DN fields.
9. Enter your organization's details in DN fields, and then click **OK**.
10. Enter an alias for the new key pair.

 **Tip** Although you can use any alias name, Edifecs recommends that you use **eamclientkey** for consistency.

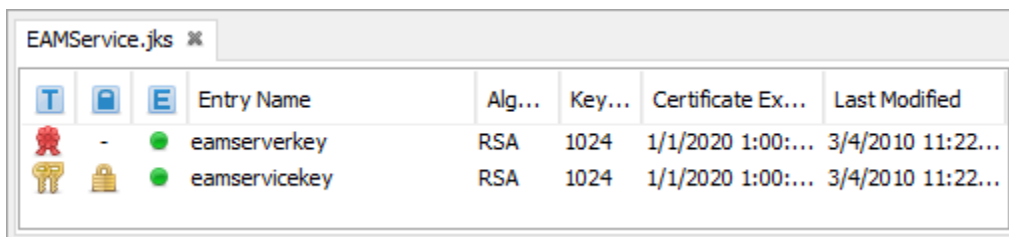
11. Enter **eamclient** as a password, and then click **OK**. This is the default password that EAM Client uses to access the key pair **eamclientkey**. If you want to use a custom password for the key pair, see [Use Custom Keystore Passwords](#).

This creates the key pair for EAM Client. Now, export the EAM Client certificate from the key pair. To do that:

12. Right-click the key pair entry you have just created, and then click **Export Certificate Chain**.
13. Enter the location where you want to store the EAM Client certificate file (*eamclientkey.cer*).
14. Import this certificate file *eamclientkey.cer* to the **EAM Server** keystore ($\${EAMRoot}\Client\configuration\security\EAMServer.jks$). Use "eamclientkey" as an alias.


EAM Service

1. Go to $\${EAMRoot}\Server\security\keystores\$, and then open *EAMService.jks* (the default password to the keystore is "eamservice"). The keystore contains two entries.



2. Delete the two entries.



Then, import the EAM Server certificate (file *eamserverkey.cer* created at [Step 10](#) in EAM Server section). To do this:

3. Click  to import the certificate.
4. Locate the EAM Server certificate file (*eamserverkey.cer*), and then click **Open**.
5. Enter an alias for the new certificate entry.



Tip Although you can use any alias name, Edifecs recommends that you use **eamserverkey** for consistency.

This imports the EAM Server trusted certificate. Now, generate a new key pair in the EAM Server keystore. To do that:

6. Click  to create a new key pair.
7. Use the default algorithm **RSA** and 2048-bit key size, and then click **OK**.
8. Set the validity range for the new key pair, and then click  to provide DN fields.
9. Enter your organization's details in the DN fields, and then click **OK**.
10. Enter an alias for the new key pair.



Tip Although you can use any alias name, Edifecs recommends that you use **eam-servicekey** for consistency.

11. Enter **eamservice** as a password and then click **OK**. This is the default password that EAM Server uses to access the key pair **eamservicekey**. If you want to use a custom password for the key pair, see [Use Custom Keystore Passwords](#).

This creates the key pair for EAM Service. Now, export the EAM Service certificate from the key pair. To do that:

12. Right-click the key pair entry you have just created, and then click **Export Certificate Chain**.
13. Enter the location where you want to store the EAM Service certificate file (*eam-servicekey.cer*).
14. Import this certificate file *eamservicekey.cer* to **EAM Server** keystore (`${EAMRoot}\Client\configuration\security\EAMServer.jks`). Use "eamservicekey" as an alias.

Test EAM Connection

To verify that the new certificates work as expected:

1. Start EAM Client and connect to EAM Server.
2. Ensure that the connection is established and no exceptions are logged.

Use Custom Keystore Passwords

This section shows you how to use non-default passwords for EAM keystores and keys. To access password-protected keystores and keys, EAM components use the passwords defined in the

.properties files:

- EAM Server: `${EAMRoot}\Server\security\config\server.properties`
- EAM Service: `${EAMRoot}\Server\security\config\service.properties`
- EAM Client. To access EAMClient.jks and its keys, EAM Client uses passwords specified in two files:
 - `${EAMRoot}\Client\configuration\security\client.properties`
 - `${EAMRoot}\Client\configuration\security\hide.properties`

These *.properties* files have a similar structure and look like the following:

Client.properties

```
#timestamp
enabled.cipher.suites=SSL_RSA_WITH_NULL_MD5
keystoretype=JKS
keystorepass=XE\ :5H+PljgCe5nBPdnCPLKV1Q\=\=
keypassword=XE\ :5H+PljgCe5nBPdnCPLKV1Q\=\=
keystorepath=${EAMRoot}/Client/configuration/security/EAMClient.jks
contextprotocol=TLSv1
tls.profiles=TLS SASL/EDIFECs_LOGIN
```

The **highlighted** properties hold the keystore and key passwords in an encoded format. If you decide to use a non-**default** password for the EAM keystores or keys, you have to specify your custom password in the corresponding *.properties* file to enable EAM access to the keystore / key.

For example, if your EAM Client's keystore and the key in this keystore have the password "foobar1234", then:

1. Open the file `${EAMRoot}\Client\configuration\security\client.properties`.
2. Specify your custom password as a plain text:


```
keystorepass=foobar1234
keypassword=foobar1234
```
3. Save the file.
4. Repeat steps 2 and 3 for the file `${EAMRoot}\Client\configuration\security\hide.properties`.

The next time that you start EAM Client and connect to the EAM Server, EAM uses the new password "foobar1234" to access the EAM Client keystore and its key.



Tip When you start EAM after changing a password in the *.properties* file, EAM overwrites plain text passwords with encoded values.

Replace the Keys for the XEServer Features



Note When you use the instructions in this section, use the following password to access the keystores and to protect new keystores and keystore entries: *password*.

Scheduler Services

XEServer services [Scheduler Server](#) and [Scheduler Client](#) use SSL to securely communicate with each other. The services use the following default keystores:

Service	Keystore Location	Passwords	
		Keystore	Key Pair
Scheduler Server	<code>\${XESRoot}/features/scheduler/security/scheduler.server.keystore.jks</code>	password	password
Scheduler Client	<code>\${XESRoot}/features/scheduler/security/scheduler.client.truststore.jks</code>	password	-

You have two options on how to replace the default key and certificate used by the two services:

- Replace the default key and certificate in the keystores that are already available,
- or-
- Create a new *.jks* keystore (with a new key pair for Scheduler Server) and a new *.jks* truststore (with Scheduler Server's certificate).

To replace the default entries in the existing keystores:

1. Open the keystore *scheduler.server.keystore.jks*.
2. Delete the key pair entry *scheduler-server*.
3. Generate a new key pair. You can use any name as an alias. The only requirement is that your keystore must hold only one entry - the key pair.



Note If you use a non-default password for the key entry (or if you change the keystore password), make sure you update the password on the [Scheduler Server](#) configuration page.

4. From the key pair you just created, export the certificate to a *.cer* file.
5. Open the truststore *scheduler.client.truststore.jks*.
6. Delete the existing certificate entry *scheduler-server*.
7. Import the *.cer* certificate from Step #4.
8. Restart the profile with Scheduler services.

XEServer SSH Server

XEServer's built-in SSH server uses SSL to secure the connection to SSH clients. The XEServer SSH server uses the following default key pair:

Component	Keystore Location	Passwords	
		Keystore	Key Pair
SSH Server	<code>\${XESRoot}\platform\security\ssh.server.keystore.jks</code>	password	password

To replace the default key pair:

1. Go to `${XESRoot}\platform\security\`.
2. Open the keystore `ssh.server.keystore.jks`.
3. Delete the default key pair entry **xes-sshserver**.
4. In the keystore `ssh.server.keystore.jks`, generate a new key pair entry with the alias **xes-sshserver**.
5. Restart XEServer.

The next time you connect to the XEServer's SSH server using an SSH client (for example, PuTTY), you will receive a warning about the server's host key mismatch which indicates that the SSH server has picked up the new key. Accept the new host key to connect to the XEServer's SSH server.

XEServer Auto-discovery

XEServer agent can send registration requests to XESManager through a protected channel secured by SSL. During the registration phase, XEServer agent acts as a client, and XESManager represents the server side. To identify XESManager as a trusted party, XEServer agent uses the default certificate in the truststore `${XESRoot}/platform/security/xes.manager.truststore.jks`.

To replace the default certificate:

1. On the computer with XESManager installed, replace the default XESManager key pair:
 1. Go to `${ECRootPath}\XESManager\workspace\config\`.
 2. Open the keystore `xes-agents.jks`. The default password is "password".
 3. Delete the key pair entry **xes-manager**.
 4. Create a new key pair and use **xes-manager** as an alias.
 5. From the key pair you just created, export a certificate to a `.cer` file.
 6. Move the `.cer` file to the computer with XEServer.
2. Open the truststore `${XESRoot}/platform/security/xes.manager.truststore.jks` and delete the default certificate entry **xesmanager**.
3. Import the `.cer` certificate to the truststore `${XESRoot}/platform/security/xes.manager.truststore.jks`.



Tip As an alternative, you can create a new `.jks` truststore, import the certificate to this truststore, and update the SSL settings on the [Auto-Discovery Tab](#).

4. Restart XEServer and XESManager.

Replace the Keys for the Jetty Components

Some of the XEServer components based on Jetty server (RESTful Web Service, Tracking and Duplicates Server Service, XEServer agent, and so on) use the default encryption keys distributed with XEServer. Edifecs recommends that you replace the default keys and certificates with your own entries to eliminate the risk of the man-in-the-middle attacks (MITM) for your organization.

A generic procedure to replace the encryption keys for Jetty-powered components comprises the following steps:

1. Go to `${XESRoot}\platform\security`.
2. Delete the existing key pair entry in `KeyringClient.jks`:

```
keytool -delete -alias {entryToDelete} -keystore KeyringClient.jks -storepass password
```

3. Generate a new key pair entry:

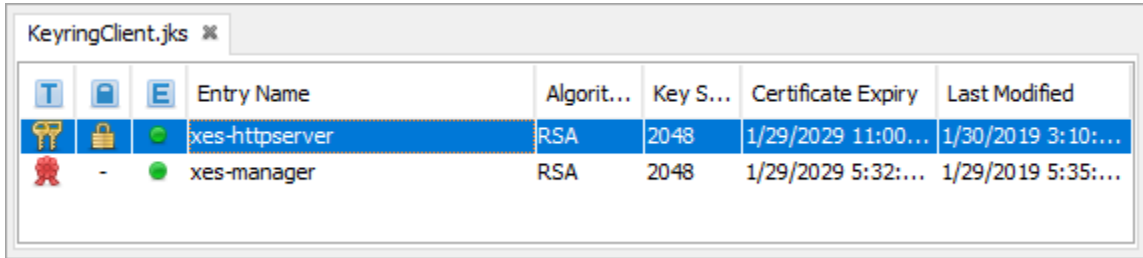
```
keytool -genkeypair -alias {newEntryAlias} -keyalg RSA -validity 365 -keysize 2048 -keystore KeyringClient.jks -storepass password -keypass password
```

For a sample scenario that shows you how to replace encryption keys for the XEServer agent (XES Agent runs in Jetty), see [Replace Keys for XEServer and XESManager](#).

Replace the Keys for XEServer and XESManager

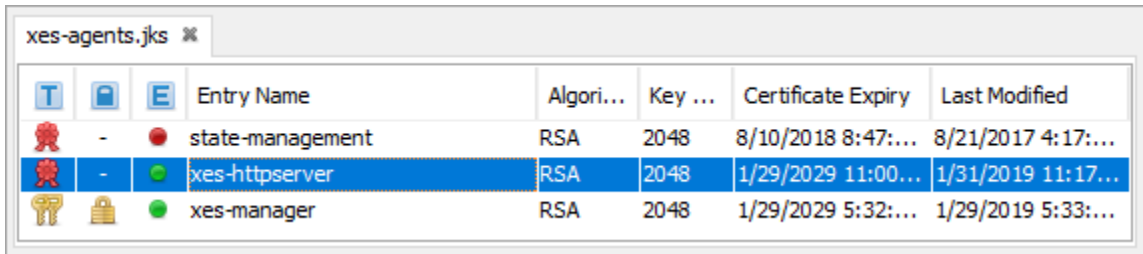
The XEServer agent and XESManager communicate with each other (the agent sends the profile statistics to XESManager, and XESManager sends commands to the agent) over a protected SSL-secured channel. To encrypt the network traffic between XEServer agent and XESManager, the following encryption keys are used:

- XEServer agent uses the default key pair **xes-httpserver** located in the keystore `${XESRoot}\platform\security`:



	Entry Name	Algorit...	Key S...	Certificate Expiry	Last Modified
	xes-httpserver	RSA	2048	1/29/2029 11:00...	1/30/2019 3:10:...
	xes-manager	RSA	2048	1/29/2029 5:32:...	1/29/2019 5:35:...

- XESManager uses the default certificate **xes-httpserver** located in the keystore: `${ECRootPath}\XESManager\workspace\config\xes-agents.jks`:



	Entry Name	Algori...	Key ...	Certificate Expiry	Last Modified
	state-management	RSA	2048	8/10/2018 8:47:...	8/21/2017 4:17:...
	xes-httpserver	RSA	2048	1/29/2029 11:00...	1/31/2019 11:17...
	xes-manager	RSA	2048	1/29/2029 5:32:...	1/29/2019 5:33:...

This keystore contains the certificates of all the XEServer agents that communicate with XESManager. Edifecs recommends that you replace the default certificate **xes-httpserver** with your own certificate to eliminate the risk of man-in-the-middle attacks (MITM).

The instructions below show you how to replace keystore entries by using the command line utility [Java Keytool](#). However, you can use other keystore management tools.



Note When you use the instructions in this section, use the following password to access the keystores and to protect new keystores and keystore entries: *password*.

Create a new key pair in XEServer

First, you have to replace the default keypair **xes-httpserver** in *KeyRingClient.jks*.

1. Go to `${XESRoot}\platform\security\`.
2. On your terminal, run the following command to remove the existing key pair **xes-**

httpserver from the keystore *KeyringClient.jks*:

```
keytool -delete -alias xes-httpserver -keystore KeyringClient.jks -storepass password
```

- Run the following command to generate a new key pair with the alias **xes-httpserver** in *KeyringClient.jks*.

```
keytool -genkeypair -alias xes-httpserver -keyalg RSA -validity 365 -keysize 2048 -keystore KeyringClient.jks -storepass password -keypass password
```

- Specify the DN values by answering each question when prompted:

DN Field	Description
First & Last Name (CN)	The domain name that you or the users of your organization use to connect to XESManager. For example, localhost .
Organizational Unit (OU)	(Optional) The name of your organization unit to differentiate your organization from other divisions.
Organization Name (O)	(Optional) The name of your organization.
City / Locality (L)	(Optional) The name of the city your organization is in.
State / Province (ST)	(Optional) The name of the place where your organization is physically located.
Country Code (C)	(Optional) A two-letter country / region code. For example, US .

Review the DN fields that you entered and proceed. The new certificate entry **xes-httpserver** is available in *KeyringClient.jks*.

Export the certificate

You have to export the certificate to a *.cer* file.

- Go to `${XESRoot}\platform\security\`.
- On your terminal, run the following command:

```
keytool -exportcert -alias xes-httpserver -file xes-httpserver.cer -keystore KeyringClient.jks
```

This exports the certificate **xes-httpserver** from the keystore *KeyringClient.jks*. The file *xes-httpserver.cer* appears at `${XESRoot}\platform\security\`.

- Move the file *xes-httpserver.cer* to the computer where XESManager is installed. You can skip this step if both XEServer and XESManager run on the same computer.

Replace the default agent certificate in XESManager

You have to import the new XEServer agent certificate file (*xes-httpserver.cer*) to the XESManager truststore.

1. Move the file *xes-httpserver.cer* to `${ECRootPath}\XESManager\workspace\config\`.
2. Open your terminal, and then change the working directory to `${ECRootPath}\XESManager\workspace\config\`.
3. Delete the existing certificate entry:

```
keytool -delete -alias xes-httpserver -keystore xes-agents.jks -storepass password
```

4. Run the following command:

```
keytool -importcert -trustcacerts -file xes-httpserver.cer -alias xes-httpserver -keystore xes-agents.jks
```



Tip You can choose any name as an alias.

Test the Connection between XEServer and XESManager

To verify that the new certificates work as expected:

1. Restart XEServer and XESManager.
2. Ensure that the connection has been established (*<https://localhost:5643/xes-manager>* page opens and your XEServer agent is online and is authorized) and no exceptions are logged.



Note If you observe that the agent has the status **Unauthorized** (you can verify the status on the XESManager Agents page), run the script `${XESRoot}\bin\agent\cleanup_security_key.bat` on the XEServer computer.

Replace the Certificate for the FHIR Service

The XEServer [FHIR Service](#) (becomes available after the installation of XESModule for FHIR) uses a default certificate to identify the FHIR server that runs on *localhost* as trusted. The certificate is located in the truststore `${XESRoot}/platform/security/xes.fhir.truststore.jks`.



Note When you use the instructions in this section, use the following password to access the keystores and to protect new keystores and keystore entries: *password*.

To replace the default certificate for the FHIR Service:

1. Generate a new key pair and export the certificate from the XESModule for FHIR keystore. For more information, see [SSL Certificate Deployment](#) in XESModule for FHIR Help Center.
2. In the truststore `${XESRoot}/platform/security/xes.fhir.truststore.jks`, delete the default certificate entry **localhost**.
3. Import the `.cer` certificate to `${XESRoot}/platform/security/xes.fhir.truststore.jks` using **localhost** as an alias.



Tip As an alternative, you can create a new `.jks` truststore, import the certificate to this truststore, and update the SSL settings on the [FHIR Service](#) configuration page.

4. Restart XEServer and XESModule for FHIR.

Replace the Certificate for the XPM Service

The XEServer [XPM Service](#) uses a default certificate to identify the XPM server that runs on *localhost* as trusted. This certificate is located in the truststore `${XESRoot}/features/route-engine/components/state-event/resources/ssl/keystore.jks`.



Note When you use the instructions in this section, use the following password to access the keystores and to protect new keystores and keystore entries: *password*.

To replace the default certificate for the XPM Service:

1. Generate a new key pair and export the certificate from the XProcess Management keystore. For more information, see the section [SSL Certificates](#) in XProcess Management Help Center.
2. In the truststore `${XESRoot}/features/route-engine/components/state-event/resources/ssl/keystore.jks`, delete the default certificate entry **localhost**.
3. Import the `.cer` certificate to `${XESRoot}/features/route-engine/components/state-event/resources/ssl/keystore.jks` using **localhost** as an alias.



Tip As an alternative, you can create a new `.jks` truststore, import the certificate to this truststore, and update the SSL settings on the [FHIR Service](#) configuration page.

4. Restart XEServer and XProcess Management.

Copyright

The product(s) described in this documentation are furnished under a license agreement and may be used only in accordance with the terms of such agreement. This document does not confer any intellectual property rights.

Copyright © 2021 Edifecs, Inc. All rights reserved.

Edifecs®, CommerceDesk®, SpecBuilder®, Encounter Management™, Enrollment Management™, XEngine™, Transaction Management™, Edifecs ICD-10 Code Management™, Edifecs ICD-10 Code Translation™, Edifecs Test Management®, Population Payment Management™, Ramp Management™, Collaborative Testing™, Population Dimensions™, Smart Decisions™, Smart Encounters™, and XProcess Management™ are trademarks or registered trademarks of Edifecs, Inc. in the United States and/or other countries (rights for any names or logos that are not listed are not waived by Edifecs, Inc.).

All other names may be trademarks of their respective owners.

Without limiting the rights under copyright, no part of this document may be reproduced in any form without prior written authorization from Edifecs, Inc. Some examples depicted herein are provided for illustration purposes only and are fictitious. No real association or connection is either intended or should be inferred. Edifecs, Inc. strives to ensure the accuracy of its documentation, and this document represents the current information available at the date of publication. Information in this document is subject to change without notice, and accuracy cannot be guaranteed.

CONFIDENTIALITY NOTICE This document is the property of Edifecs and contains Edifecs' confidential and proprietary information. This document, and the information contained herein, shall not be disclosed to third parties without written permission from Edifecs and shall not be duplicated, used, or distributed except in accordance with the limited conditions under which it was provided by Edifecs.

THIS DOCUMENT IS PROVIDED "AS IS" AND ALL EXPRESS, IMPLIED, OR STATUTORY WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE LEGALLY INVALID. EDIFECs, INC. SHALL NOT BE LIABLE FOR ANY INFORMATION CONTAINED IN, OR OMITTED FROM, THE DOCUMENTATION. NEITHER EDIFECs, INC. NOR ITS EMPLOYEES, CONTRACTORS, AGENTS, OR SUPPLIERS WHO HAVE BEEN INVOLVED IN THE CREATION, PRODUCTION, OR DELIVERY OF THESE MATERIALS SHALL BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, OR INCIDENTAL DAMAGES ARISING OUT OF THE USE OF, OR INABILITY TO USE, THESE MATERIALS. READERS ASSUME ALL RESPONSIBILITY FOR THE USE AND INTERPRETATION OF THE INFORMATION CONTAINED HEREIN.