ISC CERTAGENT (WINDOWS VERSION)

Integration Guide

Applicable Devices:
Vectera Plus

TABLE OF CONTENTS

[1] DOCUMENT INFORMATION

[1.1] DOCUMENT OVERVIEW

The purpose of this document is to provide information regarding the configuration of Futurex HSMs with CertAgent by ISC using PKCS #11 libraries. For additional questions related to your HSM, see the relevant administrator's guide.

This document will describe the steps for a basic installation of the CertAgent 7.0.5 application in Windows environments.

[1.2] APPLICATION DESCRIPTION

CertAgent, by Information Security Corp, is a Certificate Authority that allows users to issue X.509 certificates to devices and clients. When combined with the Vectera Plus, the signature certificates can be stored within the boundaries of a FIPS and PCI compliant hardware security module.

[1.3] COPYRIGHT AND TRADEMARK NOTICES

[1.4] TERMS OF USE

[1.5] GUARDIAN INTEGRATION

The Guardian Series 3 introduces mission-critical viability to core cryptographic infrastructure, including:

- Centralize device management
- Eliminates points of failure
- Distribute transaction loads

- Group-specific function blocking
- User-defined grouping systems

Please see applicable guide for configuring HSMs with the Guardian Series 3.

[2] PREREQUISITES

Supported Hardware:

- Vectera Plus, 6.7.x.x and above

Supported Operating Systems:

- Windows 7 and above

Other:

- OpenSSL

[3] INSTALL FUTUREX PKCS #11 (FXPKCS11)

In a Windows environment, the easiest way to install the Futurex PKCS #11 (FXPKCS11) module is through installing **FXTools**. FXTools can be downloaded from the Futurex Portal. Step by step installation instructions are provided below.

NOTE: The Futurex PKCS #11 module needs to be installed on the server that will be using the HSM.

[3.1] INSTRUCTIONS FOR INSTALLING THE PKCS #11 MODULE USING FXTOOLS IN WINDOWS

- Run the FXTools installer as an administrator

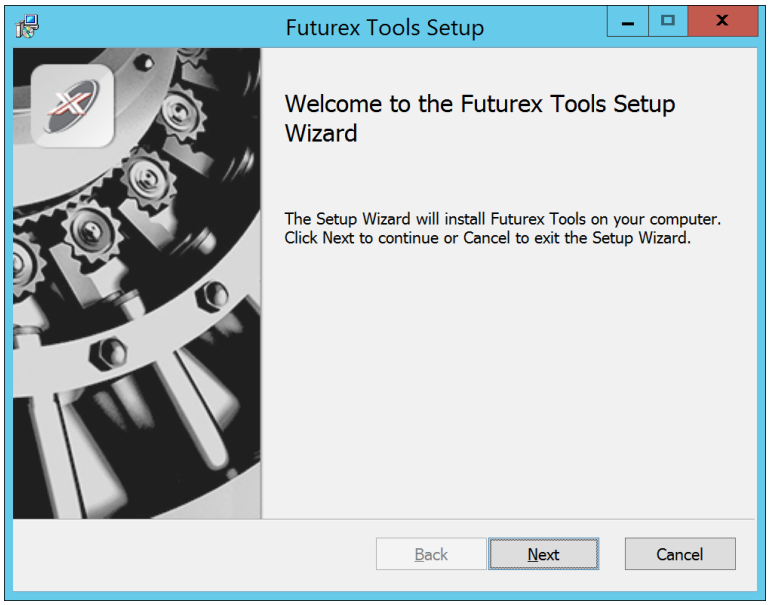


FIGURE: FUTUREX TOOLS SETUP WIZARD

By default, all tools are installed on the system. A user can overwrite and choose not to install certain modules.

- **Futurex Client Tools –** Command Line Interface (CLI) and associated SDK for both Java and C.
- **Futurex CNG Module –** The Microsoft Next Generation Cryptographic Library.
- **Futurex Cryptographic Service Provider (CSP) –** The legacy Microsoft cryptographic library.
- **Futurex EKM Module –** The Microsoft Enterprise Key Management library.
- **Futurex PKCS #11 Module –** The Futurex PKCS #11 library and associated tools.
- **Futurex Secure Access Client –** The client used to connect a Futurex Excrypt Touch to a local laptop, via USB, and a remote Futurex device.

After starting the installation, all noted services are installed. If the Futurex Secure Access Client was selected, the Futurex Excrypt Touch driver will also be installed (Note this sometimes will start minimized or in the background).

After installation is complete, all services are installed in the "*C:\Program Files\Futurex*" directory. The CNG Module, CSP Module, EKM Module, and PKCS #11 Module all require configuration files, located in their

corresponding directory with a *.cfg* extension. In addition, the CNG and CSP Modules are registered in the Windows Registry (*HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Cryptography\Defaults\Provider*) and are installed in the "*C:\Windows\System32*" directory.

[4] INSTALL EXCRYPT MANAGER (IF USING WINDOWS)

The following two sections will cover how to install the **Excrypt Manager** and **FXCLI** applications. These tools are used to configure the HSM in subsequent sections. Note that installing Excrypt Manager is optional, but installing FXCLI is required, as FXCLI is the method that is used for configuring TLS mutual authentication between the Vectera Plus and the application that is being integrated.

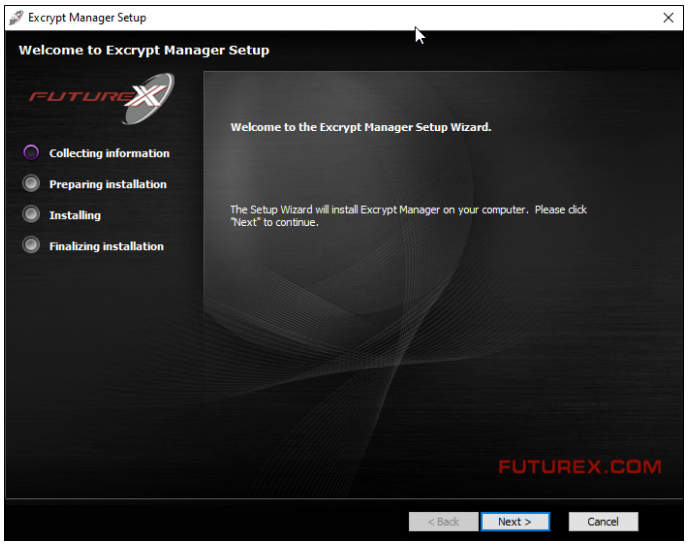
NOTE: Excrypt Manager needs to be installed on the workstation that is being used to configure the HSM.

Excrypt Manager is a Windows application that can be used to configure the HSM in subsequent sections. HSM configuration can also be completed using FXCLI, the Excrypt Touch, or the Guardian Series 3. For more information about using these tools/devices to configure the HSM, please see the relevant Administrator's Guide.

NOTE: If you plan to use a Virtual HSM for the integration, all configurations will need to be performed using either FXCLI, the Excrypt Touch, or the Guardian Series 3.

NOTE: The Excrypt Manager version must be from the 4.4.x branch or later to be compatible with the HSM firmware, which must be 6.7.x.x or later.

- Run the Excrypt Manager installer as an administrator.



The installation wizard will ask you to specify where you want Excrypt Manager to be installed. The default location is "*C:\Program Files\Futurex\Excrypt Manager*". Once that is done click "Install".

[5] INSTALL FUTUREX COMMAND LINE INTERFACE (FXCLI)

NOTE: FXCLI needs to be installed on the workstation that is being used to configure the HSM.

[5.1] INSTRUCTIONS FOR INSTALLING FXCLI IN WINDOWS

As mentioned in section 4, **Futurex Client Tools (FXCLI)** is included in the **FXTools** installation package. Just as with the **Futurex PKCS #11 (FXPKCS11)** module, the easiest way to install FXCLI on Windows is through installing FXTools. FXTools can be downloaded from the Futurex Portal.

- Run the FXTools installer as an administrator

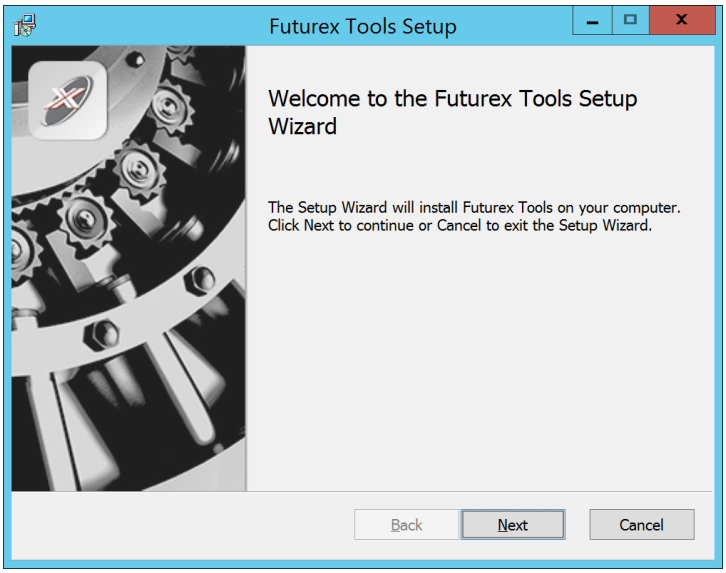


FIGURE: FUTUREX TOOLS SETUP WIZARD

By default, all tools are installed on the system. A user can overwrite and choose not to install certain modules.

NOTE: Since FXTools is only being used to install FXCLI in this case, it is not necessary to include any of the other services in the installation.

- **Futurex Client Tools –**Command Line Interface (CLI) and associated SDK for both Java and C.
- **Futurex CNG Module –**The Microsoft Next Generation Cryptographic Library.
- **Futurex Cryptographic Service Provider (CSP) –**The legacy Microsoft cryptographic library.
- **Futurex EKM Module –**The Microsoft Enterprise Key Management library.
- **Futurex PKCS #11 Module –**The Futurex PKCS #11 library and associated tools.
- **Futurex Secure Access Client –**The client used to connect a Futurex Excrypt Touch to a local laptop, via USB, and a remote Futurex device.

[5.2] INSTRUCTIONS FOR INSTALLING FXCLI IN LINUX

Download the FXCLI module

Users can download the appropriate FXCLI package files for their system from the Futurex Portal.

If the system is **64-bit**, users should select from the files marked **amd64**. If the system is **32-bit**, users should select from the files marked **i386**.

If running an OpenSSL version in the **1.0.x** branch, users should select from the files marked **ssl1.0**. If running an OpenSSL version in the **1.1.x** branch, users should select from the files marked **ssl1.1**.

Futurex offers the following features for FXCLI:

- Java Software Development Kit (**java**)
- HSM command line interface (**cli-hsm**)
- KMES command line interface (**cli-kmes**)
- Software Development Kit headers (**devel**)
- YAML parser used to parse bash output (**cli-fxparse**)

Install FXCLI

If installing an `.rpm` package, run the following command in a terminal:

```
$ sudo rpm -ivh [fxcl-xxxx.rpm]
```

If installing a `.deb` package, run the following command in a terminal:

```
$ sudo dpkg -i [fxcl-xxxx.deb]
```

After the installation is completed, system environment variables must be defined for the location of the FXCLI binaries. To do so permanently you must add the following two lines to your *.bashrc* file:

```
PATH=$PATH:/usr/bin/fxcli-hsm
PATH=$PATH:/usr/bin/fxcli-kmes
```

[6] CONFIGURE THE FUTUREX HSM

In order to establish a connection between the PKCS #11 library and the Futurex HSM, a few configuration items need to first be performed, which are the following:

NOTE: All of the steps in this section can be completed through either Excrypt Manager or FXCLI (if using a physical HSM rather than a virtual HSM). Optionally, steps 5 through 7 can be completed through the Guardian Series 3, which will be covered in Appendix A.

1. Connect to the HSM via the front USB port (NOTE: If you are using a virtual HSM for the integration you will have to connect to it over the network either via FXCLI, the Excrypt Touch, or the Guardian Series 3)
    a. Connecting via Excrypt Manager
    b. Connecting via FXCLI
2. Validate the correct features are enabled on the HSM
3. Setup the network configuration
4. Enable the EDSV multi-usage combination for asymmetric keys
5. Load the Futurex FTK
6. Configure a Transaction Processing connection and create a new Application Partition
7. Create a new Identity that has access to the Application Partition created in the previous step
8. Configure TLS Authentication. There are two options for this:
    a. Enabling server-side authentication
    b. Creating client certificates for mutual authentication

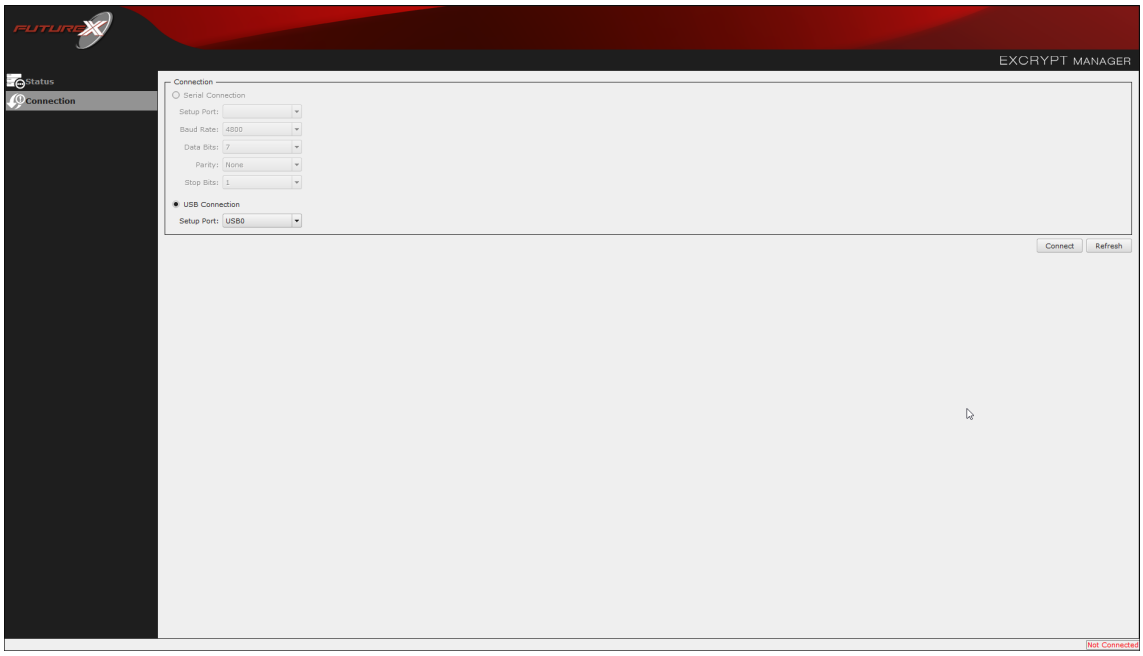
Each of these action items is detailed in the following subsections.

[6.1] CONNECT TO THE HSM VIA THE FRONT USB PORT

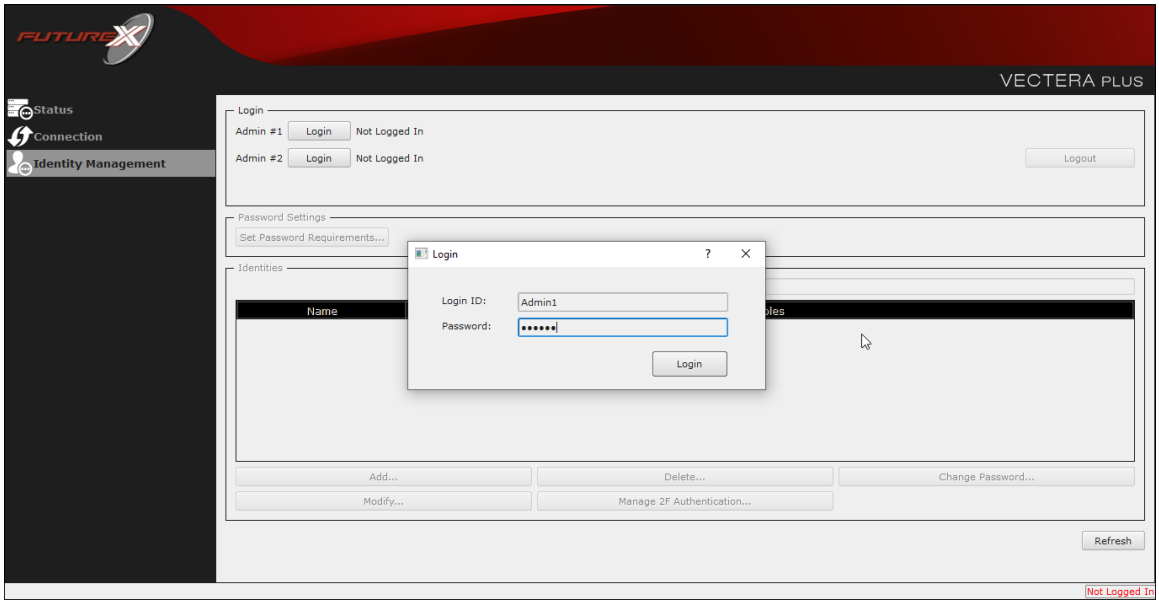
For both Excrypt Manager and FXCLI you need to connect your laptop to the front USB port on the HSM.

Connecting via Excrypt Manager

Open Excrypt Manager, click “Refresh” in the lower right-hand side of the Connection menu. Then select “USB Connection” and click “Connect”.

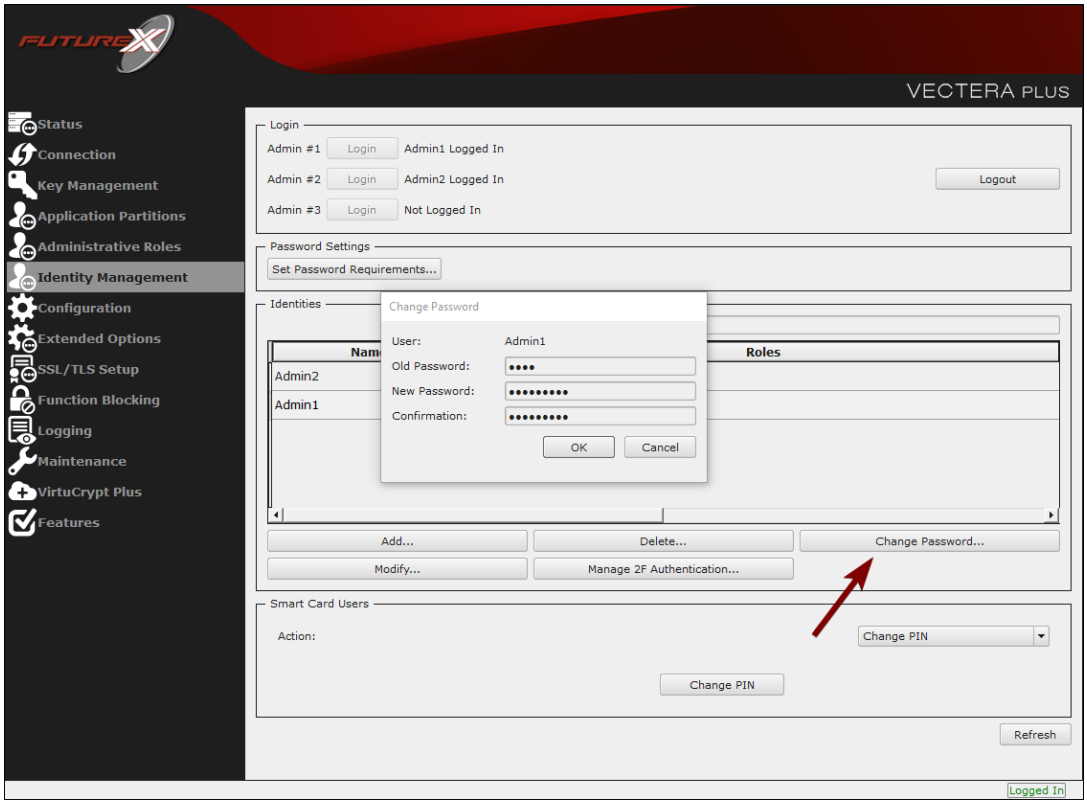


Login with both default Admin identities.



The default Admin passwords (i.e. “safe”) must be changed for both of your default Admin Identities (e.g. “Admin1” and “Admin2”) in order to load the major keys onto the HSM.

To do so via Excrypt Manager navigate to the Identity Management menu, select the first default Admin identity (e.g. “Admin1”), then click the “Change Password…” button. Enter the old password, then enter the new password twice, and click “OK”. Perform the same steps as above for the second default Admin identity (e.g. “Admin2”).



Connecting via FXCLI

Open the FXCLI application and run the following commands:

```
$ connect usb
$ login user
```

NOTE: The "login" command will prompt for the username and password. You will need to run it twice because you must login with both default Admin identities.

The default Admin passwords (i.e. “safe”) must be changed for both of your default Admin Identities (e.g. “Admin1” and “Admin2”) in order to load the major keys onto the HSM.

The following FXCLI commands can be used to change the passwords for each default Admin Identity.

```
$ user change-password -u Admin1
$ user change-password -u Admin2
```

NOTE: The user change-password commands above will prompt you to enter the old and new passwords. It is necessary to run the command twice (as shown above) because the default password must be changed for both default Admin identities.

[6.2] FEATURES REQUIRED IN HSM

In order to establish a connection between the PKCS #11 Library and the Futurex HSM, the HSM must be configured with the following features:

- **PKCS #11** -> Enabled
- **Command Primary Mode** -> General Purpose (GP)

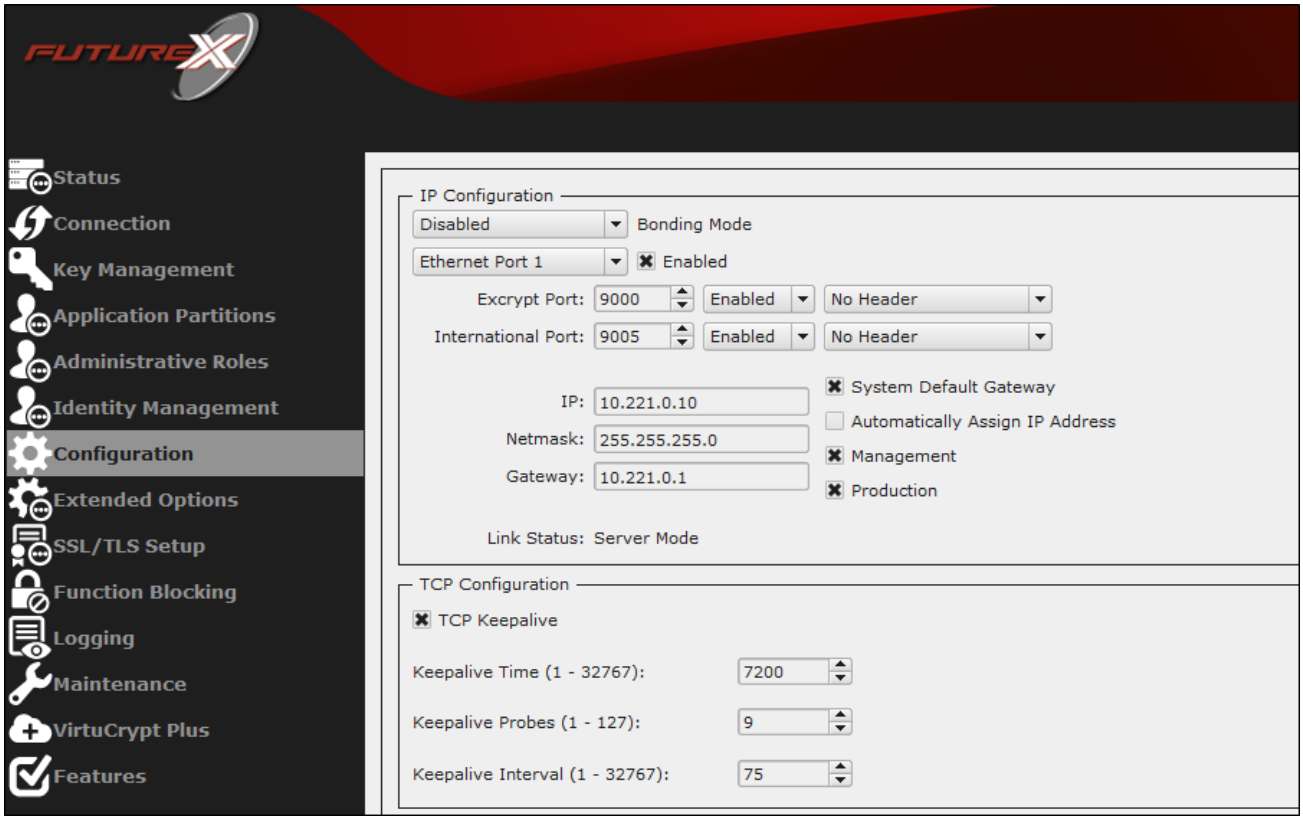
NOTE: For additional information about how to update features on your HSM, please refer to your HSM Administrator's Guide, section **"Download Feature Request File"**.

NOTE: Command Primary Mode = General Purpose, will enable the option to create the FTK major key in the HSM. This key will be required to be able to use the PKCS #11 library to communicate with the HSM. For detailed information about how to load major keys in HSMs please refer to your HSM Administrator's Guide.

[6.3] NETWORK CONFIGURATION (HOW TO SET THE IP OF THE HSM)

For this step you will need to be logged in with an identity that has a role with permissions ***Communication:Network Settings****. The default Administrator role and Admin identities can be used.*

Navigate to the *Configuration* page. There you will see the option to modify the IP configuration, as shown below:



Alternatively, the following **FXCLI** command can be used to set the IP for the HSM:

```
$ network interface modify --interface Ethernet1 --ip 10.221.0.10 --netmask 255.255.255.0 --gateway
10.221.0.1
```

NOTE: The following should be considered at this point:

- All of the remaining HSM configurations in this section can be completed using the Guardian Series 3 (please refer to Appendix A for instructions on how to do so), with the exception of the final subsection that covers how to create connection certificates for mutual authentication.

- If you are performing the configuration on the HSM directly now, but plan to add the HSM to a Guardian later, it may be necessary to synchronize the HSM after it is added to a Device Group on the Guardian.

- If configuration through a CLI is required for your use-case, then you should manage the HSMs directly.

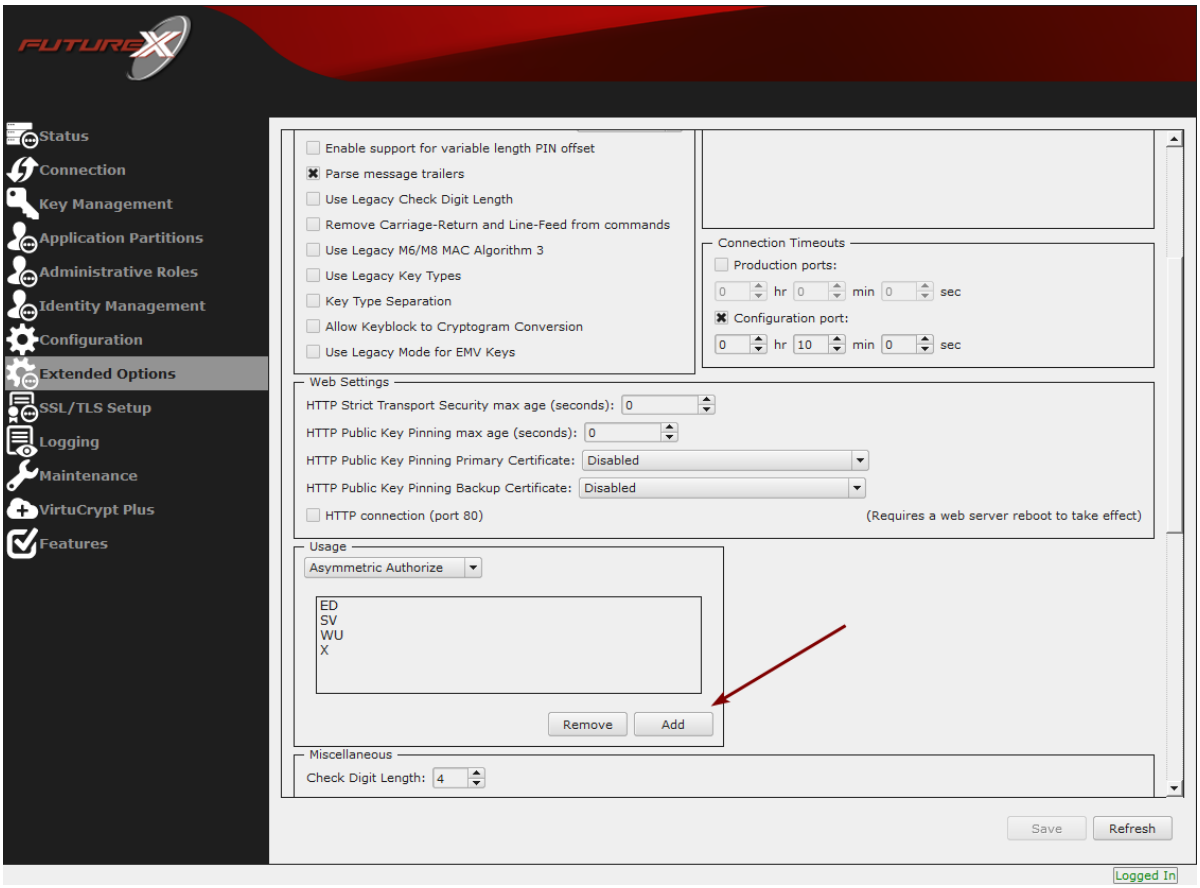
[6.4] ENABLE THE EDSV MULTI-USAGE COMBINATION FOR ASYMMETRIC KEYS

*For this step you will need to be logged in with an identity that has a role with permissions **Security:Key Settings**. The default Administrator role and Admin identities can be used.*

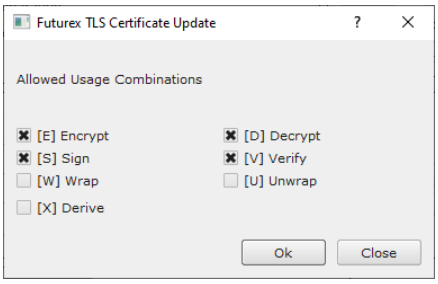
The CertAgent application requires asymmetric keys with multiple usages, which can be configured, but is not enabled by default on the Vectera Plus.

The specific multi-usage combination that CertAgent requires is EDSV. To configure this via Excrypt Manager, navigate to the *Extended Options* menu. In the "Usage" section, there is the option to add a new usage combination, as shown below.

NOTE: "Asymmetric Authorize" should be selected in the drop-down to specify that only authorized users can create asymmetric keys with the EDSV usage combination.



Select the EDSV usage combination and click "Ok".



Click the "Save" button on the bottom-right-hand side of the window to save the changes.

Alternatively, the following **FXCLI** command can be used to add the EDSV multi-usage combination for asymmetric keys for authorized users:

```
$ multi-usage add --asymmetric --auth -edsv
```

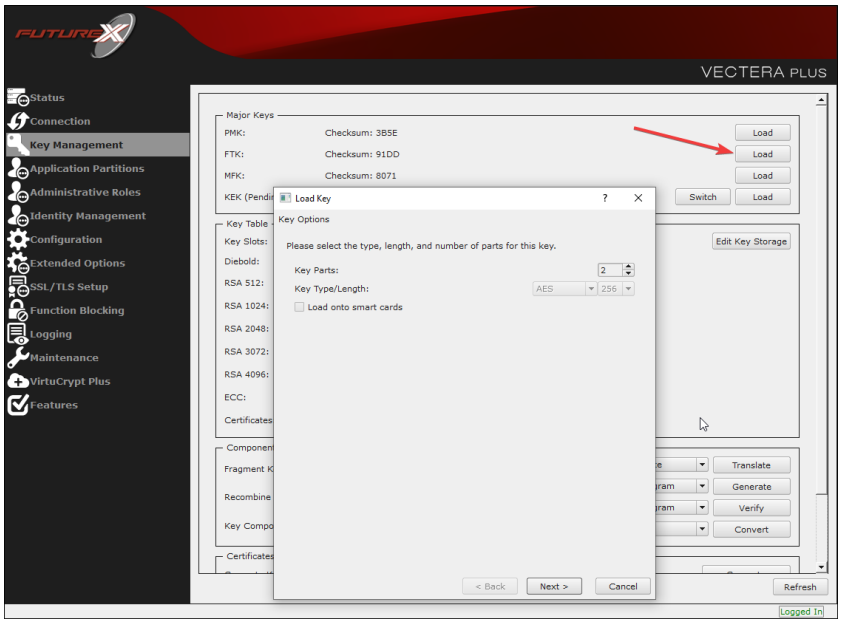
[6.5] LOAD FUTUREX KEY (FTK)

*For this step you will need to be logged in with an identity that has a role with permissions **Major Keys:Load**. The default Administrator role and Admin identities can be used.*

The FTK is used to wrap all keys stored on the HSM used with PKCS #11. If using multiple HSMs in a cluster, the same FTK can be used for syncing HSMs. Before an HSM can be used with PKCS #11, it must have an FTK.

NOTE: This process can also be completed using FXCLI, the Excrypt Touch, or the Guardian Series 3. For more information about how to load the FTK into an HSM using these tools/devices, please see the relevant Administrative Guide.

After logging in, select *Key Management*, then “Load” under FTK. Keys can be loaded as components that are XOR’d together, M-of-N fragments, or generated. If this is the first HSM in a cluster, it is recommended to generate the key and save to smart cards as M-of-N fragments.



Alternatively, the following **FXCLI** commands can be used to load an FTK onto an HSM.

If this is the first HSM you are setting up you will need to generate a random FTK. Optionally, you can also load it onto smart cards simultaneously with the -m and -n flags.

```
$ majorkey random --ftk -m [number_from_2_to_9] -n [number_from_2_to_9]
```

If it's a second HSM that you're setting up in a cluster then you will load the FTK from smart cards with the following command:

```
$ majorkey recombine --key ftk
```

[6.6] CONFIGURE A TRANSACTION PROCESSING CONNECTION AND CREATE AN APPLICATION PARTITION

For this step you will need to be logged in with an identity that has a role with permissions ***Role:Add, Role:Assign All Permissions, Role:Modify, Keys:All Slots****, and* ***Command Settings:Excrypt****. The default Administrator role and Admin identities can be used.*

NOTE: For the purposes of this integration guide you can consider the terms "Application Partition" and "Role" to be synonymous. For more information regarding Application Partitions, Roles, and Identities, please refer to the relevant Administrator's guide.

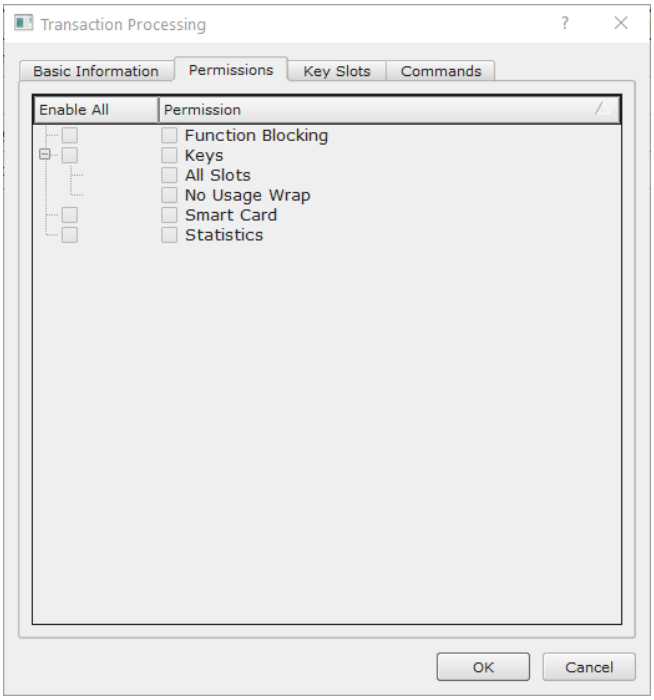
Configure a Transaction Processing Connection

Before an application logs in to the HSM with an authenticated user, it first connects via a “Transaction Processing” connection to the **Transaction Processing** Application Partition. For this reason, it is necessary to take steps to harden this Application Partition. The following three things need to be configured for the Transaction Processing partition:

1. It should not have access to the “All Slots” permissions
2. It should not have access to any key slots
3. Only the PKCS #11 communication commands should be enabled

Go to *Application Partitions*, select the Transaction Processing Application Partition, and click Modify.

Navigate to the "Permissions" tab and ensure that the “All Slots” key permission is unchecked. None of the other key permissions should be enabled either.



Under the "Key Slots" tab you need to ensure that there are no key ranges specified. By default, the Transaction Processing Application Partition has access to the entire range of key slots on the HSM.

Lastly, under the “Commands” tab make sure that only the following **PKCS #11 Communication commands** are enabled:

- **ECHO**: Communication Test/Retrieve Version
- **PRMD**: Retrieve HSM restrictions
- **RAND**: Generate random data
- **HASH**: Retrieve device serial
- **GPKM**: Retrieve key table information
- **GPKS**: General purpose key settings get/change
- **GPKR**: General purpose key settings get (read-only)

Alternatively, the following **FXCLI** commands can be used to remove all permissions and key ranges that are currently assigned to the **Transaction Processing** role and enable only the PKCS #11 Communication commands:

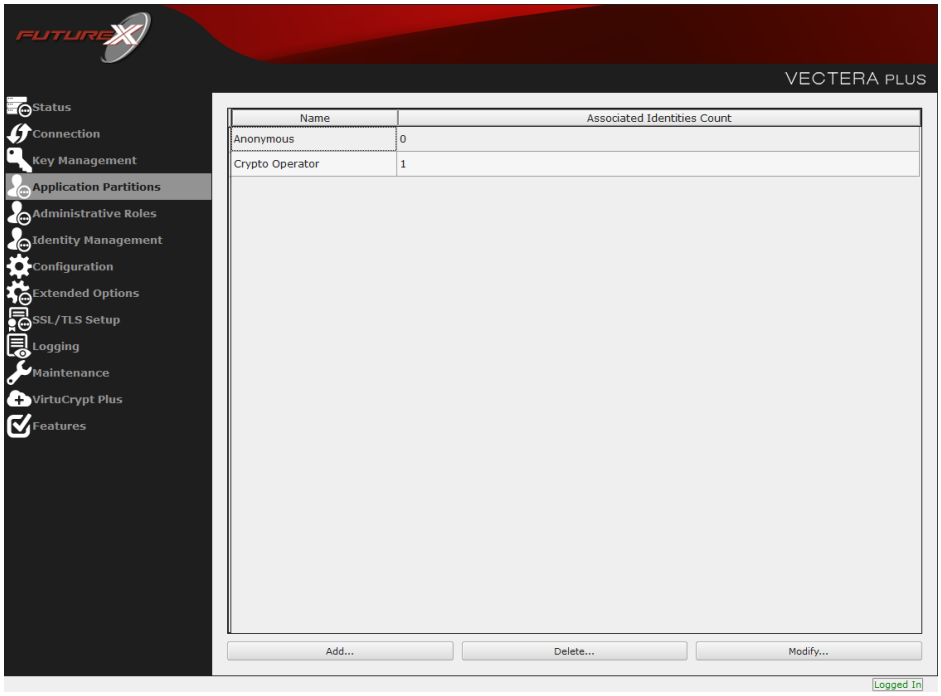
```
$ role modify --name Anonymous --clear-perms --clear-key-ranges
```

```
$ role modify --name Anonymous --add-perm Excrypt:ECHO --add-perm Excrypt:PRMD --add-perm Excrypt:RAND
--add-perm Excrypt:HASH --add-perm Excrypt:GPKM --add-perm Excrypt:GPKS --add-perm Excrypt:GPKR
```

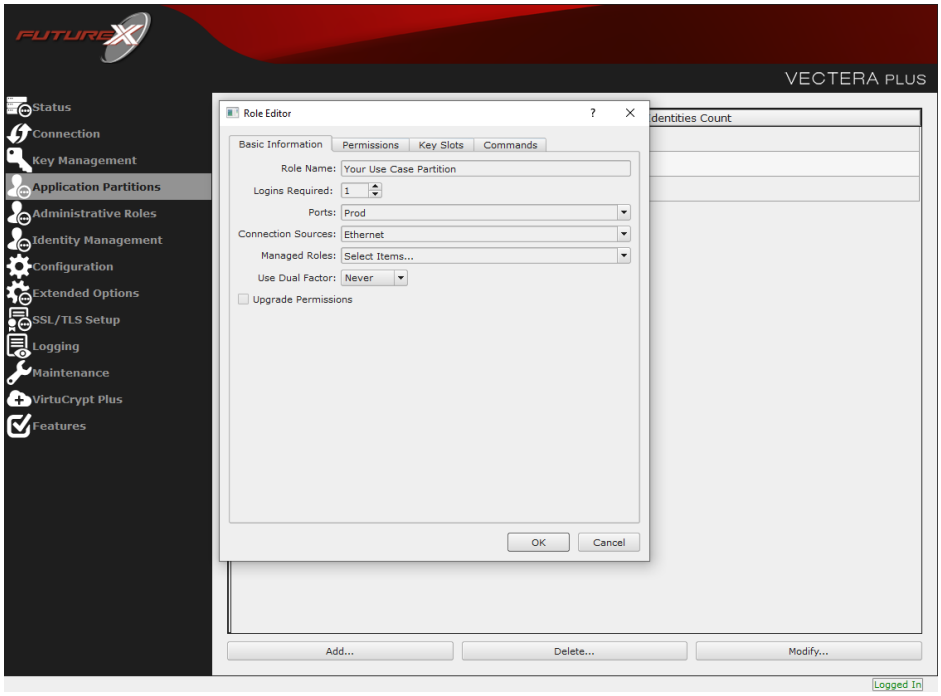
Create an Application Partition

In order for application segregation to occur on the HSM, an Application Partition must be created specifically for your use case. Application partitions are used to segment the permissions and keys on an HSM between applications. The process for configuring a new application partition is outlined in the following steps:

Navigate to the *Application Partitions* page and click the "Add" button at the bottom.

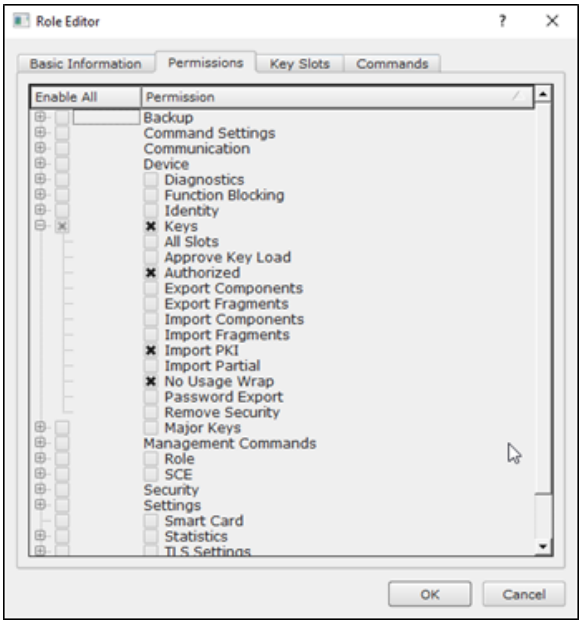


Fill in all of the fields in the *Basic Information* tab exactly how you see below (except for the *Role Name* field). In the *Role Name* field, specify any name that you would like for this new Application Partition. *Logins Required* should be set to "1". *Ports* should be set to "Prod". *Connection Sources* should be configured to "Ethernet". The *Managed Roles* field should be left blank because we'll be specifying the exact Permissions, Key Slots, and Commands that we want this Application Partition/Role to have access to. Lastly, the *Use Dual Factor* field should be set to "Never".

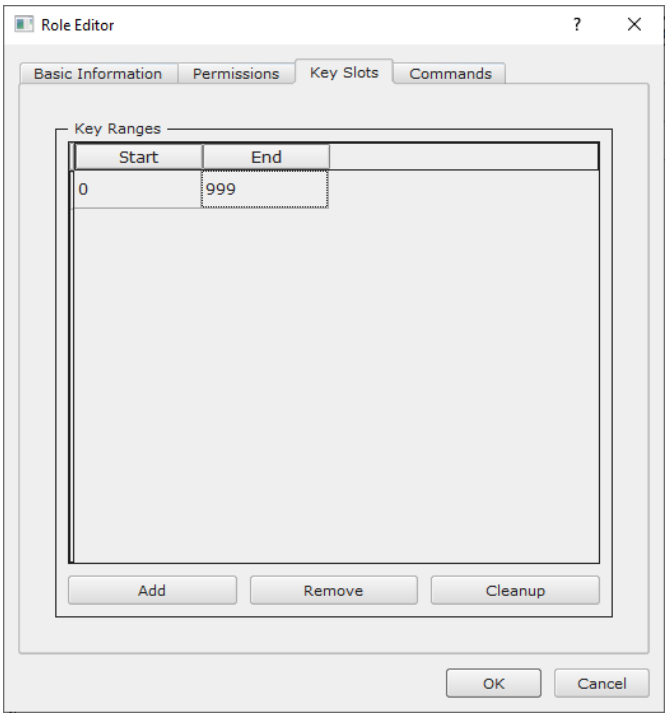


Under the "Permissions" tab, select the key permissions shown in the screenshot below. The **Authorized** permission allows for keys that require login. The **Import PKI** permission allows trusting an external PKI, which

is used by some applications to allow for PKI symmetric key wrapping (It is not recommended to enable unless using this use case). The **No Usage Wrap** permission allows for interoperable key wrapping without defining key usage as part of the wrapped key (This is only recommended if exchanging keys with external entities or using the HSM to wrap externally used keys).



Under key Slots, it is recommended that you create a range of 1000 total keys (here we've specified the key range 0-999), which do not overlap with another Application Partition. Within this range, there must be ranges for both symmetric and asymmetric keys. If more keys are required by the application, configure accordingly.



Based on application requirements there are particular functions that need to be enabled on the Application Partition in order to utilize the HSMs functionality. The most often used commands are included below. These

can be enabled under the "Commands" tab.

PKCS #11 Communication Commands

- ECHO: Communication Test/Retrieve Version
- PRMD: Retrieve HSM restrictions
- RAND: Generate random data
- HASH: Retrieve device serial
- GPKM: Retrieve key table information
- GPKS: General purpose key settings get/change
- GPKR: General purpose key settings get (read-only)

Key Operations Commands

- APFP: Generate PKI Public Key from Private Key
- ASYL: Load asymmetric key into key table
- GECC: Generate an ECC Key Pair
- GPCA: General purpose add certificate to key table
- GPGS: General purpose generate symmetric key
- GPKA: General purpose key add
- GPKD: General purpose key slot delete/clear
- GRSA: Generate RSA Private and Public Key
- LRSA: Load key into RSA Key Table
- RPFP: Get public components from RSA private key

Interoperable Key Wrapping

- GPKU: General purpose key unwrap (unrestricted)
- GPUK: General purpose key unwrap (preserves key usage)
- GPKW: General purpose key wrap (unrestricted)
- GPWK: General purpose key wrap (preserves key usage)

Data Encryption Commands

- ADPK: PKI Decrypt Trusted Public Key
- GHSH: Generate a Hash (Message Digest)
  *Starting in firmware version 7.x, this function is enabled by default and does not need to be specified.
- GPED: General purpose data encrypt and decrypt
- GPGC: General purpose generate cryptogram from key slot
- GPMC: General purpose MAC (Message Authentication Code)
- GPSR: General purpose RSA encrypt/decrypt or sign/verify with recovery
- HMAC: Generate a hash-based message authentication code
- RDPK: Get Clear Public Key from Cryptogram

Signing Commands

- ASYS: Generate a Signature Using a Private Key
- ASYV: Verify a Signature Using a Public Key
- GPSV: General purpose data sign and verify
- RSAS: Generate a Signature Using a Private Key

Alternatively, the following **FXCLI** commands can be used to create the new Application Partition and enable all of the functions that are needed:

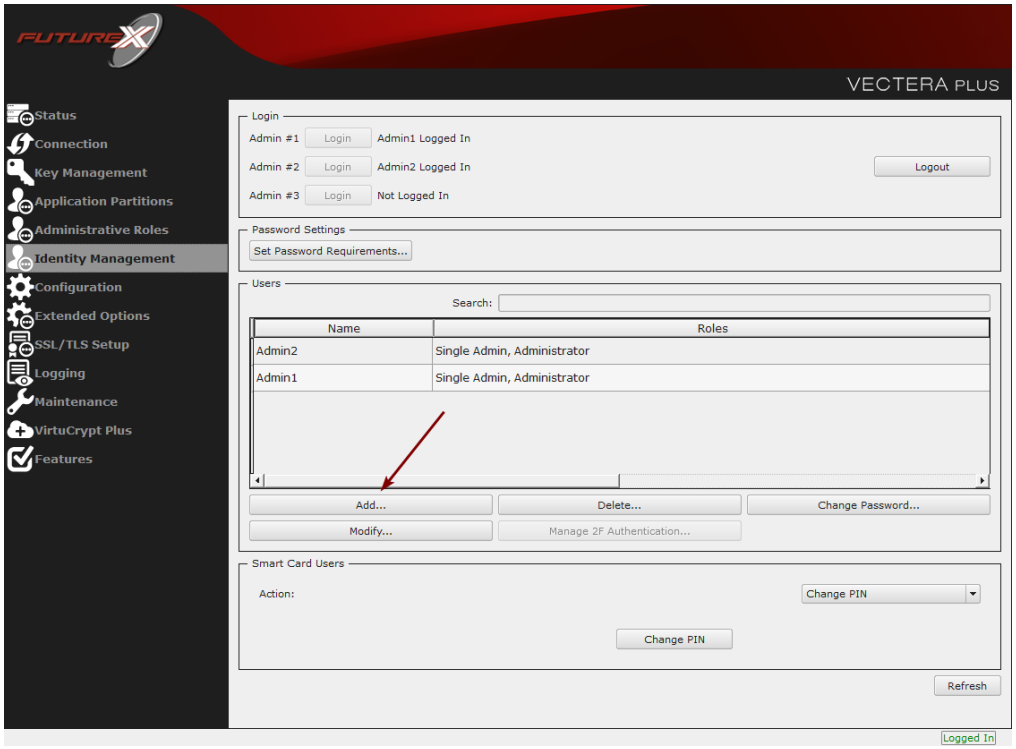
```
$ role add --name Role_Name --application --key-range (0,999) --perm "Keys:Authorized" --perm "Key-
s:Import PKI" --perm "Keys:No Usage Wrap"
```

```
$ role modify --name [role_name] --clear-perms --add-perm Excrypt:ECHO --add-perm Excrypt:PRMD --add-
perm Excrypt:RAND --add-perm Excrypt:HASH --add-perm Excrypt:GPKM --add-perm Excrypt:GPKS --add-perm
Excrypt:GPKR --add-perm Excrypt:APFP --add-perm Excrypt:ASYL --add-perm Excrypt:GECC --add-perm
Excrypt:GPCA --add-perm Excrypt:GPGS --add-perm Excrypt:GPKA --add-perm Excrypt:GPKD --add-perm
Excrypt:GRSA --add-perm Excrypt:LRSA --add-perm Excrypt:RPFP --add-perm Excrypt:GPKU --add-perm
Excrypt:GPUK --add-perm Excrypt:GPKW --add-perm Excrypt:GPWK --add-perm Excrypt:ADPK --add-perm
Excrypt:GHSH --add-perm Excrypt:GPED --add-perm Excrypt:GPGC --add-perm Excrypt:GPMC --add-perm
Excrypt:GPSR --add-perm Excrypt:HMAC --add-perm Excrypt:RDPK --add-perm Excrypt:ASYS --add-perm
Excrypt:ASYV --add-perm Excrypt:GPSV --add-perm Excrypt:RSAS
```

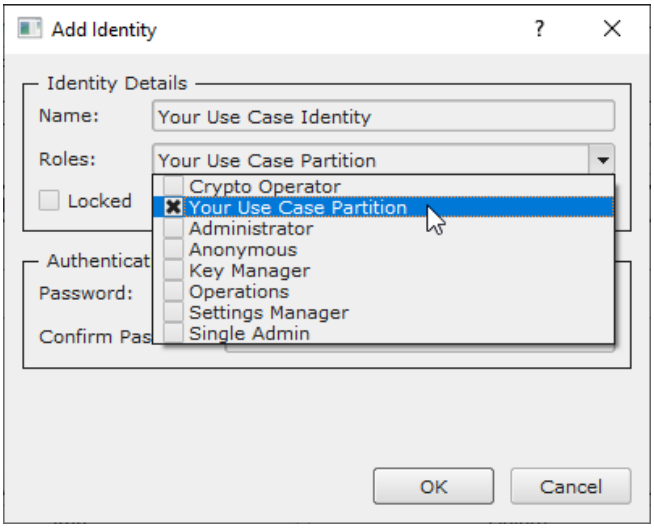
[6.7] CREATE NEW IDENTITY AND ASSOCIATE IT WITH THE NEWLY CREATED APPLICATION PARTITION

*For this step you will need to be logged in with an identity that has a role with permissions **Identity:Add**. The default Administrator role and Admin identities can be used.*

A new identity must be created, which will need to be associated with the Application Partition created in the previous step. To create this new identity, go to *Identity Management*, and click "Add".



Specify a name for the new identity, and in the Roles dropdown select the name of the Application Partition created in the previous step. This will associate the new Identity with the Application Partition that you created.



Alternatively, the following **FXCLI** command can be used to create a new Identity and associate it with the role that was created:

```
$ identity add --name Identity_Name --role Role_Name --password safest
```

This new identity must be set in fxpkcs11.cfg file, in the following section:

```
#HSM crypto operator identity name
<CRYPTO-OPR>     [insert name of identity that you created]     </CRYPTO-OPR>

# Production connection
<PROD-ENABLED>    YES        </PROD-ENABLED>
<PROD-PORT>        9100         </PROD-PORT>
```

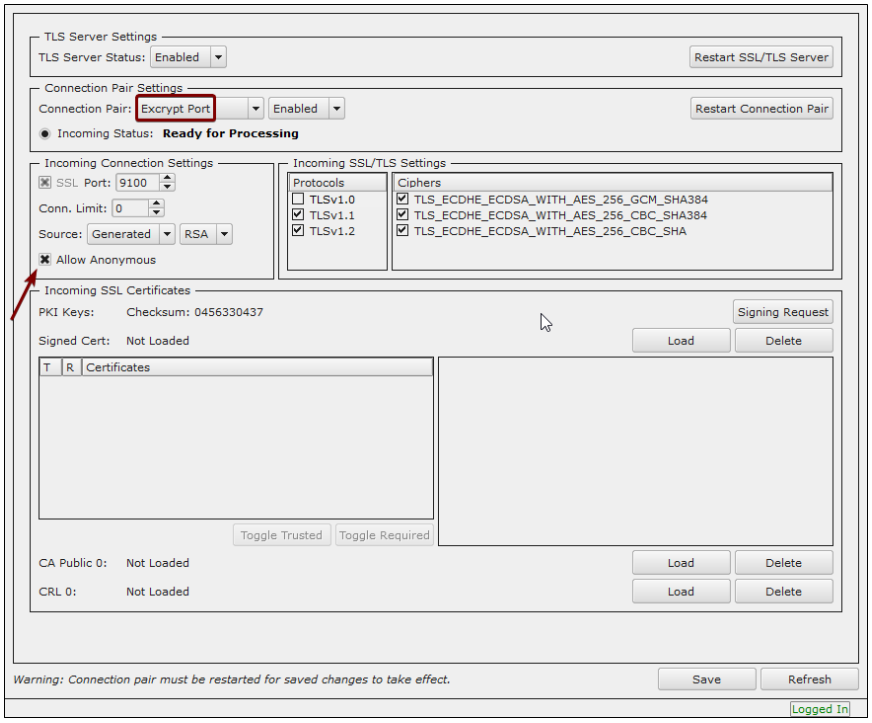
NOTE: Crypto Operator in the fxpkcs11.cfg file must match exactly the name of the identity created in the HSM.

[6.8] CONFIGURE TLS AUTHENTICATION

*For this step you will need to be logged in with an identity that has a role with permissions **Keys:All Slots, Management Commands:Certificates, Management Commands:Keys, Security:TLS Sign**, and **TLS Settings:Upload Key**. The default Administrator role and Admin identities can be used.*

Enable Server-Side Authentication (Option 1)

Mutually authenticating to the HSM using client certificates is recommended, but server-side authentication is also supported. To enable server-side authentication go to *SSL/TLS Setup*, then select the Excrypt Port and enable the "Allow Anonymous" setting.



Alternatively, the following **FXCLI** command can be used to enable server-side authentication with the "Allow Anonymous" SSL/TLS setting:

```
$ tls-ports set -p "Excrypt Port" --anon
```

Create Connection Certificates for Mutual Authentication (Option 2)

Mutually authenticating to the HSM using client certificates is recommended, and enforced by default. In the example below, FXCLI is utilized to generate a CA that then signs the HSM server certificate and a client certificate. The client keys and CSR are generated in Windows PowerShell with OpenSSL. For other options for managing certificates required for mutual authentication with the HSM, please review the relevant Administrator's guide.

Find the **FXCLI** program that was installed with FXTools, and run it as an administrator.

Things to note:

- For this example, the computer running FXCLI is connected to the front port of the HSM. Remote management is possible however, using the HSMs Web Portal, or the Excrypt Touch.
- For commands that create an output file, if you do not specify a file path (as is the case here) it will save the file to the directory from which the FXCLI program is executed.
- Using user-generated certificates requires a PMK to be loaded on the HSM.
- If you run **help** by itself it will show a full list of available commands. You can see all of the available options for any given command by running the command name followed by **help**.

```
# Connect your laptop to the HSM via the USB port on the front, then run this command.
$ connect usb
```

```
# Log in with both default Admin identities. This command will prompt for the username and password.
You will need to run this command twice.
$ login user
```

```
# Generate TLS CA and store it in an available key slot on the HSM
$ generate --algo RSA --bits 2048 --usage mak --name TlsCaKeyPair --slot next
```

```
# Create root certificate
$ x509 sign \
    --private-slot TlsCaKeyPair \
    --key-usage DigitalSignature --key-usage KeyCertSign \
    --ca true --pathlen 0 \
    --dn 'O=Futurex\CN=Root' \
    --out TlsCa.pem
```

```
# Generate the server keys for the HSM
$ tls-ports request --pair "Excrypt Port" --file production.csr --pki-algo RSA
```

```
# Sign the server CSR with the newly created TLS CA
$ x509 sign \
    --private-slot TlsCaKeyPair \
    --issuer TlsCa.pem \
    --csr production.csr \
    --eku Server --key-usage DigitalSignature --key-usage KeyAgreement \
    --ca false \
    --dn 'O=Futurex\CN=Production' \
    --out TlsProduction.pem
```

```
# Push the signed server PKI to the production port on the HSM
$ tls-ports set --pair "Excrypt Port" \
    --enable \
    --pki-source Generated \
    --clear-pki \
    --ca TlsCa.pem \
    --cert TlsProduction.pem \
    --no-anon
```

***NOTE**: The following OpenSSL commands will need to be run from Windows PowerShell, rather than from the FXCLI program.*

```
# Generate the client keys
$ openssl genrsa -out privatekey.pem 2048
```

```
# Generate client CSR
$ openssl req -new -key privatekey.pem -out ClientPki.csr -days 365
```

Using FXCLI, sign the CSR that was just generated using OpenSSL.

```
# Sign the client CSR under the root certificate that was created
$ x509 sign  \
--private-slot TlsCaKeyPair \
 --issuer TlsCa.pem \
 --csr ClientPki.csr \
 --eku Client --key-usage DigitalSignature --key-usage KeyAgreement \
 --dn 'O=Futurex\CN=Client' \
  --out SignedPki.pem
```

Switch back to Windows PowerShell for the remaining commands.

```
## Make PKCS12 file
# Concatenate the signed client cert and private key into one pem file
$ cat SignedPki.pem >> Tree.pem
```

```
$ cat privatekey.pem >> Tree.pem
```

```
# Use OpenSSL to create a PKCS#12 file that can be used to authenticate, as a client, using our PKCS
#11 library
$ openssl pkcs12 -export -in Tree.pem -out PKI.p12 -name "ClientPki" -password pass:safest
```

[7] EDIT THE FXPKCS11 CONFIGURATION FILE

The *fxpkcs11.cfg* file allows the user to set the PKCS #11 library to connect to the HSM. To edit, run a text editor as an Administrator and edit the configuration file accordingly. Most notably, the fields shown below must be set inside the **<HSM>** section (note that the full *fxpkcs11.cfg* file is not included).

NOTE: Our PKCS #11 library expects the PKCS #11 config file to be in a certain location (*C:\Program Files\Futurex\fxpkcs11\fxpkcs11.cfg* for Windows and */etc/fxpkcs11.cfg* for Linux), but that location can be overwritten using an environment variable (FXPKCS11_CFG).

```
# Connection information
<ADDRESS>             10.0.5.58           </ADDRESS>

# Load balancing
<FX-LOAD-BALANCE>           YES               </FX-LOAD-BALANCE>

# Log configuration
<LOG-FILE> C:\Program Files\Futurex\fxpkcs11\fxpkcs11.log </LOG-FILE>

# HSM crypto operator identity name
<CRYPTO-OPR>          [identity_name]     </CRYPTO-OPR>

# Production connection
<PROD-ENABLED>        YES                 </PROD-ENABLED>
<PROD-PORT>           9100                </PROD-PORT>

# Production SSL information
<PROD-TLS-ANONYMOUS>    NO                </PROD-TLS-ANONYMOUS>
<PROD-TLS-CA>     C:\Program Files\Futurex\fxpkcs11\TlsCa.pem    </PROD-TLS-CA>
<PROD-TLS-CA>     C:\Program Files\Futurex\fxpkcs11\TlsProduction.pem    </PROD-TLS-CA>
<PROD-TLS-KEY>     C:\Program Files\Futurex\fxpkcs11\PKI.p12     </PROD-TLS-KEY>
<PROD-TLS-KEY-PASS>      safest           </PROD-TLS-KEY-PASS>
```

In the **<ADDRESS>** field, the IP of the HSM that the PKCS #11 library will connect to is specified.

If a Guardian is being used to manage HSMs in a cluster, the **<FX-LOAD-BALANCE>** field must be defined as "YES". If a Guardian is not being used it should be set to "NO".

In the **<LOG-FILE>** field, set the path to the PKCS #11 log file.

In the **<CRYPTO-OPR>** field, the name of the identity created in step 7.6 needs to be specified.

The **<PROD-ENABLED>** and **<PROD-PORT>** fields declare that the PKCS #11 library will connect to Production port 9100.

The **<PROD-TLS-ANONYMOUS>** field defines whether the PKCS #11 library will be authenticating to the server or not.

The **<PROD-TLS-KEY>** field defines the location of the client private key. Supported formats for the TLS private key are PKCS #1 clear private keys, PKCS #8 encrypted private keys, or a PKCS #12 file that contains the private key and certificates encrypted under the password specified in the **<PROD-TLS-KEY-PASS>** field.

Because a PKCS #12 file is defined in the **<PROD-TLS-KEY>** field in this example, it is not necessary to define the signed client cert with the **<PROD-TLS-CERT>** tag, or the CA cert/s with one or more instances of the **<PROD-TLS-CA>** tag.

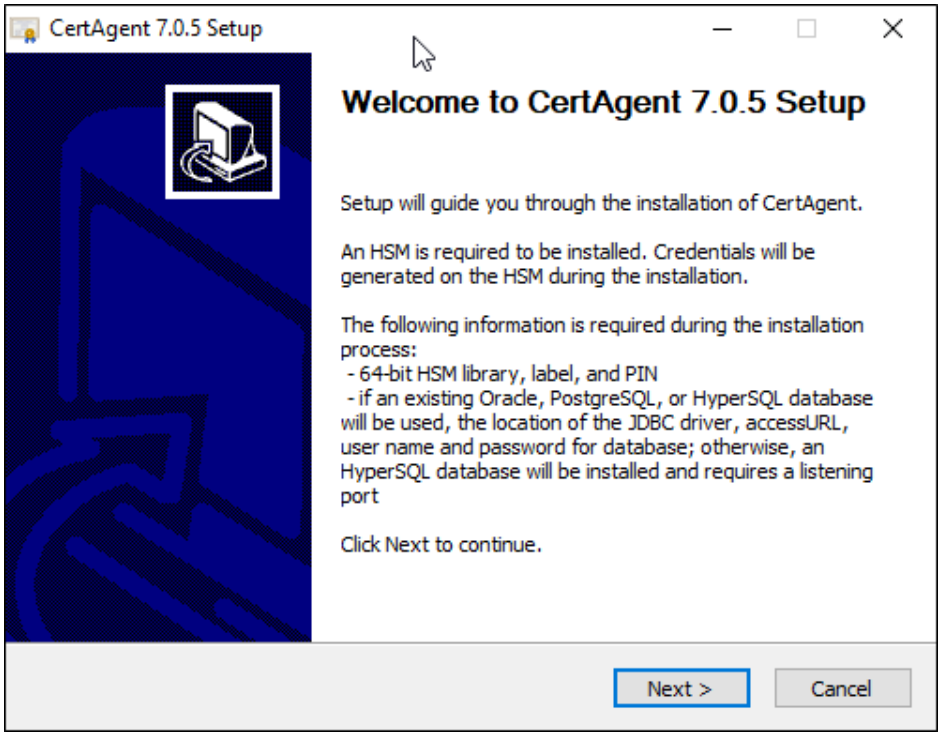
For additional details reference the Futurex PKCS #11 technical reference found on the Futurex Portal.

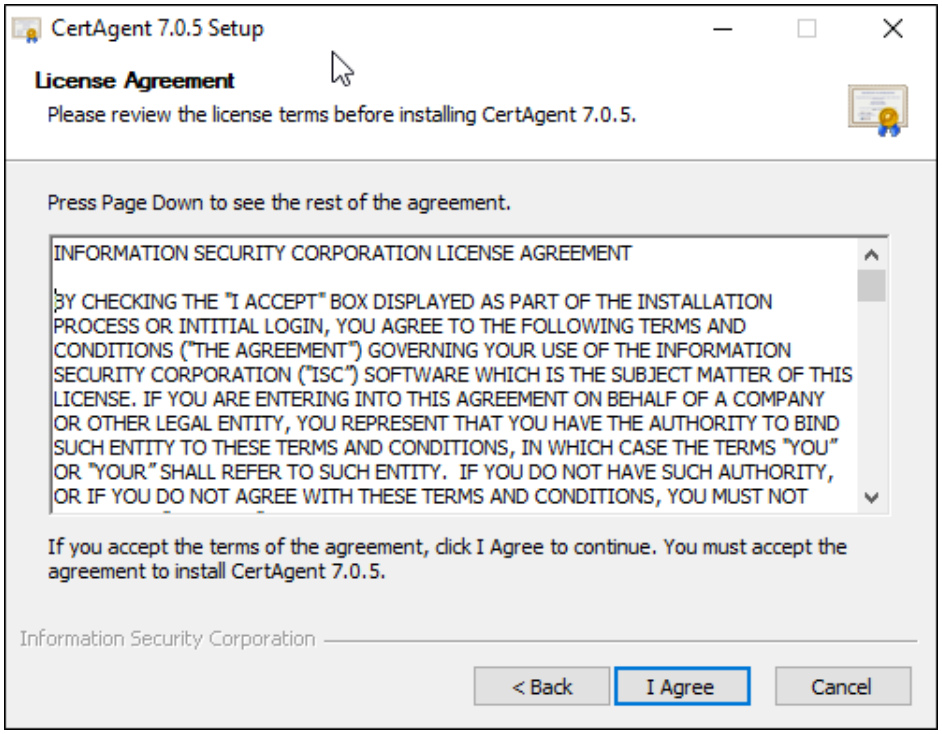
Once the *fxpkcs11.cfg* is edited, run the *PKCS11Manager* file to test the connection against the HSM, and check the *fxpkcs11.log* for errors and information. For more information, see our Administrator's Guide.

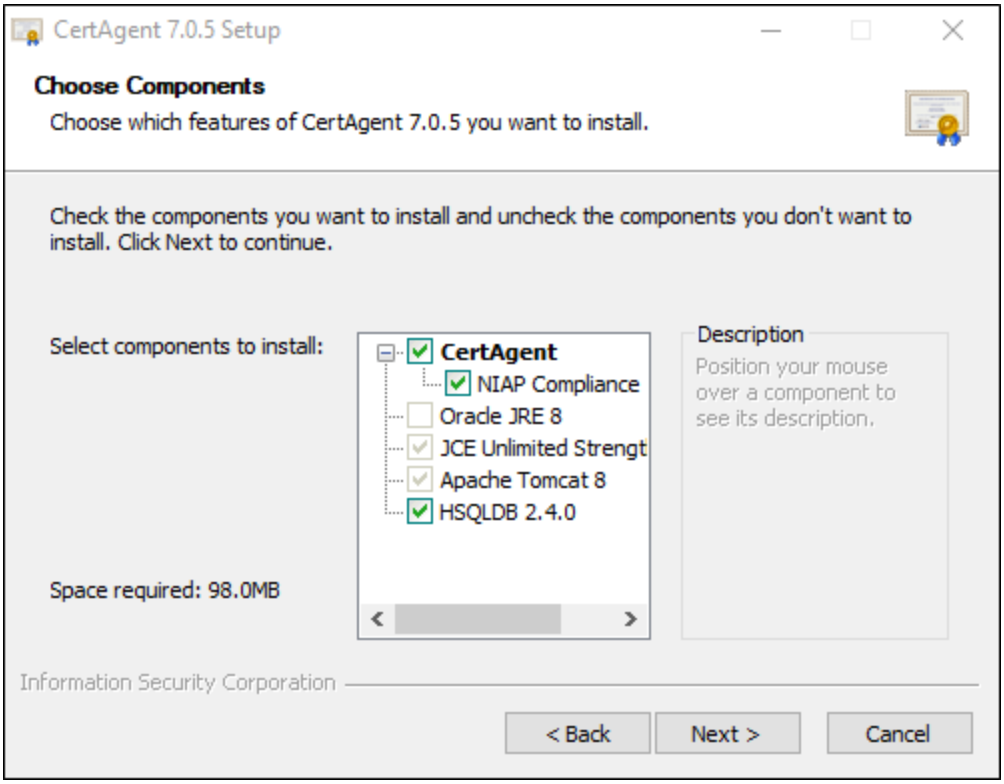
[8] STEPS TO LOAD THE FUTUREX PKCS #11 LIBRARY INTO CERTAGENT

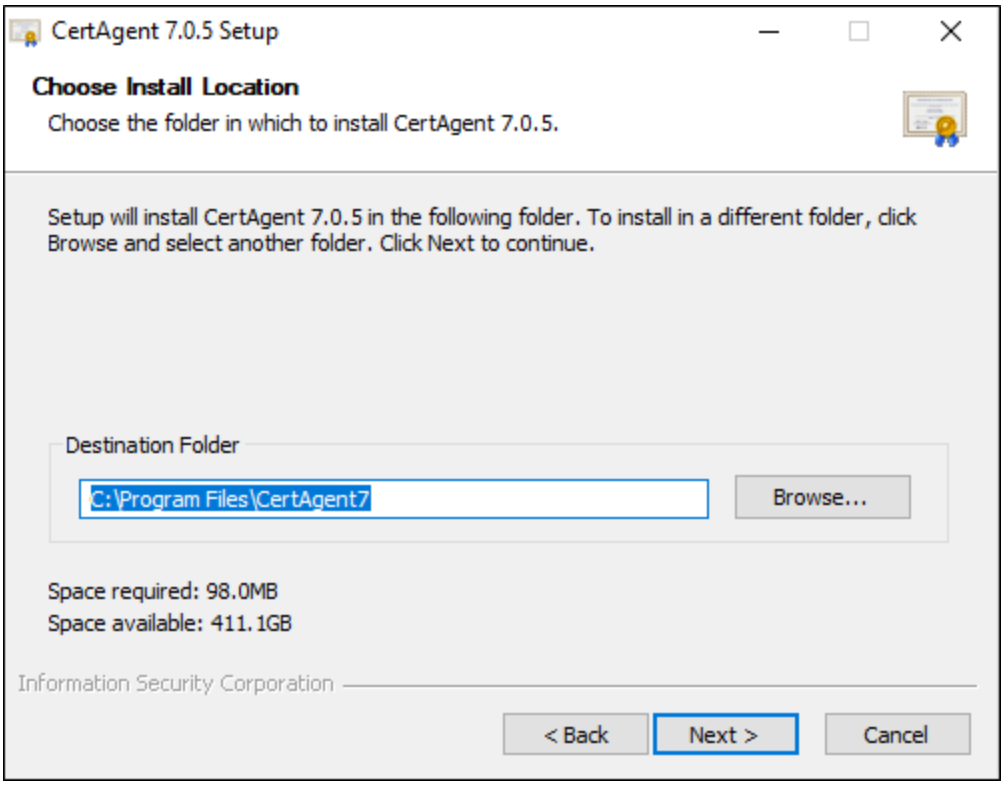
The Web based interface used by CertAgent is supported by Internet Explorer and Firefox.

1. Double click on: Certagent.7.0.5.x64.exe and follow the on-screen instructions:

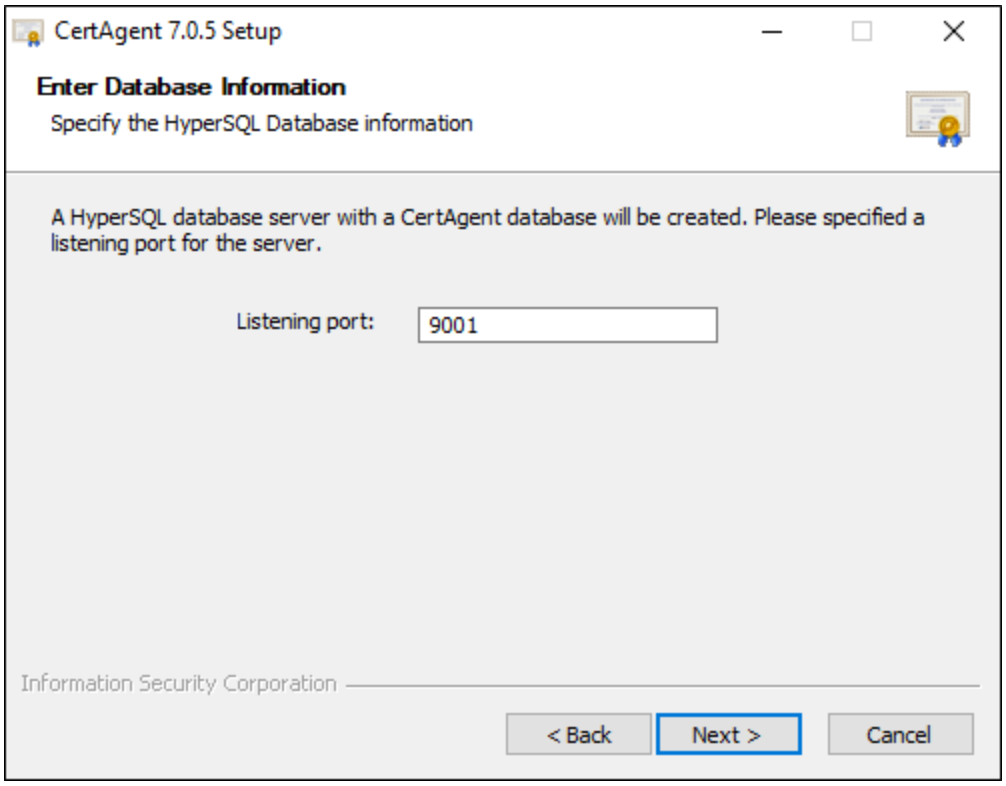




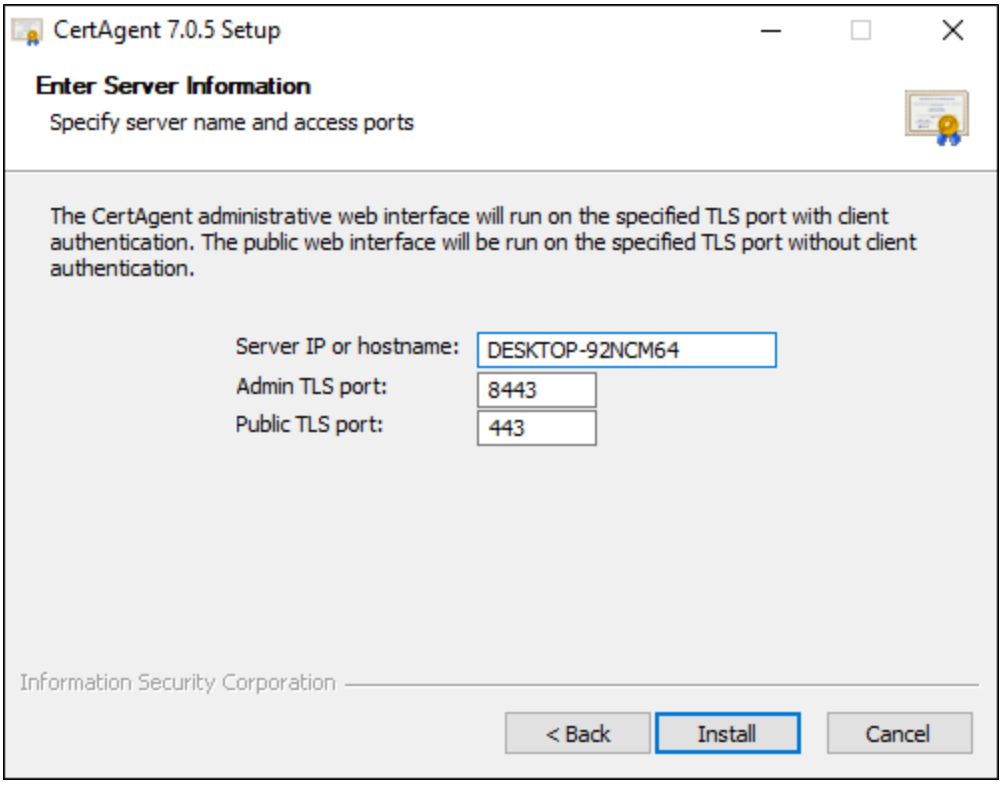

CertAgent 7.0.5 Setup
Choose Components
Choose which features of CertAgent 7.0.5 you want to install.
Check the components you want to install and uncheck the components you don't want to install. Click Next to continue.
Select components to install:
CertAgent
NIAP Compliance
Oracle JRE 8
JCE Unlimited Strengt
Apache Tomcat 8
HSQLDB 2.4.0
Description
Position your mouse over a component to see its description.
Space required: 98.0MB
Information Security Corporation
< Back
Next >
Cancel

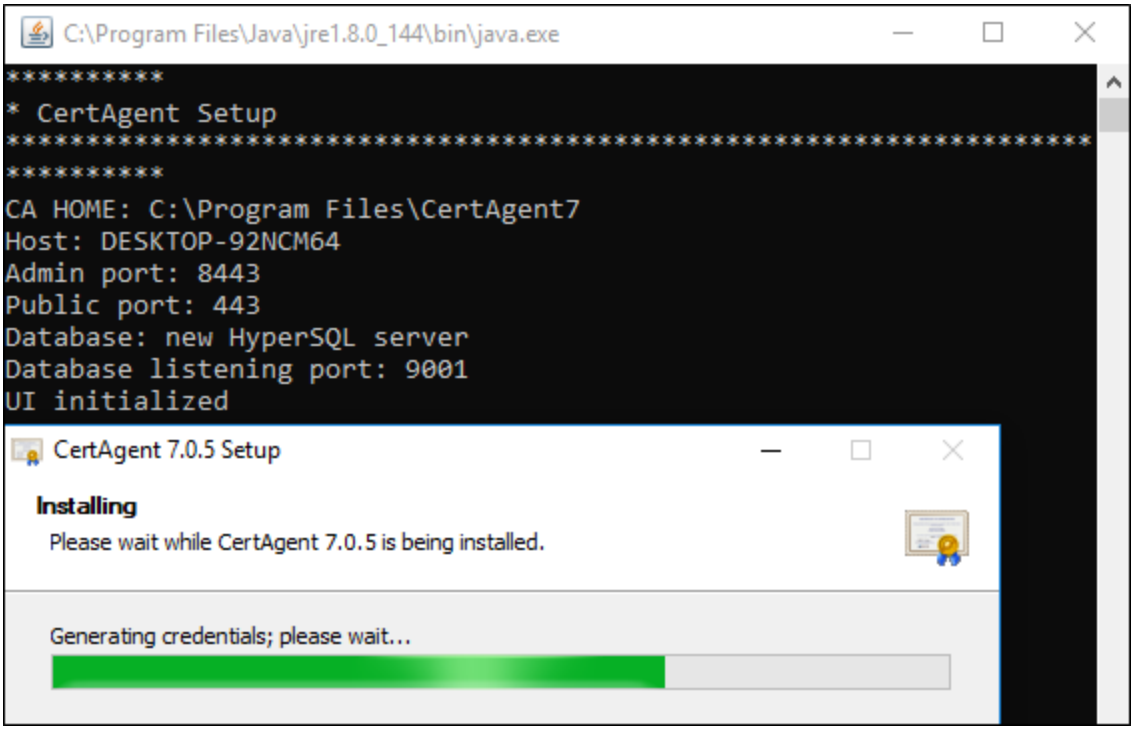

CertAgent 7.0.5 Setup
Choose Install Location
Choose the folder in which to install CertAgent 7.0.5.
Setup will install CertAgent 7.0.5 in the following folder. To install in a different folder, click Browse and select another folder. Click Next to continue.
Destination Folder
C:\Program Files\CertAgent7
Browse...
Space required: 98.0MB
Space available: 411.1GB
Information Security Corporation
< Back
Next >
Cancel

2. Setup will ask for the listening port for the HyperSQL database that will be created. If 9001 is already in use, 9002 or 9003 can also be used.



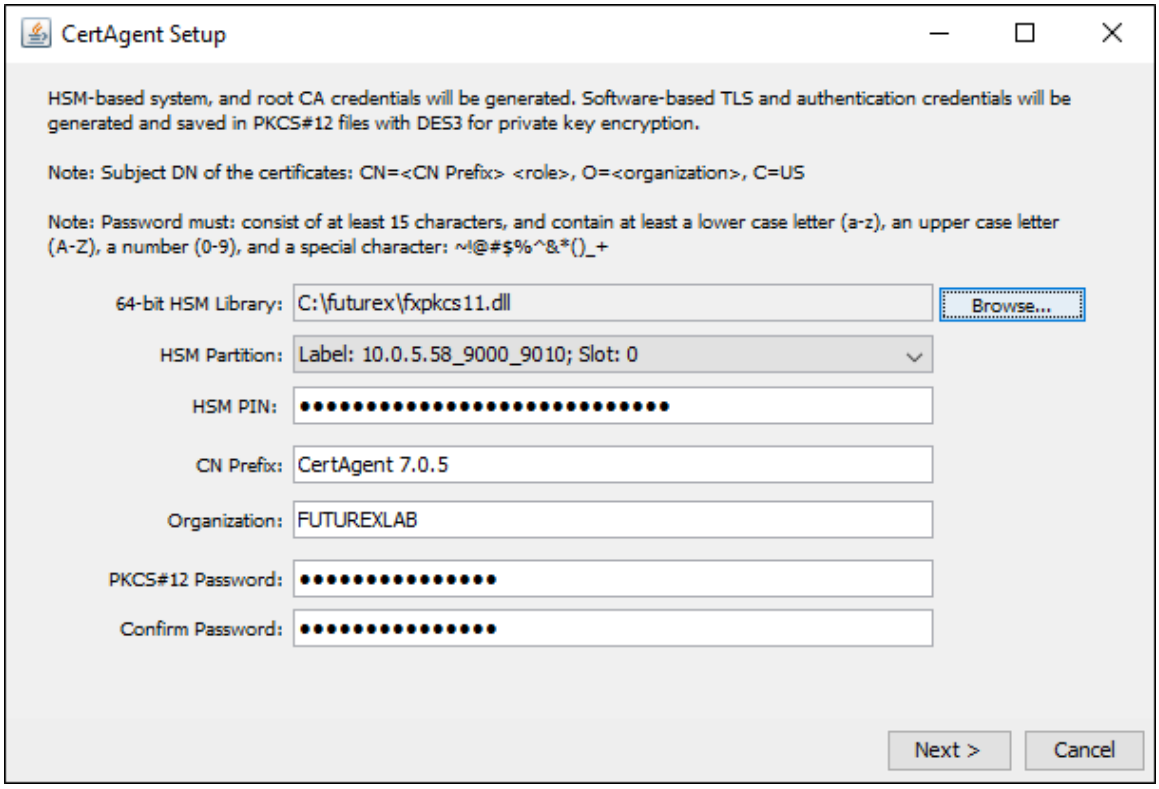
3. CertAgent will ask to create TLS ports and credentials for 'Admin' and the 'Public' web interfaces.



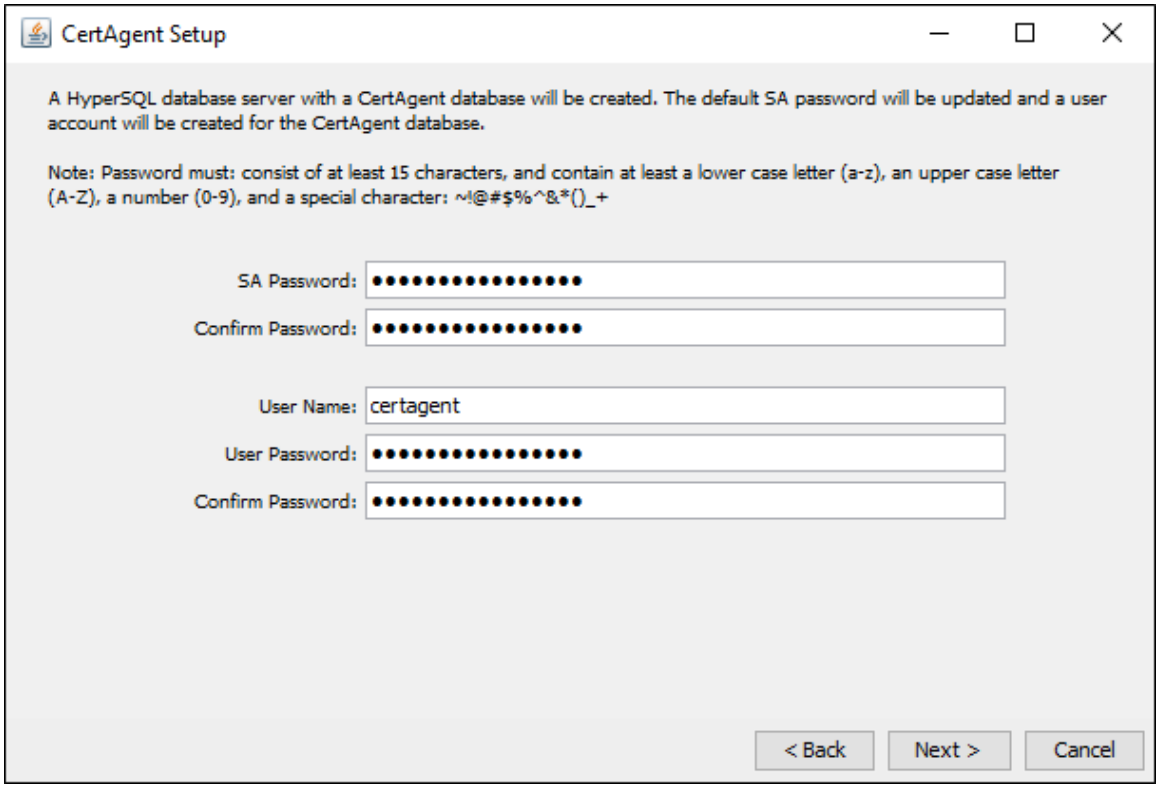


4. The following information will be required:
    - **PKCS11 library path:** Select "browse" and select the path where FXPKCS11.dll file is located in the hard drive. (Default PKCS11 install location is C:/Program Files/Futurex)
    - **HSM Partition:** Prompt to select one of the partitions found in the HSM
    - **HSM PIN:** This is the password for the identity created previously.
    - **Common Name (CN)** and **Organization Name** for the CA Root certificate that will be created by CertAgent.
    - **PKCS #12 Password:** Password to be used for PKCS #12 files generated by CertAgent and the Vectera Plus.

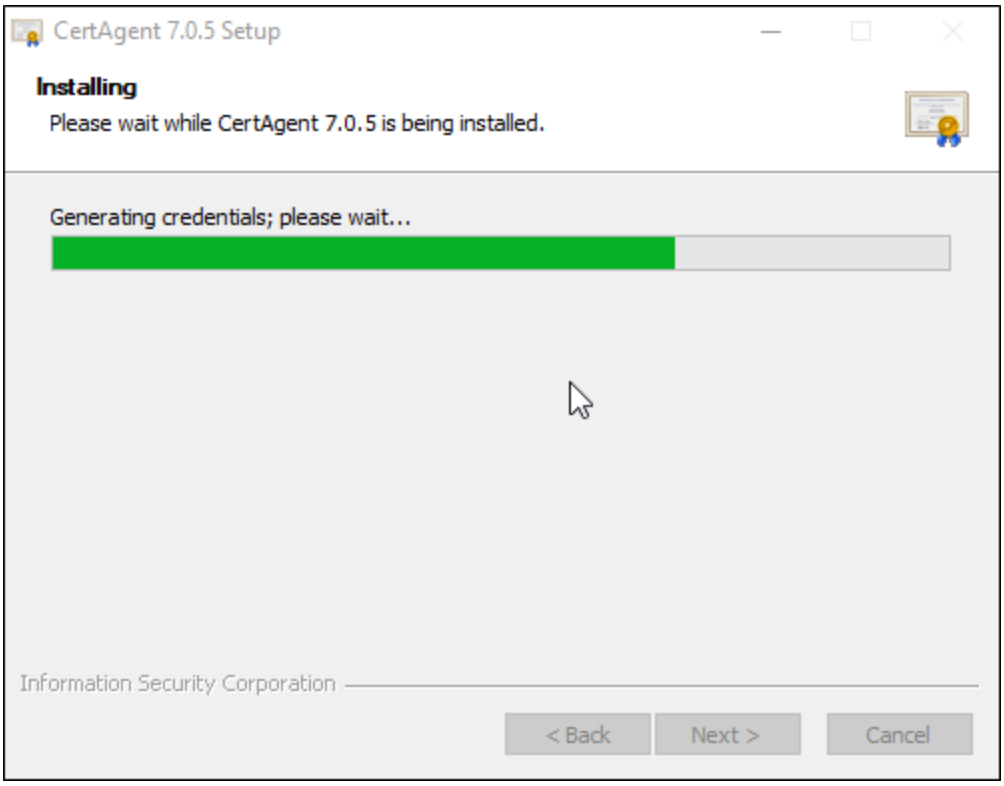
NOTE: Be sure to make note of the PKCS #12 password, admin TLS port (<admin port>) and public TLS port (<public port>) you enter during installation. This information will be required to import the Certificates for the web browsers to access the CertAgent sites (Administrator Site, Public Site, CA Site)



5. Next the SA password will be set along with a user account and password for the CertAgent database. Be sure to take note of these for future use.



6. The installer will create the credentials and will finalize the installation process.



During the Installation process we will be able to check the following logs:

- C:\Temp\ fxpkcs11.log -> for status related to all actions through the PKCS11 library.
- C:\Program Files\CertAgent7\install.log -> for CertAgent installation status.
- C:\Program Files\CertAgent7\install-hsql.log -> for HyperSQL installation status.

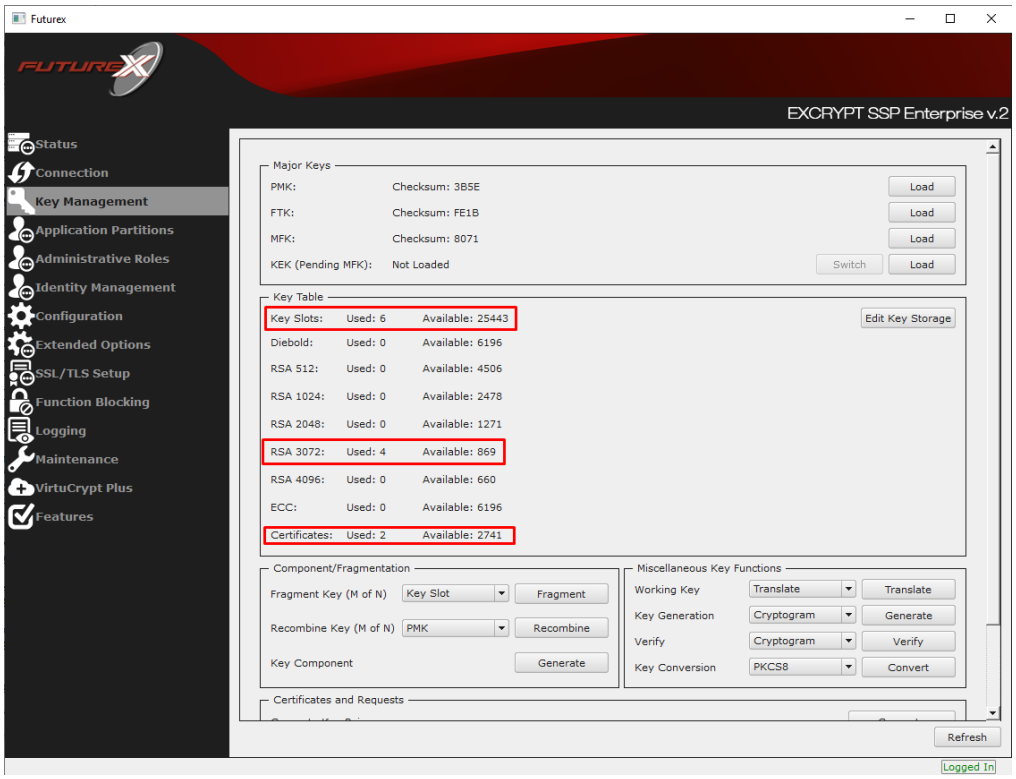
NOTE: At the end of the installation, CertAgent will create a Readme.TXT file. It is strongly recommended to read and follow instructions for POST-Installation steps.

[9] INSTALLATION VERIFICATION

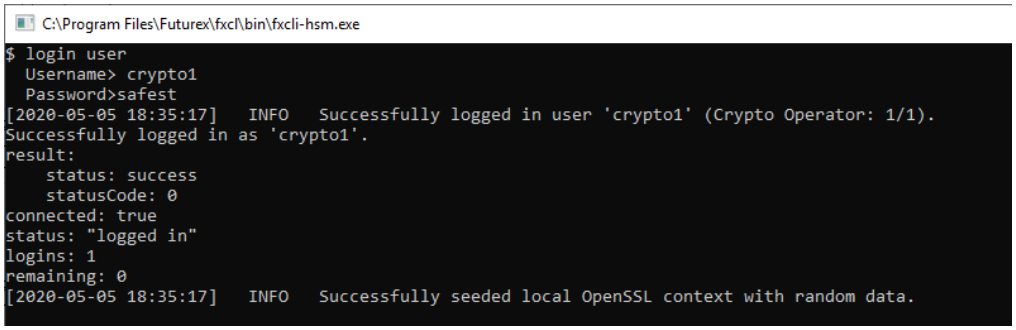
The following section are steps that can be taken to ensure CertAgent is communicating correctly with the Vectera Plus.

NOTE: The following requires the certificates installed by CertAgent to be added to the trusted list of your web browser.

1. Once the installation completes, you can login to the HSM via Excrypt Manager to verify the keys have been generated and stored on the HSM.



2. The Futurex CLI can also be used to validate this installation. Once connected using the “connect usb” command you will want to run the following commands to verify the keys exist in the Vectera.

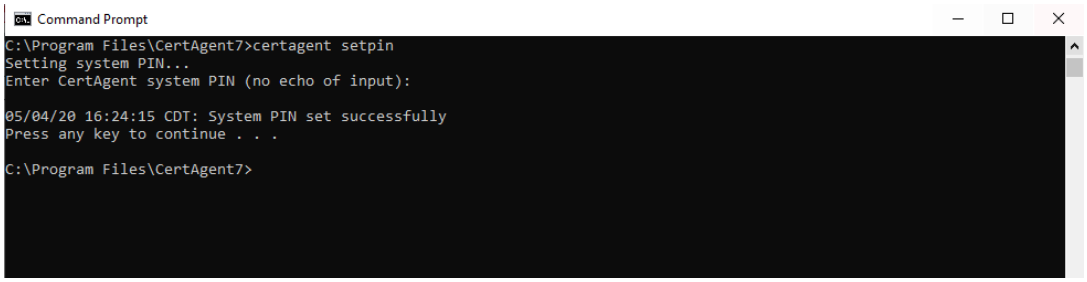


C:\Program Files\Futurex\fxcl\bin\fxcli-hsm.exe

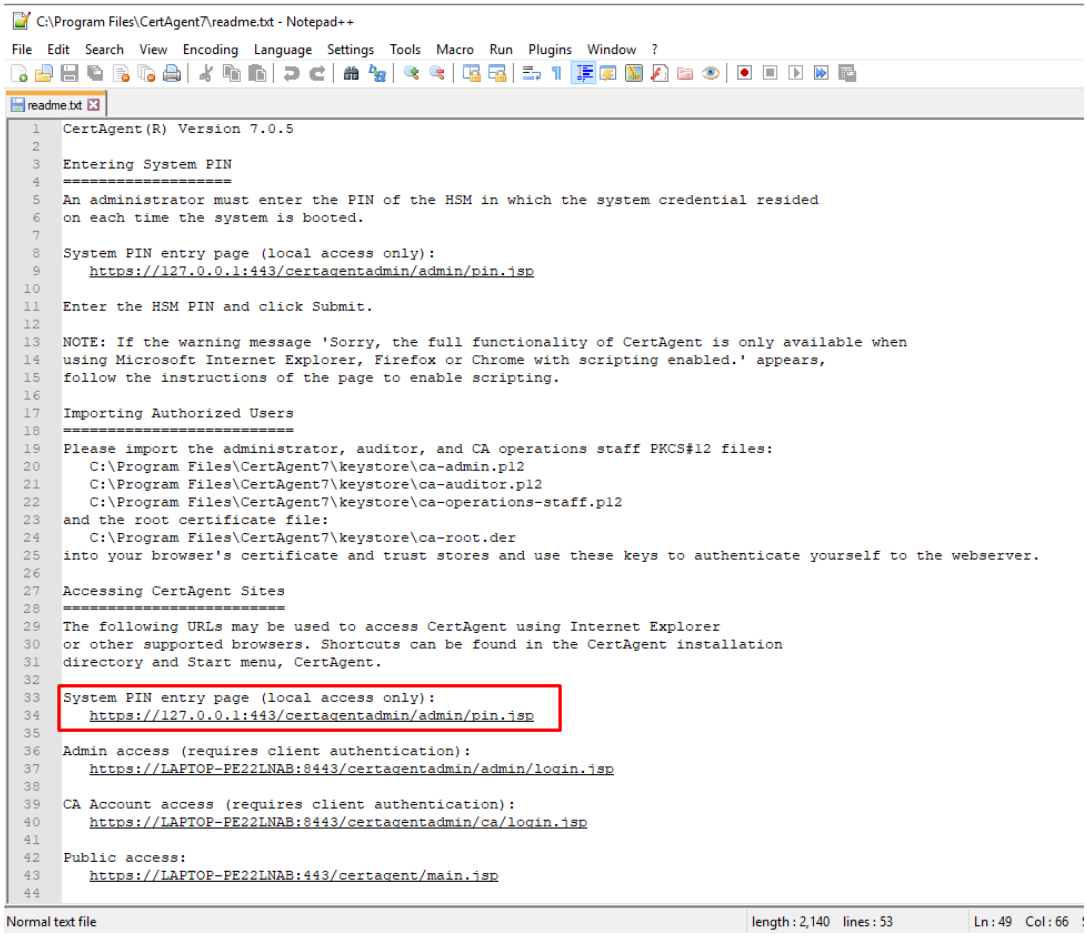
```
$ keytable list
result:
    status: success
    statusCode: 0
slots:
    -
        slot: 0
        type: "key"
        name: ""
        algorithm: RSA
        bits: 3072
        usage: Encrypt,Decrypt,Sign,Verify,Wrap,Unwrap
        majorKey: FTK
        kcv: "71AE"
        -
            slot: 1
            type: "key"
            name: ""
            algorithm: RSA
            bits: 3072
            usage: Encrypt,Verify,Wrap
            majorKey: FTK
            kcv: "8C0D"
            -
                slot: 2
                type: "certificate"
                name: ""
                algorithm: RSA
                bits: 3072
                usage: Sign,Verify,Wrap,Unwrap
                majorKey: None
                fingerprint: "3422798E22319E1E170E29837F9F0112CE1DFA5A"
                -
                    slot: 3
                    type: "key"
                    name: ""
                    algorithm: RSA
                    bits: 3072
                    usage: Encrypt,Decrypt,Sign,Verify,Wrap,Unwrap
                    majorKey: FTK
                    kcv: "70FE"
                    -
                        slot: 4
                        type: "key"
                        name: ""
                        algorithm: RSA
                        bits: 3072
                        usage: Encrypt,Verify,Wrap
                        majorKey: FTK
                        kcv: "1696"
                        -
                            slot: 5
                            type: "certificate"
                            name: ""
                            algorithm: RSA
                            bits: 3072
                            usage: Sign,Verify
                            majorKey: None
                            fingerprint: "83BC566A389AF4F34292BEA053B013A1A97BC968"
```

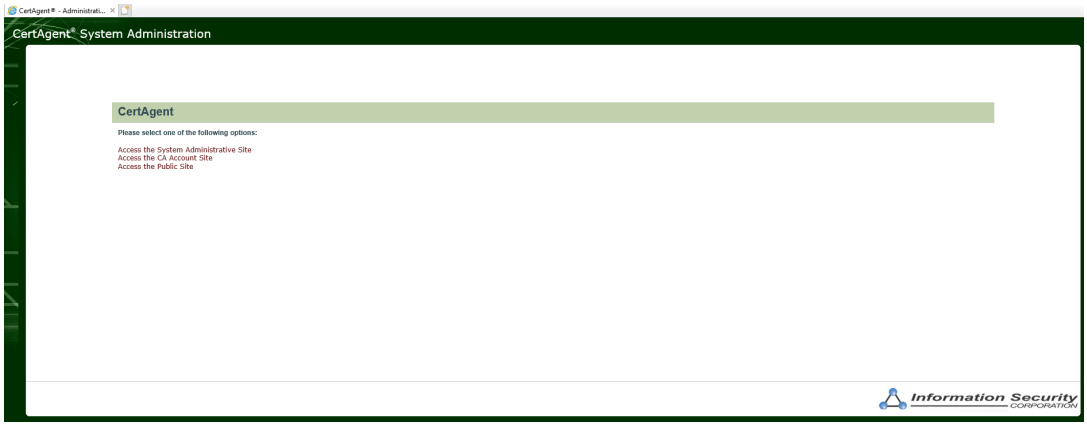
If all 6 keys are present, the installation was successful.

3. Open a command terminal and navigate to the installation location of CertAgent. Then run the command “certagent setpin”. You will then set a pin in the terminal.



4. Navigate to the System PIN Entry page shown in the README.txt.





5. The above links can then be used for the following:
    - Access the System Administrative Site
        - Admin controls over the system and server. Configuration settings can be done here as well. Must connect with the Admin certificate.
    - CA Account Site
        - Allows the certificate enrollment, management, CRL, and other settings to be set when connected with the Admin certificate.
        - Allows CSRs to be approved, signed, revoked, and other certificate enrollment tasks to be completed when connected with the Operations certificate.
    - Public Site
        - Allows users to enroll, upload, and retrieve certificates to and from the HSM when connected with the Client certificate.
6. Using the Public Site, send a certificate signing request using the “Enroll” function. Using Internet Explorer, you can generate a key for a certificate to be signed by the HSM. Firefox cannot generate a key for you.
7. After sending in a CSR, login to the CA Account Site using the Operations certificate and find the certificate in the pending section and issue it. Proper configuration of the application with the HSM will allow the certificate to be issued and retrieved all from the web.

APPENDIX A: XCEPTIONAL SUPPORT

In today's high-paced environment, we know you are looking for timely and effective resolutions for your mission-critical needs. That is why our Xceptional Support Team will help do whatever it takes to ensure you have the best experience and support possible. Every time. Guaranteed.

- 24x7x365 mission critical support
- Level 1 to level 3 support
- Extremely knowledgeable subject matter experts

At Futurex, we strive to supply you with the latest data encryption innovations as well as our best-in-class support services. Our Xceptional Support Team goes above and beyond to meet your needs and provide you with exclusive services that cannot be found anywhere else in the industry.

- Technical Services
- Onsite Training
- Virtual Training
- Customized Consulting
- Customized Software Solutions
- Secure Key Generation, Printing, and Mailing
- Remote Key Injection
- Certificate Authority Services

Toll-Free: 1-800-251-5112

E-mail: support@futurex.com

ENGINEERING CAMPUS

864 Old Boerne Road

Bulverde, Texas, USA 78163

Phone: +1 830-980-9782

+1 830-438-8782

E-mail: info@futurex.com

XCEPTIONAL SUPPORT

24x7x365

Toll-Free: 1-800-251-5112

E-mail: support@futurex.com

SOLUTIONS ARCHITECT

E-mail: solutions@futurex.com