CyberArk Trust Protection Foundation

nShield® HSM Integration Guide

2026-01-29

Table of Contents

1. Introduction . . . 1

1.1. Post-Quantum Ready . . . 1

1.2. Product configurations . . . 1

1.3. Supported nShield hardware and software versions . . . 1

1.4. Supported nShield HSM functionality . . . 2

1.5. Requirements . . . 2

1.6. Open Issues . . . 3

2. Procedures . . . 4

2.1. Prerequisites . . . 4

2.2. Enable CyberArk Advanced Key Protect . . . 4

2.3. Create an HSM (Cryptoki) connector . . . 5

2.4. HSM Central Private Key Generation . . . 8

2.5. Code signing . . . 10

3. Additional resources and related products . . . 18

3.1. nShield HSMs . . . 18

3.2. nShield as a Service . . . 18

3.3. Entrust products . . . 18

3.4. nShield product documentation . . . 18

Chapter 1. Introduction

This document describes how to integrate the CyberArk Trust Protection Foundation with the Entrust nShield hardware security module (HSM) as a Root of Trust for storage encryption, to protect the private keys and meet FIPS 140-3 or FIPS 140-2.

1.1. Post-Quantum Ready

This integration brings together the power of Entrust's next-generation nShield HSM platform and CyberArk technology to deliver security built for the quantum era. With support for NIST-selected post-quantum algorithms, the combined solution gives organizations a confident, future-proof path to quantum-safe cryptography. By enabling seamless adoption of hybrid and quantum-resistant protection today, Entrust and CyberArk empower customers to stay ahead of evolving threats, safeguard mission-critical keys and operations.

1.2. Product configurations

Entrust has successfully tested nShield HSM integration with CyberArk Trust Protection Foundation in the following configurations:

Product	Version
CyberArk Trust Protection Foundation	25.3.0.2740
Base OS	Windows Server 2025

1.3. Supported nShield hardware and software versions

Entrust has successfully tested with the following nShield hardware and software versions.

Module, OCS and softcard protection was tested in all configurations.

HSM	Security World Software	Firmware	Image
nShield 5c	13.9.3	13.8.4	13.9.3
nShield XC	13.9.3	13.8.3	13.9.3

1.4. Supported nShield HSM functionality

Feature	Support
Module	Yes
OCS cards	No
Softcards	No
nSaaS	Yes
FIPS Restricted World	Yes [1]

[1] Keys cannot be exported when using FIPS restricted World. As a result, some integration functionality (such as HSM Central Private Key Generation) will only be supported on FIPS unrestricted and non-FIPS Security Worlds.

1.5. Requirements

Familiarize yourself with:

- CyberArk Trust Protection Foundation documentation (https://docs.venafi.com).
- The nShield HSM: *Installation Guide* and *User Guide*.
- Your organizational Certificate Policy and Certificate Practice Statement, and a Security Policy or Procedure in place covering administration of the PKI and HSM:
    - The number and quorum of Administrator Cards in the Administrator Card Set (ACS), and the policy for managing these cards.
    - The number and quorum of Operator Cards in the Operator Card Set (OCS), and the policy for managing these cards.
    - The keys protection method: Module, Softcard, or OCS.

        > Currently Softcard and OCS protection methods are not supported.

    - The level of compliance for the Security World, FIPS 140-3 or FIPS 140-2.
    - Key attributes such as key size, time-out, or need for auditing key usage.

> Entrust recommends that you allow only unprivileged connections unless you are performing administrative tasks.

1.6. Open Issues

- Wrapping of PQ Keys is currently not supported (NSE-73200).

  PQ algorithms cannot be used on certificates.

- Softcard and OCS protection is not supported (NSE-75822).

  Currently being investigated so it can be supported in the future. Earlier integrations stated support for softcard and OCS, recent testing concluded they are not supported in earlier releases and the current version.

> Entrust Engineering NSE’s are used by Entrust as a reference to track open issues. If you want to discuss these issues with Entrust, use the number as a reference.

Chapter 2. Procedures

2.1. Prerequisites

Ensure the following prerequisites are implemented:

1. Install the Entrust nShield HSM using the instructions in the *Installation Guide* for the HSM.
2. Install the Entrust nShield Security World Software, and configure the Security World as described in the *User Guide* for the HSM.
3. Edit the `cknfastrc` file located in `%NFAST_HOME%\cknfastrc`.

```
CKNFAST_OVERRIDE_SECURITY_ASSURANCES=explicitness,wrapping_crypt
CKNFAST_FAKE_ACCELERATOR_LOGIN=1
CKNFAST_LOADSHARING=1
```

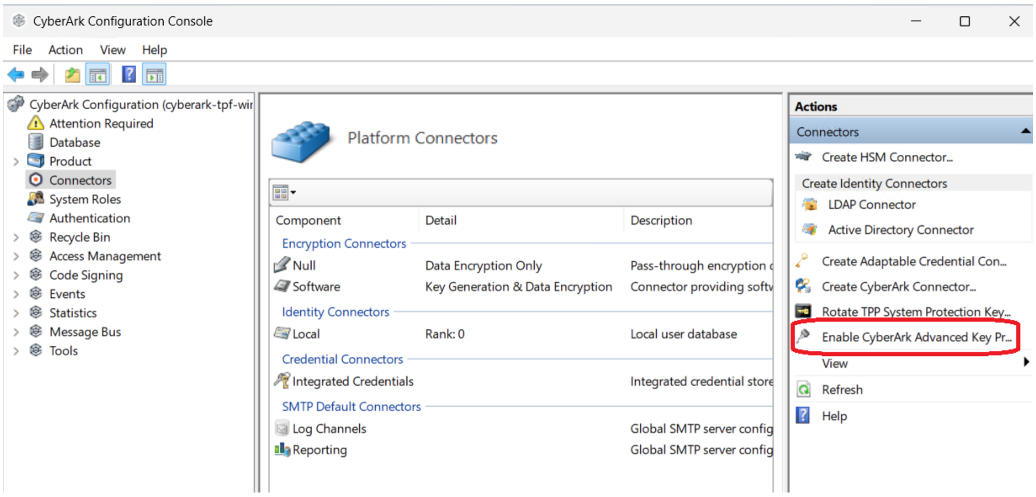
4. Install CyberArk Trust Protection Foundation. For more information, see the Online documentation.

2.2. Enable CyberArk Advanced Key Protect

CyberArk Advanced Key Protect is required for Central and Remote HSM Private Key Generation. In addition, CyberArk Code Signing Certificate Private Key Storage requires this feature to be enabled.

To enable CyberArk Advanced Key Protect:

1. Open the **CyberArk Configuration Console**.
2. Select the **Connectors** node.
3. Select **Enable CyberArk Advanced Key Protect** in the **Actions** panel.


CyberArk Configuration Console
File
Action
View
Help
CyberArk Configuration (cyberark-tpf-wir
Attention Required
Database
Product
Connectors
System Roles
Authentication
Recycle Bin
Access Management
Code Signing
Events
Statistics
Message Bus
Tools
Platform Connectors
Actions
Connectors
Create HSM Connector...
Create Identity Connectors
LDAP Connector
Active Directory Connector
Create Adaptable Credential Con...
Create CyberArk Connector...
Rotate TPP System Protection Key...
Enable CyberArk Advanced Key Pr...
View
Refresh
Help

Component	Detail	Description
Encryption Connectors		
Null	Data Encryption Only	Pass-through encryption c
Software	Key Generation & Data Encryption	Connector providing softv
Identity Connectors		
Local	Rank: 0	Local user database
Credential Connectors		
Integrated Credentials		Integrated credential store
SMTP Default Connectors		
Log Channels		Global SMTP server config
Reporting		Global SMTP server config

4. Review the information and confirm the action by selecting **Enable**.
5. Restart the platform services:

   Select the **Product** node. Under **Windows Services** do the following:

   a. Select **Website** and then select **Restart** in the **Actions** panel.
   b. Select **Trust Protection Foundation Services** and then select **Restart** in the **Actions** panel.
   c. Select **Logging** and then select **Restart** in the **Actions** panel.

2.3. Create an HSM (Cryptoki) connector

You must setup an HSM connector before the nShield HSM functionally can be used within CyberArk Trust Protection Foundation.

To create an HSM (Cryptoki) connector:

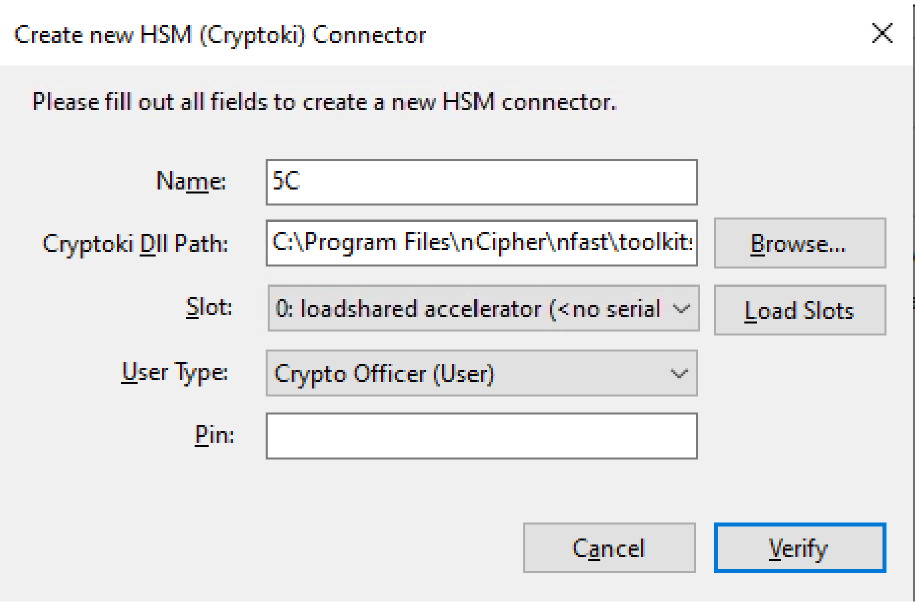
1. Open the **CyberArk Configuration Console**.
2. Select the **Connectors** node.
3. Select **Create HSM Connector** in the **Actions** panel.
4. Enter your CyberArk TPF user credentials if required.
5. For **Name**, enter any name for the HSM connector.
6. For **Cryptoki Dll Path**, select **Browse** and locate the following path to the DLL file:

   `C:\Program Files\nCipher\nfast\toolkits\pkcs11\cknfast.dll`.

7. Select **Load Slots**.
8. Select a slot to use for the intended key protection type. This is the partition on the HSM where CyberArk Trust Protection Foundation will access the encryption keys.

Since Softcard and OCS protection methods are not currently support in the integration, select the **loadshared accelerator** slot.

9. For **User Type**, select the required user to access the HSM keys on the designated partition.
10. For **Pin**, enter the passphrase of the Card Set being used. Since Module protection is being used, leave the pin blank.



Create new HSM (Cryptoki) Connector

Please fill out all fields to create a new HSM connector.

Name: 5C

Cryptoki Dll Path: C:\Program Files\nCipher\nfast\toolkit:

Slot: 0: loadshared accelerator (<no serial

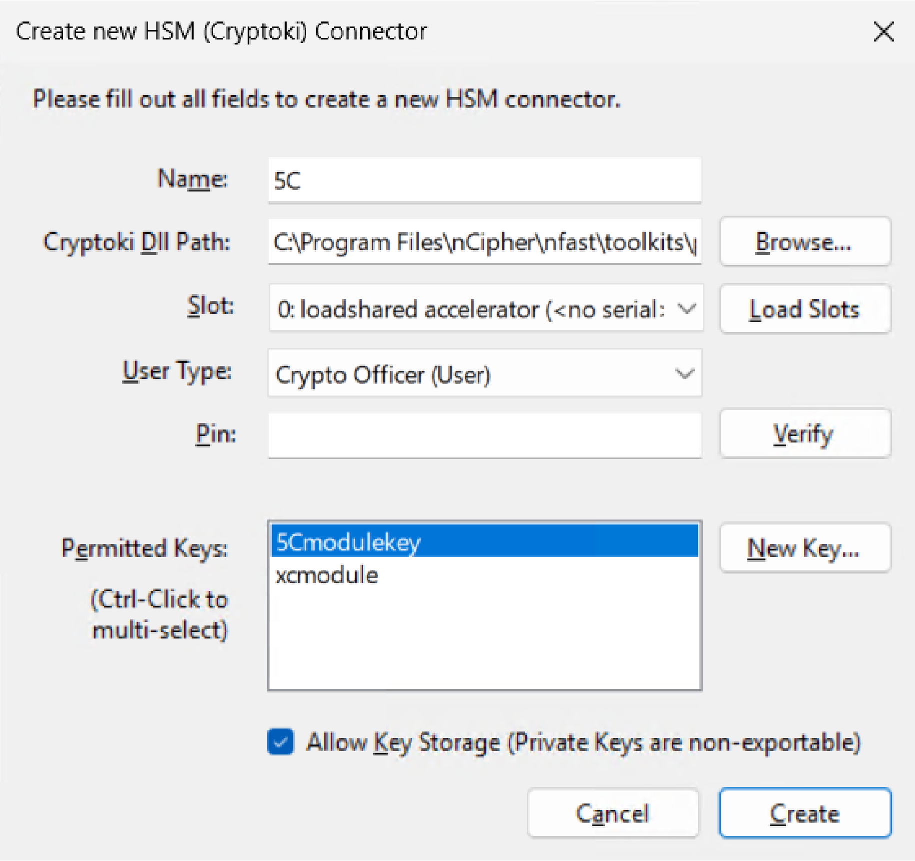
User Type: Crypto Officer (User)

Pin:

11. Select **Verify**.
12. Generate a HSM protected key.

    HSM-protected keys can be generated to encrypt data stored in the Trust Protection Foundation Secret Store.

    - Select **New Key**.
    - On the **Create New HSM Key** page, enter a **Name** and select a **Type** for the key.
    - Select **Create**.
    - The key gets shown in the **Permitted Keys** field.
13. Select **Allow Key Storage** checkbox.
14. Select the Key you just created.
15. Select **Create**.



Create new HSM (Cryptoki) Connector

Please fill out all fields to create a new HSM connector.

Name: 5C

Cryptoki Dll Path: C:\Program Files\nCipher\nfast\toolkits\

Slot: 0: loadshared accelerator (<no serial:

User Type: Crypto Officer (User)

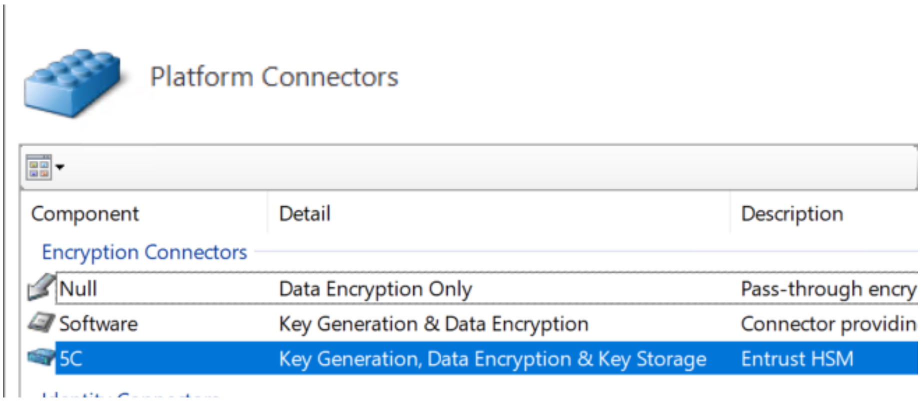
Pin:

Permitted Keys: (Ctrl-Click to multi-select)

- 5Cmodulekey
- xcmodule

☑ Allow Key Storage (Private Keys are non-exportable)

The HSM Connector gets created and displayed under the **Encryption Connectors** section under **Platform Connectors**.



Platform Connectors

Component	Detail	Description
Encryption Connectors		
Null	Data Encryption Only	Pass-through encry
Software	Key Generation & Data Encryption	Connector providin
5C	Key Generation, Data Encryption & Key Storage	Entrust HSM

Notice the details of the Connector. It should say: **Key Generation, Data Encryption & Key Storage**. This is important for the rest of the integration.

16. To list the newly created key and its protection type, open a command prompt and run the following command:

```
nfkminfo -l

Keys with module protection:
```

```
key_pkcs11_ua63b67f4b141c2babd1f5f47e7aab4a3f68fbb851 `5cmodulekey'
```

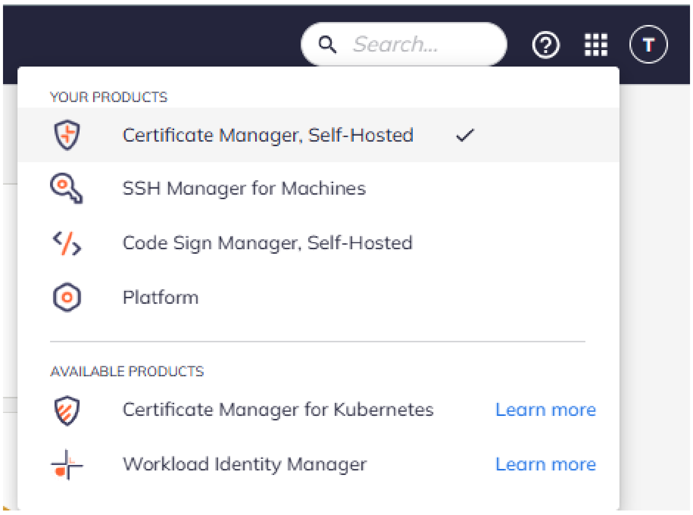
2.4. HSM Central Private Key Generation

CyberArk Trust Protection Foundation uses the Entrust nShield HSM for private key generation for SSH keys and certificates.

> Certificate Authority (CA) template objects are used in CyberArk Trust Protection Foundation to manage the certificate lifecycle. Creating one is a prerequisite to HSM Central Key Generation. For more information, see the online documentation.

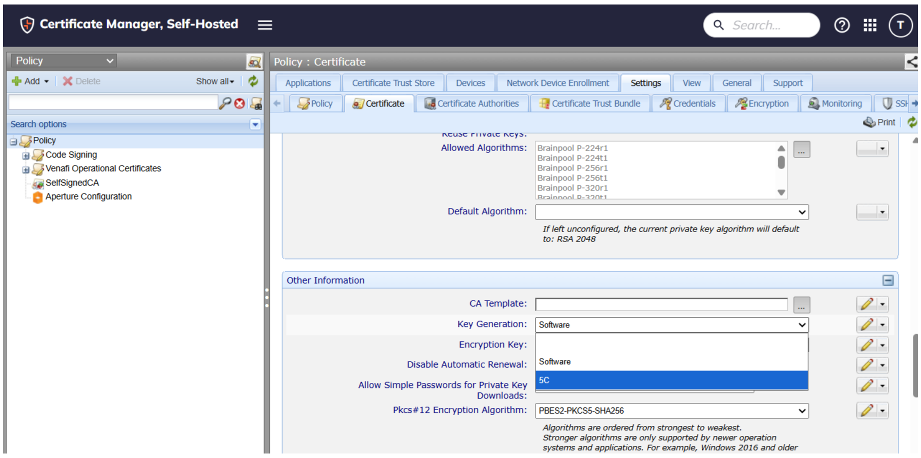
We already configured the CyberArk platform policy to enable the Entrust nShield HSM for central HSM key generation.

1. Log in to admin console: `https://[IP_address_of_CyberArk_TPF]/Aperture`.
2. In the **Application Menu** in the top right side of the application, select **Certificate Manager**.



Search...
YOUR PRODUCTS
Certificate Manager, Self-Hosted ✓
SSH Manager for Machines
Code Sign Manager, Self-Hosted
Platform
AVAILABLE PRODUCTS
Certificate Manager for Kubernetes
Learn more
Workload Identity Manager
Learn more

3. Select **Policy Tree** under the **Certificate Manager** Menu.
4. Select **Policy**, on the left pane.
5. On the right pane, select the **Certificate** tab.
6. Under **Other Information**, select your HSM Connector in the **Key Generation** drop-down menu.



Reuse Private Keys:

Allowed Algorithms: Brainpool P-224r1, Brainpool P-224t1, Brainpool P-256r1, Brainpool P-256t1, Brainpool P-320r1, Brainpool P-320t1

Default Algorithm:

If left unconfigured, the current private key algorithm will default to: RSA 2048

Other Information

CA Template:

Key Generation: Software

Encryption Key:

Disable Automatic Renewal:

Allow Simple Passwords for Private Key Downloads:

Software

5C

Pkcs#12 Encryption Algorithm: PBES2-PKCS5-SHA256

Algorithms are ordered from strongest to weakest. Stronger algorithms are only supported by newer operation systems and applications. For example, Windows 2016 and older

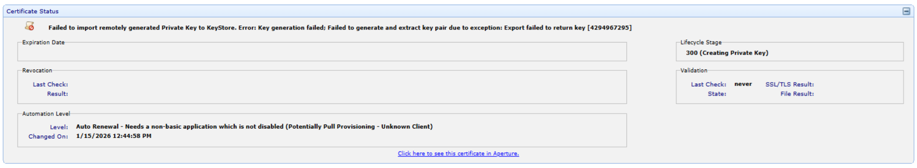
7. Select **Save**.

2.4.1. Generate the certificate:

1. Select **Policy**.
2. Select **Add** > **Certificates** > **The Certificate Type you Want**.
3. In the **General Information** tab:
   a. Enter the **Certificate Name** and another other required information.
   b. For **Management Type**, select **Provisioning** or **Enrollment**.
4. In the **CSR Handling** tab:
   a. For **CSR Generation**, select **Service Generated CSR**.
   b. For **Generate Key/CSR on Application**, select **No**.
5. In the **Subject DN** tab, enter the required information.
6. In the **Private Key** tab, enter the key information.

   For Post Quantum, we selected **ML-DSA44**.

> The current version of the Post Quantum HSM firmware does not support wrapping of Post Quantum keys. When a Post Quantum algorithm is used, an error will take place when you try to renew the certificate. The error is listed below. So for now use the system default algorithm **RSA-2048** or a non Post Quantum algorithm.



Certificate Status

Failed to import remotely generated Private Key to KeyStore. Error: Key generation failed; Failed to generate and extract key pair due to exception: Export failed to return key [4294967295]

Expiration Date

Lifecycle Stage
300 (Creating Private Key)

Revocation
Last Check:
Result:

Validation
Last Check: never SSL/TLS Result:
State: File Result:

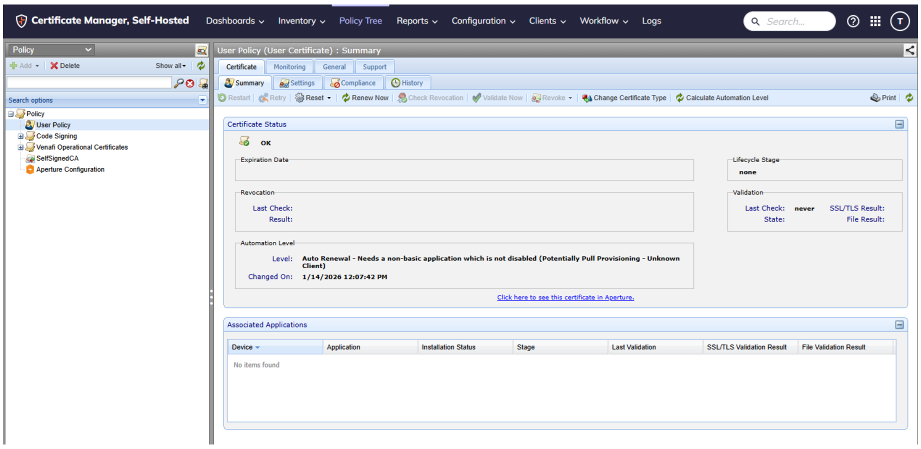
Automation Level
Level: Auto Renewal - Needs a non-basic application which is not disabled (Potentially Pull Provisioning - Unknown Client)
Changed On: 1/15/2026 12:44:58 PM

Click here to see this certificate in Aperture.

7. In the **Other Information** tab, search for the previously configured **CA Template**.

8. Select **Save**.
9. Select the newly generated certificate from the policy tree.

   The Certificate Status should be **OK**.



Certificate Status

OK

Expiration Date

Lifecycle Stage: none

Revocation
Last Check:
Result:

Validation
Last Check: never
State:
SSL/TLS Result:
File Result:

Automation Level
Level: Auto Renewal - Needs a non-basic application which is not disabled (Potentially Pull Provisioning - Unknown Client)
Changed On: 1/14/2026 12:07:42 PM

Click here to see this certificate in Aperture.

Associated Applications

Device	Application	Installation Status	Stage	Last Validation	SSL/TLS Validation Result	File Validation Result
No items found						

10. Select **Renew Now**.
11. After a minute, **Refresh** the browser window.

    Select the certificate and the certificate details will appear.

12. If you selected **Provisioning** for **Management Type**, associate the certificate to the intended application object.
13. Check to see if the certificate was installed on this application server.

2.5. Code signing

CyberArk Code Sign Manager can store private code signing keys in the Entrust nShield HSM. This section of the document describes the basic steps used to achieve this functionality for the integration. For more detailed procedures, see the online documentation.

> Certificate Authority (CA) template objects are used in CyberArk Trust Protection Foundation to manage the certificate lifecycle. Creating one is a prerequisite to CodeSign. For more information, see the online documentation.

To use an HSM for key storage, you must first enable Key Storage on the HSM Connector:

> This should have been done already when the HSM connector was setup. Here are the instructions just in case.

1. Open the **CyberArk Configuration Console**.
2. Select the **Connectors** node.
3. Select the **HSM Component** generated in an earlier step.
4. Select **Properties** in the **Actions** panel under **Encryption Driver**.
5. Enter your CyberArk Trust Protection Foundation user credentials if required.
6. Select **Allow Key Storage**.
7. Select **Apply**.
8. Select **OK**.

2.5.1. To choose a code signing Administrator:

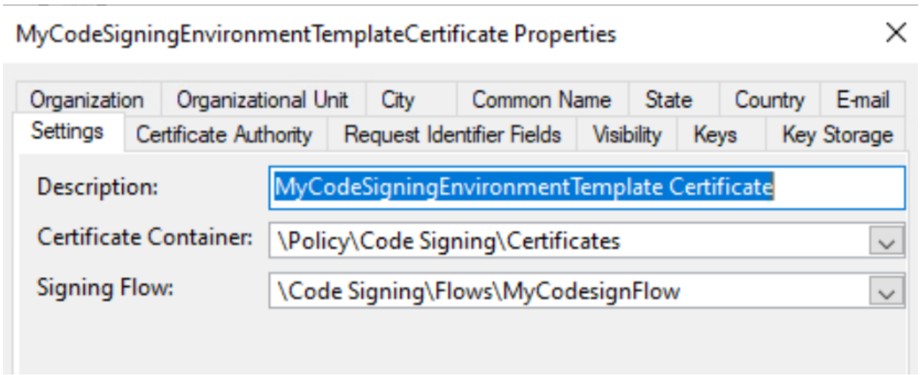
1. Open the **CyberArk Configuration Console**.
2. Select the **System Roles** node.
3. Select **Add CodeSign Manager Administrator** in the **Actions** panel.
4. Select a user to gain CodeSign Manager Administrator rights.

2.5.2. To create a code signing flow:

1. Open the **CyberArk Configuration Console**.
2. Under the **Code Signing** node, select **Custom Flows**.
3. Select **Add new Code Signing Flow** in the **Actions** panel.
4. Enter a name for the Code Signing Flow and select **Create**.
5. Select the newly created Code Signing Flow and add an approver through the **Actions** panel.
    a. Select **Standard** in the **Actions** panel.
    b. Enter the name for the Approver.
    c. When the properties window comes up, Select **OK**.

2.5.3. To create an environment template for the code signing project:

1. Open the **CyberArk Configuration Console**.
2. Under the **Code Signing** node, select **Environment Templates**.
3. Select **Certificate** in the **Actions** panel under **Add Single Template**.
4. Enter a name for the Code Signing Environment Template.
5. In the **Properties** window that appears, enter the **Description**, **Certificate Container**, and **Signing Flow** within the **Settings** tab.



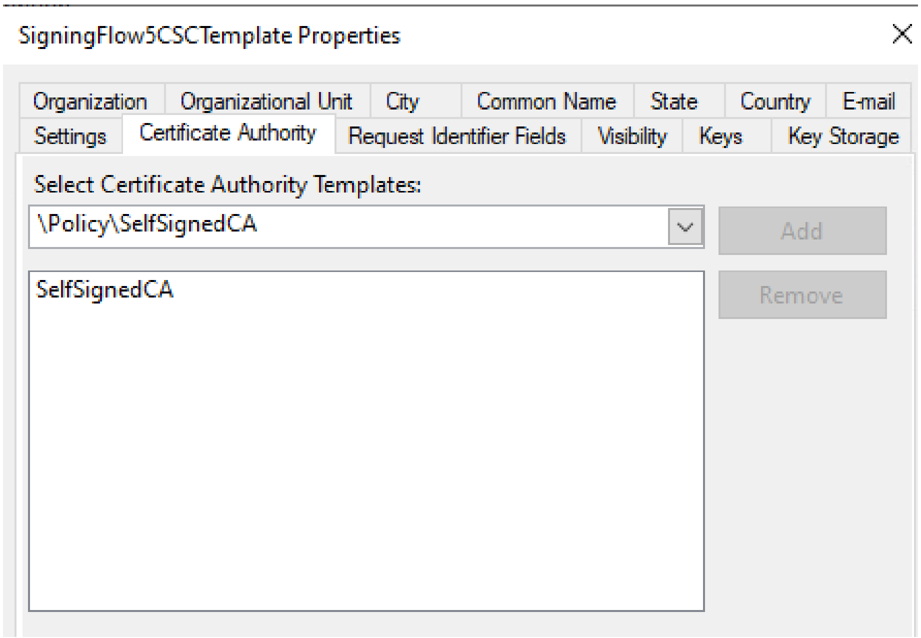
MyCodeSigningEnvironmentTemplateCertificate Properties

Description: MyCodeSigningEnvironmentTemplate Certificate

Certificate Container: \Policy\Code Signing\Certificates

Signing Flow: \Code Signing\Flows\MyCodesignFlow

6. Open the **Certificate Authority** tab and search for the previously configured **CA Template**.



SigningFlow5CSCTemplate Properties

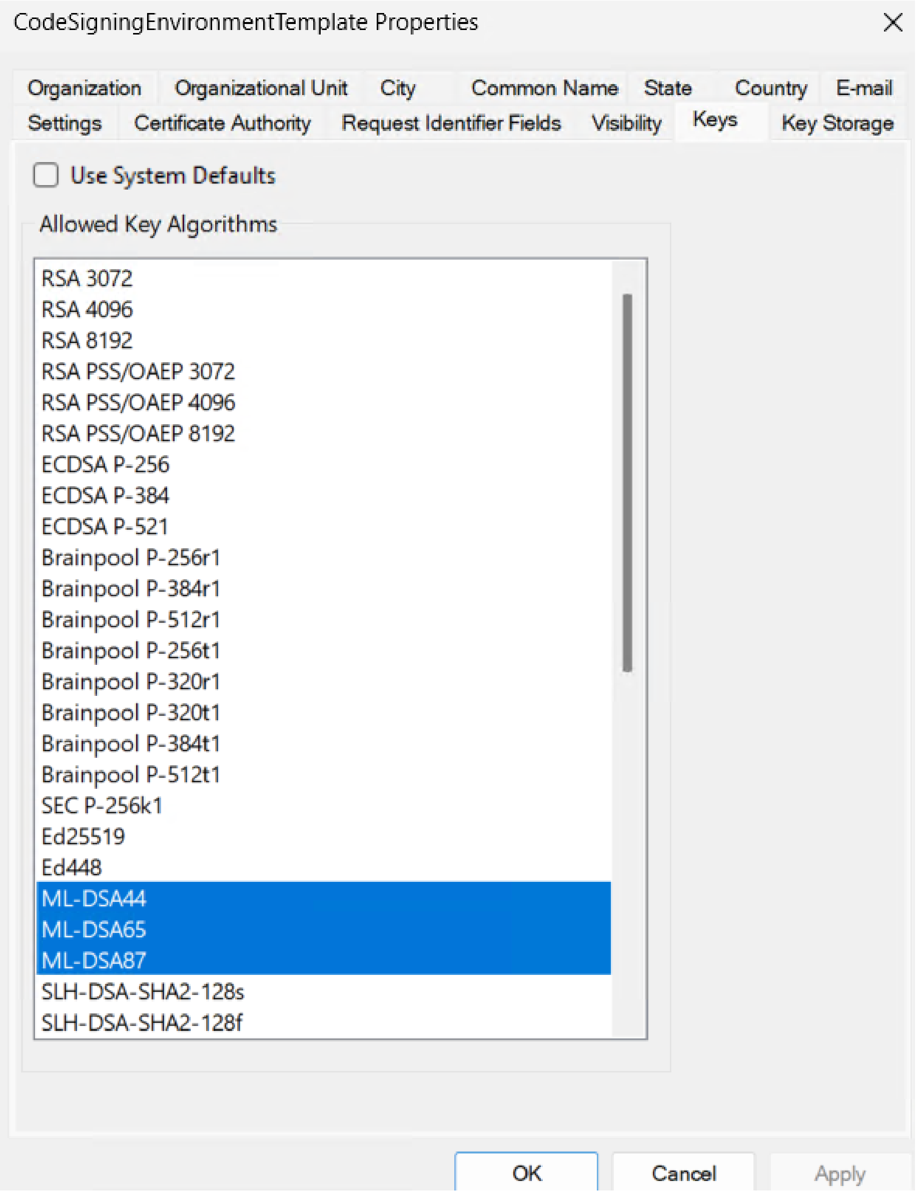
Select Certificate Authority Templates: \Policy\SelfSignedCA

SelfSignedCA

   a. Select **Add**.

7. Open the **Keys** tab and select which key sizes to allow.

> For Post Quantum testing, you can select the **ML-DSA** types.



□ Use System Defaults

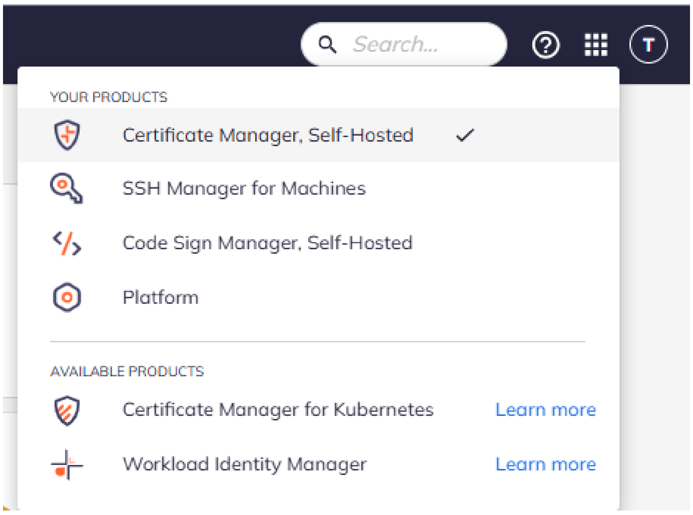
Allowed Key Algorithms

RSA 3072
RSA 4096
RSA 8192
RSA PSS/OAEP 3072
RSA PSS/OAEP 4096
RSA PSS/OAEP 8192
ECDSA P-256
ECDSA P-384
ECDSA P-521
Brainpool P-256r1
Brainpool P-384r1
Brainpool P-512r1
Brainpool P-256t1
Brainpool P-320r1
Brainpool P-320t1
Brainpool P-384t1
Brainpool P-512t1
SEC P-256k1
Ed25519
Ed448
ML-DSA44
ML-DSA65
ML-DSA87
SLH-DSA-SHA2-128s
SLH-DSA-SHA2-128f

8. Open the **Key Storage** and open the drop-down menu.
9. Select the previously created **HSM Connector**.
10. Enter any optional information in the remaining tabs.
11. Select **Apply** and then **OK**.

2.5.4. To create a new code signing project:

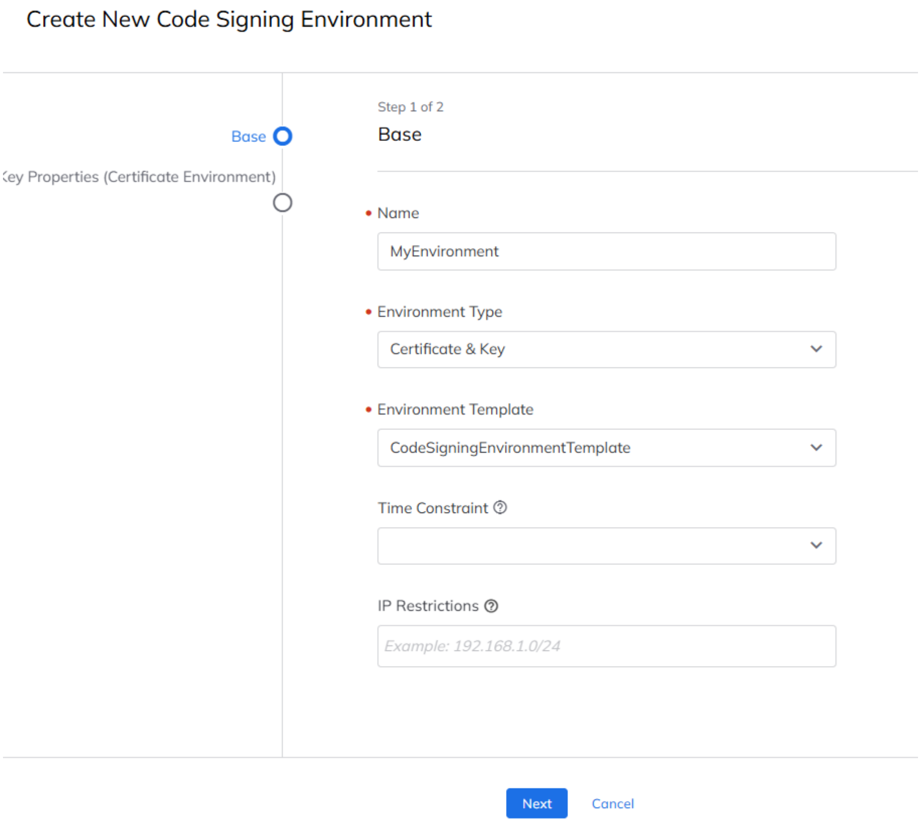
1. Log in to Aperture: `https://[IP_address_of_CyberArk_TPF]/Aperture`.
2. In the **Application Menu** in the top right side of the application, select **CodeSign Manager**.


Search...
YOUR PRODUCTS
Certificate Manager, Self-Hosted ✓
SSH Manager for Machines
Code Sign Manager, Self-Hosted
Platform
AVAILABLE PRODUCTS
Certificate Manager for Kubernetes
Learn more
Workload Identity Manager
Learn more

3. Select **Projects** under the **CodeSign Project** Menu.
4. Select **Create Project**.
5. Enter a **Project Name** and **Description**.
6. Select **Create**.

2.5.4.1. To create an environment for the project with a new HSM private key and certificate:

1. Select the **Environments** tab.
2. Select **Create**.
3. Enter the **Environment Name**.
4. For **Environment Type**, select **Certificate & Key**.
5. For **Environment Template**, select the previously created Environment Template.



Create New Code Signing Environment

Step 1 of 2

Base

• Name

MyEnvironment

• Environment Type

Certificate & Key

• Environment Template

CodeSigningEnvironmentTemplate

Time Constraint

IP Restrictions

Example: 192.168.1.0/24

6. Optionally enter a value for **Time Constraint** and **IP Restrictions**.
7. Select **Next**.
8. For **Key Storage Location**, select the HSM Connector.
9. For **Creation Type**, select **Create new key**.
10. For **Certificate Provider**, select a CA.
11. For **Key Algorithm**, select a key algorithm.

> For Post Quantum, select one of the Post Quantum algorithms that are available.

12. Enter any other necessary information for the certificate.
13. Select **Create Environment**.
14. Select **Submit for Approval** to generate a new certificate and private key once it is approved.

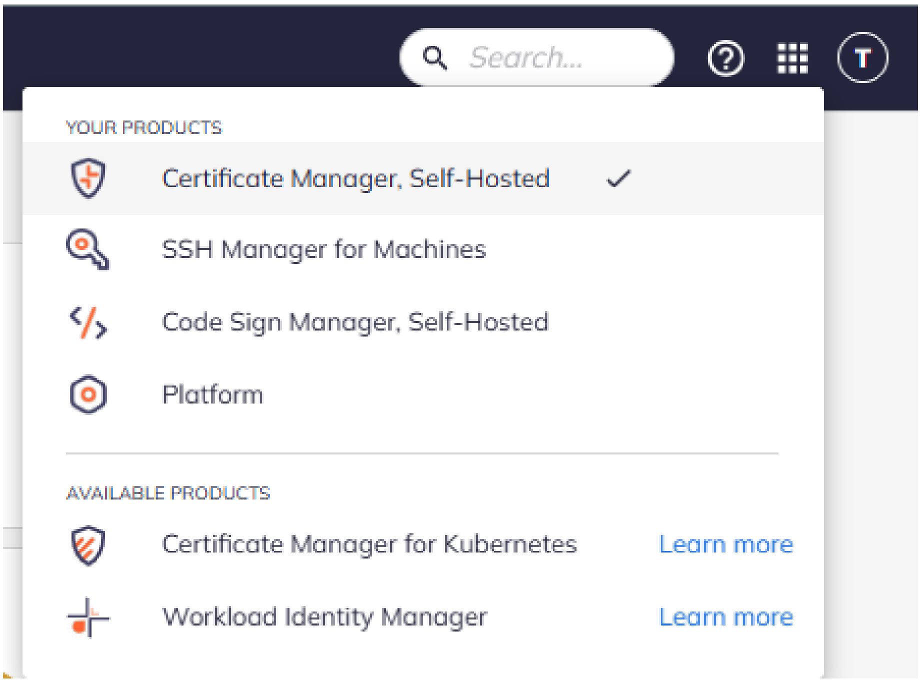
2.5.4.2. To create an environment for the project with an existing HSM private key and certificate:

1. Select the **Environments** tab.
2. Select **Create**.

3. Enter the **Environment Name**.
4. For **Environment Type**, select **Certificate & Key**.
5. For **Environment Template**, select the previously created Environment Template.
6. Optionally enter a value for **Time Constraint** and **IP Restrictions**.
7. Select **Next**.
8. **Signing Flow** should list your code signing flow and **Key Storage Location** should list your HSM Connector.
9. For **Creation Type**, select **Use Existing Key in HSM**.
10. Select an existing **Private HSM Key** and **Public HSM Key**.
11. Import an existing certificate or manually enter its details.
12. Select **Create Environment**.
13. Select **Submit for Approval** to generate a new certificate and private key once it is approved.

2.5.4.3. To approve the project:

1. Log in to Aperture: `https://[IP_address_of_CyberArk_TPF]/Aperture`.
2. In the **Application Menu** in the top right side of the application, select **CodeSign Manager**.


Search...
YOUR PRODUCTS
Certificate Manager, Self-Hosted ✓
SSH Manager for Machines
Code Sign Manager, Self-Hosted
Platform
AVAILABLE PRODUCTS
Certificate Manager for Kubernetes
Learn more
Workload Identity Manager
Learn more

3. Select **Approvals** under the **CodeSign Project** Menu.
4. Select **Pending Approvals**.

5. Select the request.
6. Select **Approve/Reject**.
7. Enter a **Comment** for the approval.
8. Select **Approve**.
9. If you selected the option to generate new keys, the keys are now created on the Entrust nShield HSM. To list it, open a command prompt and run the following command:

   ```
   nfkminfo -l

   Keys with module protection:
    key_pkcs11_ua0b301946b84b585e6d90fa676e24400a3ccb1d7f 'ML-DSA44 41b427bcf9a241b18d455a3584df5c1f'
   ```

Chapter 3. Additional resources and related products

3.1. nShield HSMs

3.2. nShield as a Service

3.3. Entrust products

3.4. nShield product documentation