nShield Security World

nShield Security World v13.6.15 Management Guide

17 February 2026

Table of Contents

1. Introduction

You must create a Security World before using the HSM to manage keys.

You normally create a Security World after installing and configuring the module and its software. For more information, see:

- nShield v13.6.15 Hardware Install and Setup Guides
- nShield v13.6.15 Software Install Guide
- nShield v13.6.15 HSM User Guide

You create a Security World with a single HSM. If you have more than one module, select one module with which to create the Security World, then add additional modules to the Security World after its creation. If you create a Security World with the audit logging feature enabled, all additional HSMs added to this Security World will also have audit logging enabled.

> ℹ To use the module to protect a different set of keys, you can replace an existing Security World with a new Security World.

> ❗ All Security Worlds rely on you using the security features of your operating system to control the users who can access the Security World and, for example, write data to the host.

(**Network-attached HSMs**) Other nShield HSMs can also use a Security World created on an nShield HSM using client cooperation. For more information, see Client cooperation.

1.1. Display information about your Security World

To display information about the status of your Security World:

- Run the `nfkminfo` command-line utility. See nfkminfo.
- Run the `kmfile-dump` command-line utility. See kmfile-dump.
- **Network-attached HSMs:** Run the `nethsmadmin` command-line utility. See Copy a Security World to a network-attached HSM and check the current version.
- **Network-attached HSMs:** Select **Security World mgmt** > **Display World info** from the front panel main menu.

2. Security World infrastructure

The Security World infrastructure provides secure life-cycle management of cryptographic keys. It gives you control over the procedures and protocols you need to create, manage, distribute and, in the event of disaster, recover keys.

A Security World provides you with the following features:

- Security
- Application independence
- Platform independence
- Flexibility
- Scalability
- Robustness
- Audit logging

A Security World comprises:

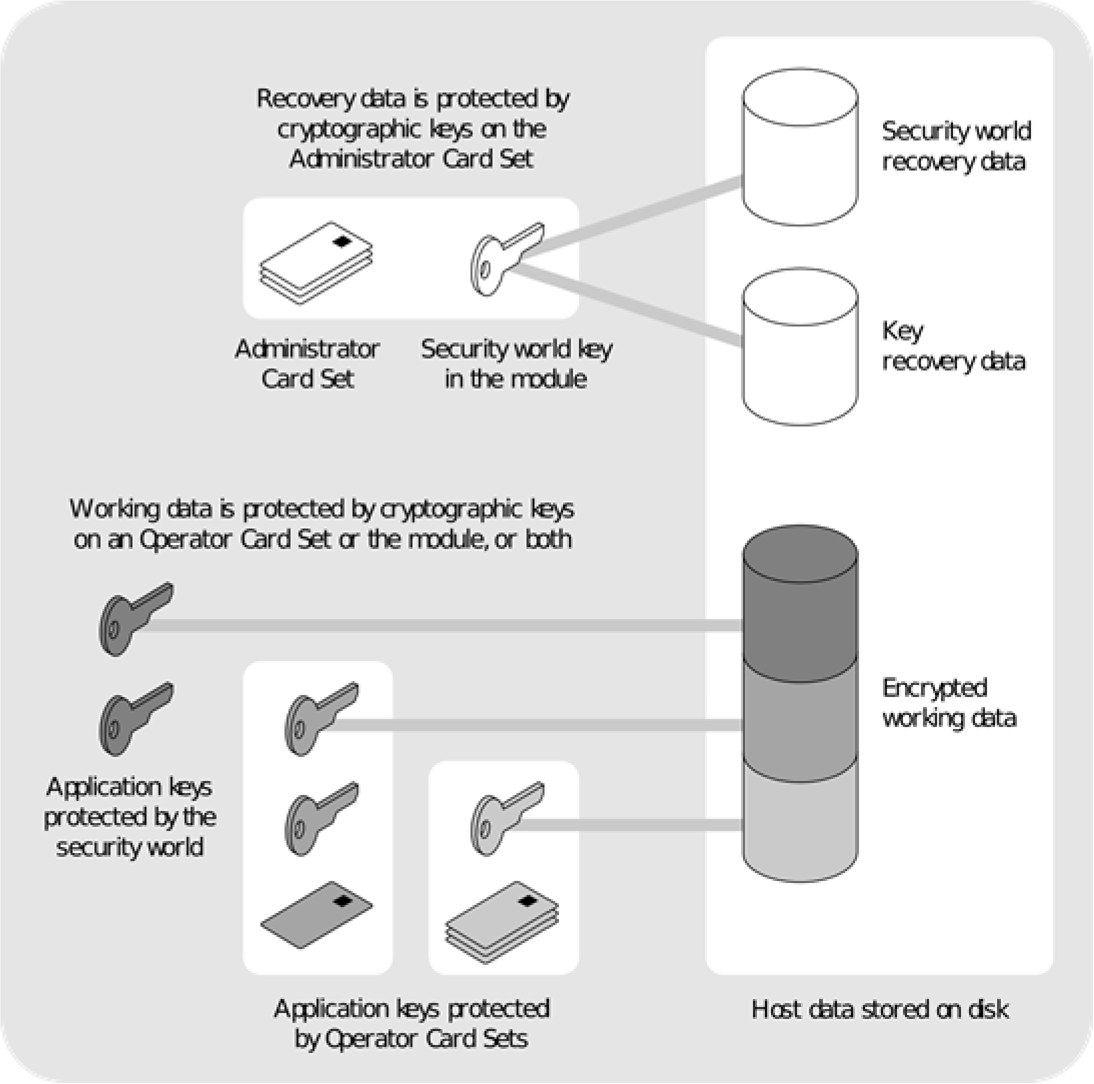
- One or more Entrust nShield HSMs.
- An Administrator Card Set (ACS).
  A set of Administrator smart cards used to control access to the Security World configuration, as well as in recovery and replacement operations.

  > ❗ Store the ACS in a secure location, separate from the HSMs, when you are not using them to provide authorization for administrative tasks.

- Optionally, one or more Operator Card Sets (OCSs).
  A set or sets of Operator smart cards used to control access to application keys.
- Some cryptographic key and certificate data that is encrypted using the Security World key and stored on a host computer or computers.

You can add or remove cards, keys, and even hardware security modules at any time. These components are linked by the Security World key, which is unique to each world. To see how these components are related to one another, see the image below.

Distributing the keys used for different tasks within the Security World over different storage media means that the Security World can recover from the loss of any one component. It also increases the difficulties faced by an attacker, who needs to obtain all the components before gaining any information.


Recovery data is protected by cryptographic keys on the Administrator Card Set
Security world recovery data
Administrator Card Set
Security world key in the module
Key recovery data
Working data is protected by cryptographic keys on an Operator Card Set or the module, or both
Encrypted working data
Application keys protected by the security world
Application keys protected by Operator Card Sets
Host data stored on disk

2.1. Security

The Security World technology is designed to ensure that keys remain secure throughout their life cycle. Every key in the Security World is always protected by another key, even during recovery and replacement operations.

Because the Security World is built around nShield key-management modules, keys are only ever available in plain text on secure hardware.

> All Security Worlds rely on you using the security features of your operating system to control the users who can access the Security World and, for example, write data to the host.

2.1.1. Smart cards

The Security World uses an ACS to control access to recovery and replacement functionality. It can also use one or more OCSs to control access to application keys.

> In FIPS 140 Level 3 Security Worlds, you require a card from either the ACS or an OCS to authorize most operations, including the creation of keys and OCSs.

Each card set consists of a number of smart cards, *N*, of which a smaller number, *K*, is required to authorize an action. The required number *K* is known as the *quorum*.

> The value for *K* should be less than *N*. We do not recommend creating card sets in which *K* is equal to *N* because an error on one card would render the whole card set unusable. If your ACS became unusable through such an error, you would have to replace the Security World and generate new keys. In Common Criteria CMTS Security Worlds the minimum value of *K* for the ACS is 2.

An ACS is used to authorize several different actions, each of which can require a different value for *K*. All the card sets are distinct: a smart card can only belong to the ACS or to one OCS.

Each user can access the keys protected by the Security World and the keys protected by their OCS. They cannot access keys that are protected by another OCS.

Operator Cards employ the Security World key to perform a challenge-response protocol with the hardware security module. This means that Operator Cards are only useable by an HSM that belongs to the same Security World.

2.1.1.1. Smart card types

Multiple smart card types have been introduced over time and are currently supported by the HSMs:

- Type 1 and 2 smart cards can only be presented locally to the HSM and are not compatible with the Remote Administration. These cards are no longer accepted for ACS and OCS creation with the latest Security World cipher suite `ECp521mAES`.
- Type 3 smart cards are compatible with the Remote Administration. These cards offer a higher level of security by establishing a secure channel between the card and the HSM. Two versions are available for the Type 3 smart cards, v1.0 and v1.1 which supports a secure channel protocol compliant with NIST SP800-56Ar3.

The type of a smart card inserted into the HSM card reader can be displayed with the `slot-info` utility.

2.1.2. Remote Operator

The Remote Operator feature is used to load a key protected by an OCS onto a machine to which you do not have physical access (for example, because it is in a secure area).

2.1.3. Remote Administration

Remote Administration is a collection of features that allow you to configure and operate HSMs without being physically present, including creating and presenting ACS and OCS cards.

For more information, refer to Remote Administration v13.6.15 User Guide.

2.1.4. Client cooperation

The client cooperation feature allows nShield HSM host computers to automatically update the Security World and key data stored on a remote file system (RFS). For more information, see Client cooperation.

2.1.5. NIST SP800-131A

When a new Security World is created it will be SP800-131A compliant.

2.1.5.1. Compatibility issues (network-attached HSMs)

In order to comply with the latest encryption standards, Entrust has adopted an enhanced NIST SP800-131A compliant encryption protocol between nShield network-attached HSMs and their clients with Security World software installed. In some cases, this change may have an impact on the compatibility of network-attached HSMs in environments with mixed HSM deployments.

In most cases where versions of Security World software of v11.50 or later are deployed in conjunction with v11.40 software or lower, no action is required. However, there are two cases in which communication cannot be established between the HSM and clients or hosts:

- v11.50 or higher clients communicating with a v11.40 or lower nShield HSM, where the HSM client uses an nToken.
- v11.50 or higher nShield HSM communicating with a Remote File System (RFS) using v11.40 or lower.

Release version	Image versions nShield HSM	Security World Software[1] v11.40	Security World Software[1] v11.50 and later
Up to 11.40	Up to image version 0.3.5.	Supported.	For deployments with nTokens, please upgrade the nShield HSM netimage. As a less preferred option, you can downgrade the client-side software.
11.50 or later	Image 0.3.6 or later.	RFS and client software upgrade required.	Supported.

[1] Previously known as nCipher software, or nCSS.

2.1.6. FIPS 140 compliance

All Security Worlds are compliant with the Federal Information Processing Standards (FIPS) 140 specification. The default setting for Security Worlds complies with Level 2 of FIPS 140.

A Security World that complies with the roles and services section of FIPS 140 Level 2 does not require any authorization to create an OCS or an application key.

2.1.6.1. FIPS 140 Level 3 compliance

When you create a Security World, you can choose whether the Security World is compliant with the roles and services section of either:

- FIPS 140 at Level 2
- FIPS 140 at Level 3

The FIPS 140 Level 3 option is included for those customers who have a regulatory requirement for compliance with FIPS 140 at Level 3.

If you choose to create a Security World that complies with FIPS 140 Level 3, the nShield HSM initializes in that mode, conforming with the roles and services, key management, and self-test sections of the FIPS validation certificate.

For more details about the Security World options to comply with FIPS 140 Level 3, see Security World options.

Before you can create or erase an OCS in a Security World that complies with FIPS 140 Level 3, you must authorize the action with a card from the ACS or an OCS from that Security World.

For more details about FIPS 140, see https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.140-

3.pdf.

2.1.7. Common Criteria compliance (nShield XC HSMs)

The nShield Connect XC and Solo XC HSMs are Common Criteria certified to Common Criteria v3.1 EAL4+ AVA_VAN.5 and to eIDAS.

To configure and operate the module in its evaluated configuration, the separate Common Criteria guides should be followed. Please contact Entrust nShield Support, https://nshield-support.entrust.com.

2.2. Platform independence

The Security World is completely platform independent. All key information is stored in a proprietary format that any computer supported by Security World Software can read, regardless of the native format used by that computer. This enables you to:

- Safely move a Security World between platforms with differing native formats. For example, you can move a Security World between Windows and Linux operating environments.
- Include hosts running different operating systems in the same Security World.

> When copying host data between computers using different operating systems or disk formats, use a mechanism that preserves the original data format and line endings (such as `.tar` file archives).

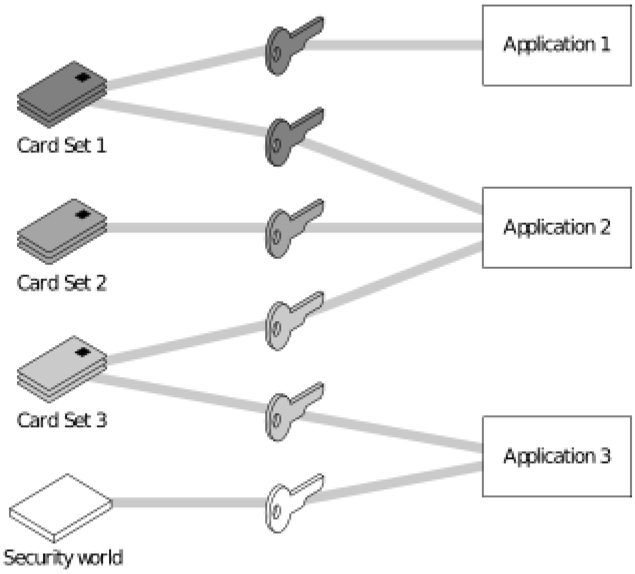
2.3. Application independence

A Security World can protect keys for any applications correctly integrated with the Security World Software. Each key belongs to a specific application and is only ever used by that application. Keys are stored along with any additional data that is required by the application.

You do not need to specify:

- Which applications you intend to use. You can add a key for any supported application at any time.
- How the key is used by an application. A Security World controls the protection for the key; the application determines how it is used.

Although keys belong to a specific application, OCSs do not. You can protect keys for differ ent applications using the same OCS.



In the image above:

- **Card Set 1** protects multiple keys for use with **Application 1** and **Application 2**
- **Card Set 2** protects a single key for use with **Application 2**
- **Card Set 3** protects multiple keys for use with **Application 2** and **Application 3**
- The *Security World key* protects a single key for use with **Application 3**.

2.4. Flexibility

Within a Security World, you can choose the level of protection for each application key that you create.

When you create a Security World, a cryptographic key is generated that protects the appli cation keys and the OCSs in the Security World.

See Security World keys for more information.

2.5. Scalability

A Security World is scalable. You can add multiple hardware security modules to a server and share a Security World across multiple servers. You can also add OCSs and application keys at any time. You do not need to make any decisions about the size of the Security World when you create it.

To share a Security World across multiple clients or servers:

Network-attached HSMs

- Ensure each client has at least one hardware security module configured
- Copy the Security World data to each client
- Load the Security World onto the hardware security modules for each client.

PCIe and USB HSMs

- Ensure each server has at least one hardware security module fitted
- Copy the host data to each server, or make it available on a shared disk
- Use the recovery and replacement data with the ACS to load the required cryptographic keys securely onto every hardware security module.

If you create cards or keys in a Security World from a client rather than on the hardware security module (using the command line, the Microsoft CSP wizard), you must transfer the files from the client to the remote file system, unless the client is already on the same computer as a remote file system.

To provide access to the same keys on every server:

- **Network-attached HSMs**
  You must ensure that all changes to the client data are propagated to the remaining servers. If your clients are part of a cluster, then the tools provided by the cluster should synchronize the data.
- **PCIe and USB HSMs**
  To provide access to the same keys on every server, you must ensure that all changes to the data are propagated to the remaining servers. If your servers are part of a cluster, then the tools provided by the cluster should synchronize the data. If the servers are connected by a network, then they could all access the same copy of the data.

There is no risk of an attacker obtaining information by snooping on the network, as the data is only ever decrypted inside a hardware security module.

Alternatively, you can maintain copies of the data on different clients (**network-attached HSMs**) or servers (**PCIE and USB HSMs**).

You can configure the host computer of an nShield PCIe or USB HSM to:

- Access a Remote File System (RFS) as used by an nShield HSM.
- Share Security World and key data stored in `/opt/nfast/kmdata/local` (**Linux**) or `%NFAST_KMDATA%\local` (**Windows**).

Client hardware security modules that access data in this way are described as *cooperating clients*. For more information, see Client cooperation.

> Use the `rfs-sync` command-line utility to synchronize `opt/nfast/kmdata/local` (**Linux**) or `%NFAST_KMDATA%\local` (**Windows**) between a cooperating client and the remote file system it is configured to access. Run `rfs-sync` whenever a cooperating client is initialized, to retrieve data from the remote file system, and also whenever a client needs to update its local copy of the data (or, if the client has write access, to commit changes to the data).

If you want to make cards or keys which are normally created from the client available from the front panel of a network-attached HSM, we recommend that you use client cooperation to automate the copying of files to the device.

2.5.1. Load-sharing

If you have more than one hardware security module on your system or configured with a client, your applications (that have been integrated with the Security World Software) can make use of the load-sharing features in the Security World Software to share the cryptography between them.

The following approaches are supported:

- API specific load-sharing modes
- HSM Pool mode: a more generic load-sharing approach for module protected keys introduced with module firmware version 2.65.2.

> Some applications may not be able to make use of these features.

HSM Pool mode is supported on all major APIs except in nCipherKM JCA/JCE CSP. When HSM Pool mode is enabled for an API, the application sees the HSMs in the Security World as a single resource pool. A significant benefit is that when a failed HSM is restored to the Security World or a new HSM is added to the Security World, it is automatically added to the resource pool making it available for cryptographic operations as failback support, that is, without restarting the application. The pool of HSMs can be viewed as a single resource using the command `enquiry --pool`.

> For certain situations, `enquiry` and `nfkminfo` fields for modules in the pool have a particular meaning:
>
> - `Module #1: Not Present` indicates that there are no HSMs in the

> pool.
>
> - `hardware status: network error` on a module indicates that the Security World has changed, for example another HSM has been removed from the Security World. Run the same utility (`enquiry` or `nfkminfo`) directly on the module to obtain the hardware status for the module.

2.6. Audit Logging

The Audit Logging facility can be enabled on nShield HSMs where there is a requirement to provably log events. This facility provides the following features:

- Tamper evident logging of relevant nCore command execution on the HSM
- Tied to Security World
- Traceability of cryptographic key lifetime
  - Authorization for key usage
  - Key loading onto HSM
  - Optional logging of key usage
  - Key destruction
- Public key log verification

The format of the audit logs depends on the version of firmware loaded on the HSMs.

For further information, see Audit Logging.

2.7. Applications and Security Worlds

A Security World can protect keys for a range of industry standard applications. For details of the applications that are currently supported, visit https://nshieldsupport.entrust.com.

We have produced Integration Guides for many supported applications. The Integration Guides describe how to install and configure an application so that it works with Entrust hardware security modules and Security Worlds.

For more information about the Entrust range of Integration Guides:

- Visit https://nshieldsupport.entrust.com.
- Contact Support.

2.8. The nShield PKCS #11 library and Security Worlds

> Do not use PKCS #11 to perform any task that requires an Administrator Card. Use the equivalent nShield utilities instead.

Many applications use a PKCS (Public Key Cryptography Standard) #11 library to generate and manage cryptographic keys. We have produced an nShield version of the PKCS #11 library that uses the Security World to protect keys.

Enabling a PKCS #11 based application to use nShield hardware key protection involves configuring the application to use the nShield PKCS #11 library.

The nShield PKCS #11 library treats a smart card from an OCS in the current Security World as a PKCS #11 token. The current PKCS #11 standard only supports tokens that are part of a 1-of-*N* card set, however the nShield PKCS #11 library has vendor specific extensions that support K/*N* card sets, see PKCS#11 library with the preload utility.

A Security World does not make any distinction between different applications that use the nShield PKCS #11 library. Therefore, you can create a key in one PKCS #11 compliant application and make use of it in a different PKCS #11 compliant application.

2.9. Risks

Even the best-designed tools cannot offer security against every risk. Although a Security World can control which user has access to which keys, it cannot prevent a user from using a key fraudulently. For example, although a Security World can determine if a user is authorized to use a particular key, it cannot determine whether the message that is sent with that key is accurate.

A Security World can only manage keys that were created inside the Security World. Keys created outside a Security World, even if they are imported into the Security World, may remain exposed to a security risk.

Most failures of security systems are not the result of inherent flaws in the system, but result from user error. The following basic rules apply to any security system:

- Keep your smart cards safe.
- Always obtain smart cards from a trusted source: from Entrust or directly from the smart card manufacturer.

  > nShield Remote Administration Cards can only be supplied by Entrust.

- Never insert a smart card used with key management products into a smart card reader you do not trust.
- Never insert a smart card reader you do not trust into your hardware security module.
- Never tell anyone your passphrase.
- Never write down your passphrase.
- Never use a passphrase that is easy to guess.

> If you have any doubts about the security of a key and/or Security World, replace that key and/or Security World with a newly generated one.

3. Robustness

Cryptography must work 24 hours a day, 7 days a week, in a production environment. If something does go wrong, you must be able to recover without compromising your security. A Security World offers all of these features.

3.1. Backup and recovery

The Security World data stored on the file system and remote file system of a network-attached HSM or the host of a PCIe or USB HSM is encrypted using the Security World key.

You should regularly back up the data stored in the Key Management Data directory with your normal backup procedures. It would not matter if an attacker obtained this data because it is worthless without the Security World key, stored in your hardware security module, and the Administrator cards for that Security World.

When you create a Security World, it automatically creates recovery data for the Security World key. As with all host data, this is encrypted with the same type of key as the Security World key. The cryptographic keys that protect this data are stored in the ACS. The keys are split among the cards in the ACS using the same *K*/*N* mechanism as for an OCS. The ACS protects several keys that are used for different operations.

The cards in the ACS are only used for recovery and replacement operations and for adding extra hardware security modules to a Security World. At all other times, you must store these cards in a secure environment.

> In FIPS 140 Level 3 Security Worlds, the ACS or an OCS is needed to control many operations, including the creation of keys and OCSs.

3.2. Replacing a hardware security module

If you have a problem with an HSM, you can replace it with another hardware security module of the same type by:

- **Network-attached HSMs:** loading the Security World data on the remote file system onto the replacement device.
  Alternatively, you may be able to erase the Security World from the device that has the problem, return the device to its default state and then reload the Security World on the same device.
- **PCIe and USB HSMs:** using the ACS and the recovery data to load the Security World key securely.

Use the same mechanism to reload the Security World key if you need to upgrade the firmware in the hardware security module or if you need to add extra hardware security modules to the Security World.

If you have more than one hardware security module on your system or configured with a client and you use one of the load-sharing modes identified above, then your system or client application is resilient to the failure of individual hardware security modules.

If you use HSM Pool mode, then an nShield network-attached HSM can be replaced and returned to the HSM pool without restarting the client application.

For information about replacing a hardware security module, see Adding or restoring an HSM to the Security World.

3.3. Replacing the Administrator Card Set

If you lose one of the smart cards from the ACS, or if the card fails, you must immediately create a replacement set using one of the following methods:

- The front panel controls of an nShield network-attached HSM.
- The `racs` utility (see racs).

> ℹ You should also use one of these methods to migrate the ACS from standard nShield cards to nShield Remote Administration Cards. Authorization needs to take place using the local slot of an HSM.

> ℹ When using the `racs` utility, you cannot redefine the quantities in a K of N relationship for an ACS. The K of N relationship defined in the original ACS persists in the new ACS.

A hardware security module does not store recovery data for the ACS. Provided that *K* is less than *N* for the ACS, and you have at least *K* cards available, a hardware security module can re-create all the keys stored on the device even if the information from other cards is missing.

The loss or failure of one of the smart cards in the ACS means that you must replace the ACS. However, you cannot replace the ACS unless you have:

- The required number of current cards
- Access to their passphrases.

> ❗ Although replacing the ACS deletes the copy of the recovery data on your host, you can still use the old ACS with the old host data, which

> you may have stored on backup tapes and other hosts. To eliminate any risk this may pose, we recommend erasing the old ACS as soon as you create a new ACS.

3.4. Replacing an Operator Card Set or recovering keys to softcards

If you lose an Operator Card, you lose all the keys that are protected by that card. To prevent this, you have the option to store a second copy of the working key that the recovery key protects in a Security World. Similarly, you can recover keys protected by one softcard to another softcard.

> The ability to replace an OCS is an option that is enabled by default during Security World creation (see OCS and softcard replacement). You can only disable the OCS replacement option during the Security World creation process. You cannot restore the OCS replacement option, or disable this option, after the creation of the Security World.

> You can only recover keys protected by an OCS to another OCS, and not to a softcard. Likewise, you can only recover softcard-protected keys to another softcard, and not to an OCS.

Network-attached HSMs: To create new copies of the keys protected by the recovery key on an OCS, you can use either:

- The front panel controls of the nShield HSM
- The `rocs` command-line utility.

> It is not possible to recover PKCS #11 keys using the front panel controls of the nShield HSM. You must use the `rocs` command-line utility.

PCIe and USB HSMs: To create new copies of the keys protected by the recovery key on a given card set, and to recover keys protected by one softcard to another softcard, use the `rocs` command-line utility.

3.4.1. The security of recovery and replacement data

Replacing OCSs and softcards requires authorization. To prevent the duplication of an OCS or a softcard without your knowledge, the recovery keys are protected by the ACS.

However, there is always some extra risk attached to the storage of any key-recovery or

OCS and softcard replacement data. An attacker with the ACS and a copy of the recovery and replacement data could re-create your Security World. If you have some keys that are especially important to protect, you may decide:

- To issue a new key if you lose the OCS that protects the existing key
- Turn off the recovery and replacement functions for the Security World or the recovery feature for a specific key.

You can only generate recovery and replacement data when you create the Security World or key. If you choose not to create recovery and replacement data at this point, you cannot add this data later. Similarly, if you choose to create recovery and replacement data when you generate the Security World or key, you cannot remove it securely later.

If you have not allowed recovery and replacement functionality for the Security World, then you cannot recover any key in the Security World (regardless of whether the key itself was created as recoverable).

The recovery data for application keys is kept separate from the recovery data for the Secu rity World key. The Security World always creates recovery data for the Security World key. It is only the recovery of application keys that is optional.

4. Security World keys

4.1. Using the Security World key: module-protected keys

You can use the Security World key to protect an application key that you must make available to all your users at all times. This key is called a *module-protected key*. Module-protected keys:

- Have no passphrase
- Are usable by any instance of the application for which they were created, provided that this application is running on a server fitted with a hardware security module belonging to the correct Security World.

This level of protection is suitable for high-availability Web servers that you want to recover immediately if the computer resets.

4.2. Using Operator Card Sets: OCS-protected keys

An OCS belongs to a specific Security World. Only a hardware security module within the Security World to which the OCS belongs can read or erase the OCS. There is no limit to the number of OCSs that you can create within a Security World.

An OCS stores a number of symmetric keys that are used to protect the application keys. These keys are of the same type as the Security World key.

Each card in an OCS stores only a fragment of the OCS keys. You can only re-create these keys if you have access to enough of their fragments. Because cards sometimes fail or are lost, the number of fragments required to re-create the key (*K*) are usually less than the total number of fragments (*N*).

To make your OCS more secure, we recommend that you make the value of *K* relatively large and the value of *N* less than twice that of *K* (for example, the values for *K*/*N* being 3/5 or 5/9). This practice ensures that if you have a set of *K* cards that you can use to recreate the key, then you can be certain that there is no other such card set in existence.

> Some applications restrict *K* to 1.

4.2.1. Using Operator Card Sets to share keys securely

You can use OCSs to enable the same keys for use in a number of different HSMs at the same time.

If you have a non-persistent OCS, you must leave one of the cards in an appropriate card slot of each HSM. This should only be done if it is in accordance with the security policies of your organization.

To use OCS-protected keys across multiple HSMs, set:

- *K* to 1
- *N* at least equal to the number of the HSMs you want to use.

You can then insert single cards from the OCS into the appropriate card slot of each HSM to authorize the use of that key.

To issue the same OCS-protected key to a set of users, set:

- *K* to 1
- *N* equal to the number of users.

You can then give each user a single card from the OCS, enabling those users to authorize the use of that key.

> If you have created an OCS for extra security (in which *K* is more than half of *N*), you can still share the keys it protects simultaneously amongst multiple modules as long you have enough unused cards to form a *K*/*N* quorum for the additional hardware security modules. For example, with a 3/5 OCS, you can load keys onto 3 hardware security modules because, after loading the key on the first device, you still have 4 cards left. After loading the key on a second device, you still have 3 cards left. After loading the key onto a third device, you have only 2 cards left, which is not enough to create the quorum required to load the key onto a fourth device.

If a card becomes damaged, you can replace the whole OCS if you have authorization from the ACS belonging to that Security World.

> You can only replace OCSs that were created by Security Worlds that have the OCS/softcard replacement option enabled. For more information, see OCS and softcard replacement.

4.2.2. Using Operator Card Sets for high availability

If you cannot risk the failure of a smart card, but some keys must remain accessible at all times, you can create a 1/2 OCS.

Use the first card as the working card and store the second card in a completely secure envi ronment. If the working card fails, retrieve the spare second card from storage, and use it until you re-create a new set of 2 cards (see Replace OCS and softcards).

> You can only replace OCSs that were created by Security Worlds that have the OCS/softcard replacement option enabled. For more information, see OCS and softcard replacement.

4.2.3. Using persistent Operator Card Sets

If you create a standard (non-persistent) OCS, you can only use the keys protected by that OCS while the last required card of the quorum remains loaded in the card reader. The keys protected by this card are removed from the memory of the hardware security module as soon as the card is removed from the card reader, which provides added security.

If you create a *persistent* OCS, the keys protected by a card from that OCS persist after the card is removed from the smart card reader.

This enables:

- The use of the same smart card in several hardware security modules at the same time
- Several users to load keys onto the same hardware security module at the same time.

The Security World Software maintains strict separation between the keys loaded by each user, and each user only has access to the keys protected by their OCS.

Keys protected by a persistent card are automatically removed from the hardware security module:

- When the application that loaded the OCS closes the connection to the hardware secu rity module
- After a time limit that is specified when the card set is created
- When an application chooses to remove a key
- When the HSM is cleared. See Manually removing keys from an HSM for more information
- If there is a power loss to the module, for example, due to power outage.

> Some applications automatically remove a key after each use, reloading it only when required. Such applications do not benefit from persistent OCSs. The only way of sharing keys between hardware security modules for such applications is by having multiple smart cards in an OCS.

Although the hardware security module stores the key, the key is only available to the application that loaded it. To use keys protected by this card in another application, you must re-insert the card, and enter its passphrase if it has one. Certain applications only permit one user at a time to log in, in which case any previously loaded persistent OCS used in that application is removed before the user is allowed to log in with a new OCS.

4.2.4. Manually removing keys from an HSM

You can manually remove all keys protected by persistent cards by clearing the hardware security module. For example, you could:

- Run the command `nopclearfail --clear --all`
- Press the Clear button of the hardware security module (**nShield Connect XC**)
- Turn off power to the hardware security module (**network-attached HSMs**)

Any of these processes removes all keys protected by OCSs from the hardware security module. In such cases, all users of any applications using the hardware security module must log in again.

Persistence is a permanent property of the OCS. You can choose whether or not to make an OCS persistent at the time of its creation, but you cannot change a persistent OCS into a non-persistent OCS, or a non-persistent OCS into a persistent OCS.

A Security World can contain a mix of persistent and non-persistent card sets.

4.3. Using passphrases for extra security

You can set individual passphrases for some or all the cards in an OCS.

You can change the passphrase for a card at any time provided that you have access to the card, the existing passphrase, and a hardware security module that belongs to the Security World to which the card belongs.

> Some applications do not support the use of passphrases.

4.3.1. Maximum passphrase length

> The maximum passphrase length limitation is not applicable to software versions before Security World Software v11.72.

passphrases are limited to a maximum length of 254 characters, when using the following commands:

- new-world
- createocs
- cardpp
- ppmk
- racs

Other commands are unaffected.

You can still use and edit existing passphrases that are longer than 254 characters.

Prior to Security World Software v11.72, we set no absolute limit on the length of passphrases, although individual applications may not accept passphrases longer than a specific number of characters. Likewise, the Security World does not impose restrictions on which characters you can use in a passphrase, although some applications may not accept certain characters.

Entrust recommends that your password only contains 7-bit ASCII characters:

```
A-Z, a-z, 0-9, ! @ # $ % ^ & * - _ + = [ ] { } | \ : ' , . ? / ` ~ " < > ( ) ;
```

4.3.2. passphrase penalty timer

The HSM maintains a penalty time, measured in seconds and based on the number of failed PINs. Each failed attempt to enter a passphrase adds 4 seconds to the penalty time.

The penalty timer has a 14s penalty threshold, the first 3 failed passphrase verifications do not incur a penalty delay. Before verifying a passphrase, the HSM waits for the current penalty timer to be below 14s. The penalty time decays over time.

> A HSM only has a small number of command processing threads, related to the kind of hardware in use (for example, 9 threads on an nShield Solo). Once all of these are waiting for a penalty to expire, any other submitted commands will be forced to wait. This can mean that even if penalty time isn’t large, the total delay experienced by clients may be substantial.

4.4. Using softcard-protected keys

If you want to use passphrases to restrict key access but avoid using physical tokens (as required by smart-card protection), you can create a *softcard-protected key*.

A *softcard* is a file containing a logical token that you cannot load without a passphrase. You must load the logical token to authorize the loading of any key that is protected by the softcard. Softcard files:

- Are stored in `/opt/nfast/kmdata/local` (**Linux**) or `%NFAST_KMDATA%\local` (**Windows**).
- Have names of the form `softcard_hash` (where `hash` is the hash of the logical token share).

Softcard-protected keys offer better security than module-protected keys and better availability than OCS-protected keys. However, because softcard-protected keys do not require physical tokens to authorize key-loading, OCS-protected keys offer better security than softcard-protected keys.

The passphrase of a softcard is set when you generate it, and you can use a single softcard to protect multiple keys. Softcards function as persistent 1/1 logical tokens, and after a softcard is loaded, it remains valid for loading its keys until its `KeyID` is destroyed.

4.5. NVRAM key storage (network-attached HSMs)

Application keys protected by an nShield network-attached HSM are stored in an encrypted format and copied automatically to the remote file system. A hardware security module, such as an nShield HSM, and/or an OCS protects the keys, as described in the preceding sections. You can also store application keys within the nonvolatile memory of a suitable hardware security module.

NVRAM-stored keys are encrypted in exactly the same way as application keys that are protected by the unit. The encrypted application key on the unit is replaced by a file containing the name of the NVRAM file that contains the application key. This file allows applications to use NVRAM-stored keys in the same way as keys stored in the remote file system. You can protect an NVRAM-stored key with either the Security World or an OCS.

> NVRAM-stored keys differ from standard application keys only in their storage location. They still require protection by the unit or an OCS.

Use of an NVRAM-stored key is the same as for any other key protected by an nShield HSM Security World.

NVRAM key storage:

- Offers no additional security benefits above those offered by the standard Security World Software mechanisms
- Is available for only a limited number of keys

- Reduces a Security World's ability to offer load-balancing and recovery
- Requires backup and recovery procedures in addition to any that you implement for data stored on the client computer.

> ❗ Do not store keys in NVRAM unless you must do so to satisfy regulatory requirements.

NVRAM key storage was introduced only for those users who must store keys within the physical boundary of a hardware security module to comply with regulatory requirements. NVRAM-stored keys provide no additional security benefits and their use exposes your ACS to increased risk. Storing keys in nonvolatile memory also reduces load-balancing and recovery capabilities. Because of these factors, we recommend you always use standard Security World keys unless explicitly required to use NVRAM-stored keys.

5. Security World options

Decide what kind of Security World you need before you create it. Depending on the kind of Security World you need, you can choose different options at the time of creation. For convenience, Security World options can be divided into the following groups:

- Basic options, which must be configured for all Security Worlds
  - Optionally enable Audit Logging for the Security World
- Recovery and replacement options, which must be configured if the Security World, keys, or passphrases are to be recoverable or replaceable
- SEE options, which only need be configured if you are using CodeSafe
- Options relating to the replacement of an existing Security World with a new Security World.

Security World options are highly configurable at the time of creation but, so that they will remain secure, not afterwards. For this reason, we recommend that you familiarize yourself with Security World options, especially those required by your particular situation, before you begin to create a Security World.

5.1. Security World basic options

When you create a Security World, you must always configure the basic options described in this section.

5.1.1. Cipher suite

The following cipher suites are available:

Cipher suite	Release version	Supported smart card types	Internal mechanisms
`ECp521mAES`	v13.7 or higher	Type 3	ECDSA signatures and ECC for key recovery
`DLf3072s256mAESc-SP800131Ar1` (default)	v12.50 or higher	Type 1, 2 and 3	DSA signatures and RSA for key recovery

5.1.1.1. KML type selection

KML is the module signing key. The type of key used can be selected when a Security World is created and/or loaded using the `--kml-type` option with the `new-world` utility, otherwise the default KML type for the cipher suite is used. The KML types supported are listed in the following table:

KML types	Description
NISTp521hSHA1	ECDSA KML on NIST P-521
NISTp256hSHA1	ECDSA KML on NIST P-256
DSAp3072s256	DSA KML with a 3072-bit modulus in a 256-bit order subgroup
DSAp1024s160	DSA KML with a 1024-bit modulus in a 160-bit order subgroup

Note that the KML type selected when loading a Security World can be different from the KML type used when the Security World was created.

The KML type used when the Security World was created and the active KML type are both reported in the nfkminfo output, respectively in the World and Module information (see nfkminfo utility).

5.1.1.1.1. Acceptability rules

The following table lists the KML types acceptable for each cipher suites:

Cipher suite	Default KML type	Acceptable KML types
ECp521mAES	NISTp521hSHA1	NISTp256hSHA1, NISTp521hSHA1
DLf3072s256mAEScSP800131Ar1	DSAp3072s256	DSAp3072s256, NISTp256hSHA1[2] [3]
DLf3072s256mRijndael[1]	DSAp3072s256	DSAp3072s256, NISTp256hSHA1[2] [3]
DLf1024s160mRijndael[1]	DSAp1024s160	DSAp1024s160
DLf1024s160mDES3[1]	DSAp1024s160	DSAp1024s160

The KML type selection has an impact on performances and security, for more information see KML type and security strength.

[1] Legacy cipher suite.

[2] In FIPS 140 Level 3 mode, ECDSA KML types are not supported with DSA Security Worlds.

[3] It is not recommended to use the module file with legacy Security World software - i.e. prior to v13.7 - when a DSA Security World has been loaded with an ECDSA KML.

5.1.2. ACS quorum

You must decide the total number of cards (*N*) in a Security World's ACS and must have that many blank cards available before you start to create the Security World. You must also decide how many cards from the ACS must be present (*K*) when performing administrative functions on the Security World.

> ℹ We recommend that you do not create ACSs for which *K* is equal to *N*, because you cannot replace such an ACS if even 1 card is lost or damaged.

> ℹ In Common Criteria CMTS Security Worlds the minimum value of K for the ACS is 2.

In many cases, it is desirable to make *K* greater than half the value of *N* (for example, if *N* is 7, to make *K* 4 or more). Such a policy makes it harder for a potential attacker to obtain enough cards to access the Security World. Choose values of *K* and *N* that are appropriate to your situation.

The total number of cards used in the ACS must be a value in the range 1 – 64.

5.1.3. FIPS 140 Level 3 compliance

By default, Security Worlds are created to comply with the roles and services, key management, and self-test sections of the FIPS 140 standard at Level 2. However, you can choose to enable compliance with the FIPS 140 standard at Level 3.

> ℹ This option provides compliance with the roles and services of the FIPS 140 Level 3 standard. It is included for those customers who have a regu latory requirement for compliance.

If you enable compliance with FIPS 140 Level 3 roles and services, authorization is required for the following actions:

- Generating a new OCS
- Generating or importing a key, including session keys
- Erasing or formatting smart cards (although you can obtain authorization from a card you are about to erase).

In addition, you cannot import or export private or symmetric keys in plain text.

5.1.3.1. FIPS 140 Level 3 approved configurations

In order to comply with FIPS 140 Level 3, the HSM must be initialized with a Security World in FIPS 140 Level 3 mode, and with the appropriate cipher suite as detailed in the following table:

Product	Firmware version	FIPS standard	Security World cipher suite	Revisions
nShield 5s	v13.7.X (upcoming)	FIPS 140-3	`ECp521mAES`	FIPS 186-5 revision
	v13.2, v13.4		`DLf3072s256mAESc-SP800131Ar1`	NIST SP 800-131Ar1 transition
nShield Solo XC	v12.72.1	FIPS 140-2		
nShield Edge	v12.72.0			

5.1.4. Common-Criteria-CMTS Support (nShield Solo XC only)

You can choose to enable support for EN 419 221-5 Protection profiles for TSP Cryptographic modules - Part 5 - Cryptographic Module for Trust Services using the common-criteria-cmts mode, see https://www.commoncriteriaportal.org/files/ppfiles/ANSSI-CC-PP-2016_05%20PP.pdf

If you enable support for the EN 419 221-5 Protection Profile the following constraints and facilities are enabled:

- Constraints:
  - The minimum quorum for the ACS cardset is 2
  - You cannot import or export private or symmetric keys in plain text
  - Remote Operator feature is disabled.
- Facilities:
  - `generatekey` and `mkaclx` utilities support generating EN 419 221-5 Assigned Keys
  - `nfkmverify` supports the verification of EN 419 221-5 Assigned Keys.

In order to meet the requirements of this Protection Profile the HSM must be operated in accordance with the *nShield Solo XC Common Criteria Evaluated Configuration Guide*.

5.1.5. UseStrongPrimes Security World setting

From firmware version 12.70, the nShield HSM always targets FIPS 186-4 compliance when generating RSA keys of 1024 bits or more. It typically does this using a "strong primes" strat egy, however Entrust only guarantees this strategy if the `UseStrongPrimes` setting is enabled.

If your firmware is version 12.70 or higher, you do not need this setting enabled for FIPS 186-4 compliance.

If you are using an older version of firmware, meaning it has a version number *lower than* 12.70, then you need the `UseStrongPrimes` setting enabled to grant FIPS 186-2 compliance.

If your Security World is FIPS 140 Level 3, then this setting is on by default. If your Security World is not FIPS 140 Level 3, then you can disable the `UseStrongPrimes` setting for faster RSA key generation, however this removes FIPS 186-2 compliance.

5.1.6. Remote Operator

To use a module without needing physical access to present Operator Cards, you must enable the Remote Operator feature on the module. For more information, see Optional features.

By default, modules are initialized into Security Worlds with remote card set reading enabled. If you add a module for which remote card reading is disabled to a Security World for which remote card reading is enabled, the module remains disabled.

5.2. OCS and softcard replacement

By default, Security Worlds are created with the ability to replace one OCS or softcard with another. This feature enables you to transfer keys from the protection of the old OCS of softcard to a new OCS or softcard.

> You can replace an OCS with another OCS, or a softcard with another softcard, but you cannot replace an OCS with a softcard or a softcard with an OCS. Likewise, you can transfer keys from an OCS to another OCS, or from a softcard to another softcard, but you cannot transfer keys from an OCS to a softcard or from a softcard to an OCS.

You can choose to disable OCS and softcard replacement for a Security World when you create it. However, in a Security World without this feature, you can never replace lost or damaged OCSs; therefore, you could never recover the keys protected by lost or damaged OCSs, even if the keys themselves were generated as recoverable (which is the default for key generation).

> OCS and softcard replacement cannot be enabled after Security World creation without reinitializing the Security World and discarding all the existing keys within it.

For an overview of Security World robustness and OCS or softcard replacement, see Robustness. For details about performing OCS and softcard replacement operations, see Replace OCS and softcards and Replace the ACS.

5.3. passphrase replacement

By default, Security Worlds are created so that you cannot replace the passphrase of a card or softcard without knowing the existing passphrase.

However, you can choose to enable passphrase replacement at the time you create a Security World. This option makes it possible to replace the passphrase of a card or softcard even if you do not know the existing passphrase. Performing such an operation requires authorization from the Security World's ACS.

For details about performing passphrase replacement operations, see Changing unknown or lost passphrase.

5.4. Nonvolatile memory (NVRAM) options

Enabling nonvolatile memory (NVRAM) options allows keys to be stored in the module's NVRAM instead of in the Key Management Data directory of the host computer. Files stored in the module's non-volatile memory have Access Control Lists (ACLs) that control who can access the file and what changes can be made to the file. NVRAM options are rele vant only if your module's firmware supports them, and you can store keys in your module's NVRAM only if there is sufficient space.

> When the amount of information to be stored in the NVRAM exceeds the available capacity, you can instead store this data in a blob encrypted with a much smaller key that is itself then stored in the NVRAM. This functionality allows the amount of secure storage to be limited only by the capacity of the host computer.

5.5. Security World SEE options

You must configure `SEE options` if you are using the nShield Secure Execution Engine (SEE). If you do not have SEE installed, the SEE options are irrelevant.

5.5.1. SEE debugging

SEE debugging is disabled by default, but you can choose whether to enable it for all users or whether to make it available only through use of an ACS. In many circumstances, it is useful to enable SEE debugging for all users in a development Security World but to disable SEE debugging in a production Security World. Choose the SEE debugging options that best suit your situation.

5.5.2. Real-time clock (RTC) options

Real-time clock (RTC) options are relevant only if you have purchased and installed the CodeSafe Developer kit. If so, by default, Security Worlds are created with access to RTC operations enabled. However, you can choose to control access to RTC operations by means of an ACS.

5.6. Security World replacement options

Options relating to Security World replacement are relevant only if you are replacing a Security World.

If you replace an existing Security World, its `/opt/nfast/kmdata/local` (**Linux**) or `%NFAST_KMDATA%\local` (**Windows**) directories are not overwritten but renamed to `/opt/nfast/kmdata/local_N` (**Linux**) or `%NFAST_KMDATA%\local_N` (**Windows**), (where *N* is an integer assigned depending on how many Security Worlds have been previously saved during overwrites). A new Key Management Data directory is created for the new Security World. If you do not wish to retain the `/opt/nfast/kmdata/local_N` (**Linux**) or `%NFAST_KM-DATA%\local_N` (**Windows**) directory from the old Security World, you must delete it manually.

6. Create a new Security World

You can use the following to create Security Worlds:

- The new-world command line utility, see Create a Security World using new-world.
- The front panel controls, see Create a Security World using the nShield HSM front panel (network-attached HSMs).

When you create a Security World:

- The HSM is erased.
- A new HSM key for this Security World is generated.
- A new ACS to protect this HSM key is created.
- The Security World information, including encrypted key material and related data, is stored in either the file system of the nShield HSM operating system and on the RFS (**network-attached HSMs**, see RFS) or on the hard disk of the host computer (**PCIe and USB HSMs**).
  For multiple clients or hosts to use the same Security World, the system administrator must ensure that these files are copied to all the clients or hosts and updated when required. For more information about Security World files, see Security World Files.
  - The information is encrypted using the secrets stored on the ACS.
- The HSM and Security World are configured for Audit Logging, if selected.

❗ If you want to re-use the physical cards created in a previous Security World, you must erase all Operator Cards, except for nShield Remote Administration Cards, while the previous Security World still exists. See Erasing cards and softcards.

❗ We recommend that you regularly back up the entire contents of the RFS. Either the `%NFAST_KMDATA%` directory on Windows, or the `kmdata` directory on Linux, is required to restore an nShield HSM or its replacement, to the current state in case of failure.

❗ HSMs enrolled in a Security World with audit-logging enabled (including a Common-Criteria (CMTS) compatible Security World) will continuously generate audit-logs and if these audit-logs are not transferred and removed by the nshieldauditd service, then the HSMs may run out of disk space and will not process certain commands until their audit-logs are transferred and removed. See nShield Audit Log Service

> Due to the additional primality checking required by SP800-131A, Security World generation will take longer when using the new default Ciphersuite (from v12.40 onwards) - on nShield USB-attached HSMs, this could be up to 45 minutes.

6.1. Prerequisites

- Familiarise yourself with the Security World file structure and logic: Security World Files.
- Decide what kind of Security World you need and the options that you want to enable on creation: Security World options.
- Before configuring the Security World, you should know:
  - The security policy for the HSM
  - The number and quorum of Administrator Cards and Operator Cards to be used.
- You must have enough smart cards to form the Security World's card sets.

6.2. Create a Security World using new-world

6.2.1. Before you start

Before you create a Security World:

- The HSM must be in pre-initialization mode.
- **PCIe and USB HSMs:** You must be logged in to the host computer as one of the following users:
  - `root` (**Linux**).
  - A user who is permitted to create privileged connections (**Windows**).
  - A user in the group `nfast`.
- **Network-attached HSMs:** You must be logged in to the computer that is running the RFS. The RFS should be a privileged client that has the client tools installed.
- **Windows:** You must have set the `NFAST_HOME` environment variable.
  This variable is set by default during product software installation.

Linux: You must install the Security World Software in `/opt/nfast/`.` Installation to other locations is not generally supported nor recommended, but if the attempt is made, you must create a symbolic link from `/opt/nfast/` to the directory in which the software is actually installed.

When you have finished creating a Security World, you must change the mode to "Operational" using `nopclearfail -I m1` or `nopclearfail -O -m1`.

If you are using a PCIe or a USB-attached HSM, follow the directions in this section to create a Security World from the command line with the new-world utility.

6.2.2. Run the new-world command-line utility

This example creates a Security World supporting FIPS140 Level 3 with a ACS quorum of 3/5 and with audit logging enabled.

```
new-world --mode=fips-140-level-3 --acs-quorum=3/5 --audit-logging
```

6.2.3. Copy a Security World to a network-attached HSM and check the current version

If a Security World is created using `new-world`, the `nethsmadmin` command-line utility enables you to copy the resultant files to a nShield HSM. Run the command:

```
nethsmadmin --module=<MODULE> --update-world
```

`nethsmadmin` can also be used to check if the Security World files have been copied to the nShield HSM. Run the command:

```
nethsmadmin --module=<MODULE> --check-world
```

In these commands:

`--module=<MODULE>`

Specifies the HSM to use, by its `ModuleID` (default = 1).

Follow the directions in this section to create a Security World from the command line with the new-world utility.

6.3. Create a Security World using the nShield HSM front panel (network-attached HSMs)

When initiated from the nShield HSM front panel, while a Security World is being created the nShield HSM disconnects itself from the network to ensure that the operation is not interrupted. This means that the

> Remote Administration feature cannot be used to present cards from a remote location when creating a Security World from the front panel.

6.3.1. Before you start

Before you start to create a Security World:

- The `/opt/nfast/kmdata/local` (**Linux**) or `%NFAST_KMDATA%\local` (**Windows**) directory must exist on the remote file system and be empty.

To create a Security World from the nShield HSM Front Panel:

1. From the main menu, select **Security World mgmt** > **Module initialization** > **New Security World**.
2. Specify the Security World mode:
   a. **FIPS 140 Level 3** creates a Security World compliant with FIPS 140 requirements for roles and services at Level 3.
   b. **Common Criteria CMTS** creates a Security World supporting Common Criteria Protection Profile EN 419 221-5.
   c. **Unrestricted** creates a Security World which doesn't impose any particular conformance. With appropriate environmental constraints, an unrestricted Security World can be compliant with FIPS 140 Level 2.
3. Select the Cipher suite for the Security World. Currently only one option is available for the Security World key, AES (SP800-131AR1).
4. Enter the default quorum for the ACS. This consists of:
   a. The maximum number of cards from the ACS required by default for an operation. This number must be less than or equal to the total number of cards in the set.
   b. The total number of cards to be used in the ACS. This must be a value in the range 1 – 64 except for the Common Criteria CMTS Security World mode, for which the range is 2 – 64.

   > We recommend that you do not create an ACS for which the required number of cards is equal to the total number of cards because you cannot replace such an ACS if even a single card is lost or damaged.

5. If you answer the question **Specify all quorums?** by selecting:
   a. **no** - all operations and features (with the exception of passphrase recovery) will be enabled and require the maximum number of cards
   b. **yes** - you can specify which operations and features you want to enable (including

passphrase recovery) and what the required number of cards for each of these will be.

6. If you chose to disable individual features or require a lower number of cards required for an operation, specify these parameters now. You can select a different number of Administrator Cards (K) to be required for each operation. You can also disable recovery and replacement operations and choose to use KNSO to authorize SEE (Secure Exe cution Engine) operations. The options for which you can specify a separate value of K are as follows:

Operation	Action allowed on HSM
Module reprogramming	Initializing an HSM into a Security World. You must specify a value of K for this operation.
passphrase replacement	Replacement of passphrases from backup files when recovering an OCS. You can disable this operation, see Passphrase replacement. This operation is disabled in Common Criteria CMTS mode and cannot be enabled.
OCS/softcard replacement	Recovery of keys from backup files when replacing an OCS. You can disable this operation if you are using the nShield HSM, see OCS and softcard replacement.
NVRAM access	Reading from and writing to the NVRAM. You can choose to authorize this operation with K_{NSO}, see Nonvolatile memory (NVRAM) options.
RTC access	Updating the real time clock. This is not applicable for nShield 5c. You can choose to authorize this operation with K_{NSO}, see Real-time clock (RTC) options.
SEE debugging	Viewing full SEE debug information. You can specify a value of K for this operation, all it for all users or authorize it with K_{NSO}, see SEE debugging. This operation is disabled in Common Criteria CMTS mode.
FTO	Use of an Foreign Token Open (FTO) Delegate Key (ISO Smart Card Support). You can specify a value of K for this operation or authorize it with K_{NSO}. This operation is disabled in Common Crite ria CMTS mode.

7. Specify if audit logging should be enabled.

   In Common Criteria CMTS mode, audit logging is automatically enabled and cannot be disabled.

8. Specify whether the HSM is a valid target for remote shares (that is, whether it can import slots), see Remote Operator. This option is disabled for Common Criteria CMTS

mode.

9. For Common Criteria CMTS mode only, choose whether to specify the maximum number of times an Assigned key can be used since it was authorized. A use limit compatible with the specified maximum will be imposed at key creation time and can be verified for Assigned keys. If you choose to specify a maximum key usage limit:

    a. Enter the key usages allowed, up to a maximum of 9999.

10. For Common Criteria CMTS mode only, choose whether to specify a maximum timeout for Assigned keys since key authorization. A time limit compatible with the specified maximum will be imposed when the key is created, and can be verified for Assigned keys. If you choose to specify a key timeout:

    a. Select the units from **Seconds**, **Minutes**, **Hours**, or **Days**.

    b. Enter a value up to a maximum of 9999 in your selected unit.

11. Format a card for the ACS as follows:

    a. Insert a card for the ACS and confirm that you want to use it.

    b. If the card is not blank, choose whether to overwrite it or to use a different card.

    c. Choose whether to specify a passphrase for the card. If you choose to specify a passphrase:

        i. Enter the passphrase.

        ii. Enter the passphrase again to confirm it. If the two passphrases do not match, you must enter the correct passphrase twice.

    d. When prompted, remove the card.

12. Repeat the previous step to format additional cards for the ACS, setting their passphrases as described, until the ACS is complete. Each prompt screen shows how many cards are required and how many have been used.

13. At completion, a message confirms that the Security World has been created.

6.4. After you have created a Security World

If you enrolled a network-attached HSM into a security world, propagate the world and module files to client machines, then run `hsc_configurepoolmodule -mN` on each machine enrolled as a client (`N` is the module number of the newly enrolled HSM). If there is more than one module to configure and the Security World software version is v13.6 or later, you can run `hsc_configurepoolmodule` without any parameters to add all eligible modules to the pool.

Store the ACS in a safe place.

> ❗ If you lose more than *N* minus *K* of these Administrator Cards you can-

> not restore the Security World or lost Operator Cards. For example, if you have a 2/3 ACS and you lose more than one card, you cannot restore the Security World. If you have created an Administrator card set where *K* = *N*, then the loss of one card stops you from being able to restore the Security World.

To prevent this situation from occurring, replace lost or damaged cards from the ACS as soon as you discover the loss or damage. For more information, see Replace the ACS.

> The security of the keys that you create within this Security World is wholly dependent on the security of these smart cards.

In **Network-attached HSMs**, the Security World data is stored on the HSM and on the RFS. For more information see Security World Files.

In **PCIe** and **USB HSMs**, the Security World host data is stored in the directory to which the `NFAST_KMLOCAL` environment variable points (see Security World Files). The data in this directory is encrypted. You should:

- Ensure that this directory is backed up regularly.
- Check the file permissions for this directory.
    - Ensure that the nFast Administrator role, and any user that you want to be able to create Operator Cards or keys, have write permission for this directory.
    - All other valid users must have read permission.

> Installation of Security World Software must be performed by a user with Administrator rights that allow read and write operations, and applications to be started and stopped.

The HSM can now be used to create Operator Cards and keys for the new Security World.

> If you need to migrate existing keys into the Security World, see nShield Security World v13.6.15 Key Management Guide.

7. Add HSMs to a Security World

7.1. Pre-initialization backup (PCIe and USB HSMs)

Initialization removes any data stored in an HSM's nonvolatile memory (for example, data for an SEE program or NVRAM-stored keys). To preserve this data, you must back it up before initializing the HSM and restore it after the HSM has been reprogrammed. We provide the `nvram-backup` utility to enable data stored in nonvolatile memory to be backed up and restored.

In order to continue using existing keys and Operator Cards, you must reprogram the HSM:

- After you upgrade the firmware
- If you replace the HSM
- If you need to add an HSM to an existing Security World.

7.2. Add an HSM to a Security World with the CSP or CNG wizard (Windows)

To add an HSM to an existing Security World:

1. Ensure the HSM is in initialization mode and run the wizard by double-clicking its short cut in the Windows Start menu: **Start > Entrust nShield Security World**.
2. Click the **Next** button.

   The wizard allows you to configure HSM Pool mode for CAPI/CNG.

3. Click the **Next** button.

   If the wizard finds an existing Security World, it prompts you to specify whether you want to use the existing Security World or create a new Security World.

   If the wizard displays any other windows:

   a. Cancel the operation.
   b. Check that you have correctly set the environment variable `NFAST_KMDATA`.
   c. Copy the `local` sub-directory from the Key Management Data of another computer in the same Security World or from a backup tape of this computer to the Key Management Data directory of this computer.
   d. Run the wizard again.

4. Ensure that the **Use the existing security world** option is selected, and click the **Next** button.

You can then proceed to add HSMs in the same manner that you add multiple HSMs when you create a Security World.

7.3. Add an HSM to a Security World with new-world

1. Run:

   ```
   new-world [-l|--program] [-S|--no-remoteshare-cert] [-m|--module=<MODULE>]
   ```

   If you intend to initialize the HSM into a new Security World, run `new-world` with the `-i` option.

   If the HSM is not in the pre-initialization state, `new-world` displays an error and exits.

   The HSM must be in pre-initialization mode.

   If the HSM is in the pre-initialization state, `new-world` prompts you for cards from the Security World's ACS and to enter their passphrases as required.

2. After `new-world` has reprogrammed the HSM, restart the HSM in the operational state.
3. If you enrolled a network-attached HSM into a security world, propagate the world and module files to all client machines.
4. If using HSM Pool Mode, additionally run `hsc_configurepoolmodule -mN` on each machine enrolled as a client (`N` is the module number of the newly enrolled HSM).

   If there is more than one module to configure and the Security World software version is v13.6 or later, it's enough to run `hsc_configurepoolmodule` without any parameters to add all eligible modules to the pool.

5. Store the ACS in a safe place.

> If any error occurs (for example, if you do not enter the correct passphrases), the HSM is reset to the factory state. The HSM does not form part of the Security World unless you run `new-world` again.

7.4. Add an HSM to a Security World using the nShield HSM front panel (network-attached HSMs)

To add an HSM to a Security World:

1. If the HSM already belongs to a Security World, erase it from the Security World to which it belongs, as described in Remove modules and delete Security Worlds.
2. From the main menu, select **Security World mgmt** > **Module initialization** > **Load Security World**.
3. Specify whether the HSM can use the Remote Operator feature import slots. For more information, see Remote Operator.
4. At the prompt, insert an Administrator Card, and enter its passphrase if required.
5. Continue to insert Administrator Cards when prompted until you have inserted the number required to authorize HSM reprogramming.

7.5. Add or restore an HSM to the Security World

When you have created a Security World, you can add additional HSMs to it. The HSMs may have previously been removed from the same Security World, that is, the Security World can be restored on an HSM by adding the HSM to the Security World again.

(**PCIe and USB HSMs**) These additional HSMs can be on the same host computer as the original HSM or on any other host.

You can also restore an HSM to a Security World to continue using existing keys and Operator Cards:

- After you upgrade the firmware
- If you replace the HSM.

> The additional HSMs can be any nShield HSMs.

To add an HSM to a Security World, you must:

- Have installed the additional HSM hardware.
- **PCIe and USB HSMs:** After installing additional HSM hardware and restarting host machine, you must stop and then restart the hardserver as described in Stopping and restarting the client hardserver. This ensures that the added HSM is recognized and accessible.
- **Network-attached HSMs:** Have a copy of the Security World data on the HSM’s remote file system in the Key Management Data directory.
- **PCIe and USB HSMs:** Have a copy of the Security World data on this host. This is the host data written by new-world when you created the Security World. This data is stored in the `local` directory within the Key Management Data directory.

  > If the Key Management Data directory is not in the default location,

> ensure that the `NFAST_KMDATA` environment variable is set with the correct location for your installation.

- **PCIe and USB HSMs:** Be logged in to the host computer as `root` (**Linux**) or as a user who is permitted to create privileged connections (**Windows**). See Hardserver start-up settings and server_startup.
- Have started the HSM in pre-initialization mode.
- Possess a sufficient number of cards from the ACS and the appropriate passphrases.

Adding or restoring an HSM to a Security World:

- Erases the HSM (**PCIe and USB HSMs**) or the Security World data on the HSM's internal file system (**network-attached HSMs**).
- Reads the required number of cards (*K*) from the ACS so that it can re-create the secret
- Reads the Security World data from the RFS (**network-attached HSMs**) or the computer's hard disk (**PCIe and USB HSMs**).
- Uses the secret from the ACS to decrypt the Security World key
- Stores the Security World key in the HSM's nonvolatile memory
- Configures the HSM for audit logging if the Security World was created with audit logging selected.

After adding an HSM to a Security World:

- You cannot access any keys that were protected by a previous Security World that con tained that HSM.
- You have to sync the module file to the clients by one of the following methods:
  - Copy the files manually to the clients.
  - Run `rfs-sync -update`. See rfs-sync.

> ℹ It is not possible to program an HSM into two separate Security Worlds simultaneously.

8. Remove modules and delete Security Worlds

8.1. Erase a module from a Security World

Erasing a module from a Security World deletes from the module all of the secret information that is used to protect your Security World. Provided that you still have the ACS and the host data, you can restore the secrets by adding the module to the Security World.

Erasing a module removes any data stored in its nonvolatile memory (for example, data for an SEE program or NVRAM-stored keys). To preserve this data, you must back it up before erasing the module. We provide the `nvram-backup` utility to enable data stored in nonvolatile memory to be backed up and restored.

> ! You do not need the ACS to erase a module. However, unless you have a valid ACS and the host data for this Security World, you cannot restore the Security World after you have erased it.

After you have erased a module, it is in the same state as when it left Entrust (that is, it has a random module key and a known K_{NSO}).

(**PCIe and USB HSMs**) In order to erase a module, you must:

- Be logged into your computer as `root` (**Linux**) or as a user who is permitted to create privileged connections (**Windows**).
- Have started the module in the pre-initialization mode.

> ℹ (**nShield 5s**) You may additionally run `hsmadmin factorystate` to restore the HSM to factory state in addition to erasing the Security World. This will remove any other user information on the HSM besides the Security World, such as loaded CodeSafe 5 applications and NVMemPersist files stored in nCore.

If you are physically removing the module from the machine where it is installed, run `hsmadmin enroll` after removing the module. This refreshes the list of nShield HSM modules currently installed on the machine.

8.1.1. Erasing a module from the unit front panel (network-attached HSMs)

To erase a module from a Security World, from the main menu, select **Security World mgmt**

> **Module initialization** > **Erase Security World**.

When you erase a Security World in this way, the Security World files remain on the remote file system. Delete these files if you wish to remove Security World completely. For more information, see Security World Files.

To erase all configuration files, audit logs and (if applicable) SEE machine files from a network-attached HSM, additionally restore it to factory state with the main menu option **System** > **Factory state**. This is recommended if decommissioning the unit. Note that factory state will reset the KNETI key.

8.1.2. Erase a module with new-world

The `new-world` command-line utility can erase any modules that are in the pre-initialization mode.

To erase modules with the `new-world` utility, run the command:

```
new-world [-e|--factory] [-m|--module=<MODULE>]
```

If `new-world` successfully erased a module, it displays a message that it restored the module to factory state.

For more information, see new-world.

8.1.3. Erase a module with initunit

The `initunit` command-line utility erases any modules that are in the pre-initialization state.

To erase modules with the `initunit` utility, run the command:

```
initunit [-m|--module=<MODULE>] [-s|--strong-kml]
```

In the `initunit` command, `--module=<MODULE>` specifies the ID of the module you want to erase. If you do not specify this option, all modules in the pre-initialization state are erased. `--strong-kml` specifies that the module generates an AES (SP800-131A) module signing key, rather than the default key.

> The `--disablepkcs1pad` option will only work on SP800-131A Security Worlds.

8.1.3.1. Output

If `initunit` is successful, for each module that is in the pre-initialization state, it returns a message similar to this:

```
 Initialising Unit
># Setting dummy HKNSO Module Key Info:
 HKNSO
>################### HKM
>###################
```

Otherwise, `initunit` returns an error message.

8.2. Replacing an existing Security World (network-attached HSMs)

When you erase a Security World from the module's front panel, all long-term key material is deleted from the module's memory and all Security World data is removed from the module's internal file system.

This operation does not remove any files from the remote file system or client machines. You should remove the files manually from the `/opt/nfast/kmdata/local` (**Linux**) or `%NFAST_KMDATA%\local` (**Windows**) directory on the remote file system and any client computers to which the Security World was copied.

> Any Operator Cards created in a previous Security World cannot be used in the new Security World. If you are replacing a Security World, you must erase all the Operator Cards created in the previous Security World before you create the new Security World. See Erasing cards and softcards.

8.3. Deleting a Security World

You can remove an existing Security World and replace it with a new one if, for example, you believe that your existing Security World has been compromised. However:

- You are not able to access any keys that you previously used in a deleted Security World
- It is recommended that you reformat any nShield Remote Administration Cards that were used as Operator Cards within this Security World *before* you delete it. For more information about reformatting (or erasing) Operator Cards, see Erasing cards and soft cards.

> Except for nShield Remote Administration Cards, if you do not reformat the smart cards used as Operator Cards before you delete your Security World, you must throw them away because they cannot be used, erased, or reformatted without the old Security World key.

> You can, and should, reuse the smart cards from a deleted Security World's ACS. If you do not reuse or destroy these cards, then an attacker with these smart cards, a copy of your data (for example, a weekly backup) and access to any nShield key management HSM can access your old keys.

To delete an existing Security World:

1. Remove all the HSMs from the Security World.
2. Delete the Security World data files, see Security World Files.

   > There may be copies of the Security World data archive saved on your backup media. If you have not reused or destroyed the old ACS, an attacker in possession of these cards could access your old keys using this backup media.

   > If audit logging was enabled for the Security World then audit logs can still be verified provided that the audit log data is maintained as this contains all the information needed to verify the logs. For further information see *Audit Logging*.

8.3.1. Deleting the Security World using the nShield HSM front panel (network-attached HSMs)

When you erase a Security World using the unit front panel, all long-term key material is deleted from the HSM's memory and all Security World data is removed from the HSM's internal file system.

- You will not be able to access any of the keys that you have previously used
- Before you remove an old Security World, you must reformat any smart cards that were used previously as Operator Cards within this Security World.

> If you do not reformat the smart cards used as Operator Cards before you reinitialize your HSM, you must throw them away because they can not be used, erased, or reformatted without the old Security World key.

You can, and should, reuse the smart cards from the old ACS. If you do not reuse or destroy these cards, then an attacker with these smart cards, a copy of your data (for example, a weekly backup) and access to any nShield key management HSM, can access your old keys.

To erase a Security World using the front panel of the unit, from the main menu select **Security World mgmt** > **Module initialization** > **Erase Security World**.

This operation does not remove any files from the RFS or client machines. You should remove the files manually from the `/opt/nfast/kmdata/local` (**Linux**) or `%NFAST_KMDATA%\local` (**Windows**) directory on the RFS and any client computers to which the Security World was copied.

9. Security World Files

9.1. Location of Security World files

The logic for finding the security world data directory is:

1. If `NFAST_KMLOCAL` is set, use that.
2. Otherwise, if `NFAST_KMDATA` is set, use `${NFAST_KMDATA}/local` on Linux, `%NFAST_KMDATA%\local` on Windows.
3. Otherwise, if `NFAST_HOME` is set, use `${NFAST_HOME}/kmdata/local` on Linux, `%NFAST_HOME%\kmdata\local` on Windows.
4. Otherwise, use `/opt/nfast/kmdata/local` on Linux, `C:\nfast\kmdata\local` on Windows.

> By default, the Key Management Data directory, and sub-directories, inherit permissions from the user that creates them. Installation of the Security World Software must be performed by a user with Administrator rights that allow read and write operations, and the starting and stopping of applications.

Security World operations create or modify Security World files as follows:

Operation	creates/modifies	file(s)
Create a Security World	creates	• `world` • `module_ESN` (see note) • `module_ESN_HKML` (see note)
Load a Security World	creates or modifies	• `module_ESN` (see note) • `module_ESN_HKML` (see note)
Replace an ACS	modifies	`world`
Create an OCS	creates	• `card_HASH` • `cards_HASH_NUMBER`
Create a softcard	creates	`softcard_HASH`
Generate a key	creates	`key_APPNAME__IDENT`
Recover a key	modifies	`key_APPNAME` (for each key that has been recovered)

- <ESN> - Electronic serial number of the module on which the Security World is created.

- <HKML> - Hash of the long term key in the module.
- <IDENT> - Identifier given to the card set or key when it is created.
- <NUMBER> - Number of the card in the card set.
- <APPNAME> - Name of the application by which the key was created. It's a 40-character string that represents the hash of the card set's logical token. It's either user supplied or a hash of the key's logical token, depending on the application that created the key.

> `module_ESN` and `module_ESN_HKML` are created or modified for each module in the Security World.
>
> `module_ESN_HKML` and `module_ESN.old` might exist as backups of `module_ESN`, taken when a Security World was previously loaded.

9.1.1. Make keys and cards available from the front panel (network-attached HSMs)

If you want to make cards or keys which are normally created from the client available from the module's front panel, we recommend that you use client co-operation to automate the copying of files to the module. For information about configuring client co-operation, see Client cooperation.

If you do not use client cooperation, you must manually copy the appropriate card and key files from the client or host on which the card set or key was created to the remote file system of `/opt/nfast/kmdata/local` (**Linux**) or `%NFAST_KMDATA%\local` (**Windows**). These files must then be updated on the module by selecting **Security World mgmt** > **RFS operations** > **Update World files** from the main menu.

To be able to create Operator Cards or keys, the user on the client must have write permission for this directory. All other valid users must have read permission.

9.2. Required files

The following files must be present and up to date in the `/opt/nfast/kmdata/local` (**Linux**) or `%NFAST_KMDATA%\local` (**Windows**) directory, or the directory specified by the `NFAST_KMLOCAL` environment variable, for a client or host to use a Security World:

- `world`
- A `module_ESN` file for each module that this host uses
- A `module_ESN_HKML` file for each module that this host uses

- A `cards_<IDENT>` file for each card set that is to be loaded from this host
- A `card_<IDENT>_NUMBER` file for each card in each card set that is to be loaded from this host
- A `key_<APPNAME>_<IDENT>` file for each key that is to be loaded from this host.

These files are not updated automatically. You must ensure that they are synchronized whenever the Security World is updated on the module.

10. Managing card sets and softcards

When you create a Security World, an Administrator Card Set (ACS) is created at the same time. You use the ACS to:

- Control access to Security World configuration
- Authorize recovery and replacement operations.

The Security World is used to create and manage keys, and the Operator Card Sets (OCSs) and softcards you create with the Security World are used to protect those keys.

A Security World offers three levels of key protection:

Level of protection	Description
Direct protection	Keys that are directly protected by the Security World are usable at any time without further authorization.
Softcard	Keys that are protected by a softcard can only be used by the operator who possesses the relevant passphrases.
OCS	Keys that are protected by an OCS can only be used by the operator who possesses the OCS and any relevant passphrases (if set).

For more information about creating a Security World, see Create a new Security World.

For more information about key management, see Working with keys.

After a Security World has been created, you can use it to create and manage OCSs and softcards (as described in this chapter), as well as to create and manage the keys it protects (see Working with keys).

> **Network-attached HSMs**
>
> To perform the tasks described in this chapter, we recommend using the unit front panel or a client on the same computer that contains the RFS. To perform these tasks on a different client, you must transfer the card data to the RFS.
>
> If you are sharing the Security World across several client computers, you must ensure that the changes are propagated to all your computers. One way to achieve this is to use client cooperation. For more information, see Client cooperation.

If you want to use the Remote Operator feature to configure smart cards for use with a remote unit or module, see Remote Operator.

10.1. View cards and softcards

It is often necessary to obtain information from card sets, usually because for security reasons they are left without any identifying markings.

To view details of all the Operator Cards in a Security World or details of an individual Opera tor Card, you can use:

- nfkminfo
- the front panel (only on network-attached HSMs)

To check which passphrase is associated with a card, you can use:

- cardpp
- the front panel (only on network-attached HSMs)

To list all softcards in a Security World or to show details of an individual softcard, you can use the `ppmk` or `nfkminfo` command-line utilities. To check which passphrase is associated with a softcard, you can use the `ppmk` command-line utility.

10.1.1. View card sets using an nShield network-attached HSM front panel

You can use the unit front panel to view details of all the Operator Cards in a Security World or to view details of an individual Operator Card.

To view a list of all the card sets in the Security World, from the front panel select **Security World mgmt** > **Cardset operations** > **List cardsets**.

To view details of a single card using the unit front panel:

1. Insert the card into the unit.
2. From the main menu, select **Security World mgmt** > **Card operations** > **Card details**.
3. The type of the card (Administrator or Operator) is displayed with the number of the card in the card set.

10.1.2. View card sets using the command line

You can use the `nfkminfo` command-line utility to view details of either all the Operator Cards in a Security World or of an individual Operator Card.

To list the OCSs in the current Security World from the command line, open a command window, and give the command:

```
nfkminfo --cardset-list
```

In this command, `--cardset-list` specifies that you want to list the operator card sets in the current Security World.

`nfkminfo` displays output information similar to the following:

```
Cardset summary - 1 cardsets:              (in timeout, P=persistent, N=not)
 Operator logical token hash                    k/n timeout name
hash                                           1/1 none-N name
```

To list information for a specific card, use the command:

```
nfkminfo <TOKENHASH>
```

In this command, `<TOKENHASH>` is the `Operator logical token hash` of the card (as listed when the command `nfkminfo --cardset-list` is run).

This command displays output information similar to the following:

```
name            "name"
k-out-of-n      1/1
flags           NotPersistent
timeout         none
card names      ""
hkltu           794ada39038fa8c4e9ea46a24136bbb2b8b337f2
```

Not all software can give names to individual cards.

10.1.3. View softcards

10.1.4. View softcards with nfkminfo

To list the softcards in the current Security World using the `nfkminfo` command-line utility, give the command:

```
nfkminfo --softcard-list
```

In this command `--softcard-list` specifies that you want to list the softcards in the current Security World.

To show information for a specific softcard using the `nfkminfo` command-line utility, give the command:

```
nfkminfo --softcard-list <IDENT>
```

In this command <IDENT> is the softcard's logical token hash (as given by running the command nfkminfo --softcard-list). This command displays output information similar to the following:

```
SoftCard
 name       "mysoftcard"
 hkltu      7fb95888ea2850d4e3ffcc8f0c22100937344308
Keys protected by softcard 7fb95888ea2850d4e3ffcc8f0c22100937344308:
 AppName simple              Ident mykey
 AppName simple              Ident myotherkey
```

10.1.4.1. View softcards with ppmk

To list the softcards in the current Security World using the ppmk command-line utility, use the command:

```
ppmk --list
```

In this command --list specifies that you want to list the softcards in the current Security World.

In order to view the details of a particular softcard using the ppmk command-line utility, give the command:

```
ppmk --info <NAME>|<IDENT>
```

In this command, you can identify the softcard whose details you want to view either by its name (<NAME>) or by its logical token hash (as given by running the command nfkminfo --softcard-list).

10.2. Erase cards and softcards

Erasing a card or softcard removes all the secret information from the card or softcard and deletes information about the card or softcard from the host.

> In the case of an OCS that uses nShield Remote Administration Cards, it is possible to reformat the cards at any time using slotinfo --ignore-auth. In the case of an OCS that uses standard nShield cards, it is only possible to erase or format the cards within the Security World in which they were created.

You can erase Operator Cards using

- the unit front panel (only network-attached HSMs)
- createocs

You can also use these methods to erase Administrator Cards other than those in the current Security World's ACS (for example, you could use these methods to erase the remaining Administrator Cards from an incomplete set that has been replaced or Administrator Cards from another Security World).

> None of these tools erases cards from the current Security World's ACS.

If you erase an Operator Card that is the only card in an OCS, information about the card set is deleted. However, if you erase one card from an OCS of multiple cards, you must remove the card information from the `opt/nfast/kmdata/local/` (**Linux**) or `%NFAST_KMDATA\local%` (**Windows**) directory after you have erased the last card.

10.2.1. FIPS 140 Level 3-compliant Security Worlds

When you attempt to erase cards for a Security World that complies with FIPS 140 Level 3, you are prompted to insert an Administrator Card or Operator Card from an existing set. You may need to specify to the application the slot you are going to use to insert the card. You need to insert the card only once in a session. You can therefore use one of the cards that you are about to erase.

10.2.2. Erase card sets using an nShield network-attached HSM front panel

To erase a card set using the front panel, follow this procedure:

1. From the main menu select: **Security World mgmt** > **Card operations** > **Erase card**
2. Insert the card set that you want to erase. The card is read.
3. You are asked to confirm that you want to erase this card from the card set.
4. To confirm, press the right-hand navigation button.
5. You are asked once again if you want to erase this card.
6. To confirm, press the right-hand navigation button.

10.2.3. Erase cards using the command line

To erase a card from the command line, run the command:

```
createocs -m|--module=<MODULE> -e|--erase
```

If you have more than one card reader and there is more than one card available, `createocs` prompts you to confirm which card you wish to erase. Use **[Ctrl][X]** to switch between cards.

If you have created a FIPS 140 Level 3 compliant Security World, you must provide authorization in order to erase or create Operator Cards. You can obtain this authorization from any card in the ACS or from any Operator Card in the current Security World, including cards that are to be erased. After you insert a card containing this authorization, `createocs` prompts you to insert the card to be erased.

As an alternative, you can reformat using `slotinfo --format`.

10.2.4. Erase softcards

Erasing a softcard deletes all information about the softcard from the host. You can erase softcards with the `ppmk` command-line utility.

10.2.4.1. Erasing softcards with ppmk

To erase a softcard with `ppmk`, open a command window, and give the command:

```
ppmk --delete <NAME>|<IDENT>
```

In this command, you can identify the softcard to be erased either by its name (`NAME`) or by its logical token hash as listed by `nfkminfo` (`<IDENT>`).

If you are working within a FIPS 140 Level 3 compliant Security World, you must provide authorization to erase softcards; `ppmk` prompts you to insert a card that contains this authorization. Insert any card from the ACS or any Operator Card from the current Security World.

If you insert an Administrator Card from another Security World or an Operator Card that you have just created, `ppmk` displays an error message and prompts you to insert a card with valid authorization. When `ppmk` has obtained the authorization from a valid card or if no authorization is required, it completes the process of erasing the softcard.

10.3. Passphrases

10.3.1. Verify the passphrase of a card or softcard

10.3.1.1. Verify the passphrase of a card using the nShield HSM front panel (only on network-attached HSMs)

To verify the passphrase associated with a card using the unit front panel:

1. Insert the card into the unit.
2. From the main menu, select **Security World mgmt** > **Card operations** > **Check PIN**.

   The type of the card (Administrator or Operator) is displayed with the number of the card in the card set.

3. If this is the card that you want to check, press the right-hand navigation to confirm.
4. Enter the passphrase.

   If the passphrase that you entered is correct, a confirmation message is shown. Otherwise, an error is reported.

10.3.2. Verify the passphrase of a softcard with ppmk

In order to verify the passphrase of a particular softcard, open a command window, and give the command:

```
ppmk --check <NAME>|<IDENT>
```

In this command, you can identify the softcard whose passphrase you want to verify either by its name (`<NAME>`) or by its logical token hash (as given by running the command `nfkminfo --softcard-list`).

`ppmk` prompts you to enter the passphrase and then tells you whether the passphrase you entered is correct for the specified softcard.

10.3.3. Change card and softcard passphrase

Each softcard or card of a card set can have its own individual passphrase: you can even have a card set in which some cards have a passphrase and others do not, and you can have distinct softcards that nevertheless use the same passphrase. A passphrase can be of any length and can contain any characters that you can type.

Normally, in order to change the passphrase of a card or softcard, you need the card or soft card and the existing passphrase. Known card passphrase can be changed using the front panel (only on network-attached HSMs), or the cardpp command-line utility; softcard passphrase can be changed with the ppmk command-line utility. You can also add a passphrase to a card or softcard that currently does not have one or remove a passphrase from a card that does currently have one.

If you generated your Security World with the passphrase replacement option, you can also replace the passphrase of a card or softcard even if you do not know the existing passphrase. Such a passphrase replacement operation requires authorization from the ACS.

10.3.3.1. Change known passphrase

To change a card passphrase, you need the card and the old passphrase.

Each card in a set can have its own individual passphrase. You can even have a set in which some cards have a passphrase and others do not.

> Prior to Security World Software v11.72, we set no absolute limit on the length of a passphrase. However, some applications may not accept a passphrase longer than 255 characters. Likewise, the Security World does not impose restrictions on which characters you can use, although some applications may not accept certain characters. Entrust recommends that your password only contains 7-bit ASCII characters:
>
> A-Z, a-z, 0-9, ! @ # $ % ^ & * - _ + = [ ] { } | \ : ' , . ? / ` ~ " < > ( ) ;
>
> See Maximum passphrase length for more about passphrase length when using Security World Software v11.72.

10.3.3.1.1. Change known passphrase from than nShield network-attached HSM front panel

To change the passphrase of a card using the unit front panel:

1. Insert the card.
2. From the main menu, select **Security World mgmt** > **Card operations** > **Change PIN**.
3. Select the card whose passphrase you want to change.
4. Enter the old passphrase, and then enter it again to confirm it.
5. Enter the new passphrase. If you do not want this card to have a passphrase, select **NO** at the prompt.

10.3.3.1.2. Change known passphrase with cardpp

Each card in a card set can have its own individual passphrase. You can even have a set in which some cards have a passphrase and others do not. A passphrase can be of any length and can contain any characters that you can type.

> With Security World Software v11.72 and later, passphrases are limited to a maximum length of 254 characters, when using `cardpp`. See Maximum passphrase length.

To change a known card's passphrase with `cardpp`:

1. Run:

   ```
   cardpp --change [-m|--module=<MODULE>]
   ```

2. If prompted, insert the card whose passphrase you want to change. If there is a card already in the slot, you are not prompted.
3. If prompted, enter the existing passphrase for the card. If the card has no current passphrase you are not prompted.
   If you enter the passphrase correctly, `cardpp` prompts you to enter the new passphrase.
4. Enter a new passphrase, and then enter it again to confirm it.

10.3.3.1.3. Change known softcard passphrase with ppmk

> With Security World Software v11.72 and later, passphrases are limited to a maximum length of 254 characters, when using `ppmk`. See Maximum passphrase length for more information.

To change a known softcard's passphrase when you know the passphrase, follow these steps:

1. Give the following command:

   ```
   ppmk --change <NAME>|<IDENT>
   ```

   In this command, you can identify the softcard whose passphrase you want to change either by its name (`<NAME>`) or by its logical token hash as listed by `nfkminfo` (`<IDENT>`).

   `ppmk` prompts you to enter the old passphrase.

2. Type the old passphrase, and press **Enter**. If you enter the old passphrase correctly,

   `ppmk` prompts you to enter the new passphrase.

3. Type the old passphrase, and press **Enter**. Type the new passphrase again, and press **Enter** to confirm it.

   After you have confirmed the new passphrase, `ppmk` then changes the softcard's passphrase.

10.3.3.2. Change unknown or lost passphrase

10.3.3.2.1. Change unknown card passphrase with cardpp

If you generated your Security World with the passphrase replacement option, you can change the passphrase of a card even if you do not know its existing passphrase. Such a passphrase replacement operation requires authorization from the ACS.

To change an unknown card passphrase with `cardpp`:

1. Run:

   ```
   cardpp --recover [--module=<MODULE>]
   ```

2. As prompted, insert the appropriate number of cards from the ACS required to authorize passphrase replacement.
3. When prompted, insert the Operator Card whose passphrase you want to replace.
4. When prompted, type the new passphrase, and then press **Enter**.
5. When prompted, type the new passphrase again to confirm it, and then press **Enter**.

   `cardpp` sets the new passphrase, and then prompts you for another Operator Card.

6. Repeat the process in the previous step to change the passphrase on further cards, or press **Q** to quit.

10.3.3.2.2. Replace unknown passphrase with ppmk

If you generated your Security World with the passphrase replacement option, you can change the passphrase of a softcard even if you do not know its existing passphrase. Such a passphrase replacement operation requires authorization from the ACS.

To change an unknown softcard passphrase with the `ppmk` command-line utility:

1. Run a command of the form:

```
ppmk --recover <NAME>|<IDENT>
```

In this command, you can identify the softcard by its `<NAME>` or by its `<IDENT>` (its logical token hash as shown in output from the `nfkminfo` command-line utility).

2. As prompted, insert the appropriate number of cards from the ACS required to authorize passphrase replacement.
3. When prompted, type the new passphrase, and then press **Enter**.
4. When prompted, type the new passphrase again to confirm it, and then press **Enter**.

   If the passphrase does not match, `ppmk` prompts you to input and confirm the passphrase again.

After you successfully confirm the new passphrase, `ppmk` finishes configuring the softcard to use the new passphrase.

> Only insert Administrator Cards into a hardware security module that is connected to a trusted server.

10.4. Operator Card Sets (OCS)

> To delete a card set, see Erase cards and softcards

10.4.1. Create Operator Card Sets (OCSs)

You can use an Operator Card Set (OCS) to control access to application keys. OCSs are optional, but if you require one, create it before you start to use the hardware security module with applications. You must create an OCS before you create the keys that it is to protect.

You can create OCSs that have:

- Names for individual cards, as well as a name for the whole card set
- Specific *K/N* policies
- Optional passphrases for any card within a given set
- Formal FIPS 140 Level 3 compliance.

> Some third-party applications impose restrictions on the OCS smart card quorums (*K/N*) or the use of smart card passphrases. For more information, see the appropriate integration guide for the application. Integration guides for third-party applications are available from

> https://nshieldsupport.entrust.com/.

OCSs belong to the Security World in which they are created. When you create an OCS, the smart cards in that set can only be read by hardware security modules belonging to the same Security World.

You can use the following tools to create an OCS:

- The createocs command-line utility.
- (**Network-attached HSMs**) The unit front panel.
- (**Windows**) The nShield CSP wizard, as described in Create an Operator Card Set with the CSP or CNG wizard (Windows).
- (**Windows**) The nShield CNG wizard, as described in Microsoft Cryptography API: Next Generation (CNG).

10.4.2. Persistent Operator Card Sets

If you create a standard (non-persistent) OCS, the keys it protects can only be used while the last required card of the quorum remains loaded in the local slot of the HSM, or one of its Dynamic Slots. The keys protected by this card are removed from the memory of the device as soon as the card is removed from the smart card reader. If you want to be able to use the keys after you have removed the last card, you must make that OCS persistent.

Keys protected by a persistent card set can be used for as long as the application that loaded the OCS remains connected to the hardware security module (unless that application removes the keys).

For more information about persistent OCSs, see Using persistent Operator Card Sets.

> **Network-attached HSMs**
>
> An OCS to be used to authorize login on a unit must be persistent and not loadable remotely. It is recommended that such an OCS is not used to protect sensitive keys.

10.4.3. Time-outs

OCSs can be created with a time-out, so that they can only be used for limited time after the OCS is loaded. An OCS is loaded by most applications at start up or when the user supplies the final required passphrase. After an OCS has timed out, it is not loadable by another application unless it is removed and reinserted. Time-outs operate independently of OCS persistence.

10.4.4. FIPS 140 Level 3-compliant Security Worlds

When you attempt to create an OCS for a Security World that complies with FIPS 140 Level 3, you are prompted to insert an Administrator Card or Operator Card from an existing set. You may need to specify to the application the slot you are going to use to insert the card. You need to insert the card only once in a session.

10.4.5. Create an Operator Card Set using an nShield network-attached HSM front panel

To create an OCS, follow these steps:

1. From the main menu, select **Security World mgmt** > **Cardset operations** > **Create OCS**.

   You are prompted to enter the name of the OCS.

2. Enter a name and press right-hand navigation button.
3. Enter the quorum for the OCS, using the touch wheel to move from one field to the other. The quorum consists of:
   - The maximum number of cards from the OCS required by default for an operation. This number must be less than or equal to the total number of cards in the set.
   - The total number of cards to be used in the OCS. This must be a value in the range 1 – 64.
4. Press the right-hand navigation button to move to the next screen.
5. If you wish to specify a time out for the card set, enter the time out in seconds.
6. Choose whether to create a persistent card set. You can select:
   - **Not persistent** (which is the default)
   - **Persistent**
   - **Remoteable/Persistent**
7. Choose whether to name individual cards and enable passphrase replacement by answering **Yes** or **No** to each question and then pressing the right-hand navigation button.
8. Insert a smart card to be formatted for the OCS.

   If the card is not blank, choose whether to overwrite it or to use a different card. (If the card is an Operator Card from another Security World, you cannot overwrite it and are prompted to enter a different card.)

9. If you have chosen to name individual cards, you are prompted to enter the name for

the card.

10. You are asked whether you wish to specify a passphrase for the card. If you choose **Yes**, you are prompted to enter the passphrase twice.

    While the Operator Card is being created, the screen displays the message **Processing**.

    If there are further cards from this OCS to be processed, the screen changes to **Waiting**. Remove the card, and repeat steps 8 through 10 for each of the remaining cards.

When all the cards in the set have been processed, you are told that the card set has been created successfully.

10.4.6. Creating an Operator Card Set using the command line

To create an OCS from the command line:

1. Run createocs.
2. Insert the smart card to use.

   If you insert an Administrator Card from another Security World or an Operator Card that you have just created, `createocs` displays the following message:

   ```
   Module x slot n: unknown card +
   Module x slot n: Overwrite card ? (press Return)
   ```

   where `x` is the hardware security module number and `n` is the slot number. If you insert an Operator Card from another Security World, `createocs` displays the following message:

   ```
   Module x slot n: inappropriate Operator Card (TokenAuthFailed).
   ```

   When you insert a valid card, `createocs` prompts you to type a passphrase.

   > ℹ The nShield PKCS #11 library requires Operator Cards with passphrases.

   > ℹ Some applications do not have mechanisms for entering passphrases. Do not give passphrases to Operator Cards that are to be used with these applications.

3. Type a passphrase and press **Enter**. Alternatively, press **Enter** if you do not want this card to have a passphrase.

A passphrase can be of any length and can contain any character that you can type.

If you entered a passphrase, `createocs` prompts you to confirm it.

4. Type the passphrase again and press **Enter**.

   If the passphrases do not match, `createocs` prompts you to input and confirm the passphrase again.

5. When the new card has been created, if you are creating a card set with more than one card in it, `createocs` prompts you to insert another card.
6. For each additional card in the OCS, follow the instructions from step 2 through 4.

10.4.7. Create an Operator Card Set with the CSP or CNG wizard (Windows)

You can use the nShield CSP or CNG wizard to create a *K/N* OCS that is suitable for use with the nShield Cryptographic Service Provider (CSP) or Cryptography API: Next Generation (CNG), as appropriate. You can only create an OCS using the CSP or CNG wizard if you already have a Security World and have an ACS available for that Security World.

To create an OCS using the CSP or CNG wizard, follow these steps:

1. Ensure that you have created the Security World and that at least one HSM is in the operational state.
2. Run the wizard by double-clicking its shortcut in the Start menu: **Start > Entrust nShield Security World**.
3. The wizard displays the welcome screen.
4. Click the **Next** button. The wizard allows you to configure HSM Pool mode for CAPI/CNG.

   > Do not enable HSM Pool mode when creating an Operator Card Set because HSM Pool mode only supports module-protected keys.

5. Click the **Next** button.

   The wizard determines what actions to take based on the state of the Security World and of the HSMs that are attached to your computer:

   - If the wizard cannot find the Security World, it prompts you to create a new Security World or to install cryptographic acceleration only.

     In such a case, you should:

        - Cancel the operation
        - Check that the environment variable `NFAST_KMDATA` is set correctly
        - Copy the `local` sub-directory from the Key Management Data directory of another computer in the same Security World or from a backup tape of this computer to the Key Management Data directory of this computer.
        - run the wizard again.
    - If there is an existing Security World, the wizard gives you the option of using the existing Security World, creating a new Security World or installing cryptographic acceleration only.
        - In order to use the existing Security World, ensure that the **Use the existing security world** option is selected, and click the **Next** button.
        - If there are any HSMs in the pre-initialization state, the wizard adds them to the Security World; see Adding or restoring an HSM to the Security World.

6. When at least one hardware security module is in the operational state, the wizard prompts you to select a method to protect private keys generated by the CSPs.
7. Ensure that the Operator Card Set option is enabled. If you are running the CNG wizard (not the CSP wizard) click the **Next** button. Then select the **Create a new Operator Card Set** option.

   If you want the OCS to be persistent, select the **Persistent** option. Persistence is described in Persistent Operator Card Sets.

8. Click the **Next** button, and if you have a FIPS world, the wizard prompts you to insert a card created with the current Security World.

   > This shows that your Security World is compliant with the roles and services of the FIPS 140 Level 3 standard. It is included for those customers who have a regulatory requirement for compliance.

   Under the constraints of level 3 of the FIPS 140 standard, Operator Cards cannot be created without authorization. To obtain authorization, insert any card from the ACS or any Operator Card belonging to the current Security World.

   The wizard does not enable the next world, the wizard warns you and prompts you for another card.

9. Click the **Next** button.

   The wizard prompts you for a smart card to use as the first card in the OCS.

10. Insert a blank smart card to be used as the Operator Card, and click the **Next** button.

> Do not use a card from the ACS or an existing Operator Card.
> If you insert a card that is not blank, the wizard asks you if you want to erase it.

11. When you have inserted an appropriate card, the wizard prompts you for the name of the card and, if required, a passphrase.

    If you want to protect this card with a passphrase, turn on the **Card will require a passphrase** option, and enter the passphrase. You must enter the passphrase in both fields to ensure that you have typed it correctly.

    > Operator Cards with passphrases are required by the nShield PKCS #11 library.

12. If you have not yet written all the smart cards in the OCS, the wizard prompts you for another card. Repeat the appropriate preceding steps of the OCS creation process for all smart cards in the set.
13. When the wizard has finished creating the OCS, it displays a screen telling you this. If you want to create another OCS, click the **Back** button on this screen.

    When you have created all the OCSs that you require, click the **Next** button to install the CAPI CSP or register the CNG CSP. For more information, see Microsoft CryptoAPI Guide for nShield Security World v13.6.15 or Microsoft CNG Guide for nShield Security World v13.6.15.

[

10.5. Replace OCS and softcards

10.5.1. Replace Operator Card Sets

> Replacing an OCS requires authorization from the ACS of the Security World to which it belongs. You cannot replace an OCS unless you have the required number of cards from the appropriate ACS.

If you have lost a card from a card set, or you want to migrate from standard nShield cards to nShield Remote Administration Cards, you should use one of the following:

- The `rocs` utility
- The front panel of the nShield HSM

> You cannot mix standard nShield cards with nShield Remote Administra

> tion Cards in the same set.

We recommend that after you have replaced an OCS, you then erase the remaining cards in the old card set and remove the old card set from the Security World. For more information, see Erase cards and softcards

Deleting the information about an OCS from the client or host does not remove the data for keys protected by that card set.

To prevent you from losing access to your keys if the smart card you are using as the Operator Card is lost or damaged, `rocs` command-line utility provides an interactive method or a command-line only method that can recover the keys protected by the lost Operator Card to another OCS or token.

Replacing one OCS with another OCS also transfers the keys protected by the first OCS to the protection of the new OCS.

When you replace an OCS or softcard and recover its keys to a different OCS or softcard, the key material is not changed by the process. The process deletes the original Security World or host data (that is, the encrypted version of the key or keys and the smart card or softcard data file) and replaces this data with host data protected by the new OCS or softcard.

To replace an OCS or softcard, you must:

- Have enabled OCS and softcard replacement when you created the Security World

  > If you did not enable OCS and softcard replacement, or if you created the Security World with an early version of the `pkcs-init` command-line utility that did not support OCS and softcard replacement, you cannot recover keys from lost or damaged smart cards or softcards.

- Have created the original OCS using
  - the front panel of a network-attached HSM
  - createocs
  - `createocs-simple`
  - the nShield PKCS #11 library version 1.6 or later

  > If you initialized the token using `ckinittoken` from the nShield PKCS #11 library version 1.5 or earlier, you must contact Support to arrange for them to convert the token to the new format while you still possess a valid card.

- Have a sufficient number of cards from the ACS to authorize recovery and replacement

  > All recovery and replacement operations require authorization from the ACS. If any of the smart cards in the ACS are lost or damaged, immediately replace the entire ACS.

- Have initialized a second OCS using
  - the front panel of a network-attached HSM
  - createocs
  - `createocs-simple`
  - the nShield PKCS #11 library version 1.6 or later

  > The new OCS need not have the same *K/N* policy as the old set.

If you are sharing the Security World across several client computers (for network-attached HSMs) or host computers (for PCIe HSMs), you must ensure that the changes to the host data are propagated to all your computers. One way to achieve this is to use client coopera tion. For more information, see Client cooperation.

10.5.2. Replace OCS from a network-attached HSM front panel

To replace an OCS from the unit front panel, follow these steps:

1. From the main menu, select **Security World mgmt** > **Admin operations** > **Recover keys**.
2. Select **all** to recover all keys in the Security World, or select the application for which you want to recover the keys.
3. If you selected an application, select the keys that you want to recover.
4. Insert the required number of Administrator Cards to recover keys, and enter their passphrases if required.
5. Insert the required number of Operator Cards, and enter their passphrases if required.

   When you have inserted the required number of cards, details of the recovered key are displayed.

6. Check the key details are correct and then scroll down and select **Recover key**.

   You can also select **More info** to see more information about the keys.

A message is displayed when the keys are recovered.

10.5.3. Replace OCS or softcards with rocs

You can use the `rocs` command-line utility interactively, or you can supply all the parameters using the command line.

10.5.3.1. Using rocs interactively

Refer to rocs for more details about each of the available commands.

To replace an OCS or recover keys to a softcard:

To exit without completing the replacement or recovery process, press **Q** and then **Enter**. The `rocs` utility returns you to the `rocs >` prompt with out processing any keys.

1. Launch the `rocs` interactive mode prompt:

```
rocs
```

2. Enter the following commands, in order:

   a. `module <number>`
   The number of the HSM you want to use.

   b. `list cardsets`
   Note the number (`No.`) of the OCS or softcard to which you want to transfer the keys ('target').

   c. `target <cardset-spec>`
   `<cardset-spec>` is the number of the target OCS or softcard you obtained in the previous step.

   Keys protected by an OCS can only be recovered to another OCS, and not to a softcard. Likewise, softcard-protected keys can only be recovered to another softcard, and not to an OCS.

   d. `list keys`
   Note the number (`No.`) of the keys you want to recover.

   e. `mark <key-spec> [<key-spec> […]]`
   `<key-spec>` is the number (`No.`) of the key you want to recover. To recover multiple keys, leave a space between each `<key-spec>`.

   Only mark keys from a different OCS or softcard to the one you selected as the tar get.

If you selected any keys by mistake, deselect them with `unmark ..<key-spec>`.

f. `recover`
Transfers the marked keys to the target OCS or softcard.

The operation is not permanent at this stage.

3. When prompted, insert a card from the ACS and enter the passphrase.
Repeat this step until you have loaded the required number of cards.
If you do not have the required number of cards from the ACS, exit the process.

❗ Only insert Administrator Cards into a hardware security module that is connected to a trusted server.

4. If you are recovering keys to:

an **OCS**:

+

a. `rocs` prompts you to insert a card from the first OCS that you have selected as the target. OCSs are processed in ascending numerical order as listed by the `list cardsets` command.
b. Insert a card from this OCS.
c. `rocs` prompts you for the passphrase for this card. This action is repeated until you have loaded the required number of cards from the OCS.

a **Softcard**:

+ If you are recovering keys to a softcard, `rocs` prompts you for the passphrase for the softcard that you have selected as the target.

When you have loaded the target softcard or the required number of cards from the tar get OCS, `rocs` transfers the selected keys to the target OCS or softcard.

If you have selected other target OCSs or softcards, `rocs` prompts for a card from the next OCS.

ℹ Repeat this step for each selected target.

5. Enter `save [<key-spec> […]]`.
Write the key blobs to disk. If you specify one or more `<key-spec>` values, only those keys will be saved. If you do not specify a `<key-spec>`, all keys will be saved.

ℹ If you have transferred a key by mistake, you can restore it to its original protection with `revert <key-spec> [<key-spec> […]]`.

10.5.3.1.1. Using rocs from the command line

Refer to rocs for more details about each of the available options.

You can select all the options for rocs using the command line by running a command of the form:

```
rocs -m|--module=<MODULE> [-t|--target=<CARDSET-SPEC>] [-k|--keys=<KEYS-SPEC>] [-c|--cardset=<CARDSET-SPEC>] [-
i|--interactive]
```

Set the values as follows:

- <MODULE>: The HSM to use.
- (target) <CARDSET-SPEC>: The OCS or softcard to use to protect the keys.
- <KEYS-SPEC>: The keys to recover.
- (cardset) <CARDSET-SPEC>: This selects all keys that are protected by the named OCS or softcard.

-i\|--interactive starts rocs in interactive mode, even if keys have been selected.

You must specify the target before you specify keys.

You can use multiple --keys=<KEYS-SPEC> and --cardset=<CARDSET-SPEC> options, if necessary.

You can specify multiple targets on one command line by including separate --keys=<KEYS-SPEC> or --cardset=<CARDSET-SPEC> options for each target. If a key is defined by --keys=<KEYS-SPEC> or --cardset=<CARDSET-SPEC> options for more than one target, it is transferred to the last target for which it is defined.

If you have selected a hardware security module, a target OCS or softcard, and keys to recover but have not specified the --interactive option, rocs automatically recovers the keys. rocs prompts you for the ACS and OCS or softcard. For more information, see Using rocs interactively.

If you use rocs from the command line, all keys are recovered and saved automatically. You cannot revert the keys unless you still have cards from the original OCS.

If you do not specify the target and keys to recover, or if you specify the --interactive option, rocs starts in interactive mode with the selections you have made. You can then use further rocs commands to modify your selection before using the recover and save commands to transfer the keys.

10.6. Softcards

> To delete a softcard, see Erase cards and softcards.

10.6.1. Create softcards

You must create a softcard before you create the keys that it is to protect.

A softcard is a file containing a logical token that cannot be loaded without a passphrase; its logical token must be loaded in order to authorize the loading of any key that is protected by the softcard. Softcard files are stored in the Key Management Data directory and have names of the form `softcard_<hash>` (where `<hash>` is the hash of the logical token share). Softcards belong to the Security World in which they are created.

A softcard's passphrase is set when you generate it, and you can use a single softcard to protect multiple keys. Softcards are persistent; after a softcard is loaded, it remains valid for loading the keys it protects until its `KeyID` is destroyed.

> It is possible to generate multiple softcards with the same name or passphrase. For this reason, the hash of each softcard is made unique (unrelated to the hash of its passphrase).

> Softcards are not supported for use with the nCipherKM JCA/JCE CSP in Security Worlds that are compliant with FIPS 140 Level 3.

> To use softcards with PKCS #11, you must have `CKNFAST_LOADSHARING` set to a nonzero value. When using pre-loaded softcards or other objects, the PKCS #11 library automatically sets `CKNFAST_LOADSHARING=1` (load-sharing mode on) unless it has been explicitly set to `0` (load-sharing mode off).

> As with OCSs, if debugging is enabled, a softcard's passphrase hash is available in the debug output (as a parameter to a `ReadShare` command).

You can create softcards from the command-line, see Create a softcard with ppmk.

10.6.1.1. Create a softcard with ppmk

To create a new softcard using the `ppmk` command-line utility:

1. Decide whether you want the new softcard's passphrase to be replaceable or non-replaceable. To create a softcard with a replaceable passphrase, run the command:

```
ppmk --new --recoverable <NAME>
```

To create a softcard with a non-replaceable passphrase, run the command:

```
ppmk --new --non-recoverable <NAME>
```

In these commands, `<NAME>` specifies the name of the new softcard to be created.

2. **PCIe HSMs:** If you are working within a FIPS 140 Level 3 compliant Security World, you must provide authorization to create new softcards. The `ppmk` utility prompts you to insert a card that contains this authorization. Insert any card from the ACS. If you insert an Administrator Card from another Security World, `ppmk` displays an error message and prompts you to insert a card with valid authorization.

   When `ppmk` has obtained the authorization from a valid card, or if no authorization is required, it prompts you to type a passphrase.

3. When prompted, type a passphrase for the new softcard, and press **Enter**.

   A passphrase can be of any length and contain any characters that you can type except for tabs or carriage returns (because these keys are used to move between data fields).

4. When prompted, type the passphrase again to confirm it, and press **Enter**.

   If the passphrases do not match, `ppmk` prompts you to input and confirm the passphrase again.

After you have confirmed the passphrase, `ppmk` completes creation of the new softcard.

10.6.1.2. Create a softcard with the CNG wizard (Windows)

You can use the nShield CNG wizard to create a Softcard that is suitable for use with the nShield Cryptography API: Next Generation (CNG), as appropriate. You can only create an Softcard using the CNG wizard if you already have a Security World and have an ACS available for that Security World.

To create an Softcard using the CNG wizard, follow these steps:

1. Ensure that you have created the Security World and that at least one HSM is in the operational state.
2. Run the wizard by double-clicking its shortcut in the Windows Start menu: **Start > Entrust nShield Security World**.

3. The wizard displays the welcome screen.
4. Click the **Next** button. The wizard allows you to configure HSM Pool mode for CAPI/CNG.

   > Do not enable HSM Pool mode when creating a Softcard because HSM Pool mode only supports module-protected keys.

5. Click the **Next** button.

   The wizard determines what actions to take based on the state of the Security World and of the HSMs that are attached to your computer:

   - If the wizard cannot find the Security World, it prompts you to create a new Security World or to install cryptographic acceleration only.

     In such a case, you should:

     - Cancel the operation
     - Check that the environment variable `NFAST_KMDATA` is set correctly
     - Copy the `local` sub-directory from the Key Management Data directory of another computer in the same Security World or from a backup tape of this computer to the Key Management Data directory of this computer.
     - Run the wizard again.
   - If there is an existing Security World, the wizard gives you the option of using the existing Security World, creating a new Security World or installing cryptographic acceleration only.
     - In order to use the existing Security World, ensure that the **Use the existing security world** option is selected, and click the **Next** button.
     - If there are any hardware security modules in the pre-initialization state, the wizard adds them to the Security World; see Adding or restoring an HSM to the Security World.
6. When at least one hardware security module is in the operational state, the wizard prompts you to select a method to protect private keys generated by the CSPs.
7. Ensure that the Softcard option is enabled. Click the **Next** button. Then select the **Create a new Softcard** option, and enter the name and passphrase of the Softcard in the boxes provided.
8. Click the **Next** button, and if you have a FIPS world, the wizard prompts you to insert a card created with the current Security World.

   > This shows that your Security World is compliant with the roles and services of the FIPS 140 Level 3 standard. It is included for those

> customers who have a regulatory requirement for compliance.

Under the constraints of level 3 of the FIPS 140 standard, Softcards cannot be created without authorization. To obtain authorization, insert any card from the ACS or any OCS belonging to the current Security World.

9. On the Software Installation screen when you are informed **You now have a valid security world and key protection mechanism**, click the Back button if you want to create another Softcard, or if you want to change the default protection for new CNG keys to a different protection option. When you have created all the Softcards that you require, click the **Next** button on this screen to register the CNG providers. For more information, see Microsoft CNG Guide for nShield Security World v13.6.15.

10.7. Replace the ACS

Replacing the ACS requires a quorum of cards from the current ACS (*K*/*N*) to perform the following sequence of tasks:

1. loading the secret information that is to be used to protect the archived copy of the Security World key.
2. creating a new secret that is to be shared between a new set of cards.
3. creating a new archive that is to be protected by this secret.

If you discover that one of the cards in the current ACS has been damaged or lost, or you want to migrate from standard nShield cards to nShield Remote Administration Cards, you should use one of the following to create a new set:

- The `racs` utility.

  > When using the `racs` utility, you cannot redefine the quantities in a K of N relationship for an ACS. The K of N relationship defined in the original ACS persists in the new ACS.

- The front panel of an nShield network-attached HSM.

> If further cards are damaged, you may not be able to re-create your Security World.

> You cannot mix nShield cards with nShield Remote Administration Cards in the same set.

> Replacing the ACS modifies the `world` file. In order to use the new ACS

on other machines in the Security World, you must copy the updated `world` file to all the machines in the Security World after replacing the ACS. Failure to do so could result in loss of administrative access to the Security World.

We recommend that you erase your old Administrator Cards as soon as you have created the new ACS. An attacker with the old ACS and a copy of the old host data could still re-create all your keys. With a copy of a current backup, they could even access keys that were created after you replaced the ACS.

Before you start to replace an ACS, you must ensure that you have enough blank cards to create a complete new ACS. If you start the procedure without enough cards, you will have to cancel the procedure part way through.

10.7.1. Replace an ACS using an nShield network-attached HSM front panel

To replace an ACS:

1. From the main menu, select **Security World mgmt** > **Admin operations** > **Replace ACS**.
2. Insert one of the remaining cards from the card set that you want to replace and press the right-hand navigation button.

   Continue to insert cards until you have inserted the number of cards required to authorize the process.

3. When prompted, insert a card for the replacement card set and press the right-hand navigation button.
4. If required, specify a passphrase for the card.
5. Insert cards until the card set is complete. A message confirms that the card set has been created.
6. At this point the modified **world** file has been pushed to the RFS, so make a backup of the modified **world** file on the RFS, preferably in a way that distinguishes it from previous backups.
7. Copy the **world** file to any other HSMs in the same Security World, either using the **Security World mgmt** > **RFS operations** > **Update World files** option on the HSM concerned or using the `nethsmadmin` utility, see Using nethsmadmin to copy a Security World to a nShield HSM and check the current version.

8. If client cooperation is not enabled, copy the modified `world` file onto any client machines where it is needed.
9. Check that the new Administrator Cards are usable and that their passphrases have been set as intended, see Passphrases
10. Erase the Administrator Cards from the old card set. For more information, see Erase cards and softcards.

10.7.2. Replace an ACS using racs

The racs utility creates a new ACS to replace a set that was created with the new-world utility.

> When using the `racs` utility, you cannot redefine the quantities in a K of N relationship for an ACS. The K of N relationship defined in the original ACS persists in the new ACS.

1. Ensure the HSM is in operational mode.
2. Run the `racs` utility:

   ```
   racs [-m|--module=<MODULE>]
   ```

   In this command: ** `<MODULE>`: the `ModuleID` of the module to use.

3. When prompted, insert the appropriate quorum of Administrator Cards to authorize the replacement.
4. When prompted that `racs` is writing the new ACS, insert blank cards as necessary on which to write the replacement Administrator Cards.
5. **Additional steps for network-attached HSMs:**
   a. If you ran `racs` on a client machine, ensure that there is a copy of the modified **world** file on the RFS.
   b. Make a backup of the **world** file, preferably in a way that distinguishes it from previous backups.
   c. Copy the **world** file to any other HSMs in the same Security World, for example using the `nethsmadmin` utility, see Using nethsmadmin to copy a Security World to an nShield HSM and check the current version.
   d. If client cooperation is not enabled, copy the modified **world** file onto any other client machines where it is needed.
   e. Check that the new Administrator Cards are usable and that their passphrases have been set as intended, see Passphrases.

6. When you have finished replacing the ACS, erase the old Administrator Cards. For more information, see Erase cards and softcards.

11. Application interfaces

You can use the `generatekey` utility to generate or import keys for use with your applications (see Working with keys).

On **Linux**, you must add the user of any application that uses an nShield HSM to the group **nfast** before the application runs.

On **Windows**, by default any user is allowed to use any application that uses an nShield HSM.

Network-attached HSMs only: If you create keys on a client that is not on the same computer as the RFS, you must copy the key data to the RFS before the nShield HSM can use these keys.

11.1. nShield native and custom applications

Use the nShield native option for applications that were written using nShield key management software and that expect keys to be both protected by the Security World and stored in the Security World data structure.

Use the `custom` external application option for applications that were written using nShield key management software and that expect their keys to be in standalone files.

You must make sure that your application is capable of loading the card set.

11.2. Other types of application

See the following links for help using HSMs with other types of application:

- nCipherKM JCA/JCE CSP
- PKCS #11 applications
- Microsoft CAPI CSP
- Microsoft CNG CSP

12. Environment variables

This appendix describes the environmental variables used by Security World Software.

> When you are using these environment variables on Windows to configure nShield services such as the hardserver (nFast Server service), these must be set as System variables only; not as User Variables. Any service for which the environment variable changes are intended must be restarted for the change to take effect.

Variable	Description	Win	Lnx
KERNEL_HEADERS	This variable allows you to specify the path to kernel headers (if, for example, they are not in the default directory). It is necessary for the configuration script to be able to find the kernel headers when building the PCI driver during software installation.	n	y
NFAST_CERTDIR	This variable specifies the path to the dynamic feature enabling Feature Certificates directory. You only need to change the value of this variable if you move the Installation directory. See NFAST_HOME, NFAST_KMDATA, and NFAST_LOGDIR.	y	y
NFAST_DEBUG	This variable enables debug logging for the PKCS #11 library. You must set NFAST_DEBUG equal to a value in the range 1 – 7 for debug messages to be logged (see Hardserver debugging).	y	y
NFAST_DEBUGSYSLOG	This variable redirects debug logging to syslog. The value of the environment variable should be one of the syslog facilities to be used. Prefixing the facility name with + enables traditional logging and syslog simultane ously.	y	Y
NFAST_HOME	This variable specifies the path to the Installation direc tory, which is set automatically in the system environment variables by the installer on Windows. It is not required on Linux because installation to locations other than /opt/nfast is not generally supported. See NFAST_KMDATA, NFAST_CERTDIR, and NFAST_LOGDIR.	y	y
NFAST_KMDATA	This variable sets the location of the Key Management Data directory. You only need to change the value of this variable if you move the Key Management Data directory. See NFAST_HOME, NFAST_CERTDIR, NFAST_LOGDIR, and NFAST_KMLOCAL.	y	y

Variable	Description	Win	Lnx
NFAST_KMLOCAL	This variable specifies the location of the Key Management and Security World Data directory. If this environ ment variable is not set, by default the module looks for the Security World data in the local subdirectory of the Key Management Data directory. See NFAST_KMDATA.	y	y
NFAST_LOGDIR	This variable specifies the location of the Log Files directory. You only need to change the value of this variable if you move the Log Files directory. See NFAST_HOME, NFAST_KMDATA, and NFAST_CERTDIR.	y	y
NFAST_USER_LOGDIR	This variable specifies the location of log files that are specific to each user. In Security World versions before 12.60.3, the default is the user's home directory (Linux) or user profile folder (Windows). From 12.60.3, the default is the subdirectory nshieldlogs in the home directory or user profile folder.	y	y
NFAST_NFKM_TOKENSFILE NFAST_NFKM_TOKENSSELECT	This variable sets the default values for a file in which ClientID and KeyIDs are stored by the preload command-line utility.	y	y
NFAST_SEE_MACHINEENCKEY_DEFAULT	This variable is the name of the SEEConf key needed to decrypt SEE-machine images. Running the command loadmache --encryptionkey=`IDENT (or `loadmache --unencrypted) overrides any value set by this variable.	y	y
NFAST_SEE_MACHINEENCKEY_<module>	This variable is the name of the SEEConf key needed to decrypt the SEE-machine image targeted for the spec ified module. It overrides NFAST_SEE_MACHINEENCKEY_DEFAULT for the specified module. Running the command loadmache --encryptionkey=<IDENT> (or loadmache --unencrypted) overrides any value set by this variable.	y	y
NFAST_SEE_MACHINEIMAGE_DEFAULT	This variable is the path of the SEE machine image to load on to any module for which a specific image is not defined. Supplying the machine-filename parameter when running the loadmache command-line utility overrides this variable. This variable is not affected when running the loadsee-setup or hsc_loadseemachine utilities.	y	y

Variable	Description	Win	Lnx
NFAST_SEE_MACHINEIMAGE_<module>	This variable is the path of the SEE machine image to load on to the specified module. If set, this variable overrides the use of NFAST_SEE_MACHINEIMAGE_DEFAULT for the specified module. Supplying the <machine-file name> parameter when running the loadmache command-line utility overrides the NFAST_SEE_MACHINEIMAGE_<module> variable. This variable is not affected when running the loadsee-setup or hsc_loadseemachine utilities.	y	y
NFAST_SEE_MACHINESIGHASH_DEFAULT	This variable is the default key hash of the vendor sign ing key (seeinteg) that signs SEE machine images. This variable is only required if you are using a dynamic SEE feature with an encrypted SEE machine. Running the command loadmache --sighash=<HASH> any value set in this variable.	y	y
NFAST_SEE_MACHINESIGHASH_<module>	This variable is the key hash of the vendor signing key (seeinteg) that signs SEE machine images for the spec ified module. It overrides NFAST_SEE_MACHINESIGHASH_DEFAULT for the specified module. This variable is only required if you are using a dynamic SEE fea ture with an encrypted SEE machine. Running the com mand loadmache --sighash=<HASH> any value set in this variable.	y	y
NFAST_SERVER NFAST_PRIVSERVER	If these variables are set in the hardserver's environment, the values specify: **On Linux**, the pathnames of the UNIX domain sockets that the hardserver uses for ordinary/privileged client connections to the hardserver. **On Windows**, the names of the Windows named pipes for ordinary/privileged client connections to the hardserver. These variables are available for this purpose for backward compatibility only; you should configure sockets in the hardserver configuration file. If you set these variables for a network-attached HSM, they override the values in the hardserver configuration file. See nShield HSM configuration files.	y	y

Variable	Description	Win	Lnx
NFAST_SERVER_PORT NFAST_SERVER_PRIVPORT	If these variables are set in the hardserver's environment, the values specify the TCP port numbers that the nFast server uses for connections over TCP sockets. These variables are available for this purpose for backward compatibility only: you should configure ports in the hardserver configuration file, as described in server_startup. If you set these variables, they override the values in the hardserver configuration file.	y	y
NFLOG_CATEGORIES	This variable is used to filter log messages by supplying a colon-separated list of allowable message categories; see Logging, debugging, and diagnostics. If no value is supplied, all message categories are logged.	y	y
NFLOG_SEVERITY	This variable is used to filter log messages by supplying a minimum severity level to be logged; see Logging, debugging, and diagnostics. If no value is supplied, the default severity level is WARNING.	y	y
NFLOG_DETAIL	This variable is used to filter log messages by supplying a bitmask of detail flags; see Logging, debugging, and diagnostics. The default is time+severity+writeable.	y	y
NFLOG_FILE	This variable is used to specify a filename (or file descriptor) in which log messages are to be written; see Logging, debugging, and diagnostics. The default is stderr (the equivalent of file descriptor &2).	y	y