THALES

Thales Luna Network HSM 7

PARTITION ADMINISTRATION GUIDE

Document Information

Last Updated	2025-12-03 12:18:22 GMT-05:00

Disclaimer
All information herein is either public information or is the property of and owned solely by Thales Group and/or its subsidiaries who shall have and keep the sole right to file patent applications or any other kind of intellectual property protection in connection with such information.

Nothing herein shall be construed as implying or granting to you any rights, by license, grant or otherwise, under any intellectual and/or industrial property rights of or concerning any of Thales Group's information.

This document can be used for informational, non-commercial, internal, and personal use only provided that:

- The copyright notice, the confidentiality and proprietary legend and this full warning notice appear in all copies.
- This document shall not be posted on any publicly accessible network computer or broadcast in any media, and no modification of any part of this document shall be made.

Use for any other purpose is expressly prohibited and may result in severe civil and criminal liabilities.

The information contained in this document is provided "AS IS" without any warranty of any kind. Unless otherwise expressly agreed in writing, Thales Group makes no warranty as to the value or accuracy of information contained herein.

The document could include technical inaccuracies or typographical errors. Changes are periodically added to the information herein. Furthermore, Thales Group reserves the right to make any change or improvement in the specifications data, information, and the like described herein, at any time.

Thales Group hereby disclaims all warranties and conditions with regard to the information contained herein, including all implied warranties of merchantability, fitness for a particular purpose, title and non-infringement. In no event shall Thales Group be liable, whether in contract, tort or otherwise, for any indirect, special or consequential damages or any damages whatsoever including but not limited to damages resulting from loss of use, data, profits, revenues, or customers, arising out of or in connection with the use or performance of information contained in this document.

Thales Group does not and shall not warrant that this product will be resistant to all possible attacks and shall not incur, and disclaims, any liability in this respect. Even if each product is compliant with current security standards in force on the date of their design, security mechanisms' resistance necessarily evolves according to the state of the art in security and notably under the emergence of new attacks. Under no circumstances, shall Thales Group be held liable for any third party actions and in particular in case of any successful attack against systems or equipment incorporating Thales products. Thales Group disclaims any liability with respect to security for direct, indirect, incidental or consequential damages that result from any use of its products. It is further stressed

that independent testing and verification by the person using the product is particularly encouraged, especially in any application in which defective, incorrect or insecure functioning could result in damage to persons or property, denial of service, or loss of privacy.

Regulatory Compliance

This product complies with the following regulatory regulations. To ensure compliancy, ensure that you install the products as specified in the installation instructions and use only Thales-supplied or approved accessories.

USA, FCC

This equipment has been tested and found to comply with the limits for a “Class B” digital device, pursuant to part 15 of the FCC rules.

Canada

This class B digital apparatus meets all requirements of the Canadian interference-causing equipment regulations.

Europe

This product is in conformity with the protection requirements of EC Council Directive 2014/30/EU. This product satisfies the CLASS B limits of EN55032.

CONTENTS

PREFACE: About the Partition Administration Guide

This document describes the operational and administrative tasks you can perform to maintain the functionality and efficiency of your application partitions. It contains the following chapters:

The preface includes the following information about this document:

For information regarding the document status and revision history, see "Document Information" on page 2.

Customer Release Notes

The Customer Release Notes (CRN) provide important information about specific releases. Read the CRN to fully understand the capabilities, limitations, and known issues for each release. You can view the latest version of the CRN at www.thalesdocs.com.

Audience

This document is intended for personnel responsible for maintaining your organization's security infrastructure. This includes Luna HSM users and security officers, key manager administrators, and network administrators.

All products manufactured and distributed by Thales are designed to be installed, operated, and maintained by personnel who have the knowledge, training, and qualifications required to safely perform the tasks assigned to them. The information, processes, and procedures contained in this document are intended for use by trained and qualified personnel only.

It is assumed that the users of this document are proficient with security concepts.

Document Conventions

This document uses standard conventions for describing the user interface and for alerting you to important information.

Notes

Notes are used to alert you to important or helpful information. They use the following format:

> **NOTE** Take note. Contains important or helpful information.

Cautions

Cautions are used to alert you to important information that may help prevent unexpected results or data loss. They use the following format:

> **CAUTION!** Exercise caution. Contains important information that may help prevent unexpected results or data loss.

Warnings

Warnings are used to alert you to the potential for catastrophic data loss or personal injury. They use the following format:

> ****WARNING** Be extremely careful and obey all safety and security measures. In this situation you might do something that could result in catastrophic data loss or personal injury.**

Command syntax and typeface conventions

Format	Convention
bold	The bold attribute is used to indicate the following: > Command-line commands and options (Type **dir /p**.) > Button names (Click **Save As**.) > Check box and radio button names (Select the **Print Duplex** check box.) > Dialog box titles (On the **Protect Document** dialog box, click **Yes**.) > Field names (**User Name**: Enter the name of the user.) > Menu names (On the **File** menu, click **Save**.) (Click **Menu** > **Go To** > **Folders**.) > User input (In the **Date** box, type **April 1**.)
italics	In type, the italic attribute is used for emphasis or to indicate a related document. (See the *Installation Guide* for more information.)
<variable>	In command descriptions, angle brackets represent variables. You must substitute a value for command line arguments that are enclosed in angle brackets.
[**optional**] [<optional>]	Represent optional **keywords** or <variables> in a command line description. Optionally enter the keyword or <variable> that is enclosed in square brackets, if it is necessary or desirable to complete the task.
{**a**|**b**|**c**} {<a>|<b>|<c>}	Represent required alternate **keywords** or <variables> in a command line description. You must choose one command line argument enclosed within the braces. Choices are separated by vertical (OR) bars.
[**a**|**b**|**c**] [<a>|<b>|<c>]	Represent optional alternate keywords or variables in a command line description. Choose one command line argument enclosed within the braces, if desired. Choices are separated by vertical (OR) bars.

Support Contacts

If you encounter a problem while installing, registering, or operating this product, please refer to the documentation before contacting support. If you cannot resolve the issue, contact your supplier or Thales Customer Support. Thales Customer Support operates 24 hours a day, 7 days a week. Your level of access is governed by the support plan negotiated between Thales and your organization. Please consult this plan for details regarding your entitlements, including the hours when telephone support is available to you.

Customer Support Portal

The Customer Support Portal, at https://supportportal.thalesgroup.com, is where you can find solutions for most common problems and create and manage support cases. It offers a comprehensive, fully searchable database of support resources, including software and firmware downloads, release notes listing known problems and workarounds, a knowledge base, FAQs, product documentation, technical notes, and more.

> **NOTE** You require an account to access the Customer Support Portal. To create a new account, go to the portal and click on the **REGISTER** link.

Telephone

The support portal also lists telephone numbers for voice contact (Contact Us).

CHAPTER 1: Luna HSM Client Software Installation

You can install the client for all Luna General Purpose HSMs, or for a specific type (Network, PCIe, or USB). Install the client as follows:

- For Luna Network HSM 7, install the Luna HSM Client on any computer that must connect to the appliance as a client.
- For Luna PCIe HSM 7, install the Luna HSM Client on the workstation into which the Luna PCIe HSM 7 is installed.
- For Luna USB HSM 7, install the Luna HSM Client on the workstation connected to the Luna USB HSM 7.
- Install the Luna HSM Client on any computer that is to have a Remote Luna PED connected.
- Install the Luna HSM Client on any computer that is to serve as a Remote Backup server.

For a list of supported operating systems by client version, refer to the CRN:

- Customer Release Notes

Choose the instructions for your operating system:

Windows Luna HSM Client Installation

This section describes how to invoke the Windows Luna HSM Client perform unattended or scripted installations on Windows platforms.

> **NOTE** The GUI interactive installer (see "Windows Interactive Luna HSM Client Installation" on page 28) is deprecated, and will be removed from a future release.

Use the **/quiet** switch (see below) to ensure no pauses or prompting during installation. The following procedures are described:

If you want to perform an interactive installation, using the graphical, interactive installer, see "Windows Interactive Luna HSM Client Installation" on page 28

> **NOTE** Unattended installation stores the root certificate in the certificate store and marks the publisher (Thales) as trusted for future installations. You are not prompted to trust Thales as a driver publisher during unattended installation.

Command line options overview

The following command-line options are available:

Option	Values	Description
addlocal=	Various (see below)	Takes one-or-more device values, and one-or-more feature values, as a comma-separated list. Case insensitive. Values may be quoted or not.
installdir=	A fully qualified folder path to install the client software	Case insensitive. Default value is “c:\program files\safenet\lunaclient”. Enclose paths containing spaces in “”.
/install	N/A	Install the product and features.

Option	Values	Description
/uninstall	N/A	Remove the product and features.
/quiet	N/A	Performs a silent installation; no prompts or messages. **NOTE** Windows defaults to launching the interactive graphical installer, unless you specify **/quiet** at the command line. Always include the **/quiet** option for scripted/unattended Luna HSM Client installation.
/norestart	N/A	Prevents a reboot, post-installation. Any reboots must be performed manually.
/log	The name of a log file	Generates a highly detailed series of logs of the installation progress. This is required only for product support.

The following devices or components are available for use with the addlocal= option:

Device identifier value	Can be used with these installable features
NETWORK	CSP_KSP, JSP, SDK, JCProv*
PCI	CSP_KSP, JSP, SDK, JCProv, SNMP
USB	CSP_KSP, JSP, SDK, JCProv, SNMP
BACKUP	SNMP (this device performs backup and restore operations and is not enabled for cryptographic applications)
PED	N/A (Used for remotely authenticating to multifactor quorum-authenticated HSMs; not used by cryptographic applications - use of this device requires hands-on presence)

The device names are not case-sensitive.

* The Luna Network HSM 7 appliance contains its own SNMP support; therefore the SNMP feature is not installed on clients where the Luna Network HSM 7 is the only HSM to be used.)

The following features are available for use with the addlocal= option:

Feature identifier value	Can be installed with these Luna devices	Description
CSP_KSP	NETWORK, PCI, USB	Microsoft CSP and KSP
FMSDK	NETWORK, PCIe *	Functionality Modules Software Development Kit

Feature identifier value	Can be installed with these Luna devices	Description
FMTOOLS	NETWORK, PCIe *	Tools for use when preparing Functionality Modules
JCProv	NETWORK, PCIe, USB	JCPROV PKCS#11
JSP	NETWORK, PCIe, USB	Java Provider component
SDK	NETWORK, PCIe, USB	Software SDK – Java / C++ samples

The features can be installed together with the listed device(s) only - they cannot be installed separately - and need to be included only once in the command line. For example, if you are installing the NETWORK and PCI devices and you wish to install the CSP / KSP feature, specify CSP_KSP one time. The feature names are not case-sensitive.

NOTE * If you install FMTOOLS for NETWORK only, then just **mkfm** and the **library** are installed.

If you install FMTOOLS for PCI, then **mkfm** and the **library** along with **ctfm** and **fmrecover** are installed.

If you install FMTOOLS for both NETWORK and PCIe devices, then all four elements are installed.

If you install the FM SDK, the Luna SDK is installed as well, to satisfy dependencies.

Options for **addlocal=** are separated by spaces. Device and feature values are separated by commas, with no spaces, unless the whole list is enclosed between quotation marks. If a space is encountered, outside of paired quotation marks, the next item found is treated as a command option.

Installing all components and features

> **NOTE** CSP or KSP registration includes a step that verifies the DLLs are signed by our certificate that chains back to the DigiCert root of trust G4 (in compliance with industry security standards).
>
> This step can fail if your Windows operating system does not have the required certificate. If you have been keeping your Windows OS updated, you should already have that certificate.
>
> If your Luna HSM Client host is connected to the internet, use the following commands to update the certificate manually:
>
> **certutil -urlcache -f** http://cacerts.digicert.com/DigiCertTrustedRootG4.crt
>
> **certutil -addstore -f root** DigiCertTrustedRootG4.crt
>
> **To manually update a non-connected host**
>
> 1. Download the DigiCert Trusted Root G4 (http://cacerts.digicert.com/DigiCertTrustedRootG4.crt) to a separate internet-connected computer.
> 2. Transport the certificate, using your approved means, to the Luna HSM Client host into a <downloaded cert path> location of your choice
> 3. Add the certificate to the certificate store using the command:
>
>   **certutil -addstore -f root** <downloaded cert path>

Subsequent sections detail how to install the Luna HSM Client software, drivers (if necessary), and optional features (like Java support and the SDK), for individual HSMs. This section describes how to install everything at once, so that all Luna HSMs and Remote PED are supported and all the optional features are available.

Use the **ADDLOCAL=** option together with the value **all** to install the base client software and the drivers for all Luna devices, along with all the features.

To install the Luna HSM Client software and drivers for *all* Luna devices and *all* features

From the location of **LunaHSMClient.exe** run the following command:

> Install the full Luna HSM Client software with drivers for all Luna HSMs (Luna Network HSM 7, Luna PCIe HSM 7, Luna Backup HSM, Remote PED), as well as all the features (CSP/KSP, JSP, JCProv, C++ SDK, SNMP Subagent)
>
> **LunaHSMClient.exe /install /quiet ADDLOCAL=all**
>
> > **NOTE** You can omit the **/quiet** option to see all options in the GUI dialog.

> [Optional logging] Install the full Luna HSM Client software with drivers for all Luna HSMs (Luna Network HSM 7, Luna PCIe HSM 7, Luna Backup HSM, Remote PED, as well as all the features (CSP/KSP, JSP, JCProv, C++ SDK, SNMP Subagent), and log the process.
>
> **LunaHSMClient.exe /install /log install.log /quiet ADDLOCAL=all**

> **NOTE** The setting **/log** is optional and saves the installation logs to the file named **install.log** in the example. The **install.log** file (whatever name you give it) is required only if troubleshooting an issue with Thales GroupTechnical Support.

Installing the Luna HSM Client for the Luna Network HSM 7

Use the **ADDLOCAL=NETWORK** option to install the base client software for the Luna Network HSM 7. Include the values for any optional, individual software components you desire. The base software must be installed first.

To install the Luna HSM Client for the Luna Network HSM 7

From the location of **LunaHSMClient.exe** run one of the following commands:

> Install the base Luna HSM Client software necessary to communicate with Luna Network HSM 7

LunaHSMClient.exe /install /quiet ADDLOCAL=NETWORK

[Optional] Install the base Luna HSM Client software and any of the optional components for the Luna Network HSM 7 that you desire:

For example, the following command installs the base software and all of the optional components:

LunaHSMClient.exe /install /quiet ADDLOCAL=NETWORK,CSP_KSP,JSP,SDK,JCProv

If you wish to install only some of the components, just specify the ones you want after the product name (NETWORK in this example).

Installing the Luna HSM Client for the Luna PCIe HSM 7

Use the **ADDLOCAL=PCI** option to install the base client software for the Luna PCIe HSM 7. Include any features you desire. The base software must be installed first.

To install the Luna HSM Client for the Luna PCIe HSM 7

From the location of **LunaHSMClient.exe** run one of the following commands:

> Install the base Luna HSM Client software for Luna PCIe HSM 7

LunaHSMClient.exe /install /quiet ADDLOCAL=PCI

> Install the base Luna HSM Client software and any of the optional features for the Luna PCIe HSM 7 that you desire:

For example, the following command installs the base software and all of the optional components:

LunaHSMClient.exe /install /quiet ADDLOCAL=PCI,CSP_KSP,JSP,SDK,JCProv,SNMP

If you wish to install only some of the components, just specify the ones you want after the product name (PCI in this example).

Installing the Luna HSM Client for the Luna USB HSM 7

Use the **ADDLOCAL=USB** option to install the base client software for the Luna USB HSM 7. Include any features you desire. The base software must be installed first.

To install the Luna HSM Client for the Luna USB HSM 7

From the location of **LunaHSMClient.exe** run one of the following commands:

> Install for Luna USB HSM 7

LunaHSMClient.exe /install /quiet ADDLOCAL=USB

> Install the base Luna HSM Client software and any of the optional features for the Luna USB HSM 7 that you desire:

For example, the following command installs the base software and all of the optional components:

LunaHSMClient.exe /install /quiet ADDLOCAL=USB,CSP_KSP,JSP,SDK,JCProv

If you wish to install only some of the components, just specify the ones you want after the product name (USB in this example).

Installing the Luna HSM Client for the Luna Backup HSM

Use the **ADDLOCAL=BACKUP** option to install the base client software for the Luna Backup HSM, and the optional feature, if desired. For the Backup HSM, which performs backup and restore operations and is not enabled for use with cryptographic applications, the feature you might add is SNMP, if applicable in your environment.

To install the Luna HSM Client for the Luna Backup HSM

From the location of **LunaHSMClient.exe** run one of the following commands:

> Install the base Luna HSM Client software for Luna Backup HSM

LunaHSMClient.exe /install /quiet /norestart ADDLOCAL=BACKUP

> Install the base Luna HSM Client software and an optional component for the Luna Backup HSM:

For example, the following command installs the base software and the optional component:

LunaHSMClient.exe /install /quiet /norestart ADDLOCAL=backup

Installing the Luna HSM Client for Remote PED

Use the **ADDLOCAL=** option with component value **PED**to install the client software for the Remote PED Server.

To install the Luna HSM Client for the Remote PED Server

> From the location of **LunaHSMClient.exe** run the following command:

LunaHSMClient.exe /install /quiet addlocal=ped

Installation Location

Specify the installation location, if the default location is not suitable for your situation.

This applies to installation of any Luna Device. Provide the **INSTALLDIR=** option, along with a fully qualified path to the desired target location. For example:

LunaHSMClient.exe /install /quiet addlocal=all installdir=c:\lunaclient

That command silently installs all of the Luna device software and features to the folder c:\lunaclient (in this example). The software is installed into the same subdirectories per component and feature, under that named folder, as would be the case if **INSTALLDIR** was not provided. That is, **INSTALLDIR** changes the prefix or primary client installation folder to the one you specify, and the libraries, devices, tools, certificate folders, etc. are installed in their predetermined relationship, but under the new main folder location.

ChrystokiConfigurationPath Environment Variable

During installation of Luna HSM Client components, a new entry is added to the Windows environment variables: **ChrystokiConfigurationPath**. This variable contains the path to the Luna HSM Client configuration file, **Chrystoki.ini** (see "Configuration File Summary" on page 76 for a full description).

> **NOTE** After first-time installation or a re-installation where the path to **Chrystoki.ini** changed, any open command prompts must be closed and reopened to recognize the new **ChrystokiConfigurationPath** environment variable setting.

Logging

If problems are encountered during installation or uninstallation of the software and you wish to determine the reason, or if Thales Technical Support has requested you to do so, detailed logs can be generated and captured by specifying the /log option and providing a filename to capture the log output. Two logs are generated – one according to the name given and the other similarly named, with a number appended. Both log files must be sent to Thales support if assistance is required.

Example commands that include logging are:

LunaHSMClient.exe /install /quiet /log install.log /norestart ADDLOCAL=backup,snmp

LunaHSMClient.exe /uninstall /quiet /log uninstall.log

Uninstalling the Luna HSM Client

You can also perform scripted/unattended uninstallation.

To uninstall the Luna HSM Client

> From the location of **LunaHSMClient.exe** run the following command:

LunaHSMClient.exe /uninstall /quiet

> To log the uninstallation process, run the following command:

LunaHSMClient.exe /uninstall /quiet /log uninstall.log

Windows Interactive Luna HSM Client Installation

> **NOTE** The GUI interactive installer (see "Windows Interactive Luna HSM Client Installation" above) is deprecated, and will be removed from a future release.

This section describes how to install the Luna HSM Client software on Windows, using the GUI interactive installer. It contains the following topics:

Applicability to specific versions of Windows is summarized in the Customer Release Notes for this release.

> **NOTE** Before installing a Luna HSM system, confirm that the product you have received is in factory condition and has not been tampered with in transit. Refer to the Startup Guide included with your product shipment. If you have any questions about the condition of the product that you have received, contact Technical Support immediately.

Required Client Software

Each computer that connects to a Luna Network HSM 7 as a Client must have the cryptoki library, the **vtl** client shell and other utilities and supporting files installed.

Each computer that contains, or is connected to a Luna PCIe HSM 7 or a Luna USB HSM 7 must have the cryptoki library and other utilities and supporting files installed.

Prerequisites

The Luna HSM Client installer requires the Microsoft Universal C Runtime (Universal CRT) to run properly. Universal CRT requires your Windows machine to be up to date. Before running the installer, ensure that you have the Universal C Runtime in Windows (KB2999226) update and its prerequisites installed on your machine. The following updates must be installed in order:

1. March 2014 Windows servicing stack update (see https://support.microsoft.com/en-us/help/2919442)
2. April 2014 Windows update (see https://support.microsoft.com/en-us/help/2919355)

3. Visual C++ Redistributable for Visual Studio 2015 (see https://www.microsoft.com/en-in/download/details.aspx?id=481450)

> **NOTE** CSP or KSP registration includes a step that verifies the DLLs are signed by our certificate that chains back to the DigiCert root of trust G4 (in compliance with industry security standards).
>
> This step can fail if your Windows operating system does not have the required certificate. If you have been keeping your Windows OS updated, you should already have that certificate.
>
> If your Luna HSM Client host is connected to the internet, use the following commands to update the certificate manually:
>
> **certutil -urlcache -f** http://cacerts.digicert.com/DigiCertTrustedRootG4.crt
>
> **certutil -addstore -f root** DigiCertTrustedRootG4.crt
>
> **To manually update a non-connected host**
>
> 1. Download the DigiCert Trusted Root G4 (http://cacerts.digicert.com/DigiCertTrustedRootG4.crt) to a separate internet-connected computer.
> 2. Transport the certificate, using your approved means, to the Luna HSM Client host into a <downloaded cert path> location of your choice
> 3. Add the certificate to the certificate store using the command:
>
>   **certutil -addstore -f root** <downloaded cert path>

Installing the Luna HSM Client Software

Luna HSM Client can be installed on 64-bit Windows operating systems. Hardware drivers are 64-bit only. Older client versions include 32-bit libraries and binaries.

> **NOTE** Luna HSM Client 10.1.0 and newer includes libraries for 64-bit operating systems only.

For compatibility of our HSMs with Windows CAPI we have Luna CSP, and for the newer Windows CNG we have Luna KSP. See "Luna CSP and KSP" on page 33 for more information.

Interactive (prompted, this page) and non-interactive (no prompts "Windows Luna HSM Client Installation" on page 21) installation options are available.

> **NOTE Compatibility of Luna PCIe HSM 7 version, Client version, and Windows OS versions**
>
> Luna HSM Client 10.3.0 was the last client version to support Windows Server 2012 R2, which accepts the Luna PCIe HSM 7 6.x driver.
>
> If you have Windows Server 2012 R2 computer with a Luna PCIe HSM 7 6.x onboard, do not install Luna HSM Client 10.4.0 or newer there; these client versions will not load the Luna PCIe HSM 7 6.x driver.
>
> If you have Luna PCIe HSM 7 6.x and 7.x HSM card in the same system, failure of the 6.x driver would prevent loading of the 7.x driver as well. If your application works with Linux, the Luna PCIe HSM 7 6.x will continue to work there, and will not block Luna PCIe HSM 7 7.x.

To install the Luna HSM Client software

1. Log into Windows as **Administrator**, or as a user with administrator privileges (see "Troubleshooting" on page 37).
2. Uninstall any previous versions of the Client software before you proceed (see "Uninstalling the Luna HSM Client Software" on page 35).

   > **NOTE** If you do not uninstall previous Luna HSM Client versions, you might face installation issues, such as failure to install the new client.

3. Download the Luna HSM Client from the Thales Support Portal at https://supportportal.thalesgroup.com.

   > **TIP** Thales recommends verifying the integrity of the Luna HSM Client packages, by calculating their SHA256 hash values and comparing with the hash values posted on the Support Portal, before installing them on your client machines.
   >
   > You can use the sha256sum tool on Linux machines to calculate the SHA256 hash values.

4. Extract the .zip to an appropriate folder.
5. In the extracted directory, locate the folder for your Windows architecture and double click **LunaHSMClient.exe**.
6. The Custom Setup dialog allows you to choose which software components you wish to install. Click a product to select the components to install, or click Select All to install all available components.

   The installer includes the Luna SNMP Subagent as an option with any of the Luna HSMs, except Luna Network HSM 7, which has agent and subagent built in. After installation of the Luna SNMP Subagent is complete, you will need to move the SafeNet MIB files to the appropriate directory for your SNMP application, and you will need to start the SafeNet subagent and configure for use with your agent, as described in SNMP Monitoring.
7. The Custom Setup dialog allows you to choose which software components you wish to install. Click a product to select the components to install, or click Select All to install all available components.

   The installer includes the Luna SNMP Subagent as an option with any of the Luna HSMs, except Luna Network HSM, which has agent and subagent built in. After installation of the Luna SNMP Subagent is complete, you will need to move the SafeNet MIB files to the appropriate directory for your SNMP application, and you will need to start the SafeNet subagent and configure for use with your agent, as described in SNMP Monitoring.

> **NOTE Dependencies and considerations when installing:**
> - The FM Tools and FM SDK are useful to you only if you will be using or creating Functionality Modules, to add custom abilities to your HSMs.
> - The FM SDK requires that you install Luna PCIe HSM 7 software and drivers.
> - Similarly, if you are using third-party software to make standard cryptographic calls to the HSM, and are not creating application programs, then you can forego loading the Software Development Kit.
> - There is no harm in installing unneeded components; they do not conflict.
> - The FM SDK option remains gray/unselectable until "Software SDK" is selected, because some of the FM SDK samples have dependencies on General Cryptoki Samples that are part of "Software SDK".

After you select the components you want to install, click **Install**.

a. Agree to the terms of the License Agreement to proceed with installation. To view the agreement text, click the link in the dialog. The installer loads a PDF version if a PDF reader is available; otherwise it launches a text editor and a plain-text version of the agreement.

b. If Windows presents a security notice asking if you wish to install the device driver from Thales, click "Always trust software from Thales DIS CPL USA, Inc." and click **Install** to accept.

c. If you choose not to install the driver(s), your Luna HSM Client cannot function with any locally-connected Luna hardware (which includes Luna PCIe HSM 7, Luna USB HSM 7, or Luna Backup HSMs).

8. When the installation completes, the button options are Uninstall, Modify, or Quit; click **Quit** to finish.
9. [Optional] For easy use of the Luna HSM Client command-line tools, add the directory to the system PATH variable.

   "C:\Program Files\SafeNet\Lunaclient"

Modifying the Installed Windows Luna HSM Client Software

If you wish to modify the installation (perhaps to add a component or product that you did not previously install), you must re-run the current installer and ensure that the desired options are selected.

> **NOTE** This feature requires minimum Luna HSM Client 7.2.0.

To modify the installed Luna HSM Client software

1. Run the **LunaHSMClient.exe** program again. Because the software is already installed on your computer, the following dialog is displayed (in this example, devices and features were previously installed, and the task is to uninstall a couple of items):

2. Select or deselect individual Devices or Features, as desired.
3. Click **Modify**. The client software is updated (items are added or removed).

   If you are uninstalling some items, or if you are adding features, the dialog shows a progress bar briefly, and then shows the current status.

   If you are adding a Luna Device, then you might be prompted with the operating system pop-up to accept/trust the driver.
4. Click **Quit** when the modification is complete.

> **NOTE** You can also use **Programs and Features** in the Windows Control Panel to launch the Uninstall/Modify dialog for the client software.

Java

If you install the Luna Java Security Provider (JSP), refer to Luna JSP Overview and Installation for additional setup procedures for your operating system.

Luna CSP and KSP

Thales provides Luna CSP for applications running in older Windows crypto environments running Microsoft Certificate Services (CAPI), and Luna KSP for newer Windows clients running Cryptography Next Generation (CNP). Consult Microsoft documentation to determine which one is appropriate for your client operating system.

> Luna CSP Registration Utilities

> Luna KSP for CNG Registration Utilities

If the **Luna CSP (CAPI) / Luna KSP(CNG)** option is selected at installation time, the **SafeNetKSP.dll** file is installed in **C:\Windows\System32** (used for 64-bit KSP). If you are installing a Luna HSM Client version older than 10.1, **SafeNetKSP.dll** is also installed in **C:\Windows\SysWOW64** (used for 32-bit KSP).

> **NOTE** The **cryptoki.ini** file, which specifies many configuration settings for your HSM and related software, includes a line that specifies the path to the appropriate libNT for use with your application(s). Verify that the path is correct.

USB-powered PED

The USB-powered Luna PIN Entry Device (PED) contains new hardware that enables the PED to be powered by the USB connection; there is no longer a requirement for an external power Adapter. It is functionally equivalent to your existing (previous-generation) Luna PEDs and is compatible with HSM versions, 5.x, 6.x, and 7.x.

The USB-powered Luna PED ships with Luna PED Firmware 2.8.0. Note that you cannot upgrade older, adapter-powered Luna PEDs to 2.8.0; existing PEDs continue to need a separate power adapter for remote PED and upgrade use. The model number on the manufacturer's label identifies the refreshed PED: PED-06-0001. An installed driver is required; see step 1, below.

To use the new USB-powered PED

1. Ensure the Luna HSM Client software is installed on the Windows computer that will act as the Remote PED server to your Luna HSM. Installing the Remote PED component of the Luna HSM Client installs the required driver.

   > **NOTE** A USB connection, without the driver software, only illuminates the Luna PED screen, with no menu. An installed and running PED driver, on the connected computer, is required for the PED to fully boot and to display its menu.

2. Connect the PED to the computer where you installed the Remote PED component of the Luna HSM Client, using the USB micro connector on the PED and a USB socket on your computer.
3. After you connect the PED to the host computer, it will take 30 to 60 seconds for initial boot-up, during which time a series of messages are displayed, as listed below:

   BOOT V.1.1.0-1

   CORE V.3.0.0-1

   Loading PED...

   Entering...
4. After the boot process is complete, the PED displays **Local PED mode** and the **Awaiting command...** prompt. Your new Luna PED is now ready for use.
5. To enter Remote PED mode, if needed, exit Local PED mode with the "< " key, and from the **Select Mode** menu, select option **7 Remote PED**.

Modifying the Number of Luna Backup HSM Slots

By default, the Luna HSM Client allows for three slots reserved for each model of Luna Backup HSM. You can edit **crystoki.ini** to modify the number of reserved slots. See also "Configuration File Summary" on page 76.

To modify the number of reserved Backup HSM slots

1. Navigate to the **crystoki.ini** file and open in a text editor.

2. Add the following line(s) to the **CardReader** section of the file:
   - For Luna Backup HSM G5:

     LunaG5Slots = <value>**;**
   - For Luna Backup HSM 7:

     LunaG7Slots = <value>**;**

Uninstalling the Luna HSM Client Software

You need to uninstall Luna HSM Client before installing a new version. If you wish to modify the installation (perhaps to add a component or product that you did not previously install), you must uninstall the current installation and re-install with the desired options. If you have a Luna Backup HSM connected to the client workstation, either disconnect it or stop the PEDclient service ("pedclient -mode stop" on page 330) before you proceed.

To uninstall the Luna HSM Client software

1. Run the **LunaHSMClient.exe** program again. Because the software is already installed on your computer, the following dialog is displayed, showing which components are currently installed (for this example, all Devices and all Features were previously installed):

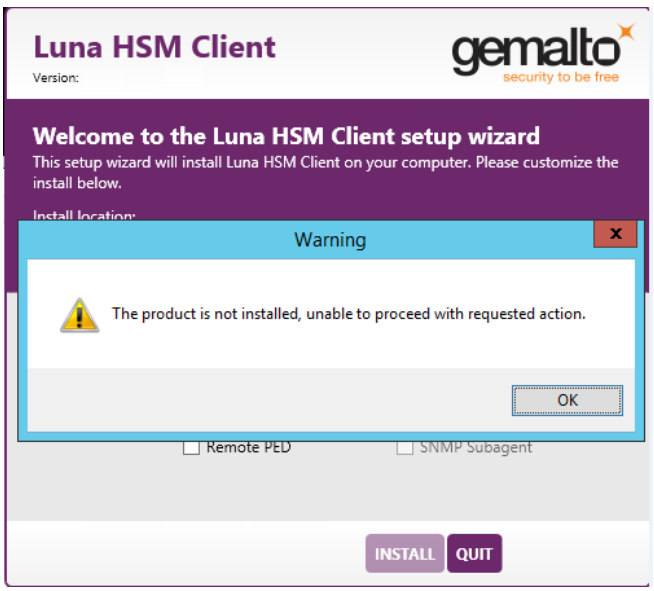
2. Click **Uninstall**. The client software is uninstalled.

3. When the uninstallation is complete, click **OK** to dismiss the operating system's confirmation dialog.

> **NOTE** You can also use **Programs and Features** in the Windows Control Panel to uninstall the client software.

Uninstall if not present

If the Luna HSM Client software has been uninstalled, and you launch the installer in uninstall mode, from the command line, the installer starts, looks for the installed software, fails to find it, and presents a Windows dialog to that effect.



If the Luna HSM Client software has been uninstalled, nothing related to the client appears in Windows Control Panel, so nothing exists to launch from that avenue.

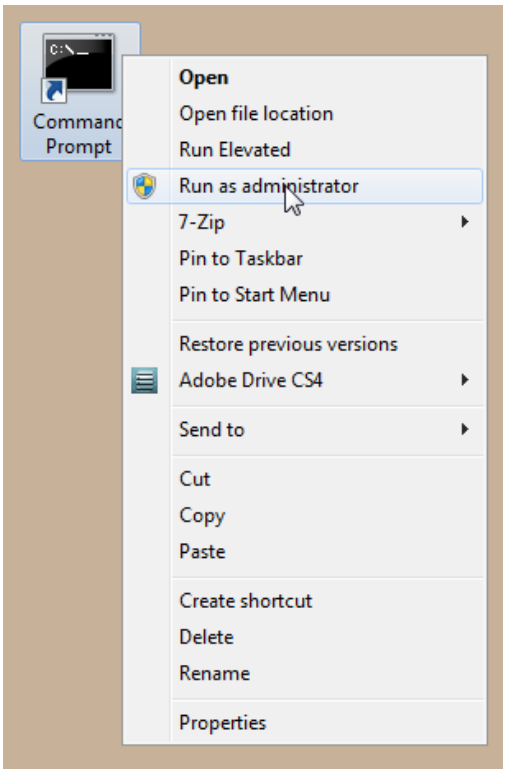
After Installation

Open a new command-line/console window to allow the library path to be found before you run LunaCM or other utilities that require the library.

Troubleshooting

If you are not the Administrator of the computer on which Luna HSM Client is being installed, or if the bundle of permissions in your user profile does not allow you to launch the installer with "Run as Administrator", then some services might not install properly. One option is to have the Administrator perform the installation for you.

Another approach might be possible. If you have sufficient elevated permissions, you might be able to right-click and open a Command Prompt window as Administrator.



If that option is available, then you can use the command line to move to the location of the **LunaHSMClient.exe** file and launch it there, which permits the needed services to load for PEDclient. See "Windows Luna HSM Client Installation" on page 21 for instructions on how to install the client software from the command line.

Linux Luna HSM Client Installation

You must install the Luna HSM Client software on each client workstation you will use to access a Luna HSM. This section describes how to install the client on a workstation running Linux, and contains the following topics:

Refer to the Customer Release Notes for a complete list of the supported Linux operating systems. These instructions assume that you have already acquired the Luna HSM Client software.

Prerequisites

Before starting the installation, ensure that you have satisfied the following prerequisites:

CentOS 8.4 Missing Dependency

Due to a missing dependency on CentOS 8.4 [specifically the symlink (libnsl.so.1) to libnsl was removed], when installing Luna HSM Client 10.5.0 or newer, you must install an additional rpm package first:

Run **yum install libnsl** before invoking the **install.sh** script.

Components Required to Build the PCIe Driver and the Backup HSM Driver

On Linux, the PCIe driver module (and optionally the Backup HSM driver) is built by the client as part of the installation if you choose to install the Luna PCIe HSM 7 component or the Backup HSM. To build the driver, the client requires the following items:

> Kernel headers for build
> **kernel-devel** package
> **rpmbuild** package
> C and C++ compilers
> **make** command
> **libelf-dev**, **libelf-devel**, or **elfutils-libelf-devel**

If any one of these items is missing, the driver build will fail and the client software will not be installed.

> **NOTE** The installed *kernel* and *kernel-devel* versions on the Client system must match, in order for the drivers to compile successfully. In general, if the versions do not match, or if you are not sure, use this command **yum install kernel-devel-`uname -r`** before installing Luna HSM Client. Note the required backticks, (the key to the left of the 1/! key on the keyboard) surrounding **`uname –r`** (or equivalent command **yum install kernel-devel-$(uname -r)**).
>
> To check installed versions related to the currently running kernel: **rpm -qa kernel * | grep $(uname -r)**.

For the PCIe, USB, or Backup HSM drivers to be available immediately after installation on RHEL, you must install **chkconfig** *before* the Luna HSM Client software. If this package is not installed, you must reboot the client computer after installing the client.

Debian Requires alien

The Luna HSM Client software is provided as RPM packages. If you are installing on a Debian system, you must have **alien** installed to allow the Luna HSM Client installation script to convert the RPM packages to DEB packages. The installation script will stop with a message if you attempt to install on a Debian system without **alien** installed. This applies to any other supported Debian-based Linux distribution, such as Ubuntu.

SUSE Linux on IBM PPC

JCE un-restriction files must be downloaded from IBM, not from SUN, for this platform. Attempting to use SUN JCE un-restriction files on IBM PowerPC systems with SUSE Linux causes signing errors.

Where to install, and SELinux

The instructions on this page assume that much of the installation goes into /usr. You can change that install location (see "Flexible Install paths" on page 41). There might be some interaction with SELinux that you would need to consider. Security Enhanced Linux or SELinux is a security mechanism built into the Linux kernel used by RHEL-based distributions. By default, in CentOS 8 and newer, SELinux is enabled and in enforcing mode.

SELinux adds an additional layer of security to the system by allowing administrators and users to control access to objects based on policy rules. SELinux policy rules specify how processes and users interact with each other as well as how processes and users interact with files. When there is no rule explicitly allowing access to an object, such as for a process opening a file, access is denied.

SELinux has three modes of operation:

- Enforcing: SELinux allows access based on SELinux policy rules.
- Permissive: SELinux only logs actions that would have been denied if running in enforcing mode. This mode is useful for debugging and creating new policy rules.
- Disabled: No SELinux policy is loaded, and no messages are logged.

So if, for example, your non- /usr installation completes uneventfully, but pedclient errors show up in the logs, then consider setting SELinux to "Permissive" mode. Or set explicit rules that will comply with SELinux's Enforcing mode.

NOTE **MutexFolder**: For Luna clients on Linux, the callback service (CBS) employed by pedclient originally placed mutex entries in /tmp. This was fine in most cases, but could be an issue if operating system services cleared the /tmp folder, causing the cbs process to stop. The workaround was to restart the service. A solution was provided that moved the mutex folder to /var/log. However, this was found to be an issue for installations by non-root users, where the service did not have permission to write into /var/log.

Beginning with Luna HSM Client 10.4.0, a chrystoki.conf entry "MutexFolder =" is added. If access to the default folder /tmp is not desired or is restricted, the MutexFolder= entry allows an administrator to specify an accessible folder.

```
Misc = {
…
  MutexFolder = /usr/lock;
}
```

However, the indicated folder must exist. If this is set to a non-existent folder, the service fails to start properly, such as in this example of logs for the cbs service:

(MutexFolder = /nosuchfolder/lock)

```
..  daemon info  systemd: Starting CallBack Server...
...  user notice  root: cbs started.
...  daemon info  cbs: Starting cbs:[  OK  ]#015LunaNamedSystemMutex: open()
failed: No such file or directory
...  daemon info  cbs: LOGGER_init failed
...  daemon info  cbs: Failed to initialize the logger.  Exiting.
...  user crit  pedClient: Failed to initialize the logger.  Exiting.
...  daemon info  systemd: Started CallBack Server.
```

About Installing the Luna HSM Client Software

It is recommended that you refer to the Customer Release Notes for any installation-related issues or instructions before installing the client software.

CAUTION! You must install the client software using root-level privileges. For security reasons, we recommend that you do not log in as root (or use su root) to run the installation script, but instead use the sudo command to run the installation script, as detailed below.

The installation script

The installation script is **install.sh** and is usually launched with **sh install.sh** followed by any options or parameters.

> interactive: **sh install.sh** [**-install_directory** <prefix>]

> all: **sh install.sh all** [**-install_directory** <prefix>]

> scriptable: **sh install.sh -p** [**network**|**pci**|**usb**|**backup**|**ped**] [**-c sdk**|**jsp**|**jcprov**|**snmp**]|**fmsdk**|**fm_tools** [**-install_directory** </usr>]

The options on the script are:

> device(s)

- **network** is the Luna Network HSM 7 (software only, no drivers)
- **pci** is the Luna PCIe HSM 7 (software plus PCI driver)

- **usb** is the Luna USB and Backup HSMs (software plus driver for the USB-connected HSMs)
- **backup** is software to enable Remote Backup
- **ped** is software for the Luna Remote PED

> components include the optional Software Development kit, Java providers, SNMP instance (not needed for Luna Network HSM 7 which has it built in), Functionality Module tools, and the Functionality Module SDK

Install.sh syntax and options:

```
[myhost]$ sh install.sh help

usage:
install.sh      - Luna HSM Client install through menu
install.sh help - Display scriptable install options
install.sh all  - Complete Luna HSM Client install

install.sh -p [network|pci|usb|backup|ped] [-c sdk|jsp|jcprov|snmp|fmsdk|fm_tools] [-install_
directory </usr>]

  -p <list of Luna products>
  -c <list of Luna components|all> - Optional. Default components are installed if not provided
  -install_directory <Defaults to /usr> - Optional. Sets the installation directory prefix.
Non-root install is restricted to installation of Luna Network HSM
product and Luna SDK, Luna JSP (Java) and Luna JCPROV (Java) components.

Luna products options
   network - Luna Network HSM
   pci     - Luna PCIe HSM
   usb     - Luna USB HSM
   backup  - Luna Backup HSM
   ped     - Luna Remote PED

Luna components options
   sdk     - Luna SDK
   jsp     - Luna JSP (Java) --> Luna Network HSM, Luna PCIe HSM and Luna USB HSM default
component
   jcprov  - Luna JCPROV (Java) --> Luna Network HSM, Luna PCIe HSM and Luna USB HSM default
component
   snmp    - Luna SNMP subagent
```

By default, the Client programs are installed in the **/usr/safenet/lunaclient** directory.

Flexible Install paths

An administrative (root) user, in charge of installing and uninstalling the software, has access wherever the installed material eventually resides. However, the operational, application-level use of Luna HSM Client might be assigned to a non-root user with constrained access and privileges. That non-root user might be a person or a departmental function or an application. By changing the install path to (for example) **%home/bigapplication/safenet/luna** you allow that non-root user access to tools and files for connecting to the HSM and using HSM partitions.

You can change the installation path for scriptable (non-interactive) installs by changing the prefix with the script option **-install_directory** <prefix>

The prefix, or major location is your choice, and replaces the /usr default portion. (See mention of SELinux, earlier on this page)

> **NOTE** This feature requires minimum Luna HSM Client 7.2.0.
> Avoid the use of space characters in directory names.

The script option **-install_directory** <prefix> is available for scriptable installation, where either "all" or a list of products and components is specified on the command line. The script option **-install_directory** <prefix> is not used with interactive installation; instead, you are prompted.

The **/safenet/lunaclient** portion is appended by the install script, and provides a predictable structure for additional subdirectories to contain certificate files, and optionally STC files.

Regardless of **-install_directory** <prefix> provided, some files are not affected by that option (for example, the Chrystoki.conf configuration file goes under /etc, service files need to be in the service directory expected by Linux in order to run at boot time, and so on).

> **TIP** Thales recommends verifying the integrity of the Luna HSM Client packages, by calculating their SHA256 hash values and comparing with the hash values posted on the Support Portal, before installing them on your client machines.
> You can use the sha256sum tool on Linux machines to calculate the SHA256 hash values.

Scripted or Unattended Installation

If you prefer to provide all installation options from the command-line (script), rather than interactively, do the following.

To install the Luna HSM Client software in non-interactive or scripted fashion on a Linux workstation

1. Ensure that you have **sudo** privileges on the client workstation.
2. Access the installation software:

   Copy or move the **.tar** archive to a suitable directory where you can untar the archive and extract the contents:

   tar xvf <filename>**.tar**
3. Go to the untarred directory for your operating system (**32** or **64**-bit):

   cd /<untarred_dir>**/<32/64>**
4. To see the syntax and all available options, run the command with **help**
5. To install the software, run the **install.sh** installation script with the options **-p** <list of Luna products> and **-c** <list of Luna components>.

   ```
   install.sh -p [network|pci|usb|backup|ped] [-c sdk|jsp|jcprov|snmp|fmsdk|fm_tools] [-install_directory </usr>]
   ```

   Be sure to include the -install_directory option.

> **NOTE** Following the "-c" option, you can provide a space-separated list of components to include in the installation. If JSP and JCProv are not explicitly listed, they are installed by default, but if one is explicitly listed, then only the listed component is included.
>
> If the SNMP component is selected, it works with Luna PCIe HSM, Luna USB HSM, and Luna Backup HSM products only.
>
> Following the "-p" option, you can provide a space-separated list of HSM products to include in the installation.

For scripted/automated installation, your script will need to capture and respond to the License Agreement prompt, and to the confirmation prompt. For example:

```
[myhost]$ sudo sh install.sh all

IMPORTANT: The terms and conditions of use outlined in the software
license agreement (Document #008-010005-001_053110) shipped with the product
("License") constitute a legal agreement between you and SafeNet Inc.
Please read the License contained in the packaging of this
product in its entirety before installing this product.

Do you agree to the License contained in the product packaging?

If you select 'yes' or 'y' you agree to be bound by all the terms
and conditions se out in the License.

If you select 'no' or 'n', this product will not be installed.

(y/n) y

Complete Luna Client will be installed. This includes Luna Network HSM,
Luna PCIe HSM, Luna USB HSM, Luna Backup HSM and Luna Remote PED.

Select 'yes' or 'y' to proceed with the install.

Select 'no' or 'n', to cancel this install.

Continue (y/n)?  y
```

Interrupting the Installation

Do not interrupt the installation script in progress, and ensure that your host computer is served by an uninterruptible power supply (UPS). If you press [CTRL] [C], or otherwise interrupt the installation (OS problem, power outage, other), some components will not be installed. It is not possible to resume an interrupted install process. The result of an interruption depends on where, in the process, the interruption occurred (what remained to install before the process was stopped).

As long as the cryptoki RPM package is installed, any subsequent installation attempt results in refusal with the message "A version of Luna HSM Client is already installed."

If components are missing or are not working properly after an interrupted installation, or if you wish to install any additional components at a later date (following an interrupted installation, as described), you would need to uninstall everything first. If **sh uninstall.sh** is unable to do it, then you must uninstall all packages manually.

To install the Luna HSM Client software interactively on a Linux workstation

1. Ensure that you have **sudo** privileges on the client workstation.
2. Access the installation software:

   Copy or move the **.tar** archive to a suitable directory where you can untar the archive and extract the contents:

   tar xvf <filename>**.tar**
3. Go to the untarred directory for your operating system (**32** or **64**-bit):

   cd /<untarred_dir>**/<32/64>**
4. To install the software, run the **install.sh** installation script. You can run the script in interactive mode, or you can script the installation, as described in "Scripted or Unattended Installation" on page 42.
   - To display the help, or a list of available installer options, type:

     sudo sh install.sh -? or **sudo sh install.sh help**
   - To install all available products and optional components, type:

     sudo sh install.sh all
   - To selectively install individual products and optional components, type the command without arguments:

     sudo sh install.sh

   > **NOTE** Do not interrupt the installation script in progress. An uninterruptible power supply (UPS) is recommended. See "Interrupting the Installation" on the previous page for more information.
5. Type **y** if you agree to be bound by the license agreement. You must accept the license agreement before you can install the software.
6. A list of installable Luna devices is displayed. Select as many as you require, by typing the number of each (in any order) and pressing **Enter**. As each item is selected, the list updates, with a * in front of any item that has been selected.

   This example shows items 1 and 3 have been selected, and item 4 is about to be selected. The selections work as a toggle - if you wish to make a change, simply type a number again and press **Enter** to de-select it.

```
Products
Choose Luna Products to be installed

    *[1]: Luna Network HSM

     [2]: Luna PCIe HSM

    *[3]: Luna USB HSM

     [4]: Luna Backup HSM

     [5]: Luna Remote PED

     [N|n]: Next

     [Q|q]: Quit
Enter selection: 4
```

When selection is complete, type **N** or **n** for "Next", and press **Enter**. The "Advanced" menu is displayed.

```
Advanced
Choose Luna Components to be installed

  [1]: Luna SDK

  [2]: Luna JSP (Java)

  [3]: Luna JCProv (Java)

  [4]: Luna SNMP subagent

  [5]: Luna Functionality Module Tools

  [6]: Luna Functionality Module Software Development Kit

  [B|b]: Back to Products selection

  [I|i]: Install

  [Q|q]: Quit

 Enter selection:
```

7. Select or de-select any additional items you want to install. Selected items are indicated with a *. Some items might be pre-selected to provide the optimum experience for the majority of customers, but you can change any selection in the list. When the Components list is adjusted to your satisfaction, press **Enter**.

   > **NOTE** The installer includes the Luna SNMP Subagent as an option. If you select this option, you will need to move the SafeNet MIB files to the appropriate directory for your SNMP application after installation is complete, and you will need to start the SafeNet subagent and configure it for use with your agent.
   >
   > **Luna SDK required with FMs** - If you choose the Functionality Module (FM) options, the interactive install.sh script populates the Luna SDK as well, because of dependencies in the FM samples. If you run the installer with command-line options (non-interactive), and you choose FM items without also choosing Luna SDK, the script just gives a warning and stops.
   >
   > ELDK (the Embedded Linux Development Kit) is installed with FMs - The ELDK package is installed as part of the FM SDK component, for Linux, and must reside at /opt/eldk-5.6. It is not relocatable.

   If the script detects an existing cryptoki library, it stops and suggests that you uninstall your previous Luna software before starting the Luna HSM Client installation again.

8. The system installs all packages related to the products and any optional components that you selected.
9. [Optional] For easy use of the Luna HSM Client tools, add their directories to the $PATH.
   a. Edit your system's **bash_profile** file using an editing tool.

      vi ~/.bash_profile

   b. Add the following lines to the end of the file:

      export PATH="$PATH:/usr/safenet/lunaclient/bin"

      export PATH="$PATH:/usr/safenet/lunaclient/sbin"

c. Source the updated **bash_profile**.

source ~/.bash_profile

Installing the Minimal Client Software

The minimal client package contains the minimum run-time libraries required for a cryptography application to connect to Luna Network HSM using PKCS#11 or Java APIs, or Functionality Modules on an FM-enabled HSM. Minimal client install is intended for container instances to interact with Luna HSM partitions or services.

Installing is as simple as copying the tarball and untarring it where you want it. A copy of a configured Chrystoki.conf file, along with client and server certificate files (and optionally, STC configuration files) must be available to any instance of Luna Minimal Client at run time.

See "Luna Minimal Client Install for Linux" on page 48 for a general example using Docker.

Controlling User Access to Your Attached HSMs and Partitions

By default, only the root user has access to your attached HSMs and partitions. You can specify a set of non-root users that are permitted to access your attached HSMs and partitions, by adding them to the **hsmusers** group.

> **NOTE** The Luna HSM Client software installation automatically creates the **hsmusers** group if one does not already exist on your system. The **hsmusers** group is retained when you uninstall the client software, allowing you to upgrade your client software while retaining your hsmusers group configuration.

> **TIP** Users on your system that are not members of **hsmusers** group are not able to see the slots/partitions when using lunacm, other Luna tools, or your applications. If you open lunacm, expecting to see one or more slots, and none are visible, check that your current user is a member of **hsmusers** before doing other troubleshooting.

Adding users to hsmusers group

To allow non-root users or applications access your attached HSMs and partitions, assign the users to the **hsmusers** group. The users you assign to the **hsmusers** group must exist on the client workstation. Users you add to the **hsmusers** group are able to access your attached HSMs and partitions. Users who are not part of the **hsmusers** group are not able to access your attached HSMs and partitions.

To add a user to hsmusers group

1. Ensure that you have **sudo** privileges on the client workstation.
2. Add a user to the hsmusers group:

   sudo gpasswd --add <username> **hsmusers**

   where <username> is the name of the user you want to add to the hsmusers group.

Removing users from hsmusers group

Should you wish to rescind a user's access to your attached HSMs and partitions, you can remove them from the **hsmusers** group.

> **NOTE** The user you delete will continue to have access to the HSM until you reboot the client workstation.

To remove a user from hsmusers group

1. Ensure that you have **sudo** privileges on the client workstation.
2. Remove a user from the hsmusers group:

   sudo gpasswd -d <username> **hsmusers**

   where <username> is the name of the user you want to remove from the **hsmusers** group. You must log in again to see the change.

Uninstalling the Luna HSM Client Software or Removing Components

You may need to uninstall the client software before upgrading to a new version, or if it is no longer required.

To uninstall the client software

1. Ensure that you have **sudo** privileges on the client workstation.
2. Go to the client installation directory:

   cd /usr/safenet/lunaclient/bin
3. Run the uninstall script:

   sudo sh uninstall.sh

> **CAUTION!** The **hsmusers** group is not removed when the client software is uninstalled. Should you install the client again on the same system, all users previously in the group will have access to your attached HSMs and partitions by default. You must remove users from the group if you want to restrict their access. See "Removing users from hsmusers group" on the previous page.

To remove individual components

To uninstall the JSP component or the SDK component, you must uninstall Luna HSM Client completely, then re-run the installation script without selecting the unwanted component(s).

Java

If you install the Luna Java Security Provider (JSP), refer to Luna JSP Overview and Installation for additional setup procedures for your operating system.

Modifying the Number of Luna Backup HSM Slots

By default, the Luna HSM Client allows for three slots reserved for each model of Luna Backup HSM. You can edit **Chrystoki.conf** to modify the number of reserved slots. See also "Configuration File Summary" on page 76.

To modify the number of reserved Backup HSM slots

1. Navigate to the **Chrystoki.conf** file and open in a text editor.

2. Add the following line(s) to the **CardReader** section of the file:
   - For Luna Backup HSM G5:

     LunaG5Slots = <value>**;**
   - For Luna Backup HSM 7:

     LunaG7Slots = <value>**;**

Effects of Kernel Upgrades

If you upgrade the Linux kernel after successful installation of Luna HSM Client, then you must install the kernel-headers for the new kernel and build the UHD, K6 and K7 drivers again for the new kernel. The new kernel takes effect after reboot.

To update the kernel and then bring the system back to readiness:

1. Install development tools if not already installed.
2. Update kernel if needed.
3. Reboot.
4. Install kernel-headers for the new kernel, example: **yum install kernel-headers-$(uname -r)**
5. Rebuild the drivers for the new kernel: **rpmbuild --rebuild uhd-7.3.0-165.src**
   Do the same for k6 and k7 drivers.

Troubleshooting

Problem #1A: No slots visible for Luna Network HSM 7 = user can't read certs directory.
Problem #1B: No slots visible for Luna PCIe HSM 7 or Luna USB HSM 7 = user can't read device (/dev/k7pf0, /dev/viper0, or /dev/lunauhd0).
Solution: You might have left a user out of **hsmusers** group, or you might have set an overly restrictive umask.

Problem #2: You receive the following error: **./setenv:24: = not found**

Solution: The setenv command is only supported while using bash.

Luna Minimal Client Install for Linux

Minimal client install is intended for container instances to interact with Luna HSM partitions, and contains the minimum run-time libraries required for a cryptography application to connect to Luna Network HSM 7 using PKCS#11 or Java APIs, in addition to some configuration tools. The Luna Minimal Install is provided as a tarball that you can unpack where desired, and choose the files that you need.

> **NOTE** This feature requires minimum Luna HSM Client 7.2.0.
>
> The minimal client does not have an installer, and **omits** drivers and other material, for backup HSMs, for Luna PED, or for the Luna PCIe HSM 7. For any of those, you would use the full Luna HSM Client installer.

Mandatory files for configuration and secure communication, where to get them and where to keep them

The Luna Minimal Client, when installed on minimalist or micro-service containers, requires that you have the appropriate files and folders available:

- Chrystoki.conf configuration file (includes settings, and pointers to resources),
- certificates folders (for secure communications protocols, NTLS or STC)
- libraries and plugins required for secure communications protocols.

The Luna Minimal Client tarball includes a "template" version of the Chrystoki.conf file that you can edit for any non-default settings needed by your application, and to reflect the actual paths to resources.

Alternatively, you might already have a configured Chrystoki.conf file that you can copy into the Docker container with the minimal client, or that you can leave at an external location that is mountable from within the Docker container.

Similarly, the Docker container with the minimal client must have access to the certificates (local host certificate, and certificates from any registered application partitions or Thales Data Protection on Demand (DPoD) Luna Cloud HSM services) for secure communication. Those can reside inside the container, or can reside on an external mountable drive - either way, the paths in the Chrystoki.conf file must point to their location.

Configure and link, inside your Docker container

You will need to untar the Minimal Client tarball in your container, or open it elsewhere and copy the desired files to your container.

If you already have a Chrystoki.conf file with most, or all, of your desired settings, you can copy it into the container and edit it manually.

If you do not have suitable Chrystoki.conf file, the minimal client tarball contains a config template file that you can modify with the configurator utility.

At the same time, you can create and exchange certificates by means of the included vtl utility. Ensure that the resulting certificates are pointed-to in Chrystoki.conf file. For example instructions, see "Installing Luna Minimal Client on Linux Using Docker" on page 53.

Configure and link, exterior to your Docker container

To configure Chrystoki.conf and to establish an NTLS or STC link outside your Docker container, for later use by one-or-more Docker containers, you can

- Untar the Luna Minimal Client tarball at the desired staging location, use configurator or manually edit the Chrystoki.conf file, and use vtl to establish the secure link to Luna Network HSM appliance.

  OR

- Install the full Luna HSM Client, and follow the instructions to create/update the Chrystoki.conf file, and create and exchange certificates for a secure link to a Network HSM appliance.

The above could be done before the Docker container is created, or after one exists.

Whether you elect to pre-configure externally, with a full Luna HSM Client Installation or with a copy of the Luna Minimal Client, or from inside each Docker container after it is created (and populated and configured with the Luna Minimal Client), two general networking approaches are possible:

[Network OPTION] Dynamic *private* IP address per container

If each Docker container (default) has a *private* IP address dynamically assigned to the container at run time:

- A single set of configuration file and certificate folders is needed, that will apply to any container within that hidden/translated subnet.
- Each container can mount the needed configuration from the one location on the host.
- Because all containers have the same IP address and appear as the same client, you must *disable* **ntls ipchecking** on the Luna Network HSM 7 appliance.

[Network OPTION] Unique *public* IP address per container

If a unique *public* IP address is assigned to each Docker container, visible to the Luna Network HSM 7 appliance:

- A separate NTLS configuration is performed, either externally on the host computer, for each proposed container IP, with the resulting configuration file and certificates folders saved to unique mountable locations on the host file system, OR configuration and certificate exchange is performed from the minimal client within each container after it is created.
- Each container gets its own configuration file and unique certificates whether mounted externally or residing inside the container.
- Because each container has its own unique public IP address, and is considered its own client, keep **ntls ipcheck** *enabled* on the Luna Network HSM 7 appliance.

Luna Cloud HSM

With the additional tools included in the minimal install archive, as of Luna HSM Client 10.2.0, the expanded minimal client has the needed tools for local (in-container) configuration. If you intend to connect with DPoD Luna Cloud HSM services, see "Create a Docker Container to Access a Luna Cloud HSM Service" on page 57 for additional steps.

Included in the Minimal Client

The following components are included in the Luna Minimal Client tar ball:

Component	Used or needed for...
JCPROV	
LunaClient-Minimal-<release_ version>**.x86_64/jcprov/jcprov.jar**	JCPROV jar file
LunaClient-Minimal-<release_ version>**.x86_64/jcprov/64/libjcprov.so**	JCPROV library
JSP	
LunaClient-Minimal-<release_ version>**.x86_64/jsp/LunaProvider.jar**	JSP jar file
LunaClient-Minimal-<release_ version>**.x86_64/jsp/64/libLunaAPI.so**	JSP library
LIBRARIES	

Component	Used or needed for...
LunaClient-Minimal-<release_ version>**.x86_64/libs/64/libCryptoki2.so**	Library to address cryptographic functions of the HSM
LunaClient-Minimal-<release_ version>**.x86_64/libs/64/libCryptoki2_ 64.so**	Symbolic link pointing to libCryptoki2.so, needed for FM hostapps compiled against libCryptoki2_64.so
LunaClient-Minimal-<release_ version>**.x86_64/libs/64/libethsm.so**	Library to interact with Functionality Modules
LunaClient-Minimal-<release_ version>**.x86_64/libs/64/libSoftToken.so**	Library for STC connection (alternative to NTLS)
LunaClient-Minimal-<release_ version>**.x86_64/libs/64/libcklog2.so**	Logging library - invoked by **vtl cklog enable** command to log commands before passing them to the cryptoki library and the HSM.
PLUG-INS	
LunaClient-Minimal-<release_ version>**.x86_64/plugins/libdpod.plugin**	Enable connection protocol with Luna Cloud HSM services (See also the related XTC and REST sections of chrystoki.conf file)
CONFIGURATION FILES	
LunaClient-Minimal-<release_ version>**.x86_64/Chrystoki-template.conf**	Chrystoki.conf template in case you don't already have a conf file.
LunaClient-Minimal-<release_ version>**.x86_64/openssl.cnf**	Configuration file for OpenSSL.
BINARIES/TOOLS	
LunaClient-Minimal-<release_ version>**.x86_64/bin/64/mkfm**	Allow client to connect to Functionality Modules (if you have installed any in the HSM)
LunaClient-Minimal-<release_ version>**.x86_64/bin/64/configurator**	Configuration file management tool
LunaClient-Minimal-<release_ version>**.x86_64/bin/64/ckdemo**	Demonstrates individual, atomic, PKCS#11 operations in the HSM
LunaClient-Minimal-<release_ version>**.x86_64/bin/64/lunacm**	Partition administration tool
LunaClient-Minimal-<release_ version>**.x86_64/bin/64/cmu**	Certificate Management Utility
LunaClient-Minimal-<release_ version>**.x86_64/bin/64/multitoken**	Perform multiple crypto commands on multiple slots
LunaClient-Minimal-<release_ version>**.x86_64/bin/64/pscp** **LunaClient-Minimal-**<release_ version>**.x86_64/bin/64/plink**	Used for One Step NTLS
LunaClient-Minimal-<release_ version>**.x86_64/bin/64/salogin**	Persistent application connection tool

Component	Used or needed for...
LunaClient-Minimal-<release_ version>**.x86_64/bin/64/vtl**	Configuration tool (certificate creation and exchange, registration of clients with partitions, logging, etc.)
LICENSE AGREEMENT	
LunaClient-Minimal-<release_ version>**.x86_64/008-010068-001 _EULA_HSM7_SW_revB.pdf**	
LunaClient-Minimal-<release_ version>**.x86_64/008-010068-001 _EULA_HSM7_SW_revB.txt**	

The configuration template file is included, in case you wish to populate it via direct editing (perhaps by script). Otherwise, a configuration file is created and modified when you perform a full (non-minimal) installation and configuration elsewhere, and you can simply have your Docker containers mount the external location to make use of the resulting chrystoki.conf file and certificate folders.

Installation Prerequisites

Ensure that you have the following prerequisites before installing the Luna Minimal Client:

> A Linux host system with Docker installed (see https://www.docker.com/ for Docker download and install)

> A copy of the Luna Minimal Client tarball package

> A Luna Network HSM 7 7.x appliance, already initialized and ready to use (or an account for access to DPoD Luna Cloud HSM services) -- perform any of the actions not already done:

- Configure the Luna Network HSM 7 network settings.
- Initialize the HSM.
- Create an application partition on the Luna Network HSM 7.
- Exchange host certificates between Luna HSM Client and the Luna Network HSM 7 and register each with the other (On the client side, add the Luna Network HSM 7's certificate to the server certs folder and to the CAFile. On the Luna Network HSM 7, register the client with lunash:>**client register**).
- Start the NTLS service on the appliance with lunash:>**service restart ntls**, and assign the client to the application partition with lunash:>**client assign partition**.
- On the client side, use LunaCM to configure the application partition (see "Initializing an Application Partition" on page 332), initializing the partition and creating roles as appropriate.
- After configuring Luna HSM Client on a host system, edit the Chrystoki.conf file for use in containers, as described in "Preparing the Configuration File for Use with Luna Minimal Client and Docker" below below.

> A working knowledge of Docker.

Preparing the Configuration File for Use with Luna Minimal Client and Docker

Make the following edits to the Chrystoki.conf file before using it in the containers:

1. Change all the library paths (for example LibUNIX64) to **/usr/local/luna/libs/64**

2. Change the certificate and client token paths to the the directory you are making available to the containers at run-time (for example **/usr/local/luna/config/certs**)

Entry in Chrystoki.conf	Value in the host system	Value in the containers
ClientPrivKeyFile	/usr/safenet/lunaclient/cert/client	/usr/local/luna/config/certs
ClientCertFile	/usr/safenet/lunaclient/cert/client	/usr/local/luna/config/certs
ServerCAFile	/usr/safenet/lunaclient/cert/server	/usr/local/luna/config/certs/
PartitionPolicyTemplatePath	/usr/safenet/lunaclient/data/partition_policy_templates	/usr/local/luna/config/ppt/partition_policy_templates
LibUNIX64	/usr/safenet/lunaclient/lib/libCryptoki2_64.so	/usr/local/luna/libs/64/libCryptoki2.so
ClientTokenLib	/usr/safenet/lunaclient/lib/libSoftToken.so	/usr/local/luna/libs/64/libSoftToken.so
SoftTokenDir	/usr/safenet/lunaclient/configData/token	/usr/local/luna/config/stc/token
ClientIdentitiesDir	/usr/safenet/lunaclient/data/client_identities	/usr/local/luna/config/stc/client_identities
PartitionIdentitiesDir	/usr/safenet/lunaclient/data/partition_identities	/usr/local/luna/config/stc/partition_identities
ToolsDir	/usr/safenet/lunaclient/bin	/usr/local/luna/bin/64
SSLConfigFile	/usr/safenet/lunaclient/bin/openssl.cnf	/usr/local/luna/openssl.cnf

Ready to Install Minimal Client

For detailed instructions, see "Installing Luna Minimal Client on Linux Using Docker" below.

For additional instructions on using the minimal client with Functionality Modules, see "Create a Luna HSM Client Docker image for use with Functionality Modules" on page 58.

For additional instructions on using the minimal client with DPoD Luna Cloud HSM services, see "Create a Docker Container to Access a Luna Cloud HSM Service" on page 57.

Installing Luna Minimal Client on Linux Using Docker

The following procedure allows you to install the Luna Minimal Client in a Docker container on Linux, so that applications in that container can access Luna Network HSM 7 partitions. For an overview description of Luna Minimal Client and its prerequisites, see "Luna Minimal Client Install for Linux" on page 48.

> **NOTE** This feature requires minimum Luna HSM Client 7.2.0.
>
> If SELinux is enabled in Enforcing mode, you must assign proper permissions to any container that needs to access the config directory.

To install the Luna Minimal Client software on a Linux 64-bit Docker instance:

This example uses NTLS. The use of STC is optional. This example is based on CentOS 7; other operating systems might require adjustments to the commands and to the docker file.

1. Create a directory. In this example:

```
$HOME/luna-docker
```

The name is not important, only that you use it consistently.

2. Create the following subdirectories under that first directory:

```
$HOME/luna-docker/config
$HOME/luna-docker/config/certs
```

additionally, if you are configuring STC:

```
$HOME/luna-docker/config/stc
$HOME/luna-docker/config/stc/client_identities
$HOME/luna-docker/config/stc/partition_identities
$HOME/luna-docker/config/stc/token/001
```

and create an empty file:

- for Luna HSM Firmware 7.4.2 and older:

```
$HOME/luna-docker/config/stc/token/001/token.db
```

- for Luna HSM Firmware 7.7.0 and newer:

```
$HOME/luna-docker/config/stc/token/001/token_v2.db
```

The contents of the config directory are needed by the Docker containers.

3. Copy the Luna Minimal Client tarball to **$HOME/luna-docker**.
4. Untar the Luna Minimal Client tarball.

>tar -xf $HOME/luna-docker/LunaClient-Minimal-<release_version>**.x86_64.tar -C $HOME/luna-docker**

5. Copy the Chrystoki.conf file from the Minimal Client directory to **$HOME/luna-docker/config**.

>cp $HOME/luna-docker/LunaClient-Minimal-<release_version>**.x86_64/Chrystoki-template.conf $HOME/luna-docker/config/Chrystoki.conf**

6. Define the following environment variable:

>export ChrystokiConfigurationPath=$HOME/luna-docker/config

7. [Optional] If you choose to use STC, review the Luna Network HSM 7 documentation and modify the following instructions. The goal is to have an HSM partition created and registered with the full Luna HSM Client before you create the Docker image and containers.
8. Update the Chrystoki.conf file paths so the tools work as expected

>MIN_CLIENT_DIR=$HOME/luna-docker/LunaClient-Minimal-<release_version>**.x86_64**

>$MIN_CLIENT_DIR/bin/64/configurator setValue -s Chrystoki2 -e LibUNIX -v $MIN_CLIENT_DIR/libs/64/libCryptoki2.so

>$MIN_CLIENT_DIR/bin/64/configurator setValue -s Chrystoki2 -e LibUNIX64 -v $MIN_CLIENT_DIR/libs/64/libCryptoki2_64.so

>$MIN_CLIENT_DIR/bin/64/configurator setValue -s Misc -e ToolsDir -v $MIN_CLIENT_DIR/bin/64

>$MIN_CLIENT_DIR/bin/64/configurator setValue -s "LunaSA Client" -e SSLConfigFile -v $MIN_CLIENT_DIR/openssl.cnf

>$MIN_CLIENT_DIR/bin/64/configurator setValue -s "LunaSA Client" -e ClientPrivKeyFile -v $HOME/luna-docker/config/certs/dockerlunaclientKey.pem

>**$MIN_CLIENT_DIR/bin/64/configurator setValue -s "LunaSA Client" -e ClientCertFile -v $HOME/luna-docker/config/certs/dockerlunaclient.pem**

>**$MIN_CLIENT_DIR/bin/64/configurator setValue -s "LunaSA Client" -e ServerCAFile -v $HOME/luna-docker/config/certs/CAFile.pem**

>**$MIN_CLIENT_DIR/bin/64/configurator setValue -s "Secure Trusted Channel" -e ClientTokenLib -v $MIN_CLIENT_DIR/libs/64/libSoftToken.so**

>**$MIN_CLIENT_DIR/bin/64/configurator setValue -s "Secure Trusted Channel" -e SoftTokenDir -v $HOME/luna-docker/config/stc/token**

>**$MIN_CLIENT_DIR/bin/64/configurator setValue -s "Secure Trusted Channel" -e ClientIdentitiesDir -v $HOME/luna-docker/config/stc/client_identities**

>**$MIN_CLIENT_DIR/bin/64/configurator setValue -s "Secure Trusted Channel" -e PartitionIdentitiesDir -v $HOME/luna-docker/config/stc/partition_identities**

9. Create a Luna HSM Client certificate for the Docker containers.

   >**$MIN_CLIENT_DIR/bin/64/vtl createCert -n** <cert_name>

10. Copy the client certificate to the Luna Network HSM 7 appliance.

    >**scp $HOME/luna-docker/config/certs/**<cert_name>**.pem admin@**<Network_HSM_IP>**:**

11. Copy the appliance server certificate (**server.pem**) to **$HOME/luna-docker/config/certs**

    >**scp admin@**<Network_HSM_IP>**:server.pem $HOME/luna-docker/config/certs**

12. Register the appliance server certificate with the Client.

    >**$MIN_CLIENT_DIR/bin/64/vtl addServer -c $HOME/luna-docker/config/certs/server.pem -n** <Network_HSM_IP>

13. Connect via SSH to the Luna Network HSM 7 appliance and log in to LunaSH.

    >**ssh admin@**<Network_HSM_IP>

    From this point it is assumed that the appliance already has a valid server.pem. If not, then generate a new one via the lunash:> **sysconf regenCert** command.

14. Create a partition, if one does not already exist on the HSM.

    lunash:>**partition create -partition** <partition_name>

    The HSM must already have been initialized, via lunash:> **hsm init** command, and the HSM SO must log in via **hsm login** command, for application partitions to be created.
    For HSMs at Luna HSM Firmware 7.7.0 or newer, any new partition defaults to version zero (V0), unless you specify V1 in the **partition create** command.

15. Register the full Luna HSM Client with the appliance, and assign the partition to the client.

    lunash:> **client register -client** <client_name> {**-ip** <client_IP> | **-hostname** <client_hostname>}

    lunash:> **client assignpartition -client** <client_name> **-partition** <partition_name>

    lunash:> **ntls ipcheck disable**

    lunash:> **exit**

16. On the Client workstation, run LunaCM, set the active slot to the registered partition, and initialize it.

    >**$MIN_CLIENT_DIR/bin/64/lunacm**

lunacm:> **slot set -slot** <slotnum>

lunacm:> **partition init -label** <partition_label>

lunacm:> **exit**

17. Update the paths of the libraries, certs and general fields to their future Docker image locations within the **$ChrystokiConfigurationPath/Chrystoki.conf**.

    >sed -i -e 's#'$HOME'/luna-docker/config#/usr/local/luna/config#g' -e 's#'$HOME'/luna-docker/LunaClient-Minimal-\([0-9\.-]\+\)x86_64#/usr/local/luna#g' $ChrystokiConfigurationPath/Chrystoki.conf

Create a Luna HSM Client Docker image

The minimal client tarball includes files necessary for basic operation, and some tools; copy any additional files you want to include in the docker image to **$HOME/luna-docker/**. This example includes the entire Luna Minimal Client.

1. Create a file named Dockerfile with the following contents:

```
FROM ubuntu:xenial
#FROM centos:centos7

ARG MIN_CLIENT
COPY $MIN_CLIENT.tar /tmp
RUN mkdir -p /usr/local/luna
RUN tar xvf /tmp/$MIN_CLIENT.tar --strip 1 -C /usr/local/luna
ENV ChrystokiConfigurationPath=/usr/local/luna/config
ENV PATH="/usr/local/luna/bin/64:${PATH}"

# The package below is necessary for One-Step NTLS if you want to setup NTLS within the
Docker container.
# The only requirement beyond glibc.i686 (required by plink and pscp) would be a configured
Chrystoki.conf
# The minimal client documentation section 8 has example commands, you should modify the
value parameter ("-v")
#    to point to desired files/directories.
# One-Step NTLS uses the section “Misc” entry “ToolsDir” to find the plink/pscp binaries,
# The Chrystoki.conf needs the following entries to be updated for One-Step NTLS to work:
# Section         | Entry
# -------------------------
# Chrystoki2      | LibUNIX
# Chrystoki2      | LibUNIX64
# Misc            | ToolsDir
# "LunaSA Client" | SSLConfigFile
# "LunaSA Client" | ClientPrivKeyFile
# "LunaSA Client" | ClientCertFile
# "LunaSA Client" | ServerCAFile
# Syntax: configurator setValue –s <Section> -e <Entry> -v <value>
# Example: configurator setValue -s Misc -e ToolsDir -v /usr/local/luna/bin/64
# Ubuntu:
#RUN dpkg --add-architecture i386
#RUN apt-get update
#RUN apt-get -y install libc6:i386
# Centos:
#RUN yum install -y glibc.i686
```

```
ENTRYPOINT  /bin/bash
#End of the Dockerfile
```

2. Build a Docker image.

   >docker build . --build-arg MIN_CLIENT=LunaClient-Minimal-<release_version>**.x86_64 -t lunaclient-image**

3. Use the following command to verify the Docker image has been created:

   >docker images

Run the Docker container

1. Make the contents of the config directory available to the Containers when you create them, by mounting the config directory as a volume.

   >docker run -it --name lunaclient -v $PWD/config:/usr/local/luna/config lunaclient-image

2. From the Docker container, verify that the container has a connection to the Luna Network HSM 7 partition.

Functionality Modules (FMs) with Luna Minimal Client

To use FMs with the minimal client, see "Create a Luna HSM Client Docker image for use with Functionality Modules" on the next page.

Thales Data Protection on Demand Luna Cloud HSM Service with Luna Minimal Client

To connect to Thales Data Protection on Demand (DPoD) Luna Cloud HSM services with the minimal client, see "Create a Docker Container to Access a Luna Cloud HSM Service" below.

Create a Docker Container to Access a Luna Cloud HSM Service

This section describes the steps to connect to a Luna Cloud HSM service by running the Luna Client in a docker container. These steps require Docker and client version 10.7.2 or higher.

1. Acquire a new DPOD on Demand service from Thales Data Protection on Demand.
2. Download the Luna Cloud HSM service client configuration zip file using the DPoD user interface or API.
3. Create a new Dockerfile using the following template:

```
FROM ubuntu:20.04

RUN apt-get update && \
apt install -y unzip && \
apt install -y libcap-dev && \
#libcap required or lunacm throws libcap.so.2 not found error
apt-get install -y ca-certificates && \
#add ca-certificates to use system CA Bundle
update-ca-certificates

RUN mkdir -p /usr/local/dpodclient

#NOTE - The name of the zip file below should match the name of the downloaded file from the
web portal
```

```
COPY setup-myclient.zip /usr/local/dpodclient

RUN unzip /usr/local/dpodclient/setup-myclient.zip -d /usr/local/dpodclient
RUN tar xvf /usr/local/dpodclient/cvclient-min.tar -C /usr/local/dpodclient

WORKDIR "/usr/local/dpodclient"
```

4. Build a Docker image using the Dockerfile as context.

```
docker build -t myimage .
```

5. Run the Docker image.

```
docker run -it --entrypoint=./bin/64/lunacm  myimage
```

> **NOTE** Using Luna Cloud HSM 10.7.2 or higher, users are no longer required to run **setenv** to configure the client to connect to the Cloud HSM Service. However, **setenv** may still be used to configure the client for hybrid use cases or integrations where setting the ChrystokiConfigurationPath is required.
> The **lunacm** command will only be able to run from root of client directory if **setenv** is not executed.

Create a Luna HSM Client Docker image for use with Functionality Modules

The example "Installing Luna Minimal Client on Linux Using Docker" on page 53 uses the Luna Minimal Client to gain connection to a Luna Network HSM 7 partition. This section explores some additional steps to sign a Functionality Module (FM) from a Docker container, and also execute a Host Application in order to communicate with the Functionality Module in the Luna Network HSM 7.

> **NOTE** This feature requires minimum Luna HSM Client 7.4.0.

FMs consist of two components - the FM itself, that resides in the HSM, extending its functionality, and the Host Application component that resides with the clients that need to connect with that FM.

Due to the size of the FM SDK and ELDK, those have not been included in the Minimal Client as they would greatly expand the size of the minimal client. The assumption is that you installed the full Luna HSM Client with HSM Software Development Kit, FM Software Development Kit and other components, and then created and compiled your Functionality Modules elsewhere, and that you would be importing FM components and using FMs, but not developing and compiling them inside a Docker container.

But the above-mentioned use-cases should help in common tasks such as signing Functionality Modules or Communicating with them via Host Applications.

1. On a Linux client with the Functionality Module SDK Component installed (which also installs the Embedded Linux Development Kit (ELDK)), compile the sample FMs and Host application binaries.

   >make -C /usr/safenet/lunafmsdk/samples/pinenc all

   >make -C /usr/safenet/lunafmsdk/samples/skeleton all

   >make -C /usr/safenet/lunafmsdk/samples/wrap-comp all

2. Create a directory on the shared volume to store the Host applications and unsigned FM binaries.

>**mkdir $HOME/luna-docker/config/fm**

3. Copy the generated files over.

   >**cp /usr/safenet/lunafmsdk/samples/pinenc/fm/bin-ppc/* $HOME/luna-docker/config/fm/**

   >**cp /usr/safenet/lunafmsdk/samples/skeleton/fm/bin-ppc/* $HOME/luna-docker/config/fm/**

   >**cp /usr/safenet/lunafmsdk/samples/wrap-comp/fm/bin-ppc/* $HOME/luna-docker/config/fm/**

   >**cp /usr/safenet/lunafmsdk/samples/pinenc/host/output/bin/* $HOME/luna-docker/config/fm/**

   >**cp /usr/safenet/lunafmsdk/samples/skeleton/host/output/bin/* $HOME/luna-docker/config/fm/**

   >**cp /usr/safenet/lunafmsdk/samples/wrap-comp/host/output/bin/* $HOME/luna-docker/config/fm/**

4. Go back to the Docker container. If it is stopped you must start the container first.

   >**docker ps -a**

   >**docker star**t <container_id>

   >**docker attach** <container_id>

5. If you have not already done so, enable **LoginAllowedOnFMEnabledHSMs=1** in the Chrystoki.conf file, else you will be prompted on your first **partition init** or **role login** attempt to do so in LunaCM.

   >**configurator setValue -s Misc -e LoginAllowedOnFMEnabledHSMs -v 1**

6. Ensure that the “Partition SO” and “Crypto Officer” users are initialized via LunaCM (see "Initializing an Application Partition" on page 332 and "Initializing Crypto Officer and Crypto User Roles for an Application Partition" on page 368).

7. Generate a key pair and Self-Signed Certificate, then sign the FM binary using **mkfm** and export the Self-Signed Certificate.

   >**cmu generatekeypair -labelpublic=fmpub -labelprivate=fmpri -sign=1 -verify=1 -keytype=rsa -mech=pkcs -publicexponent=3 -modulusbits=2048 -slot** <slotnum>

   >**cmu list -slot** <slotnum>

   >**cmu selfsigncertificate -publichandle=**<public_key_handle**> -privatehandle=**<private_key_handle> **-label=FmSign -serialnumber=1 -cn=FmSign -startdate=20180606 -enddate=20201231 -slot** <slotnum>

   >**mkfm -f /usr/local/luna/config/fm/pinenc.bin -o /usr/local/luna/config/fm/pinenc.fm -kSLOTID=**<slotnum>**/fmpri**

   >**mkfm -f /usr/local/luna/config/fm/skeleton.bin -o /usr/local/luna/config/fm/skeleton.fm -kSLOTID=**<slotnum>**/fmpri**

   >**mkfm -f /usr/local/luna/config/fm/wrap-comp.bin -o /usr/local/luna/config/fm/wrapcomp.fm -kSLOTID=**<slotnum>**/fmpri**

   >**cmu export -slot** <slotnum> **-label FmSign -outputfile=/usr/local/luna/config/fm/FmSign.cert**

8. Copy the signed FMs and Self-Signed Certificate to the Luna Network HSM 7 appliance. If your Docker container supports scp, then use that. If you’ve uncommented the pre-requisites in the Dockerfile regarding **pscp** and **plink**, then you could use that as well. If the above two scenarios are not applicable, you can always copy the files from the shared fm directory volume:

   >**pscp $HOME/luna-docker/config/fm/pinenc.fm admin**@<Network_HSM_IP>:

   >**pscp $HOME/luna-docker/config/fm/skeleton.fm admin@**<Network_HSM_IP>:

>**pscp $HOME/luna-docker/config/fm/wrapcomp.fm admin@**<Network_HSM_IP>:

>**pscp $HOME/luna-docker/config/fm/FmSign.cert admin@**<Network_HSM_IP>:

9. Connect via SSH to the Luna Network HSM 7 appliance and log in to LunaSH.

   >**ssh admin@**<Network_HSM_IP>

10. Login as the HSM Admin (SO), then load the Functionality Modules.

    lunash:> **hsm login**

    lunash:> **hsm fm load -fmFile pinenc.fm -certFile FmSign.cert**

    lunash:> **hsm fm load -fmFile skeleton.fm -certFile FmSign.cert**

    lunash:> **hsm fm load -fmFile wrapcomp.fm -certFile FmSign.cert**

    lunash:> **hsm fm status**

11. If the **hsm fm status** command, in the previous step, mentioned “reboot HSM to activate” on any of the FMs, then you must reboot the HSM. Upon restarting the HSM, SO login status will be reset, thus you will have to login as SO later.

    lunash:> **hsm restart**

    lunash:> **hsm login**

12. Activate Secure Memory File System (SMFS); you must be logged in as the HSM Admin. If you check the status of the FMs, they should all be “Enabled” status now.

    lunash:> **hsm fm smfs activate**

    lunash:> **hsm fm status**

13. Verify that the Host Application can interact with the FM. If you have trouble loading the shared libraries, you can set the LD_LIBRARY_PATH environment variable.

    >**export LD_LIBRARY_PATH="/usr/local/luna/libs/64"**

    >**/usr/local/luna/config/fm/pinenctest -s**<slotnum> **gen**

    >**/usr/local/luna/config/fm/pinenctest -d**<slotnum> **test**

    >**/usr/local/luna/config/fm/skeleton -s**<slotnum> **-t "Hello all"**

    >**/usr/local/luna/config/fm/wrapcomptest -s**<slotnum>

Solaris Luna HSM Client Installation

> **NOTE** Solaris Client was not included with Luna HSM Client 10.3.0 and newer. To use Luna HSM with this operating system, use a different Client release version.

These instructions assume that you have already acquired the Luna HSM Client software, in the form of a downloaded .tar archive.

> **NOTE** Check the Customer Release Notes for Luna HSM Client versions that can be installed on Solaris.

You must install the Luna HSM Client software on each client workstation you will use to access a Luna HSM. This section describes how to install the client on a workstation running Solaris, and contains the following topics:

Applicability to specific versions of Solaris is summarized in the Customer Release Notes.

> **NOTE** Before installing a Luna system, you should confirm that the product you have received is in factory condition and has not been tampered with in transit. Refer to the Startup Guide included with your product shipment. If you have any questions about the condition of the product that you have received, contact Thales Support.

Each computer that connects to the Luna Network HSM 7 appliance as a client must have the cryptoki library, the vtl client shell and other utilities and supporting files installed.

Each computer that contains a Luna PCIe HSM 7, or is connected to a Luna USB HSM 7, must have the cryptoki library and other utilities and supporting files installed.

> **NOTE** This example shows all the Luna HSM Client products and components. Some items are not supported on all operating systems and therefore do not appear as you proceed through the installation script.

Prerequisites

Before starting the installation, ensure that you have satisfied the following prerequisites:

Random Number Generator (RNG) or Entropy Gathering Daemon (EGD)
Ensure that you have a Random Number Generator (RNG) or Entropy Gathering Daemon (EGD) on your system in one of the following locations:

- /dev/egd-pool
- /etc/egd-pool,
- /etc/entropy
- /var/run/egd-pool

RNG/EGD

Cryptographic algorithms, including those that assure the security of communication – such as in OpenSSL and other protocols – depend upon random numbers for the creation of strong keys and certificates. A readily available source of random data is the entropy that exists in complex computer processes. Utilities exist for every operating system, to gather bits of system entropy into a pool, which can then be used by other processes.

Windows and Linux have these installed by default. Other systems might not. See your system administrator.

Entropy Pool

In the case of Luna Network HSM 7, the Luna HSM Client administration tool (**vtl**) expects to find a source of randomness at **/dev/random**. If one is not found, **vtl** fails, because the link cannot be secured from the Client end.

If your system does have an entropy pool, but the random number generator (RNG) is not in the expected place, then you can create a symbolic link between the actual location and one of the following:

- /dev/random
- /dev/egd-pool
- /etc/egd-pool
- /etc/entropy
- /var/run/egd-pool

If your system does not have an entropy-gathering daemon or random number generator, please direct your system administrator to install one, and point it to one of the named devices.

Installing the Luna HSM Client Software

> **TIP** Thales recommends verifying the integrity of the Luna HSM Client packages, by calculating their SHA256 hash values and comparing with the hash values posted on the Support Portal, before installing them on your client machines.
>
> You can use the sha256sum tool on Linux machines to calculate the SHA256 hash values.

It is recommended that you refer to the Customer Release Notes for any installation-related issues or instructions before you begin the following software installation process.

> **CAUTION!** You must be logged in as **root** when you run the installation script.

By default, the Client programs are installed in the **/opt/safenet/lunaclient/bin** directory.

To install the Luna HSM Client software on a Solaris workstation

1. Log on to the client system, open a console or terminal window, and use **su** to gain administrative permissions for the installation.
2. Access the Luna HSM Client software:
   a. Copy or move the **.tar** archive to a suitable directory where you can untar the archive and launch the installation script.
   b. Extract the contents from the archive:

      tar xvf <filename>**.tar**
3. Go to the install directory for your architecture:

   NOTE Luna HSM Client 10.1.0 and newer includes libraries for 64-bit operating systems only.

Architecture	Path
Solaris Sparc 32-bit	LunaClient_7.X.0_SolarisXXSparc/32
Solaris Sparc 64-bit	LunaClient_7.X.0_SolarisXXSparc/64
Solaris x86 32-bit	LunaClient_7.X.0_SolarisXXx86/32
Solaris x86 64-bit	LunaClient_7.X.0_SolarisXXx86/64

4. To see the help, or a list of available installer options, type:

   sh install.sh -? or **sh install.sh --help**

   To install all available products and optional components, type:

   sh install.sh all

   To selectively install individual products and optional components, type the command without arguments:

   sh install.sh
5. Type **y** if you agree to be bound by the license agreement.
6. A list of installable Luna products is displayed (might be different, depending on your platform). Select as many as you require, by typing the number of each (in any order) and pressing **Enter**. As each item is selected, the list updates, with a "*" in front of any item that has been selected. The following example shows that items 1 and 3 have been selected, and item 4 is about to be selected.

```
Products
Choose Luna Products to be installed
    *[1]: Luna Network HSM
     [2]: Luna PCIe HSM
    *[3]: Luna USB HSM
     [4]: Luna Backup HSM
     [N|n]: Next
     [Q|q]: Quit
Enter selection: 4
```

7. When the selection is complete, type **N** or **n** for "Next", and press **Enter**. If you wish to make a change, simply type a number again and press **Enter** to de-select a single item.

8. The next list is titled "Advanced" and includes additional items to install. Some items might be pre-selected to provide the optimum Luna HSM experience for the majority of customers, but you can change any selection in the list. When the Components list is adjusted to your satisfaction, press **Enter**.

   > **NOTE** The installer includes the Luna SNMP Subagent as an option. If you select this option, you will need to move the SafeNet MIB files to the appropriate directory for your SNMP application after installation is complete, and you will need to start the SafeNet subagent and configure for use with your agent.

9. If the script detects an existing cryptoki library, it stops and suggests that you uninstall your previous Luna software before starting the Luna HSM Client installation again.
10. The system installs all packages related to the products and any optional components that you selected.
11. Although FMs are supported on Linux and Windows clients only in this release, the FM architecture requires a configuration file setting to allow partition login on an FM-enabled HSM. If the HSM you will be using with this client is FM-enabled (see Preparing the Luna Network HSM 7 to Use FMs for more information), you must add the following entry to the [Misc] section of the Chrystoki.conf file:

    [Misc]

    LoginAllowedOnFMEnabledHSMs=1

    > **NOTE** As a general rule, do not modify the Chrystoki.conf/crystoki.ini file, unless directed to do so by Thales Technical Support. If you do modify the file, never insert TAB characters - use individual space characters. Avoid modifying the PED timeout settings. These are now hardcoded in the appliance, but the numbers in the Chrystoki.conf file must match.

Uninstalling the Luna HSM Client Software

1. **cd /opt/safenet/lunaclient/bin**
2. **sh uninstall.sh**

Java

If you install the Luna Java Security Provider (JSP), refer to Luna JSP Overview and Installation for additional setup procedures for your operating system.

Scripted or Unattended Installation

If you prefer to run the installation from a script, rather than interactively, run the command with the options **-p** <list of Luna products> and **-c** <list of Luna components>. To see the syntax, run the command with **help** like this:

```
[myhost]$ sudo sh install.sh help
[sudo] password for fred

At least one product should be specified.

usage:
        install.sh      - Luna Client install through menu
        install.sh help - Display scriptable install options
```

```
        install.sh all  - Complete Luna Client install

        install.sh -p [sa|pci|g5|rb] [-c sdk|jsp|jcprov|ldpc|snmp]

        -p <list of Luna products>
        -c <list of Luna components>  - Optional. All components are installed if not provided

Luna products options
   sa     - Luna Network HSM
   pci    - Luna PCIe HSM
   g5     - Luna USB HSM
   rb     - Luna Backup HSM

Luna components options
   sdk    - Luna SDK
   jsp    - Luna JSP (Java)
   jcprov - Luna JCPROV (Java)
   snmp   - Luna SNMP subagent


[myhost]$
```

For scripted/automated installation, your script will need to capture and respond to the License Agreement prompt, and to the confirmation prompt. For example:

```
[myhost]$ sudo sh install.sh all

IMPORTANT: The terms and conditions of use outlined in the software
license agreement (Document #008-010005-001_053110) shipped with the product
("License") constitute a legal agreement between you and SafeNet Inc.
Please read the License contained in the packaging of this
product in its entirety before installing this product.

Do you agree to the License contained in the product packaging?

If you select 'yes' or 'y' you agree to be bound by all the terms
and conditions se out in the License.

If you select 'no' or 'n', this product will not be installed.

(y/n) y

Complete Luna Client will be installed. This includes Luna Network HSM,
Luna PCIe HSM, Luna USB HSM AND Luna Backup HSM.

Select 'yes' or 'y' to proceed with the install.

Select 'no' or 'n', to cancel this install.

Continue (y/n)?  y
```

Interrupting the installation

Do not interrupt the installation script in progress, and ensure that your host computer is served by an uninterruptible power supply (UPS). If you press [Ctrl] [C], or otherwise interrupt the installation (OS problem, power outage, other), some components will not be installed. It is not possible to resume an interrupted install process. The result of an interruption depends on where, in the process, the interruption occurred (what remained to install before the process was stopped).

As long as the cryptoki package is installed, any subsequent installation attempt results in refusal with the message "A version of Luna Client is already installed." Removing the library allows the script to clean up remaining components, so that you can install again.

What to do if installation is incomplete or damaged

1. If SNFTlibcryptoki has been installed, uninstall it manually.
2. Run the Client install script again. Now that SNFTlibcryptoki is removed, the install script removes any stray packages and files.
3. Install again, to perform a clean installation.

AIX Luna HSM Client Installation

> **NOTE** AIX Client was not included in Luna HSM Client 10.3.0 to Luna HSM Client 10.5.0. To use Luna HSM with this operating system, use a different Client release version.

These instructions assume that you have already acquired the Luna HSM Client software, usually in the form of a downloaded .tar archive.

You must install the Luna HSM Client software on each client workstation you will use to access a Luna HSM. This section describes how to install the client on a workstation running AIX, and contains the following topics:

Applicability to specific versions of AIX is summarized in the Customer Release Notes for the current release.

> **NOTE** Before installing a Luna HSM, you should confirm that the product you have received is in factory condition and has not been tampered with in transit. Refer to the Content Sheet included with your product shipment. If you have any questions about the condition of the product that you have received, please contact Thales Technical Support.

Prerequisites

Each computer that connects to the Luna Network HSM 7 appliance as a Client must have the cryptoki library, the vtl client shell and other utilities and supporting files installed. Each computer that is connected to a Luna Remote Backup HSM must have the cryptoki library and other utilities and supporting files installed - in this case, that would be a Windows or Linux computer with the "Backup" option chosen when Luna HSM Client software is installed.

> **TIP** Thales recommends verifying the integrity of the Luna HSM Client packages, by calculating their SHA256 hash values and comparing with the hash values posted on the Support Portal, before installing them on your client machines.
>
> You can use the sha256sum tool on Linux machines to calculate the SHA256 hash values.

Installing the Client Software

Check the Customer Release Notes for any installation-related issues or instructions before you begin the following software installation process.

To install the Luna HSM Client software on AIX

1. Log on to the client system, open a console or terminal window, and use **su** or **sudo** to gain administrative permissions for the installation.
2. If you downloaded the software, copy or move the .tar archive (which usually has a name like "LunaClient_7.x.y-nn_AIX.tar") to a suitable directory where you can untar the archive and launch the installation script.
3. Enter the following command to extract the contents from the archive:

   tar xvf <filename>**.tar**
4. Change directory to the software version suitable for your system.
5. Install the client software as follows:
   - To see the 'help', or a list of available installer options, type:

     sh install.sh -? or **./sh install.sh --help**
   - To install all available products and optional components, type:

     sh install.sh all
   - To selectively install individual products and optional components, type the command without arguments:

     sh install.sh

   > **NOTE** Do not interrupt the installation script in progress. An uninterruptible power supply (UPS) is recommended. See "Interrupting the Installation" on page 71 for more information.
6. Type **y** if you agree to be bound by the license agreement:

```
[mylunaclient-1 32]$ sh install.sh

IMPORTANT:  The terms and conditions of use outlined in the software
license agreement (Document #008-010005-001_EULA_HSM_SW_revN) shipped with the product
("License") constitute a legal agreement between you and SafeNet.
Please read the License contained in the packaging of this
product in its entirety before installing this product.

Do you agree to the License contained in the product packaging?

If you select 'yes' or 'y' you agree to be bound by all the terms
and conditions set out in the License.

If you select 'no' or 'n', this product will not be installed.

(y/n)
```

7. A list of installable Luna products appears (might be different, depending on your platform). Select as many as you require, by typing the number of each (in any order) and pressing Enter. As each item is selected, the list updates, with a "*" in front of any item that has been selected. This example shows item 1 has been selected.

```
Products
Choose Luna Products to be installed
    *[1]: Luna Network HSM
     [N|n]: Next
```

```
     [Q|q]: Quit

Enter selection: 1
```

> **NOTE** Although the AIX and Solaris installers display the options, Luna PCIe HSM 7 and Luna USB HSM 7 are not supported in this release. Select only **Luna Network HSM 7** during installation.

8. When selection is complete, type **N** or **n** for "Next", and press **Enter**. If you wish to make a change, simply type a number again and press **Enter** to de-select a single item.
9. The next list is called "Advanced" and includes additional items to install. Some items might be pre-selected to provide the optimum Luna HSM experience for the majority of customers, but you can change any selection in the list.

```
Products
Choose Luna Products to be installed
     [1]: Luna Network HSM
     [N|n]: Next
     [Q|q]: Quit

Enter selection: 1

Advanced
Choose Luna Components to be installed

  [1]: Luna SDK

 *[2]: Luna JSP (Java)

 *[3]: Luna JCProv (Java)

  [B|b]: Back to Products selection

  [I|i]: Install

  [Q|q]: Quit

 Enter selection:
```

   If you wish to make a change, simply type a number again and press **Enter** to select or de-select a single item.

   If the script detects an existing cryptoki library, it stops and suggests that you uninstall your previous Luna software before starting the Luna HSM Client installation again.
10. The system installs all packages related to the products and any optional components that you selected. By default, the Client programs are installed in the **/usr/safenet/lunaclient** directory.

> **NOTE** When installing, ensure that the full path of a package does not contain any space characters. (The IBM examples do not show any spaces, implying that this might be a system requirement.)

11. Although FMs are supported on Linux and Windows clients only in this release, the FM architecture requires a configuration file setting to allow partition login on an FM-enabled HSM. If the HSM you will be using with this client is FM-enabled (see Preparing the Luna Network HSM 7 to Use FMs for more information), you must add the following entry to the [Misc] section of the Chrystoki.conf file:

    [Misc]

    LoginAllowedOnFMEnabledHSMs=1

> **NOTE** As a general rule, do not modify the Chrystoki.conf/crystoki.ini file, unless directed to do so by Thales Technical Support. If you do modify the file, never insert TAB characters - use individual space characters. Avoid modifying the PED timeout settings. These are now hardcoded in the appliance, but the numbers in the Chrystoki.conf file must match.

Uninstalling the Luna HSM Client Software

You may need to uninstall the Luna HSM Client software prior to upgrading to a new release, or if the software is no longer required.

To uninstall the Luna HSM Client software:

1. Log in as root. (use sudo instead)
2. Go to the client installation directory:

    cd /usr/safenet/lunaclient/bin
3. Run the uninstall script:

    sudo sh uninstall.sh

Installing Java

If you install the Luna Java Security Provider (JSP), refer to Luna JSP Overview and Installation for additional setup procedures for your operating system.

Scripted or Unattended Installation

If you prefer to run the installation from a script, rather than interactively, run the command with the options **-p** <list of Luna products> and **-c** <list of Luna components>. To see the syntax, run the command with **help** like this:

```
[myhost]$ sudo sh install.sh help
[sudo] password for fred

At least one product should be specified.

usage:
        install.sh      - Luna Client install through menu
        install.sh help - Display scriptable install options
        install.sh all  - Complete Luna Client install

        install.sh -p [sa|pci|g5|rb] [-c sdk|jsp|jcprov|ldpc|snmp]

        -p <list of Luna products>
        -c <list of Luna components>  - Optional. All components are installed if not provided
```

```
Luna products options
   sa     - Luna Network HSM
   pci    - Luna PCIe HSM
   g5     - Luna USB HSM
   rb     - Luna Backup HSM

Luna components options
   sdk    - Luna SDK
   jsp    - Luna JSP (Java)
   jcprov - Luna JCPROV (Java)
   snmp   - Luna SNMP subagent


[myhost]$
```

For scripted/automated installation, your script will need to capture and respond to the License Agreement prompt, and to the confirmation prompt. For example:

```
[myhost]$ sudo sh install.sh all

IMPORTANT: The terms and conditions of use outlined in the software
license agreement (Document #008-010005-001_053110) shipped with the product
("License") constitute a legal agreement between you and SafeNet Inc.
Please read the License contained in the packaging of this
product in its entirety before installing this product.

Do you agree to the License contained in the product packaging?

If you select 'yes' or 'y' you agree to be bound by all the terms
and conditions se out in the License.

If you select 'no' or 'n', this product will not be installed.

(y/n) y

Complete Luna HSM Client will be installed. This includes Luna Network HSM,
Luna PCIe HSM, Luna USB HSM AND Luna Backup HSM.

Select 'yes' or 'y' to proceed with the install.

Select 'no' or 'n', to cancel this install.

Continue (y/n)?  y
```

Interrupting the Installation

Do not interrupt the installation script in progress, and ensure that your host computer is served by an uninterruptible power supply (UPS). If you press [Ctrl] [C], or otherwise interrupt the installation (OS problem, power outage, other), some components will not be installed. It is not possible to resume an interrupted install process. The result of an interruption depends on where, in the process, the interruption occurred (what remained to install before the process was stopped).

As long as the cryptoki RPM package is installed, any subsequent installation attempt results in refusal with the message "A version of Luna HSM Client is already installed."

If components are missing or are not working properly after an interrupted installation, or if you wish to install any additional components at a later date (following an interrupted installation, as described), you would need to uninstall everything first. If **sh uninstall.sh** is unable to do it, then you must uninstall all packages manually.

Because interruption of the install.sh script is not recommended, and mitigation is possible, this is considered a low-likelihood corner case, fully addressed by these comments.

Adding a Luna Cloud HSM Service

Luna HSM Client allows you to use both Luna partitions and Thales Data Protection on Demand (DPoD) Luna Cloud HSM services. Using a single client workstation, you can back up or migrate your keys between Luna and the Luna Cloud HSM service, or combine partitions and services into an HA group.

> **NOTE** Refer to the Luna HSM Client Releases for supported client versions. Thales recommends keeping your Luna HSM Client software updated to the latest version, especially if your deployment includes Luna Cloud HSM.

Prerequisites

> If Luna HSM Client is not installed at the default location, the **ChrystokiConfigurationPath** must be set for the Luna Cloud HSM service to use the correct location.

> DPoD Luna Cloud HSM services support Windows and Linux operating systems only. This procedure presumes that you have already set up Luna HSM Client on your Windows or Linux workstation:

- "Windows Luna HSM Client Installation" on page 21
- "Windows Interactive Luna HSM Client Installation" on page 28
- "Linux Luna HSM Client Installation" on page 38

> For more information on Luna/Luna Cloud HSM service compatibility, refer to "Cloning Keys Between Luna 6, Luna 7, and Luna Cloud HSM, Password or Multifactor Quorum" on page 210.

To add a DPoD Luna Cloud HSM service to an existing Luna HSM Client

1. After purchasing a Luna Cloud HSM service, refer to the DPoD Luna Cloud HSM documentation for instructions on downloading the Luna Cloud HSM service client. Transfer the zip file to your workstation using **pscp**, **scp**, or other secure means.
2. Extract the zip file into a directory on your client workstation.
3. Extract or untar the appropriate client package for your operating system. Do not extract to a new subdirectory; place the files in the Luna Cloud HSM service client install directory. The other client package can be safely deleted.
   - [Windows] **cvclient-min.zip**
   - [Linux] **cvclient-min.tar**

     # tar -xvf cvclient-min.tar

     Run the provided **setenv** script to automatically copy the necessary Luna Cloud HSM service configuration entries to the existing Luna HSM Client configuration file. The existing Luna HSM Client configuration file must be writable to execute **setenv**.

> **CAUTION!** Running **setenv** will overwrite any existing Luna Cloud HSM service configurations in the Luna HSM Client configuration file.

> **NOTE** If Luna HSM Client is not installed in the default directory, or if **setenv** was run previously, you must clear the **ChrystokiConfigurationPath** environment variable or update it to point to the location of the correct configuration file:
>
> > [Windows] In the Control Panel, search for "environment" and select **Edit the system environment variables**. Click **Environment Variables**. In both the list boxes for the current user and system variables, edit **ChrystokiConfigurationPath** to point to the **crystoki.ini** file in the correct client install directory.
>
> > [Linux] Either open a new shell session, or reset the environment variable for the current session to the location of the correct **Chrystoki.conf** file:
>
> **# export ChrystokiConfigurationPath=/etc/**

- [Windows **cmd** prompt] Open a command prompt as **Administrator** and run the script with the **-addcloudhsm** option.

  > .\setenv.cmd -addcloudhsm

- [Linux] Source the **setenv** script with the **--addcloudhsm** option.

  # source ./setenv --addcloudhsm

4. Launch or relaunch LunaCM to verify that both your Luna partitions and Luna Cloud HSM service are available. Once the Luna Cloud HSM service has been added to the Luna HSM Client, you can delete the client package downloaded from Thales DPoD.

Initializing a Luna Cloud HSM Service

You must now initialize the Luna Cloud HSM service for use with your existing Luna partitions. If your Luna HSMs are password-authenticated, the cloning domain you set on the Luna Cloud HSM service must match the partition(s) with which it will share keys.

> "Initializing an Application Partition" on page 332

> "Initializing Crypto Officer and Crypto User Roles for an Application Partition" on page 368

If you will be using the Luna Cloud HSM service with multifactor quorum-authenticated Luna partitions, LunaCM provides the option to import the credential from a red domain PED key to Luna Cloud HSM, as described below.

> **NOTE** This feature requires minimum Luna HSM Client 10.4.1, and is available for Luna Cloud HSM only. For cloning between password- and multifactor quorum-authenticated Luna HSMs, see "Universal Cloning" on page 198.

Prerequisites

> The uninitialized Luna Cloud HSM service must be available in LunaCM on a client computer with Luna HSM Client 10.4.1 or newer installed.

> The client computer must have the Luna PED driver installed:

Windows: "Modifying the Installed Windows Luna HSM Client Software" on page 32 (**Remote PED** package)

Linux: "About Installing the Luna HSM Client Software" on page 40 (**[5] Luna Remote PED** package or **-p ped** in scripted installation)

> Connect a Luna PED to the client computer and set it to **Local PED-USB** mode (see "Modes of Operation" on page 248).

> If you were previously using this client computer as a Remote PED server, you must stop PEDserver before continuing:

"pedserver -mode stop" on page 318

> If your Luna partition domain uses an M of N PED key scheme, ensure that you have enough keys on hand to provide the M of N quorum.

To initialize a Luna Cloud HSM service using an imported domain secret

1. Launch LunaCM on the client computer.
2. Set the active slot to the uninitialized Luna Cloud HSM service.

   lunacm:> **slot set -slot** <slot#>
3. Initialize the Luna Cloud HSM service, specifying an identifying label and including the **-importpeddomain** option.

   lunacm:> **partition init -label** <label> **-importpeddomain**

   Follow the prompts in LunaCM and on the Luna PED to import the domain secret and complete the initialization process.

Refer to "Cloning Keys Between Luna 6, Luna 7, and Luna Cloud HSM, Password or Multifactor Quorum" on page 210 for guidelines on using Luna Cloud HSM with Luna 7 HSMs. You can back up your partitions to Luna Cloud HSM using slot-to-slot cloning or by setting up an HA group to synchronize your partition contents with Luna Cloud HSM.

> **CAUTION!** HA failover from multifactor quorum-authenticated Luna partitions to Luna Cloud HSM requires minimum Luna HSM Client 10.5.0. Refer to known issue LUNA-23945.

> "Cloning Objects to Another Application Partition" on page 201

> "Setting Up an HA Group" on page 432

> "Partition Backup and Restore" on page 467

Dynamic Partition Loading for Luna Cloud HSM Services

Luna HSM Client 10.5.0 and newer provide access to dynamic partition loading for Luna Cloud HSM services. Dynamic partition loading allows you to add additional sets of client UserIDs (combination of unique AuthTokenClientID, AuthTokenClientSecret, AuthTokenConfigURI) to the crystoki.conf or Chrystoki.ini file and automatically access the added partitions without restarting LunaCM or impacting other applications using LunaCM. Deleted partitions will not be removed from the LunaCM list until you restart LunaCM.

The default maximum number of users that can be added to a crystoki.conf or Chrystoki.ini file is 100. For more information about configuring the maximum number of client UserIDs see MaxUserIDCount in the "Configuration File Summary" on the next page.

> **NOTE** Dynamic Partition Loading using mixed FIPS mode partitions may show incorrect results for the HSM operation mode of each partition.

Prerequisites

> Luna HSM Client 10.5.0 or newer

- An HSM client downloaded from the Thales Support Portal. If using an HSM client this procedure assumes that you have already set up your HSM client on your Windows or Linux workstation. In addition, this procedure requires the REST and XTC sections of the Luna Cloud HSM service be available in the client configuration file. See "Adding a Luna Cloud HSM Service" on page 72 for more information about adding your first Luna Cloud HSM service and the necessary configuration file entries to an existing HSM client.
  - "Windows Interactive Luna HSM Client Installation" on page 28
  - "Linux Luna HSM Client Installation" on page 38
- A minimal client downloaded from Thales Data Protection on Demand.

> A Luna Cloud HSM service partition to load dynamically.

> If HSM client is not installed at the default location, the ChrystokiConfigurationPath must be set for the Luna Cloud HSM service to use the correct location.

To dynamically load a partition

1. Open the client configuration file (the Chrystoki.conf (Linux) or crystoki.ini/crystoki-template.ini (Windows)), for the HSM client that you are adding the Luna Cloud HSM service partition to, in a text editor.
2. In the REST section, add the client UserID values for the new partition. Append the client UserID values with a unique numerical value to associate the client UserID values with each other.

> **TIP** The client UserID values can be accessed from the Chrystoki.conf (Linux) or crystoki.ini/crystoki-template.ini (Windows) configuration files included in the Luna Cloud HSM service client package.

Linux example:

```
REST = {
AuthTokenConfigURI=********
AuthTokenClientId=********
AuthTokenClientSecret=********
AuthTokenConfigURI2=********
AuthTokenClientId2=********
AuthTokenClientSecret2=********
AuthTokenConfigURI3=********
AuthTokenClientId3=********
AuthTokenClientSecret3=********
RestClient=1
ClientTimeoutSec=120
```

```
ClientPoolSize=32
ClientEofRetryCount=15
ClientConnectRetryCount=900
ClientConnectIntervalMs=1000
PartitionData00=1334054167371, na.hsm.dpondemand.io, 443
SSLClientSideVerifyFile=.\\server-certificate.pem;
}
```

Windows example:

```
[REST]
AuthTokenConfigURI=********
AuthTokenClientId=********
AuthTokenClientSecret=********
AuthTokenConfigURI2=********
AuthTokenClientId2=********
AuthTokenClientSecret2=********
AuthTokenConfigURI3=********
AuthTokenClientId3=********
AuthTokenClientSecret3=********
RestClient=1
ClientTimeoutSec=120
ClientPoolSize=32
ClientEofRetryCount=15
ClientConnectRetryCount=900
ClientConnectIntervalMs=1000
PartitionData00=1334054167371, na.hsm.dpondemand.io, 443
SSLClientSideVerifyFile=.\\server-certificate.pem;
```

3. Execute the "slot list" command in LunaCM to display the additional partitions.

> **TIP** Additional sets of client UserIDs can be exported and secured as described in "Configuration File Summary" below.

Configuration File Summary

The Luna HSM Client software installation includes a configuration file that controls many aspects of client operation. The fields in the configuration file are used to alter the default behavior of the library. So the default value is the value that would result in the normal (non-altered) behavior (but see TIP below). The configuration file can be found in the following default locations:

> **Luna HSM Client**

- Windows: C:Program Files\SafeNet\LunaClient\crystoki.ini
- Linux/UNIX: /etc/Chrystoki.conf

> **Luna Cloud HSM**

- Windows: <service_client_path>\crystoki.ini
- Linux: <service_client_path>/Chrystoki.conf

> **NOTE** The crystoki.ini and Chrystoki.conf files included with the Luna Cloud HSM service client provide a default set of configuration options.

The configuration file is organized into named sections, containing various configuration entries. It is installed with the default settings described in the table below. In addition to the default sections and entries, some additional sections/entries can be added to customize functionality. Generally, Thales does not recommend editing the configuration file directly; many entries are changed by entering commands in LunaCM or **vtl**. However, some entries can only be edited manually.

If you update the Luna HSM Client software by running the uninstaller and then installing a newer version, the existing configuration file is saved. This preserves your configuration settings, including the location of certificates necessary for your partition NTLS/STC connections for Luna products.

The following table describes all valid sections and entries in the configuration file. When editing the file, ensure that you maintain the applicable syntax conventions for your operating system (use existing sections/entries as a template for new entries). Where applicable, entries are listed with the valid range of values and the default setting.

TIP Configuration settings and Section Headings

Some of the sections and entries listed here do not appear in the base configuration file as you would receive it via download; in many use-cases, your application's employment of the HSM would not require any direct intervention in the configuration file. "Factory" settings would be sufficient. If an application would benefit from a setting that differs from the stock values, and a visible entry is not already in the configuration file on your host system, then you must add a relevant entry in order to change the behavior described in the table below.

- If the configuration file contains no explicit entries that would reside under a particular heading, then the heading itself would also not be present.
- If you intend to add a setting to the file, and there is no pre-existing section-heading for it, then create a heading as indicated in the table, below, following the format of the other sections in your config file, appropriate for the host operating system (Linux/UNIX or Windows).

The majority of fields are Boolean (true/false or 0/1), or a range of values.

Some of the entries listed include a default setting that is observed if the entry is not included in the configuration file by default; you must add the entry explicitly, only in the case where you need to change the default behavior.

Exceptions:

- If an explicit location is not set for the library location (which is normally inserted as part of installation if you use the Client installer), or if you moved the library after a path value was included, an error message is generated about the field/path/file not being found, and you must provide a proper library location in order to proceed.
- If you intend to use an NTLS connection (Luna Network HSM 7) or an XTC connection (Luna Cloud HSM), then the relevant settings must be present to define that connection, or the connection does not take place.

Do not edit the Chrystoki.conf or crystoki.ini file unless you need to do so. If you need to edit, refer to the guidance in the table below, and ask for help from our technical support engineers if this document is not sufficient.

Section/Setting	Description
Chrystoki2	
LibNT	Path to the Chrystoki2 library on Windows operating systems. **Default: C:\Program Files\SafeNet\LunaClient\cryptoki.dll**
LibNT32	Path to the Chrystoki2 library on 32-bit Windows systems only. **Default: C:\Program Files\SafeNet\LunaClient\win32\libCryptoki2.dll** **NOTE** Luna HSM Client 10.1.0 and newer includes libraries for 64-bit operating systems only.
LibUNIX64	Path to the Chrystoki2 library on 64-bit Linux/UNIX operating systems. **Default:** > **Linux/AIX: /usr/safenet/lunaclient/libs/64/libCryptoki2_64.so** > **Solaris: /opt/safenet/lunaclient/libs/64/libCryptoki2_64.so**
Luna (see * below this table)	
CloningCommandTimeout	The amount of time (in milliseconds) the library allows for the HSM to respond to a cloning command. **Default: 300000**
CommandTimeoutPedSet	This is an exception to DefaultTimeout (below). It defines the time (in milliseconds) allowed for all PED-related HSM commands. PED-related commands can take longer than ordinary commands governed by DefaultTimeOut. Generally, the following formula applies: CommandTimeOutPedSet = DefaultTimeOut + PEDTimeout1 + PEDTimeout2 + PEDTimeout3 **Default: 720000**

Section/Setting	Description
DefaultTimeOut	Defines the time (in milliseconds) the HSM driver in the host system waits for HSM commands to return a result. If a result is not returned in that time, the driver halts the HSM and returns DEVICE_ERROR to all applications using the HSM. The only exceptions are when a command's timeout is hard-coded in the Cryptoki library, or the command falls into a class governed by one of the other timeout intervals described elsewhere in this section. **Default: 500000**
DomainParamTimeout	Timeout (in milliseconds) for Domain Parameter Generation. **Default: 5400000**
KeypairGenTimeOut	Defines the time (in milliseconds) the library waits for a keypair generation operation to return a value. The randomization component of keypair generation can cause large keypairs to take a long time to generate, and this setting keeps the attempts within a reasonable time. You can change this value to manage your preferred balance between long waits and the inconvenience of restarting a keygen operation. **Default: 2700000**
PEDTimeout1	Defines the time (in milliseconds) the HSM attempts to ping the PED before sending a PED operation request. If the PED is unreachable, the HSM returns a code indicating that the PED is not connected. **Default: 100000**
PEDTimeout2	Defines the time (in milliseconds) that the HSM waits for the local PED to respond to a PED operation request. If the local PED does not respond to the request within the span of PEDTimeout2, the HSM returns an appropriate result code (such as PED_TIMEOUT). This is the timeout you might increase from the Default value if you were initializing larger MofN PED Key sets - the HSM allows M and N to each be up to 16 splits - maybe applying PED PINS, and making a duplicate set as well. **Default: 200000**

Section/Setting	Description
PEDTimeout3	Defines the additional time (in milliseconds) the HSM waits for a remote PED to respond to a PED operation request. Therefore, the actual time the firmware waits for a remote PED response is PEDTimeout2 + PEDTimeout3. **Default: 20000**
CardReader	
LunaG5Slots	Number of Luna Backup HSM G5 slots reserved so that the library will check for connected devices. **Valid Values:** > **0**: If you have no Luna Backup HSM G5s and wish to eliminate the reserved spaces in your slot list, use this setting. > **1-N**: Can be set to any number, but is effectively limited by the number of external USB devices supported by your client workstation. **Default: 3**
LunaG7Slots	Number of Luna Backup HSM 7 slots reserved so that the library will check for connected devices. **Valid Values:** > **0**: If you have no Luna Backup HSM 7s and wish to eliminate the reserved spaces in your slot list, use this setting. > **1-N**: Can be set to any number, but is effectively limited by the number of external USB devices supported by your client workstation. **Default: 3**
RemoteCommand	This setting was used when debugging older Luna products. For modern products it is ignored. **Valid Values:** > **0**: false > **1** (default): true
CKLog2	
NOTE See Using CKlog. Config is done using the vtl utility or by editing this config file directly.	

Section/Setting	Description
MaskedParams	Sensitive data is masked in cklog output, including: • Initialization Vector/IV value (user provided or HSM generated) • The clear-text data payload (input into the Encrypt or the output from decrypt) • Any private-secret keys (key-wrapping) • Any input into a hashing function All other details of the log string are retained (length, codes, etc...) **Valid Values:** > **0**: This is the legacy value using Luna HSM Client 10.9.0 or older (No masking) > **1**: Mask log data mentioned above with asterisk * (Default using Luna HSM Client 10.9.1 or newer) > **2**: Future use. **Default: 1** This option is not set by "vtl cklog enable. To set this value, use the configurator tool, or edit the configuration file manually. MaskedParams=[0|1] Passwords are always masked. **NOTE** This feature requires Luna HSM Client 10.9.1 or newer.
RBS	**NOTE** RBS is not supported with Luna Cloud HSM services.
CmdProcessor	The location of the RBS library. **Default:** > **Windows: C:\Program Files\SafeNet\LunaClient\rbs_processor2.dll** > **Linux/AIX: /usr/safenet/lunaclient/rbs/lib/librbs_processor2.dll** > **Solaris: /opt/safenet/lunaclient/rbs/lib/librbs_processor2.dll**
HostPort	The port number used by the RBS server. **Valid Values:** any unassigned port **Default: 1792**

Section/Setting	Description
ClientAuthFile	The location of the RBS Client authentication file. **Default:** > **Windows: C:\Program Files\SafeNet\LunaClient\config\clientauth.dat** > **Linux/AIX: /usr/safenet/lunaclient/rbs/clientauth.dat** > **Solaris: /opt/safenet/lunaclient/rbs/clientauth.dat**
ServerSSLConfigFile	The location of the OpenSSL configuration file used by RBS Server or Client. **Default:** > **Windows: C:\Program Files\SafeNet\LunaClient\rbs\server.cnf** > **Linux/AIX: /usr/safenet/lunaclient/rbs/server/server.cnf** > **Solaris: /opt/safenet/lunaclient/rbs/server/server.cnf**
ServerPrivKeyFile	The location of the RBS Server certificate private key file. **Default:** > **Windows: C:\Program Files\SafeNet\LunaClient\cert\server\serverkey.pem** > **Linux/AIX: /usr/safenet/lunaclient/rbs/server/serverkey.pem** > **Solaris: /opt/safenet/lunaclient/rbs/server/serverkey.pem**
ServerCertFile	The location of the RBS Server certificate file. **Default:** > **Windows: C:\Program Files\SafeNet\LunaClient\cert\server\server.pem** > **Linux/AIX: /usr/safenet/lunaclient/rbs/server/server.pem** > **Solaris: /opt/safenet/lunaclient/rbs/server/server.pem**
NetServer	Determines whether RBS acts as a server or client. **Valid Values:** > **0**: Client > **1** (default): Server

Section/Setting	Description
HostName	The hostname or IP address that the RBS server will listen on. **Valid Value:** any hostname or IP address **Default:** 0.0.0.0 (any IP on the local host)
Available	Lists the serial numbers of Luna Backup HSMs available on the RBS server.
LunaSA Client	
ReceiveTimeout	Time in milliseconds before a receive timeout. **Default:** 20000
SSLConfigFile	Location of the OpenSSL configuration file. **Default:** > **Windows: C:\Program Files\SafeNet\LunaClient\openssl.cnf** > **Linux/AIX: /usr/safenet/lunaclient/bin/openssl.cnf** > **Solaris: /opt/safenet/lunaclient/bin/openssl.cnf**
ClientPrivKeyFile	Location of the client private key. This value is set by **vtl** or lunacm:> **clientconfig deploy**. **Default:** > **Windows: C:\Program Files\SafeNet\LunaClient\cert\client** <ClientName>**Key.pem** > **Linux/AIX: /usr/safenet/lunaclient/cert/client/** <ClientName>**Key.pem** > **Solaris: /opt/safenet/lunaclient/cert/client/** <ClientName>**Key.pem**

Section/Setting	Description
ClientCertFile	Location of the client certificate that is uploaded to Luna Network HSM 7 for NTLS. This value is set by **vtl** or lunacm:> **clientconfig deploy**. **Default:** > **Windows: C:\Program Files\SafeNet\LunaClient\cert\client** <ClientName>**Cert.pem** > **Linux/AIX: /usr/safenet/lunaclient/cert/client/** <ClientName>**Cert.pem** > **Solaris: /opt/safenet/lunaclient/cert/client/** <ClientName>**Cert.pem**
LNHServerKeepAliveTimer##	Set the keepalive timer (in milliseconds) for connections to the cluster, specified by ## (refer to " LNHServer##" on the next page). **Valid Range: 10000-50000** **Default: 30000** > **NOTE** This feature requires Luna HSM Client 10.5.2 or newer, and the **cluster** package (1.0.2) included with Luna Network HSM 7 Appliance Software 7.8.2 or newer.
LNHServerLoadBalancingMode##	Selects the load-balancing mode the client will use to distribute requests among the members of the cluster, specified by ## (refer to " LNHServer##" on the next page). **Valid Values:** > **pick_first** (default): Operation requests are sent to the first cluster member where a connection can be successfully made. > **round_robin**: Connections are made to all active members, and operation requests are distributed to each active member in turn. > **NOTE** This feature requires Luna HSM Client 10.5.2 or newer, and the **cluster** package (1.0.2) included with Luna Network HSM 7 Appliance Software 7.8.2 or newer.

Section/Setting	Description
ServerCAFile	Location of the server certificate file on the client workstation. This value is set by **vtl** or lunacm:> **clientconfig deploy**. **Default:** > **Windows: C:\Program Files\SafeNet\LunaClient\cert\server\CAFile.pem** > **Linux/AIX: /usr/safenet/lunaclient/cert/server/CAFile.pem** > **Solaris: /opt/safenet/lunaclient/cert/server/CAFile.pem**
NetClient	Determines whether the library searches for network slots. **Valid Values:** > **0**: The library does not search for network slots. > **1** (default): The library searches for network slots.
TCPKeepAlive	TCPKeepAlive is a TCP stack option, available at the Luna HSM Client and the Luna Network HSM 7 appliance. It is controlled via an entry in the Luna HSM Client configuration file, and an equivalent file on the Luna Network HSM 7. On the Luna Network HSM 7 appliance, where you do not have direct access to the file system, the TCPKeepAlive= setting is controlled by lunash:> **ntls tcp_keepalive set**. The settings at the appliance and the client are independent. This allows a level of assurance, in case (for example) a firewall setting blocks communication in one direction. **Valid Values:** > **0**: false > **1** (default): true
ServerName##	These entries identify NTLS-linked Luna Network HSM 7 servers/ports, and determines the order in which they are polled to create a slot list. These values are set by **vtl** or lunacm:> **clientconfig deploy**.
ServerPort##	
LNHServer##	The IP address and port of a Luna Network HSM 7 cluster member connected to this client. See Cluster-Client Connections.
LNHServerClientCert##	The location and filename of the Luna HSM Client certificate used to access this Luna Network HSM 7 cluster. See Cluster-Client Connections.

Section/Setting	Description
LNHServerClientKey##	The location and filename of the Luna HSM Client private key used to access this Luna Network HSM 7 cluster. See Cluster-Client Connections.
LNHServerCAFile##	The location and filename of the Luna Network HSM 7 cluster server certificate. See Cluster-Client Connections.
LNHServerCN##	The Common Name for the cluster certificate. See Cluster-Client Connections.
LNHStandbyServer##	Reports the affinity group the client is directing traffic to. See Moving a Member to a Different Affinity Group. **NOTE** This feature requires Luna HSM Client 10.5.2 or newer, and the **cluster** package (1.0.2) included with Luna Network HSM 7 Appliance Software 7.8.2 or newer.
Presentation	**NOTE** This section is not created automatically. To change any of the following values, you must first create this section in the configuration file.
OneBaseSlotId	Determines whether slot listing begins at **0** or **1**. **Default: 0**
ShowAdminTokens	Determines whether the Admin partitions of locally-installed Luna PCIe HSM 7s are visible in the slot list. **Valid Values:** > **no**: Admin slots are hidden. > **yes** (default): Admin slots are visible.

Section/Setting	Description
ShowEmptySlots	Determines whether slot numbers are reserved for partitions that have not yet been created on the HSM. When this setting is enabled, slot numbers remain consistent over time, even when new partitions are created. **Valid Values:** > **no** (default): Only existing partitions are assigned slot numbers. > **yes**: Slot numbers are reserved for the maximum number of partitions that can be created on HSMs connected to the client. **NOTE** This does not apply to Luna Network HSM 7 partitions assigned to the client, which will always appear in the lowest-numbered slots, causing locally-connected and Luna Cloud HSM service slots to increment higher.
ShowUserSlots	Allows you to set permanent slot numbers for specific partitions. If you use this setting, you must specify a slot for all partitions on a specific HSM, or the partitions not listed here will not be visible to the client. **Valid Values:** Comma-delimited list in the format <slotnum>**(**<serialnum>**)** **Example:** **ShowUserSlots=1(351970018022),2(351970018021),3(351970018020),**...
VirtualToken	**NOTE** This section is created only if the HA auto-recovery mode is set to **activeEnhanced**. See "Configuring HA Auto-Recovery" on page 439.
VirtualToken##Label	The label of the HA group. This value is set by using the lunacm:> hagroup creategroup command to set up an HA group. See "Setting Up an HA Group" on page 432.
VirtualToken##SN	The pseudo serial number of the HA group. This value is set by using the lunacm:> hagroup creategroup command to set up an HA group. See "Setting Up an HA Group" on page 432.
VirtualToken##Members	The serial number of the HA group members. This value is set by using the lunacm:> hagroup addmember command to add a member to the HA group. See "Setting Up an HA Group" on page 432.

Section/Setting	Description
VirtualTokenActiveRecovery	The HA auto-recovery mode. This value is set by using the lunacm:> hagroup recoverymode command to configure the HA auto-recovery mode. See "Configuring HA Auto-Recovery" on page 439.
HAConfiguration	
AutoReconnectInterval	Specifies the interval (in seconds) at which the library will attempt to reconnect with a missing HA member, until the set number of attempts is reached. This value is set using lunacm:> **hagroup interval**. **Valid Values:** > **60-1200**: Wait the specified number of seconds between reconnection attempts. **Default: 60** seconds
HAOnly	Determines whether individual HA member slots are visible to client applications. Hiding individual members helps prevent synchronization errors by preventing applications from directing calls to individual member partitions. If a member partition fails, the other slots in the system change, which can cause applications to send calls to the wrong slot number. This setting prevents this by hiding all physical slots from applications. **Valid Values:** > **0** (default): All partitions are visible to applications as slots. > **1**: Only HA virtual slots are visible to applications. **NOTE** This setting does not affect how slots are numbered in LunaCM; you can still configure individual member partitions with HAOnly mode enabled.

Section/Setting	Description
ProbeTimeout	By default, if the HA probing thread makes a request to the Luna Network HSM 7 where the internal cryptographic module (HSM card) is locked up, the probing thread can also lock up, and failover does not occur. To deal with that possibility, this setting, with a value greater than 0, initiates a timeout (in seconds). **Valid Values:** > **0** (default): This is the same effect as the ProbeTimeout setting not existing in this configuration file - no timeout is set. > **1 - ?**: A number of seconds greater than zero. This is a balancing decision, unique to your application situation. If you wish to set a ProbeTimeout, choose a value • large enough to allow your usual crypto processes to complete normally, but • not so large that a failover, from a failed HA-group member to a healthy member, is never triggered before your application concludes that the entire HA group has failed. This feature requires Luna HSM Client 10.7.2 or newer (with the default 0 value for backward compatibility).
reconnAtt	Specifies the number of reconnection attempts the client makes to a missing HA member. Once this number is reached, you must manually reconnect the member when it becomes available (see "Manually Recovering a Failed HA Group Member" on page 452). This value is set using lunacm:> **hagroup retry**. **Valid Values:** > **-1**: Perform infinite reconnection attempts. > **0**: Disable HA auto-recovery. > **1-500**: Perform the specified number of reconnection attempts.

Section/Setting	Description
statusTimeout	This value (in seconds) is the amount of time the CA_GetCurrentHAState function will try to verify the status of HA group members, before stopping and reporting the statuses collected up to that cutoff. **Valid Values:** > **3** (default): This is the shortest reasonable value in good network conditions. > **4-60**: Any value between the default and 60 second. **NOTE** After 60 seconds, the status check could conflict with other processes, so the cap is set at 60.
Misc	
AppId = <xxx>	Application IDs are generated when the application starts, and are 16 bytes for Luna HSM Firmware 7.7.0 and newer, and and Luna HSM Client 10.3.0 and newer. Application IDs are not supported for Luna Cloud HSM services. For earlier HSM firmware or clients, see Application IDs. You can override this functionality and specify an AppId if desired. For HSM firmware version 7.8.4 onward, see **NOTE below this table.
CopyRSAPublicValuesFromPrivateTemplate	Controls whether the public exponent of an RSA key can be copied from the private key template, if the public key template does not already have a public exponent attribute set. **Valid Values:** > **0**: if no public exponent is provided in the public template, an error is returned (expected behavior). > **1**(default): if no public exponent is provided in the public template, the private exponent is copied from the private template to populate the public template. For PKCS#11 compliance, this should be set to **0**. **NOTE** This functionality requires Luna HSM Client 7.1.0 or newer.

Section/Setting	Description
ECCPointEncodingStrategy=	Allows you to force (or not) the assumption that an Elliptic Curve point is provided in RAW octet string format. **Valid Values:** > **1**: Queries HSM for correct EC Point Size, then determines whether encoding is required or not. > **2**: Does not query the HSM for EC point size. Assumes EC point is RAW and always encodes it. (Included in client since Luna HSM Client 10.5.0; available as a patch for Luna HSM Client 10.4.0 and 10.4.1. Beginning with HSM firmware version 7.8.1 this entry is no longer needed.)
FunctionBindLevel	Determines what action to take if a function binding fails during a CryptokiConnect() operation. **Valid Values:** > **0** (default): fail if not all functions can be resolved > **1**: do not fail but issue warning for each function not resolved > **2**: do not fail and do not issue warning (silent mode)
HSSKeyGenSignTimeout	Time in milliseconds before an HSS key generation and signing operation would stop unfinished. **Default:** no timeout. As an example, if you were to create 3Level, [3] LMS_SHA256_N32_H15, [1] LMOTS_SHA256_N32_W1 keypair - you might expect keygen to successfully complete within 60000 milliseconds. If you were then to sign a 1MB file, expect that signing operation to take about 120000 milliseconds but then expect to get CKR_DEVICE_ERROR if HSSKeyGenSignTimeout was set to a lower number of milliseconds. Adjust this timeout value using estimates from past runs. After the first signing when the hashes are cached, the second signing will be much faster and there would be no timeout based on the HSSKeyGenSignTimeout setting due to the shortened time for second and subsequent signings with the same private key. Requires an HSM reset to clear the cache.

Section/Setting	Description
LoginAllowedOnFMEnabledHSMs	Determines whether the client can log in to a partition on an HSM that uses Functionality Modules (FMs). FMs consist of custom-designed code that introduces new functionality, which can be more or less secure than standard HSM functions. Possible values include: > **0**: the client does not allow login to a partition on an HSM where the FM policy is enabled > **1**: the client allows login to a partition on an HSM where the FM policy is enabled This entry is added to the configuration file the first time you initialize or log in to a partition on an HSM where the FM policy is enabled using LunaCM. You are prompted to confirm that you want to allow login. **NOTE** References to Functionality Modules (FMs) do not apply to Luna USB HSM 7 as that platform does not support FMs. If you need to customize a Luna HSM beyond the extensive configuration and capability update options, consider the Luna PCIe HSM 7 and the Luna Network HSM 7 which do support tailoring or extending the HSM firmware with your own custom functionality.

MutexFolder=	For non-Windows platforms. Several Luna features write temporary files to **/tmp**. If systemd service deletes the temporary files the affected services can be disrupted - example Remote PED callback service (cbs). An administrator can use this setting to specify an alternative location like this example:
	`Misc = {`
	`…`
	`  MutexFolder = /usr/lock;`
	`}`
	The specified folder must exist, or the service fails to start properly.
	> **Linux default: <install_dir>/lock**

Section/Setting	Description
PE1746Enabled	Enables the SafeXcel 1746 security co-processor on Luna 6 HSMs, which is used to offload packet processing and cryptographic computations from the host processor. Does not apply to Luna 7 HSMs or Luna Cloud HSM services. This must be set to **0** to use Luna 6 partitions in a mixed-version HA group (see "Cloning Keys Between Luna 6, Luna 7, and Luna Cloud HSM, Password or Multifactor Quorum" on page 210). **Valid Values:** > **0**: SafeXcel co-processor is disabled on Luna 6 HSMs. > **1** (default): SafeXcel co-processor is enabled on Luna 6 HSMs.
PluginModuleDir	Specifies the location of client plugins. This setting is required to use the cloud plugin to access Luna Cloud HSM services. **Default:** > **Windows: C:\Program Files\SafeNet\LunaClient\plugins** > **Linux: /usr/safenet/lunaclient/libs/64/plugins**
ProtectedAuthenticationPathFlagStatus	Specifies which role to check for challenge request status. **Valid Values:** > **0** (default): no challenge request > **1**: check for Crypto Officer challenge request > **2**: check for Crypto User challenge request **NOTE** This functionality requires Luna HSM Client 7.1.0 or newer.

Section/Setting	Description
RSAKeyGenMechRemap	Using older versions of the Luna HSM Client software, this entry allows you to automatically remap calls to certain old, less-secure mechanisms, to new mechanisms that are FIPS-approved. This remapping allows you to operate the HSM securely without having to rewrite your applications. > **0** (default): No re-mapping is performed. > **1**: The following remapping is applied: • Calls for PKCS key generation using CKM_RSA_PKCS_KEY_PAIR_GEN are remapped to CKM_RSA_FIPS_186_3_PRIME_KEY_PAIR_GEN, which uses 186-3 Prime key generation. • Calls for X9.31 key generation using CKM_RSA_X9_31_KEY_PAIR_GEN are remapped to CKM_RSA_FIPS_186_3_AUX_PRIME_KEY_PAIR_GEN, which uses 186-3 Aux Prime key generation **NOTE** Using Luna HSM Client 10.1.0 or newer, remapping is automatic and this configuration entry is ignored. For remapping on HSMs where FIPS mode is set on individual partitions, Luna HSM Client 10.4.0 or newer is required; see Applying the Mechanism Remapping for details.
ToolsDir	The location of the Luna HSM Client tools. **Full Luna HSM Client Default:** > **Windows: C:\Program Files\SafeNet\LunaClient** > **Linux/AIX: /usr/safenet/lunaclient/bin/** > **Solaris: /opt/safenet/lunaclient/bin/** **Minimal Luna HSM Client Default:** > **Linux: /usr/safenet/lunaclient/bin/64/**
ValidateHost=	Set this flag to have the Luna HSM Client validate the server's hostname/IP against the Subject Alternate Name (SAN) values in the server's certificate. **Default: 0**
Secure Trusted Channel	**NOTE** Secure Trusted Channel is not supported with Luna Cloud HSM Services.

Section/Setting	Description
ClientIdentitiesDir	Specifies the directory used to store the STC client identity. **Default:** > **Windows: C:\Program Files\SafeNet\LunaClient\data\client_identities** > **Linux/AIX: /usr/safenet/lunaclient/data/client_identities** > **Solaris: /opt/safenet/lunaclient/data/client_identities**
ClientTokenLib (for 64-bit Windows systems)	Specifies the location of the token library on 64-bit Windows systems. This value must be correct in order to use a client token. If you are using a hard token, you must manually change this value to point to the hard token library for your operating system. The exact location of the hard token library may vary depending on your installer. **Default: C:\Program Files\SafeNet\LunaClient\softtoken.dll**
ClientTokenLib32 (for 32-bit Windows systems)	Specifies the location of the token library on 32-bit Windows systems. This entry appears on Windows only. By default, **ClientTokenLib32** points to the location of the soft token library. If you are using a hard token, you must manually change this value to point to the hard token library for your operating system. The exact location of the hard token library may vary depending on your installer. **Soft Token Default: C:\Program Files\SafeNet\LunaClient\win32\softtoken.dll** **Hard Token Default: C:\Windows\SysWOW64\etoken.dll** **NOTE** Luna HSM Client 10.1.0 and newer includes libraries for 64-bit operating systems only.
PartitionIdentitiesDir	Specifies the directory used to store the STC partition identities exported using lunacm:> **stcconfig partitionidexport**. **Default:** > **Windows: C:\Program Files\SafeNet\LunaClient\data\partition_identities** > **Linux/AIX: /usr/safenet/lunaclient/data/partition_identities** > **Solaris: /opt/safenet/lunaclient/data/partition_identities**

Section/Setting	Description
SoftTokenDir	Specifies the location where the STC client soft token (**token.db**) is stored. **Default:** > **Windows: C:\Program Files\SafeNet\LunaClient\softtoken\001** > **Linux/AIX: /usr/safenet/lunaclient/softtoken/001/** > **Solaris: /opt/safenet/lunaclient/softtoken/001/**
Session	**NOTE** This section is not created automatically. To change any of the following values, you must first create this section in the configuration file.
AutoCleanUpDisabled	Determines whether AutoCleanUp closes orphaned sessions in the event that an application leaves sessions open. Useful for Luna PCIe HSM hosts. AutoCleanUp runs during C_Finalize on the client. Luna Network HSM 7 sessions are tracked and closed by the NTLS service. **Valid Values:** > **0** (default): Run AutoCleanUp if your application leaks sessions and you cannot rewrite the application. > **1**: Disable AutoCleanUp if you have a Luna PCIe HSM 7 and your client application does proper housekeeping, or if your application is connecting via NTLS to a Luna Network HSM 7.
Shim2	**NOTE** This section is not created automatically. To change any of the following values, you must first create this section in the configuration file.
(Linux) LibUNIX64=/usr/safenet/lunaclient/lib/libCryptoki2_64.so; (Windows) LibNT=C:\Program Files\SafeNet\LunaClient\cryptoki.dll	This section is required when a shim is to be used. If the cryptoki library is not at the indicated location, then adjust the example path value to reflect the actual location.
Toggles	**NOTE** This section is not created automatically. To change any of the following values, you must first create this section in the configuration file.

Section/Setting	Description
legacy_memory_rep =	Controls the manner in which the HSM reports the available RAM space. **Valid Values:** > **0** (default): the public and private memory total/free values reported in the CK_TOKEN_INFO structure indicate the available flash memory for permanent (TOKEN) objects that are in either the public or private space respectively; this method is PKCS#11 compliant. > **1**: the public memory values indicate the total/free RAM memory; this non-standard legacy method was used by some customers to determine space available for session based objects, and must be explicitly selected in order to continue using the legacy method. **NOTE** This functionality requires Luna HSM Firmware 7.1.0 or newer.
lunacm_cv_ha_ui =	Controls whether Thales DPoD Luna Cloud HSM services can be active members of an HA group. **Valid Values:** > **0**: Luna Cloud HSM services can be added as active HA members. > **1** (default): Luna Cloud HSM services can be added to HA groups as standby members only. This is the default behavior to maximize HA performance, which may suffer due to network latency. **NOTE** This functionality requires Luna HSM Client 10.2.0 or newer.
fetch_partition_label =	Allows the client to refresh the slot list without requiring an application restart. **Valid Values:** > **0** (default): A new session must be opened (C_Initialize) to refresh the slot list cache. > **1**: Slot list cache is refreshed without requiring C_Initialize. **NOTE** This functionality requires Luna HSM Client 10.5.1 or newer.

Section/Setting	Description
map_aes_cmac_general_old=	Allows the library to automatically map relevant values to the required mechanism CKM_AES_CMAC_GENERAL. Refer to resolved issue LUNA-30232. **NOTE** This functionality requires Luna HSM Client 10.7.0 or newer.
REST	**NOTE** This section configures a connection to a Luna Cloud HSM and applies only to a Luna Cloud HSM. This section is not created automatically for clients obtained from the Thales Support Portal. See "Adding a Luna Cloud HSM Service" on page 72 for detailed instructions on adding a Luna Cloud HSM service client to a Luna HSM Client. This section governs Luna Cloud HSM service functionality only and is not related to the Luna REST API. This functionality requires Luna HSM Client 10.2.0 or newer.

Section/Setting	Description
AppLogLevel	Defines the maximum severity level of application logs to be displayed in the application console. **Valid Values:** > **trace** (unavailable for Luna HSM Client 10.4.0 and newer) > **debug** (unavailable for Luna HSM Client 10.4.0 and newer) > **error** (default setting for Luna HSM Client 10.2.0 and Luna HSM Client 10.3.0) > **warning** > **info** Application Error Logs are printed to the Event Viewer on Windows 10 and to the system and console on Linux. **Windows**: Windows operating systems print application error logs to the Event Viewer. Access the logs by opening **Event Viewer** > **Windows Logs** > **Application** and filtering the results for *LunaClientEventProvider*. **NOTE** To display Windows Logs in the Event Viewer as a non-administrator user you must register the **LunaClientEventProvider.dll**. Failure to register the **LunaClientEventProvider.dll** will result in the logs displaying to the console. **Linux**: Linux operating systems print application error logs to **/var/log/message**. Access the logs by opening **/var/log/message** and searching the results for *lunacm*. **Ubuntu**: Ubuntu operating systems print application error logs to **/var/log/syslog**. Access the logs by opening **/var/log/syslog** and searching the results for *lunacm*.
AuthTokenConfigURI	The identifier for the authentication service which issues the tokens required to validate the client's identity to the Luna Cloud HSM service. Your client host must have an internet connection to reach this resource. Do not edit the default value unless instructed to by customer support.
AuthTokenClientId	The client identity required by the authentication service to issue a token. Do not edit the default value unless instructed to by customer support.
AuthTokenClientSecret	The client passphrase required by the authentication service to issue a token. Do not edit the default value unless instructed to by customer support.

Section/Setting	Description
CurlLogsEnabled	Enables libcurl logging. This variable applies to Luna HSM Client 10.3.0 and newer. **Valid Values:** > **False**:Libcurl logging is disabled. > **True**(default): Libcurl logging is enabled. **NOTE** If using Luna HSM Client 10.4.0 or newer you **must** have AppLogLevel=info defined in your Chrystoki.conf\crystoki.ini to retrieve curl logs. See "AppLogLevel" on the previous page for more information. Curl Logs are printed to the Event Viewer on Windows 10 and to the system and console on Linux. **Windows**: Windows operating systems print application error logs to the Event Viewer. Access the logs by opening **Event Viewer** > **Windows Logs** > **Application** and filtering the results for *LunaClientEventProvider*. **NOTE** To display Windows Logs in the Event Viewer as a non-administrator user you must register the **LunaClientEventProvider.dll**. Failure to register the **LunaClientEventProvider.dll** will result in the logs displaying to the console. **Linux**: Linux operating systems print application error logs to **/var/log/message**. Access the logs by opening **/var/log/message** and searching the results for *lunacm*. **Ubuntu**: Ubuntu operating systems print application error logs to **/var/log/syslog**. Access the logs by opening **/var/log/syslog** and searching the results for *lunacm*.
ClientConnectIntervalMs	Interval in milliseconds between client connection attempts. **Default: 1000**
ClientConnectRetryCount	Maximum connection attempts between a client and a server. **Default: 900**
ClientEofRetryCount	Maximum command retries. **Default: 15**
ClientPoolSize	Number of threads in the thread pool available for client operations. **Default: 32**

Section/Setting	Description
ClientTimeoutSec	Time in seconds that a client waits for a response. This timeout applies to each retry attempt individually. **Default: 600** **NOTE** This entry does not appear in the default configuration file, but the default value applies to this timeout. You can manually add the entry if you wish to edit the timeout.

Section/Setting	Description
PartitionData00	The partition serial number, load balancer IP address or hostname, and load balancer port. Executing setenv, when configuring the Luna HSM Client, removes PartitionData00 and replaces it with values for ServerName and ServerPort. The following format is used: PartitionData00=<partition_serial_number>, <service_ip_address/hostname>, <service_port>; **NOTE** PartitionData00 is deprecated and included in the bundle to support legacy use cases. This may be removed in a future update. **CAUTION!** The Luna Cloud HSM service failover to the redundant datacenter uses a change to DNS to direct client traffic to a secondary datacenter. The client configuration file includes the FQDN for the Luna Cloud HSM service datacenter in the `REST = PartitionData00` section or the `REST = ServerName` section after executing `setenv` (`eu.hsm.dpondemand.io` or `na.hsm.dpondemand.io`). In the event of a failover the DNS record for FQDN is updated to point to the secondary datacenter. Ensure that the client is configured to use the domain name for the datacenter and to not configure any filtering based on the IP addresses. Failure to use the domain name and filtering IP addresses could result in the client being unable to failover to the secondary datacenter. **NOTE** Using Luna Cloud HSM 10.7.2 or higher, users are no longer required to run **setenv** to configure the client to connect to the Cloud HSM Service. However, **setenv** may still be used to configure the client for hybrid use cases or integrations where setting the ChrystokiConfigurationPath is required.

Section/Setting	Description
RestClient	Indicates that cvclient and associated tools are acting as REST clients. **Default: 1**
ServerName	The name of the Luna Cloud HSM server.
ServerPort	The port used for Luna Cloud HSM server traffic.
XTC	**NOTE** This section configures a connection to a Luna Cloud HSM and applies only to a Luna Cloud HSM. This section is not created automatically for clients obtained from the Thales Support Portal. See "Adding a Luna Cloud HSM Service" on page 72 for detailed instructions on adding a Luna Cloud HSM service client to a Luna HSM Client. This section governs Luna Cloud HSM service functionality only and is not related to the Luna REST API. Requires Luna HSM Client 10.2.0 or newer.
Enabled	Indicates that XTC (Transferable Token Channel) is enabled. This channel must be enabled for the client to communicate with a Luna Cloud HSM service. **Valid Values:** > **0**: XTC is disabled. > **1** (default): XTC is enabled.
TimeoutSec	Time (in seconds) before a cryptographic request expires. Timestamps are included in XTC headers, and the HSM rejects messages which have expired. **Valid Values: 1-600**
GemEngine	**NOTE** This section is not created automatically.

Section/Setting	Description
DisableCheckFinalize	Determines how the gem engine behaves for finalizing the cryptoki library. If an application has forking processes, then this causes the connection with the HSM to be shared between the parent and the child process which must be addressed for Linux/UNIX. **Valid Values:** > **0** (default): Perform pre-fork checking -- when crypto calls are made in the parent process, the cryptoki library is finalized after each crypto call. However, in the child process, the library is initialized and the connection to the HSM is maintained after crypto calls. The parent and child will have different connections to the HSM. > **1**: Perform post-fork checking -- the engine initializes the cryptoki library and maintains the connection to the HSM until the application terminates. If your application (own or 3rd party) is using OpenSSL and has forking processes, set this value to 0. Otherwise, setting the option to 1 will improve performance. Not used for Windows.

* If you intend to invoke a large number N for an M of N keyset (maximum is 16 splits), including also a backup set, you will need to increase the various PED timeout values well beyond the default values, in order to have enough time to comfortably complete the task. As a rough example, increase the PED's timeout for creating a keyset by a factor of 10. Altogether, the combined value works out to:

```
CommandTimeOutPedSet  >= ( DefaultTimeOut + PEDTimeout1 + PEDTimeout2 + PEDTimeout3 )
```

So, for example, in the Luna section of the .conf file (similar for the .ini file in Windows):

```
Luna =
{ DefaultTimeOut = 500000; PEDTimeout1 = 100000; PEDTimeout2 = 2000000; PEDTimeout3 = 20000;
KeypairGenTimeOut = 2700000; CloningCommandTimeOut = 300000; CommandTimeOutPedSet = 2620000; }
```

The longest such activity would be creating a 16-key split of a new-format orange PED Key (RPK), with duplicates, which might take a little more than half an hour at a comfortable pace with no interruptions. This is considered an extreme edge-case. Your situation will probably require settings somewhere between the defaults and the values suggested above.

**

NOTE From HSM firmware version 7.8.4 onward, Application IDs (APPId) are *encrypted*, with the following effects:

- Whenever firmware is upgraded from a non-APPID encrypted version (before firmware 7.8.4) to an encrypted APPID firmware version, the access ID shown in the logs will change.
- After the new firmware starts, the *encrypted* value of the same access ID for that application (for example, LUNACM) is now shown.
- The access ID shown also changes after every reset/restart of firmware version 7.8.4 onward because a new APPID encryption key (AEK) is created each time firmware starts up. The AEK is used by the crypto library of the APP to encrypt the access ID.
- Also whenever an Application is started it creates a new random access ID each time (unless fixed to a value [set AppId= under the Misc section] in the Configuration file).

Dynamic UserID Loading for Luna Cloud HSM Services

The UC Dynamic Loading feature is introduced in Luna HSM Client 10.5.0 for Luna Cloud HSM services. This allows each User Account and Authentication (UAA) user to have the ability to have one or more partition(s) associated with them. The DPoD tenant role can now configure multiple UAA Users and manage them in one place instead of managing each one separately. This will also allow customers to add multiple UserID's (combination of unique authtokenclientsecret, authtokenclientid and URI) without the need to restart the application after the addition of a new UserID.

The ability to load multiple partitions to the same UserID without impacting service to other users is also supported. If an attempt is made to add the same partition ID to a different user that will be ignored and a warning log will be generated.

When a new configuration is added, running the "slot list" command will display the new partition ID for that user.

NOTE The maximum amount of users to be added using the "MaxUserIDCount" variable is defaulted at 100. Multiple partitions for the same user will **NOT** have a sequential slot ID.

In Linux:

```
export AuthtokenConfigURI2=***********
export AuthTokenClientID2=**********
export AuthTokenClientSecret2=***********

export AuthtokenConfigURI3=***********
export AuthTokenClientID3=**********
export AuthTokenClientSecret3=***********
```

In Windows:

```
set var=AuthTokenConfigURI2=***********
set var=AuthTokenClientID2=**********
set var=AuthTokenClientSecret2=***********

set var=AuthTokenConfigURI3=***********
set var=AuthTokenClientID3=**********
set var=AuthTokenClientSecret3=***********
```


Updating the Luna HSM Client Software

To update the Luna HSM Client software, first uninstall any previous version of the Client. Then, run the new installer the same way you performed the original installation (refer to "Luna HSM Client Software Installation" on page 20).

The client uninstaller removes libraries, utilities, and other material related to the client, but does not remove configuration files and certificates. This allows you to install the newer version and resume operations without having to manually restore configuration settings and re-register client and appliance NTLS certificates.

> **TIP** Thales recommends verifying the integrity of the Luna HSM Client packages, by calculating their SHA256 hash values and comparing with the hash values posted on the Support Portal, before installing them on your client machines.
>
> You can use the sha256sum tool on Linux machines to calculate the SHA256 hash values.

CHAPTER 2: Client-Partition Connections

To allow clients to perform cryptographic operations, you must first give them access to an application partition on the HSM. This section contains the following information about client-partition connections:

Comparing NTLS and STC

Client access to the Luna Network HSM 7 is provided via two different types of channel:

NTLS	STC
> Ideally suited for high-performance applications and environments, executing many cryptographic operations per second. > Best used in traditional data center environments, where the client can be identified by its IP address or hostname; not recommended for use with public networks.	> Suited for higher-assurance applications requiring session protection beyond TLS; STC's message integrity and optional additional layer of encryption offers additional protection of client-to-HSM communications > Best for virtual and cloud environments where virtual machines are frequently cloned, launched, and stopped—such as when virtual machine auto-scaling is implemented to meet service-level agreements > Preferred in "HSM as a Service" environments where multiple customers, departments, or groups access partitions on a common HSM and want communication to be terminated on the cryptographic module within the Luna Network HSM 7 appliance > Suited for applications with moderate performance requirements

Network Trust Link Service

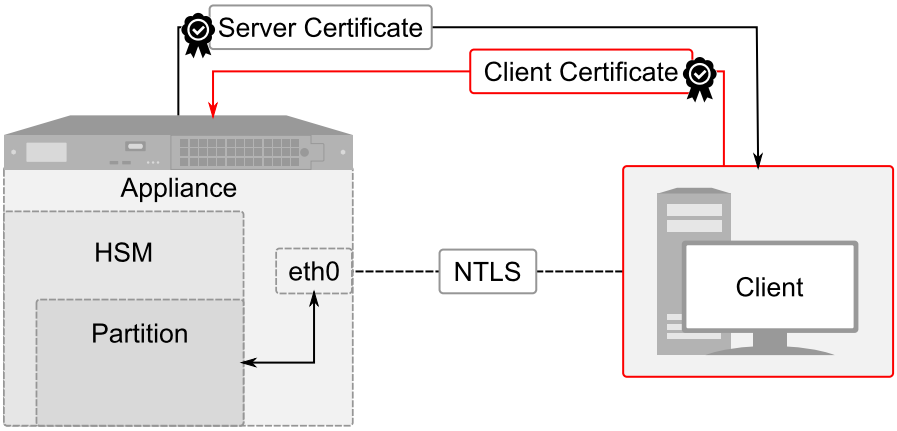
A Network Trust Link is a secure, authenticated network connection between the Luna Network HSM 7 appliance and a client computer. NTLS uses two-way digital certificate authentication and TLS data encryption to protect your sensitive data during all communications between HSM partitions on the appliance and its clients.

Using Luna Appliance Software 7.8.5 or newer, the Luna Network HSM 7 appliance can support up to 4000 simultaneous NTLS connections. Using older versions, the limit is 800 connections.

The certificates that identify appliances and clients can be self-signed or signed by a trusted Certificate Authority (CA).

NTLS Authenticated by Self-Signed Certificates

The figure below shows how a secure NTLS connection is created using self-signed certificates exchanged between the client and the appliance.



Self-signed certificates are created on both the appliance and the client. These certificates are exchanged to register the appliance and client with each other. Once registered, the client can access any partitions assigned to it in LunaSH. NTLS encrypts data between the network interfaces of the appliance (eth0 above) and client, but not between the network interface and the cryptographic module within the Luna Network HSM 7 appliance.

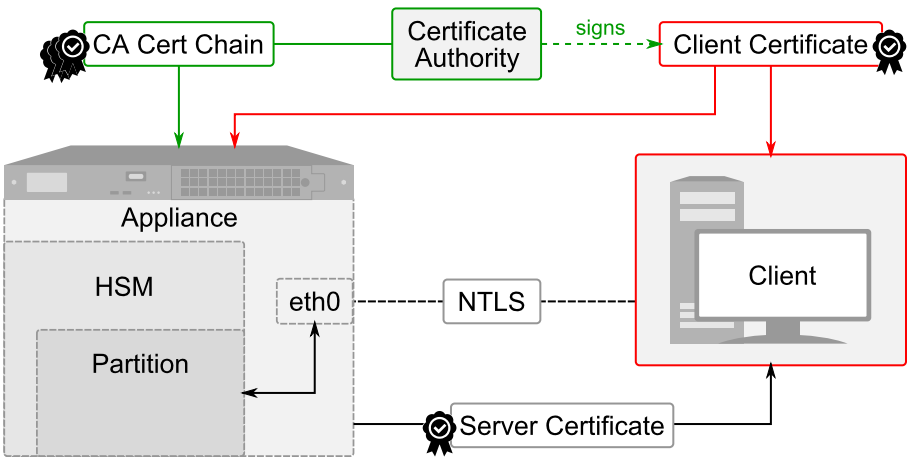
There are two methods of assigning partitions to a client via a self-signed NTLS connection:

- A multi-step procedure, performed by the appliance administrator and a client administrator
- A single-step procedure that automates the manual process. It can be used when the client administrator has **admin**-level access to the appliance, or through a custom registration account (see Creating a One-Step NTLS Registration Role).

See "Creating an NTLS Connection Using Self-Signed Certificates" on page 115.

NTLS Authenticated by a Certificate Authority on the Client Side Only

The figure below shows how a secure NTLS connection is created using a self-signed appliance certificate and a client certificate signed by a trusted CA. This can be a commercial third-party CA or your organization's own signing station. This method requires Luna HSM Client 10.1.0 or newer.

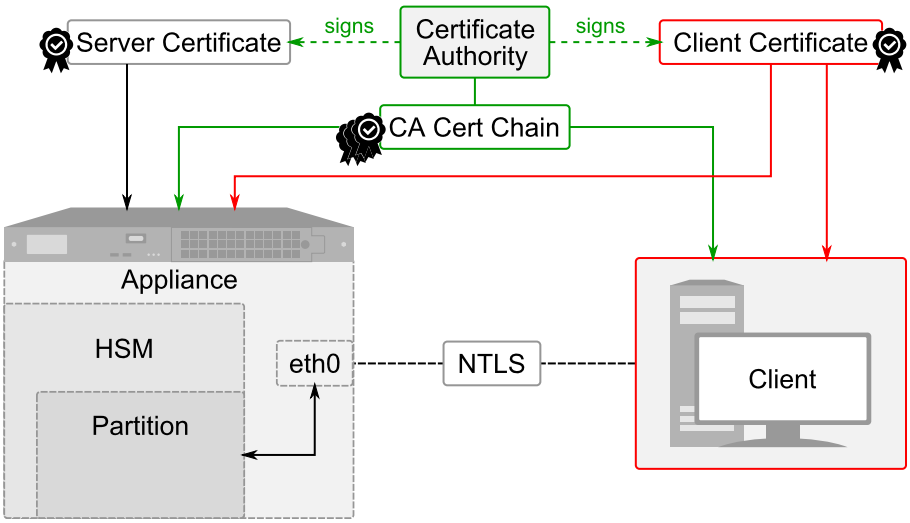


A Certificate Signing Request (CSR) is created on the client; this is an unsigned certificate that must be signed by your trusted Certificate Authority. The signed certificate is installed on the client, and the CA certificate chain is added to the trust store on the appliance. Finally, the client certificate is registered on the appliance and the client is then able to access any partitions that are assigned to it.

See "Creating an NTLS Connection Using Certificates Signed by a Trusted Certificate Authority" on page 120.

NTLS Authenticated by a Certificate Authority

The figure below shows how a secure NTLS connection is created using client and server certificates signed by a trusted Certificate Authority (CA). This can be a commercial third-party CA or your organization's own signing station. This method requires minimum Luna HSM Client 10.1.0 and Luna Network HSM 7 Appliance Software 7.7.0.



A Certificate Signing Request (CSR) is created on the appliance, the client, or both—this is an unsigned certificate that must be signed by your trusted Certificate Authority. Each signed certificate is installed on its respective appliance/client, and the required material from the CA certificate chain is added to the appliance and/or client trust stores (this step varies depending on whether you are using only CA-signed certificates or a combination of self-signed and CA-signed certificates). Finally, the client certificate is registered on the appliance and the client is then able to access any partitions that are assigned to it.

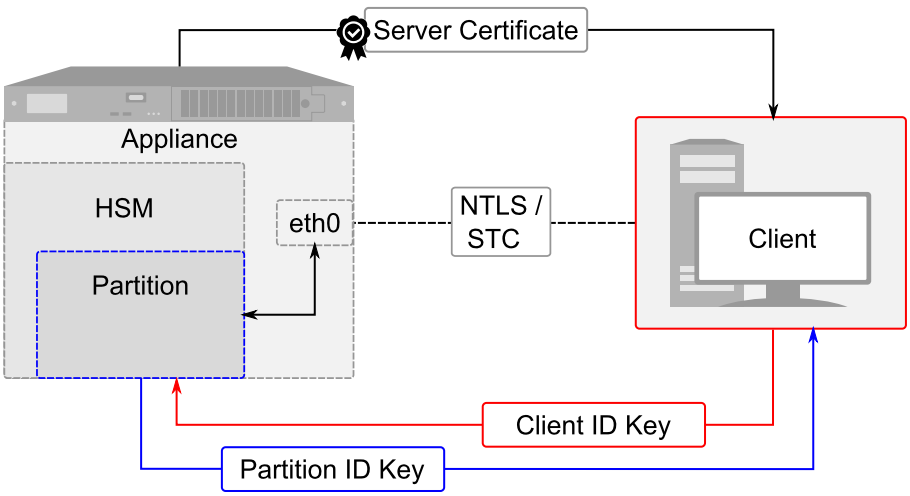
See "Creating an NTLS Connection Using Certificates Signed by a Trusted Certificate Authority" on page 120.

Secure Trusted Channel

If you require a higher level of security for your network links than is offered by NTLS, such as in cloud environments, or in situations where message integrity is paramount, you can use Secure Trusted Channel (STC) to provide very secure client-partition links, even over unsecured networks. STC offers the following features to ensure the security and integrity of your client-partition communications:

- All data is transmitted using symmetric encryption; only the end-points can decrypt messages
- Message authentication codes prevent an attacker from intercepting and modifying any command or response
- Mutual authentication of the HSM and the end-point ensure that only authorized entities can establish an STC connection

The figure below shows how an STC connection is made between the client and an application partition.



See the following procedures:

> "Creating an STC Connection" on page 128

> "Connecting an Initialized STC Partition to Multiple Clients" on page 133

> "Converting Initialized NTLS Partitions to STC" on page 137

Secure Tunnel Creation

Each STC connection is established between a client application and a specific partition on the HSM. As such, each application and partition pair goes through STC tunnel establishment individually. Before STC can create secure tunnels, trust must be established between the client and the partition through the manual exchange of public keys. Once trust has been established, unique session keys are created for each STC connection.

Session Re-Negotiation

Session keys for the tunnel are periodically renegotiated, as specified by the STC rekey threshold set for a partition. The rekey threshold specifies the number of API calls, or messages, that can be transmitted over an STC link to the partition before the session keys are renegotiated. You can adjust this value based on your application use cases and security requirements. See "Configuring STC Identities and Settings" on page 141 for more information.

Abnormal Termination

When a client shuts down a connection under normal conditions, it sends a secured message informing the HSM that the connection can be terminated. If a client terminates abnormally, or the network link is lost, the STC Daemon (STCD) detects the abnormal termination, and sends a message to the HSM informing it that the connection has ended, and the connection is closed. If the STCD sends an incorrect connection termination message, the client transparently re-establishes a new STC tunnel.

Secure Message Transport

Once a secure tunnel is established, any messages sent over the STC link are encrypted and authenticated using the unique session keys created when the tunnel is established. In addition, as with NTLS, all STC links use the TLS protocol to secure the link when it traverses a network.

Messages traversing an STC link are protected using Symmetric Encryption and Message Integrity Verification. These features are configurable for each partition and are used for each STC link to that partition. See "Configuring STC Identities and Settings" on page 141 for more information.

All messages protected outside the HSM

When STC is fully enabled on an HSM, all sensitive communications are protected all the way into the HSM. That is, any messages exchanged between a client application and the HSM use STC encryption, authentication, and verification from the client interface to the HSM interface, regardless of whether those links traverse a network, or are internal to the appliance (LunaSH to HSM) or Luna HSM Client workstation (client to HSM). All STC links that use a network connection also have the same network protection as NTLS links, that is, they are wrapped using SSL.

In addition to the STC connection between client and partition, you can also configure an STC connection between the HSM SO partition and the local services running on the appliance. This is referred to as the STC Admin channel.

See "Using the STC Admin Channel" on page 139.

Configurable options

The security features offered by STC are configurable, allowing you to specify the level of security you require, and achieve the correct balance between security and performance. Client/partition STC link parameters are configured using LunaCM. LunaSH/partition STC link parameters are configured using LunaSH.

Client and Partition Identities

The identity of a client or partition at an STC endpoint is defined by a 2048-bit RSA asymmetric public/private key pair, unique to each endpoint. Before you can establish an STC link, you must exchange public keys between the client and partition to establish trust.

The partition's private key is always kept in the HSM and is strongly associated with its partition. Only the partition security officer can retrieve the partition’s public key for delivery to a client. Upon receipt, the client administrator can use the public key hash to confirm its authenticity, before registering it. You can register multiple partition public keys to a client.

By default, the client’s identity pair is stored in a software token on the client’s file system, protected by the operating system’s access control systems. When using a software token, the client’s private key can be moved or copied to another host and used – so any client that possesses this identity pair is considered the authentic client. This enables an elastic client model for many applications.

Performance Consideration

STC introduces additional overhead to the communication channel. Depending on the application use case and cryptographic algorithms employed, this could have an impact on application performance.

Client to HSM Security Best Practices

While the Luna HSM is very secure, it is not the only component in the overall system. The HSM's application partitions become useful when client applications can communicate with those partitions, however this expands the potential attack surface. Good practices can go a long way toward minimizing that exposure.

This section suggests areas where practical choices and consistency can enhance security without sacrificing operational convenience.

Security around Password-authenticated systems

Two things must be secured:

- NTLS private key,
- partition password.

Securing the partition password

The partition password is needed when logging in, so the primary means of protecting the partition password is to protect the connection to the HSM via NTLS or STC. NTLS and STC certificates reside in a subdirectory of the Luna HSM Client directory, on every system that connects to an application partition on a Luna Network HSM.

To secure an enterprise connection to the HSM the following means are available:

- Use operating system controls and permissions on the client to prevent unauthorized users from accessing the key material.
- Use network segregation / software-defined networking or subnetting to prevent unauthorized machines from accessing the network HSM at all.
- Implement a full firewall security flow policy, to assist in preventing unauthorized network access, allowing only certain IP addresses and ports to be open to the network HSM.
- Practice proper password hygiene, in the form of a key and partition password-rotation policy, to prevent over-exposure should an NTLS key and/or partition password be compromised. (See sysconf user commands and Manage Appliance User Passwords.)

Securing the NTLS private key

To secure a PaaS*/container connection to the HSM the following means are available:

- make use of whatever vault or secret-store approach a given PaaS implementation provides but ensure that it is truly secure and not merely a pretense of "security"-by-obfuscation
- avoid bundling NTLS keys or partition passwords in VM/container images, but instead use the aforementioned PaaS vault/secret
- if the PaaS implementation provides some form of service mesh, then take advantage of it to further mitigate client private-key/partition-password vulnerability, as the service mesh would prevent an attacker from being able to use the key/password outside of the service mesh; this forces the attacker to use the exposed material in a more secure and monitored environment, where the attack could be outright prevented or at least detected much sooner.

(*PaaS = Platform as a Service)

Distinguished Name Client Certificate Verification

This section discusses how to require a Luna Network HSM 7 appliance to verify the full Distinguished Name of a registered Client, rather than just the Common Name, when validating an NTLS connection.

Using Luna Appliance Software 7.8.3 or older, NTLS client connections are validated against the Common Name (CN) field of the Distinguished Name (DN). The value for that field can be freely chosen, which can limit your organization's ability to strictly validate a given client.

Using Luna Appliance Software 7.8.4 or newer, the Luna Network HSM 7 provides the ability to have NTLS use the other attributes in the Distinguished Name (DN) field of the client certificate. The Distinguished Name is a set of key value pairs called RDN (Relative Distinguished Name) that uniquely identifies an entity holding the certificate. These attributes include:

- "C", (Country)
- "DC", (domain component - An individual attribute of the Domain Name. For example, DC=hsm, DC=thales, DC=com)
- "CN", (common name)
- "SN", (surname or family name)
- "GN", (given name)
- "ST", (state or province
- "T", (title)
- "O", (organization name)
- "OU", (organizational unit name)
- "serialNumber",
- "L", (locality)
- "emailAddress", (email address)
- "initials", (initials)
- "pseudonym", (pseudonym)

Acceptable input values are "a-zA-Z0-9_@.-" and spaces between values.

Commands are added to the Luna Shell (lunash) to implement this ability for NTLS (STC is not affected). On the Client side, no changes are made in this regard, as a client certificate can be created to match the server-side requirement (which is imposed as a filter when NTLS verifies a presented client certificate), and any deviation simply fails to authenticate.

The assignment operation is permitted only for an already registered client, thus the implication is that the Client certificate is created first, and then the DN filter in the HSM appliance is added, to match the configuration of the certificate. If you attempt to assign a DN filter for a client that is not registered, the system simply responds with "Error: Client not found".

Caveats

- The configured DN filter must be an exact match to the configuration of the client certificate, with the same RDNs in the same order.
- Only one DN filter can be configured per client.
- Multi-valued RDNs are not permitted.
- NTLS IP check validation is performed after the DN validation, and only if the DN validation is successful.

> Assigning a DN filter for a client, where a DN filter already exists, overwrites the current filter. You are warned what is about to happen and prompted to "proceed" or "quit", unless the **-force** option is used.

Workflow

1. Register a client with NTLS (client register).
2. Configure a DN filter for a client client dn assign.
3. Upon receiving a connection request, NTLS checks if a DN filter is configured for that client.
4. If a DN filter is configured, that filter is validated against the DN in the incoming client certificate.
5. If a DN filter is *not* configured, then NTLS resorts to using the CN field value in the client certificate to validate the connection.

> **NOTE** If a DN filter *is* assigned for a client, and the DN validation of that client's certificate fails, the connection is refused and *NTLS does not fall back* to CN-only validation.

To show a DN filter for a Client
To see if a DN filter already exists for a registered client, and if one exists, to show its content, use the client dn show command.

To add/assign a DN filter to a client
To add/assign a DN filter to a registered client, if one does not already exist, use the client dn assign command.

To delete a DN filter assigned to a client
To delete a DN filter assigned to a registered client, use the client dn delete command.

Creating an NTLS Connection Using Self-Signed Certificates

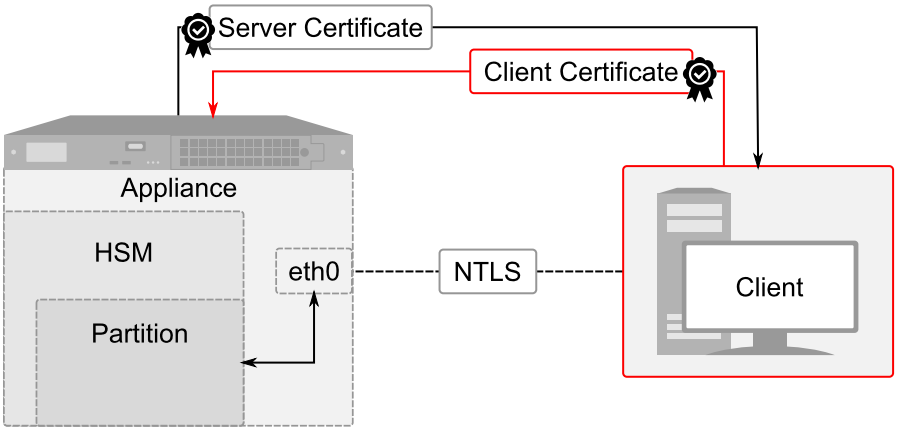
To create an NTLS connection, the Luna Network HSM 7 and the client must exchange certificates. Each registers the other's certificate in a trusted list. When both certificates are registered, the Network Trust Link is ready, and the appliance administrator can assign application partitions to the client for cryptographic operations. By default, this procedure uses self-signed certificates. To register your clients using certificates signed by a trusted Certificate Authority, see "Creating an NTLS Connection Using Certificates Signed by a Trusted Certificate Authority" on page 120.

> **NOTE** Secure Trusted Channel (STC) offers enhanced HSM-client message integrity, and an additional layer of protection for client-to-HSM communications, even over unsecured networks. To take advantage of this feature, see "Creating an STC Connection" on page 128. For more on the differences between NTLS and STC connections, see "Comparing NTLS and STC" on page 107.

There are two methods of assigning partitions to a client via a self-signed NTLS connection:

> "Multi-Step NTLS Connection Procedure" on the next page: performed by the appliance administrator and a client administrator

> "One-Step NTLS Connection Procedure" on page 118: automates the multi-step process. It can be used when the client administrator has **admin**-level access to the appliance, or through a custom registration account.



Multi-Step NTLS Connection Procedure

The multi-step procedure is performed by the appliance administrator and the client administrator.

Prerequisites

> You must have **admin**-level access to LunaSH on the appliance to register a client, or a custom account created to handle client registration (see Creating a One-Step NTLS Registration Role).

> By default, you do not need to log in as HSM SO. You can force the appliance to require HSM SO login for this procedure with lunash:> **sysconf forcesologin enable**.

> Luna HSM Client software must be installed on the client workstation (see "Luna HSM Client Software Installation" on page 20 in the *Installation Guide*)

> The client workstation must have an SSH client installed to provide secure shell access to the Luna Network HSM 7 appliance. The PuTTY SSH client (**putty.exe**) is included in the Windows client installation.

> Read/write access to the Luna HSM Client installation directory is required for the certificate exchange.

> The client workstation must have network access to the Luna Network HSM 7 appliance. The appliance auto-negotiates network bandwidth. See Recommended Network Characteristics for more information.

> **NOTE** Administration commands can take a few seconds to be noted by NTLS. If you have added or deleted a client, wait a few seconds before connecting.

To create a multi-step NTLS connection between the appliance and a client

1. On the client workstation, open a command prompt and navigate to the Luna HSM Client directory.

   > **NOTE** On Windows, ensure that you open a command prompt with Administrator privileges.

   - Windows: **C:\Program Files\SafeNet\LunaClient**
   - Linux/AIX: **/usr/safenet/lunaclient/bin**

- Solaris: **/opt/safenet/lunaclient/bin**

2. Use **pscp** or **sftp** to import the HSM Appliance Server Certificate (**server.pem**) from the appliance to the client workstation. You require **admin**- or **operator**-level account access to complete this step. If you do not have SSL access to the appliance, or a firewall blocks file transfer over the network, the appliance **admin** must provide this certificate by other secure means.

   TIP If you are importing certificates from multiple appliances to this client, rename each incoming certificate during the **pscp**/**sftp** transfer. This will prevent you from accidentally overwriting one **server.pem** certificate with another.

   TIP SCP is deprecated and SFTP is enabled by default for file transfer operations with Luna HSMs and clients. While you can continue using scp with Luna products, for the time being, eventually openSSL might discontinue scp support, and we recommend that you "future-proof" your operations by updating scripts and procedures to call sftp by preference.

   - **pscp** <user>**@**<host/IP>**:server.pem** <target_filename>
   - **sftp**<user>**@**<host/IP>**:server.pem** <target_filename>

     NOTE When using **pscp** or **sftp** over an IPv6 network, enclose addresses in square brackets.

   You must accept the SSH certificate the first time you open a **pscp**/**sftp** or SSH link. You can check the SSH fingerprint in LunaSH to confirm the secure connection.

   lunash:> **sysconf fingerprint ssh**

   If the HSM appliance IP or hostname is changed, SSH detects a mismatch in the HSM appliance's server certification information and warns you of a potential security breach. To resolve this issue, delete the server's certificate information from the client's known host file at: **/**<user home dir>**/.ssh/known_hosts2**, and re-import the server certificate.

3. Register the HSM Server Certificate with the client, using the **vtl** utility from the command line or shell prompt. If using a host name, ensure the name is reachable over the network (**ping** <hostname>). Thales recommends specifying an IP address to avoid network issues.

   >**vtl addServer -n** <Network_HSM_hostname/IP> **-c** <server_certificate>

4. Create a certificate and private key for the client. If you specify a client hostname, it must match exactly the hostname reported by the **hostname** command.

   CAUTION! If you are registering this client with multiple Luna Network HSM 7 appliances, you only need to complete this step once. Use the same client certificate for all appliances. If you recreate the client certificate and key, any existing NTLS connections will be broken.

   >**vtl createCert -n** <client_hostname/IP>

   The certificate and private key are saved to the <client_install_dir>**/cert/client** directory and are named <client_hostname/IP>**.pem** and <client_hostname/IP>**Key.pem**, respectively. The command output displays the filepath.

5. Use **pscp** or **sftp** to export the client certificate to the **admin** account (or an **admin**-level custom account) on the Luna Network HSM 7. The file arriving at the appliance is automatically placed in the appropriate directory. Do not specify a target directory.
   - **pscp** <cert_path/filename> **admin@**<host/IP>**:**[<target_filename>]
   - **sftp**<cert_path/filename> **admin@**<host/IP>**:**[<target_filename>]
6. Connect to the appliance via SSH or a serial connection, and log in to LunaSH using an **admin**- or **operator**-level account (see Logging In to LunaSH).
7. Register the client certificate with the appliance, selecting a client name that can be used to easily identify the client. Specify either the **-hostname** or **-ip** option, according to which one you used to create the certificate.

   lunash:> **client register -client** <client_name> {**-hostname** <client_hostname> | **-ip** <client_IP>}
8. [Optional] Verify the client registration.

   lunash:> **client list**

Now that the NTLS connection is established, the Luna Network HSM 7 appliance **admin** can assign partitions for the client to access (see "Assigning or Revoking NTLS Client Access to a Partition" on page 127).

One-Step NTLS Connection Procedure

The Luna HSM Client provides a one-step NTLS setup option, which automates the multi-step procedure described above.

The One-Step NTLS procedure is performed by the client administrator, and requires SSL access to an **admin**-level account (or a specialized NTLS registration account) on the Luna Network HSM 7 appliance. If you do not have SSL access to the appliance, an authorized user must provide the appliance certificate by other secure means, and you must use the multi-step procedure to manually register certificates.

This procedure uses **pscp**/**sftp** to exchange certificates over the network. If a firewall prevents this file transfer, the procedure will fail. You must exchange the certificates by other secure means and perform the manual procedure.

> **NOTE** If you are using Luna HSM Client 10.3.0 or older with Luna Network HSM 7 Appliance Software 7.8.0 or newer, you must update the client-side pscp and **plink** versions. You can copy suitable versions from a newer version of the Luna HSM Client, or you can go to PuTTY.org for the latest PuTTY version.

One-Step NTLS can only be used to create a new NTLS connection, and not to assign additional partitions to the client. If an NTLS connection already exists between the client and the appliance, or if one has already registered the other's certificate, the operation fails.

> **NOTE** ***Older Clients Can Fail to Complete One-Step NTLS with Newer Appliance Software***
>
> Newer Luna Network HSM 7 can have outdated (weaker) ciphers removed from file transfer protocols, as a security measure. If you have Luna HSM Client 7.3.0 or older installed, it might not be possible to negotiate a common cipher for a secure link. You might see an error similar to: `FATAL ERROR: Couldn't agree a host key algorithm (available: ecdsa-sha2-nistp256,ssh-ed25519).`
>
> To resolve this issue, you can download a new version of PuTTY from PuTTY.org at: https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html
>
> Copy pscp.exe and plink.exe to C:\Program Files\SafeNet\LunaClient and retry One-Step NTLS.
>
> Alternatively, install Luna HSM Client 10.4.0 or newer, which includes plink and pscp 0.76 or newer.

Luna Network HSM 7 Prerequisites

- The appliance certificate (**server.pem**) must be available on the appliance (see Generating the HSM Server Certificate).
- An application partition must be available on the HSM (see Creating or Deleting an Application Partition).
- The client must not have a certificate already registered on the appliance.

Luna HSM Client Prerequisites

- Client software must be installed (see "Luna HSM Client Software Installation" on page 20).
- The client administrator must have access to an **admin**-level account, or a specialized NTLS registration account, on the appliance (see Creating a One-Step NTLS Registration Role).
- The client administrator must know the name of an existing application partition that will be assigned to the client.
- The appliance must not have a certificate already registered with the client.
- If you are running Luna HSM Client 10.3.0 or older on a Linux platform, the **plink** executable included in the Luna HSM Client package requires one of the following 32-bit C/C++ standard libraries:
  - **glibc.i686** for Red Hat Enterprise Linux (RHEL) distributions. Install **glibc.i686** by running the following command:

    yum install glibc.i686
  - **gcc-multilib** for Ubuntu distributions. Install **gcc-multilib** by running the following command:

    sudo apt-get install gcc-multilib

  If you do not wish to install the C/C++ standard library, use the "Multi-Step NTLS Connection Procedure" on page 116 instead.

To create a One-Step NTLS connection between the appliance and a client

1. Launch LunaCM on the client workstation.

2. Initiate the One-Step NTLS procedure by specifying the appliance and client hostnames/IPs, and the name of the application partition to assign to this client. By default, the request is sent to the **admin** account, but you can specify any other account.

   lunacm:> **clientconfig deploy -server** <server_hostname/IP> **-client** <client hostname/IP> **-partition** <partition_name> [**-user** <appliance_username>] [**-password** <password>] [**-verbose**]

   > **NOTE** After you enter the account password, LunaCM appears to pause for 1-2 minutes while the registration procedure is completed. This is expected behavior.

The NTLS connection is now active, and the specified partition has been assigned to the client. If you want this client to have access to more partitions on this HSM, see "Assigning or Revoking NTLS Client Access to a Partition" on page 127.

To initialize the application partition, see "Initializing an Application Partition" on page 332.

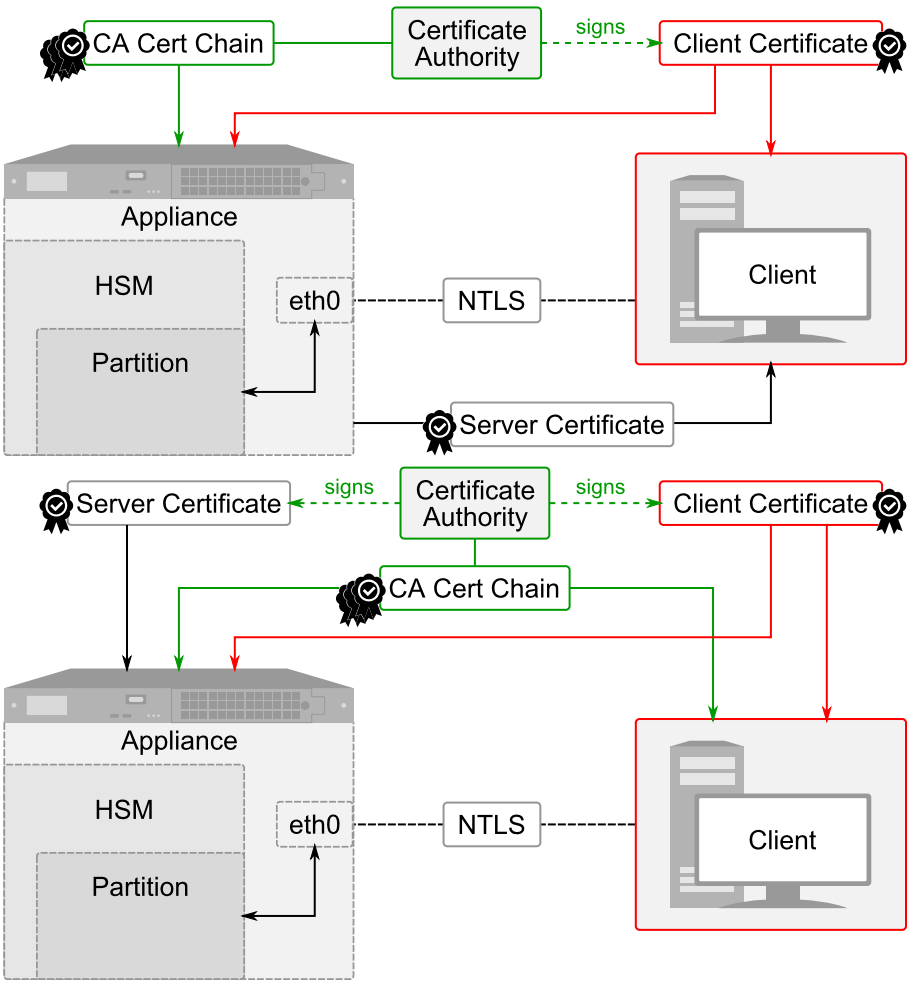
To restore a broken NTLS client connection, see "Restoring Broken NTLS or STC Connections" on page 145.

Creating an NTLS Connection Using Certificates Signed by a Trusted Certificate Authority

A trusted Certificate Authority (CA) can provide authentication for your NTLS connections. This can be a commercial third-party CA or your organization's own signing station. This type of connection is created in the following stages:

1. The Luna Network HSM 7 can be authenticated using either a self-signed certificate or a trusted CA, depending on your preference and installed Luna Appliance Software version:
   - "Registering a Self-Signed Appliance Certificate to the Client" on the next page
   - "Authenticating an Appliance Certificate With a Trusted CA and Registering the CA Chain" on page 122 (requires Luna Appliance Software 7.7.0 or newer)
2. The Luna HSM Client can be authenticated using either a self-signed certificate or a trusted CA:
   - "Creating a Self-Signed Client Certificate" on page 124
   - "Authenticating a Client Certificate With a Trusted CA and Registering the CA Chain" on page 124 (requires Luna HSM Client 10.1.0 or newer).
3. The Luna HSM Client must be registered using the self-signed client certificate, the client certificate and CA cert chain, or the CA cert chain alone (requires Luna Appliance Software 7.8.3 or newer), depending on your preference and installed Luna Appliance Software version.

   See "Registering the Client on the Appliance" on page 125.



Registering a Self-Signed Appliance Certificate to the Client

Use the following procedure to transfer the appliance's self-signed certificate to the client and register it.

Prerequisites

> You must have **admin**- or **operator**-level access to LunaSH on the appliance, or access to a custom LunaSH account.

> You must have Administrator privileges on the client workstation.

> **TIP** SCP is deprecated and SFTP is enabled by default for file transfer operations with Luna HSMs and clients. While you can continue using scp with Luna products, for the time being, eventually openSSL might discontinue scp support, and we recommend that you "future-proof" your operations by updating scripts and procedures to call sftp by preference.

To register the appliance certificate to the client

1. Use **pscp** (Windows) or **sftp** (Linux/UNIX) to import the HSM Appliance Server Certificate (**server.pem**) from the appliance to the client workstation. You require **admin**- or **operator**-level account access to complete this

step. If you do not have SSL access to the appliance, or a firewall blocks file transfer over the network, the appliance **admin** must provide this certificate by other secure means.

> **TIP** If you are importing certificates from multiple appliances to this client, rename each incoming certificate during the **pscp**/**sftp** transfer. This will prevent you from accidentally overwriting one **server.pem** certificate with another.

- **pscp** <user>**@**<host/IP>**:server.pem** <target_filename>
- **sftp** <user>**@**<host/IP>**:server.pem** <target_filename>

> **NOTE** When using **pscp**/**sftp** over an IPv6 network, enclose addresses in square brackets.

You must accept the SSH certificate the first time you open a **pscp**/**sftp** or SSH link. You can check the SSH fingerprint in LunaSH to confirm the secure connection.

lunash:> **sysconf fingerprint ssh**

If the HSM appliance IP or hostname is changed, SSH detects a mismatch in the HSM appliance's server certification information and warns you of a potential security breach. To resolve this issue, delete the server's certificate information from the client's known host file at: **/**<user home dir>**/.ssh/known_hosts2**, and re-import the server certificate.

2. Register the HSM Server Certificate with the client, using the **vtl** utility from the command line or shell prompt. If using a host name, ensure the name is reachable over the network (**ping** <hostname>). Thales Group recommends specifying an IP address to avoid network issues.

   >**vtl addServer-n** <Network_HSM_hostname/IP> **-c** <server_certificate>

Next, you must create a client certificate, either self-signed or to be signed by the CA:

> "Creating a Self-Signed Client Certificate" on page 124

> "Authenticating a Client Certificate With a Trusted CA and Registering the CA Chain" on page 124

Authenticating an Appliance Certificate With a Trusted CA and Registering the CA Chain

Use the following procedure to authenticate the appliance by having its certificate signed by your trusted CA.

Prerequisites

> You must have **admin**-level access to LunaSH on the appliance.

> **TIP** SCP is deprecated and SFTP is enabled by default for file transfer operations with Luna HSMs and clients. While you can continue using scp with Luna products, for the time being, eventually openSSL might discontinue scp support, and we recommend that you "future-proof" your operations by updating scripts and procedures to call sftp by preference.

To authenticate the appliance using a certificate signed by a trusted CA

1. Log in to LunaSH as **admin** (see Logging In to LunaSH).

2. Regenerate the Luna Network HSM 7 server certificate, specifying the **-csr** option to create a Certificate Signing Request (CSR)—an unsigned certificate to be signed by a Certificate Authority (CA). You have the option to specify other information about the certificate.

   > **CAUTION!** Regenerating the server certificate will break any existing NTLS/STC connections, when a subsequent restart of the service is performed.

   lunash:> **sysconf regenCert -csr**

3. Transfer the CSR (**serverCSR.pem**) from the appliance to a workstation using **sftp** or **pscp**.

   pscp <user>**@**<host/IP>**:serverCSR.pem** <target_filename>

   sftp<user>**@**<host/IP>**:serverCSR.pem** <target_filename>

   > **NOTE** When using **pscp** or **sftp** over an IPv6 network, enclose addresses in square brackets.

   You must accept the SSH certificate the first time you open an SFTP/PSCP or SSH link. You can check the SSH fingerprint in LunaSH to confirm the secure connection.

   lunash:> **sysconf fingerprint ssh**

4. Submit the **serverCSR.pem** certificate file to be signed by the Certificate Authority, as directed by the documentation of the particular Certificate Authority. You require the following artifacts from the CA:
   - Signed, base64-encoded, PEM-formatted client certificate
   - The CA's base64-encoded, PEM formatted certificate, including the root certificate

5. Upon receiving the signed server certificate, transfer the signed server certificate and the CA certificate chain to the **admin** user on the appliance using **sftp** or **pscp**. The files arriving at the appliance are automatically placed in the appropriate directory. Do not specify a target directory.

6. Log in to LunaSH as **admin** and register the CA certificate chain in the appliance trust store. Specify each certificate's filename, minus the **.pem** extension. Repeat this step until the entire certificate chain is registered.

   lunash:> **client addCA** <filename>

7. [Optional] Display a list of CA certificates registered on the appliance.

   lunash:> **client listCAs**

8. Install the signed appliance server certificate. This replaces the appliance's **server.pem** with the signed certificate.

   lunash:> **sysconf installCert** <filename>

9. Restart the NTLS, STC and CBS services.

   lunash:> **service restart** <service>

Next, you must create a client certificate, either self-signed or to be signed by the CA:

> "Creating a Self-Signed Client Certificate" on the next page

> "Authenticating a Client Certificate With a Trusted CA and Registering the CA Chain" on the next page

Creating a Self-Signed Client Certificate

Use the following procedure to create a self-signed client certificate.

Prerequisites

> Read/write access to the Luna HSM Client installation directory is required.

To create a self-signed client certificate

1. Create a certificate and private key for the client. If you specify a client hostname, it must match exactly the hostname reported by the **hostname** command.

   > **CAUTION!** If you are registering this client with multiple Luna Network HSM 7 appliances, you only need to complete this step once. Register the same client certificate for all appliances. If you recreate the client certificate and key, any existing NTLS connections will be broken.

   >**vtl createCert -n** <client_hostname/IP>

   The certificate and private key are saved to the <client_install_dir>**/cert/client** directory and are named <client_hostname/IP>**.pem** and <client_hostname/IP>**Key.pem**, respectively. The command output displays the filepath.

Next, you must register the client certificate on the appliance. See "Registering the Client on the Appliance" on the next page

Authenticating a Client Certificate With a Trusted CA and Registering the CA Chain

Use the following procedure to authenticate the client by having its certificate signed by your trusted CA.

Prerequisites

> You must have Administrator privileges on the client workstation.

To authenticate a client using a certificate signed by a trusted CA

1. On the client workstation, open a command prompt and navigate to the Luna HSM Client directory.

   > **NOTE** On Windows, ensure that you open a command prompt with Administrator privileges.

   - Windows: **C:\Program Files\SafeNet\LunaClient**
   - Linux/AIX: **/usr/safenet/lunaclient/bin**
   - Solaris: **/opt/safenet/lunaclient/bin**

2. Create a Certificate Signing Request (CSR) for the client—an unsigned certificate to be signed by a third-party Certificate Authority (CA). You must specify the client hostname or IP. You have the option to specify other information about the certificate.

> **CAUTION!** Regenerating the server certificate will break any existing NTLS/STC connections, when a subsequent restart of the service is performed.

> **vtl createCSR -n** <client_hostname/IP>

The certificate and private key are saved to the <client_install_dir>**/cert/client** directory and are named <client_hostname/IP>**CSR.pem** and <client_hostname/IP>**Key.pem**, respectively. The command output displays the filepath.

3. Submit the CSR file to be signed by your preferred or in-house Certificate Authority. You require the following artifacts from the CA:
   - Signed base64-encoded, PEM-formatted client certificate. The certificate must include the extension "`Enhanced Key Usage : client authentication`".
   - The CA's base64-encoded, PEM-formatted certificate chain, including the root certificate
4. Register the CA certificate chain in the client's trust store. Specify the full path and filename for each certificate. Repeat this step until the entire certificate chain is registered.

   > **vtl addCA -n** <cert_name> **-c** <cert_filepath/name>
5. Copy the signed client certificate to the following location in the Luna HSM Client directory:
   - Windows: **C:\Program Files\SafeNet\LunaClient\cert\client**
   - Linux/AIX: **/usr/safenet/lunaclient/cert/client/**
   - Solaris: **/opt/safenet/lunaclient/cert/client/**
6. Add the IP/hostname of any Luna Network HSM 7 appliance where the client will access application partitions. The CA chain used to sign the certificate must be added to the trust store of the appliance, as described in "Authenticating an Appliance Certificate With a Trusted CA and Registering the CA Chain" on page 122.

   > **vtl addServerNoCert -n** <IP/hostname>
7. [Optional] Edit **crystoki.ini**/**Chrystoki.conf** to enable server IP/hostname validation on the client. Do this only if the appliance server certificate was created with Subject Alternate Names (SANs).

```
[Misc]
ValidateHost=1
```

Next, see "Registering the Client on the Appliance" below

Registering the Client on the Appliance

Finally, you must register the client on the appliance. This is accomplished by either registering the client certificate itself, or providing its IP or hostname, depending on your installed version of the Luna Appliance Software.

Prerequisites

> The CA chain used to sign the certificate must be added to both the client's and the appliance's trust store.

> You must have **admin**-level access to LunaSH on the appliance.

NOTE The following procedure assumes that you are configuring an NTLS client-partition connection for the first time. If an NTLS client-partition connection has been established and the client certificate is being periodically replaced, for example in the case of client certificate renewals or deployment on multiple virtual machines, the new client certificate must be transferred to and registered with the appliance *only* if it was authenticated by the CA under a new host name or IP; that is, the appliance will continue trusting the CA-signed client certificate based on the registered certificate chain and maintain the NTLS client-partition connection if the new client certificate has been authenticated by the CA under a previously used client host name or IP. In such cases, where client certificates must be periodically replaced while maintaining an NTLS client-partition connection, Thales recommends that you replace the client certificate in the client and leave the expired client certificate in the appliance to avoid incurring application downtime.

TIP SCP is deprecated and SFTP is enabled by default for file transfer operations with Luna HSMs and clients. While you can continue using scp with Luna products, for the time being, eventually openSSL might discontinue scp support, and we recommend that you "future-proof" your operations by updating scripts and procedures to call sftp by preference.

To register a client to the appliance

1. If the Luna Network HSM 7 has Luna Appliance Software 7.8.1 or older installed, or if you are using a self-signed client certificate, transfer the client certificate to the **admin** account (or a custom account with **admin**-level privileges) on the Luna Network HSM 7.

   This step is not required if you are using Luna Appliance Software 7.8.3 or newer to register a CA-signed client; only the CA cert chain is required to authenticate the client.

   - **pscp** <cert_path/filename> **admin@**<host/IP>**:**[<target_filename>]
   - **sftp**<cert_path/filename> **admin@**<host/IP>**:**[<target_filename>]

2. Log in to LunaSH as **admin** or the custom **admin** account (see Logging In to LunaSH).

3. If you are registering a CA-signed client, verify that the appropriate CA is in the appliance's trust store.

   lunash:> **client listCAs**

   If the CA is not already in the appliance's trust store (for example, if you used a self-signed certificate to authenticate the appliance), you must register it now:

   a. Transfer the CA certificate chain to the **admin** user on the appliance using **sftp** or **pscp**.

   b. Register the CA certificate chain in the appliance trust store. Specify each certificate's filename, minus the **.pem** extension. Repeat this step until the entire certificate chain is registered.

      lunash:> **client addCA** <filename>

4. Register the client on the appliance. Specify the client's IP address or hostname.

   - Using Luna Appliance Software 7.8.1 or older, specify the IP or hostname that was used to name the certificate:

     lunash:> **client register -client** <clientname> {**-hostname** <hostname> | **-ip** <IPaddress>}

   - Using Luna Appliance Software 7.8.3 or newer, include the **-nocert** option:

     lunash:> **client register -client** <clientname> {**-hostname** <hostname> | **-ip** <IPaddress>} **-nocert**

You can now assign partitions to the client (see "Assigning or Revoking NTLS Client Access to a Partition" below).

Updating a Registered Client Certificate

If the client certificate is expiring, or your security policy requires you to rotate certificates on a schedule, you might prefer to perform the action without closing currently working connections. Using Luna Appliance Software 7.8.3 and newer, the client update command allows you to update the certificate such that it takes effect for all *new* connections, but current open connections remain open with the pre-update certificate. The CA issuing certificate for clients should be registered on the Luna Network HSM 7 appliance and the CA issuing certificate for the appliance should be registered on the client.

Assigning or Revoking NTLS Client Access to a Partition

Once an NTLS connection is established between the appliance and a client, the appliance **admin** must determine which application partitions the client can access. Usually this is done by the HSM Security Officer after they create the partition, but any **admin**-level appliance user can assign or revoke existing partitions to registered NTLS clients. You can assign a partition to more than one client at a time.

After you assign a partition to a client, the client can see the partition as a slot in LunaCM, initialize it, and use it for cryptographic applications.

Prerequisites

> An NTLS connection must be established between the appliance and the client (see "Client-Partition Connections" on page 107)

> The HSM SO must create the application partition on the HSM (see Creating or Deleting an Application Partition)

To assign a partition to a client

1. Connect to the appliance via SSH or a serial connection, and log in to LunaSH as **admin**, or a custom user with an **admin** role (see Logging In to LunaSH).
2. [Optional] Display a list of available partitions.

   lunash:> **partition list**
3. [Optional] Display a list of available registered clients.

   lunash:> **client list**
4. Assign a partition to a registered client.

   lunash:> **client assignPartition -client** <client_name> **-partition** <partition_name>
5. [Optional] Verify that the partition is assigned to the client.

   lunash:> **client show -client** <client_name>
6. If you registered the client by hostname, the appliance uses a DNS server to look up the device IP address. To ensure that the client is reachable in the event of a DNS failure, map the client hostname to its IP address, and save the mapping locally on the appliance.

   lunash:> **client hostip map -client** <client_name> **-ip** <client_IP>

7. Notify the client administrator that they can now access the partition and initialize it using LunaCM (see "Initializing an Application Partition" on page 332).

To revoke partition access from a client

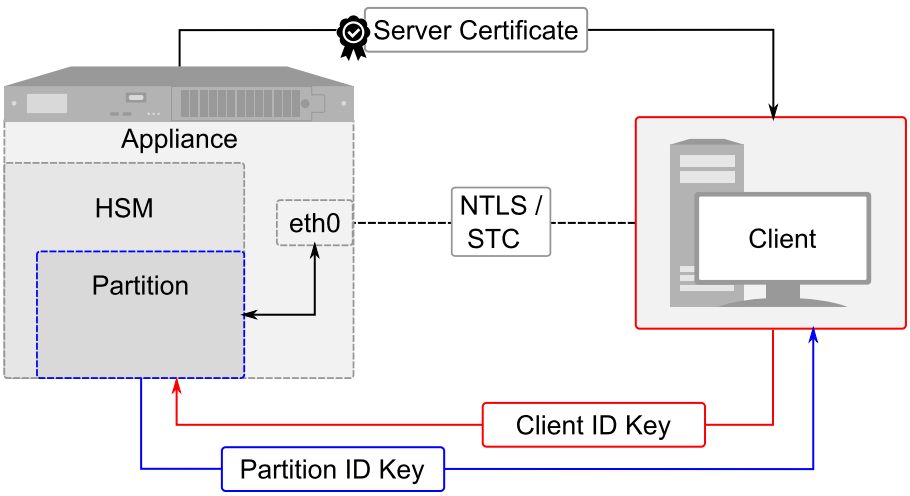
1. Connect to the appliance via SSH or a serial connection, and log in to LunaSH as **admin**, or a custom user with an **admin** role (see Logging In to LunaSH).
2. [Optional] Display a list of partitions currently assigned to the client.

   lunash:> **client show -client** <client_name>
3. Revoke the client's access to the partition.

   lunash:> **client revokePartition -client** <client_name> **-partition** <partition_name>

Creating an STC Connection

To create a Secure Trusted Channel (STC) connection, a partition identity is created directly on the partition, and the client and partition exchange identities. This allows end-to-end encryption of all communications between partition and client. This section describes how to establish an STC connection between a client and a new partition. The procedure involves the HSM SO and the administrator of the client workstation.



NOTE

> The Luna Network HSM 7 can create STC and NTLS channels to different clients as required. The client can also support both STC and NTLS links. However, all links from a specific client to a specific Luna Network HSM 7 appliance must be either STC or NTLS.

> STC links are not supported over an IPv6 network. You must use NTLS to make partition-client connections via IPv6.

> STC has been updated for the Luna 7.7.0 release, and the following dependencies apply:

- If you are using Luna HSM Firmware 7.7.0 or newer, STC requires Luna HSM Client 10.3.0 or newer.
- If you are using Luna HSM Firmware 7.4.2 or older, STC requires Luna HSM Client 10.2.0 or older.

For more information about the newer version of STC, refer to "Secure Trusted Channel" on page 161.

> To use functionality modules (FMs) with STC client connections, you require the newer version of STC, which is used in Client-V0/V1 partition connections. For more information, refer to "Secure Trusted Channel" on page 161.

1. "Preparing the HSM/Partition to Use STC" below
2. "Preparing the Client to Use STC" on page 131
3. "Creating a Client-Partition STC Connection" on page 131

Preparing the HSM/Partition to Use STC

To establish an STC connection between partition and client, you must first enable STC on the HSM (depending on your HSM firmware version), create one or more partitions and export their partition identities. These operations are performed by the HSM SO.

NOTE When you enable HSM policy 39: Allow Secure Trusted Channel on Luna HSM Firmware 7.4.2 or earlier, the following LunaSH commands are blocked to protect the integrity of any STC links that are created:

> **hsm stc identity create**

> **hsm stc identity initialize**

> **hsm stc identity delete**

> **hsm stc identity partition deregister**

If you plan to use STC on the admin channel and want to recreate the HSM identity first, see "Configuring STC Identities and Settings" on page 141 before continuing.

To prepare the HSM and partition(s) for STC connections

1. Connect to the appliance via SSH or a serial connection, and log in to LunaSH as **admin** (see Logging In to LunaSH).
2. Log in as HSM SO (see Logging In as HSM Security Officer).

   lunash:> **hsm login**

3. Enable HSM Policy 39: Allow Secure Trusted Channel. If you are using Luna HSM Firmware 7.7.0 or newer, this policy has been removed; skip this step.

   lunash:> **hsm changepolicy -policy 39 -value 1**

4. Create one or more new partitions for the client (see Creating or Deleting an Application Partition).

   lunash:> **partition create -partition** <partition_name> [**-size** <bytes>]

   > **NOTE** The following client identity storage overhead must be noted:
   > - Using Luna HSM Firmware 7.4.2 or older, and Luna HSM Client 10.2.0 or older (old STC partitions), each client identity registered to a partition uses 2392 bytes of storage on the partition.
   > - Using Luna HSM Firmware 7.7.0 or newer, and Luna HSM Client 10.3.0 or newer (updated STC partitions), each client identity registered to a partition uses 512 bytes of storage on the partition.
   >
   > Ensure that you create partitions large enough to store the identity of every client that will access the partition, in addition to cryptographic objects.

   When you create a partition, a partition identity key/key pair is automatically created.

5. For each partition, export the partition identity public key to the Luna Network HSM 7 file system. The file will be named with the partition's serial number. The command syntax is different depending on the Luna software/firmware version:

   - **Luna 7.7.0 or newer:**

     lunash:> **partition stcidentity export -partition** <partition_name>

     ```
     lunash:>partition stcidentity export -partition app_par1
     Successfully exported partition identity for partition app_par1 to file: 154438865304.pid
     ```

   - **Luna 7.4.x or older:**

     lunash:> **stc partition export -partition** <partition_name>

     ```
     lunash:>stc partition export -partition app_par1
     Successfully exported partition identity for partition app_par1 to file: 154438865304.pid
     ```

6. [Optional] View the partition identity public key hash. If you are not the client administrator, it is recommended that you provide it (via separate channel) so that the client administrator can verify the key's integrity as described in "Creating a Client-Partition STC Connection" on the next page. The command syntax is different depending on the Luna software/firmware version:

   - **Luna 7.7.0 or newer:**

     lunash:> **partition stcidentity show -partition** <partition_name>

   - **Luna 7.4.x or older:**

     lunash:> **stc partition show -partition** <partition_name>

7. If the client administrator does not have **admin** access to the appliance, or a firewall prevents you from using **pscp** or **sftp**, you must transfer these files from the HSM and provide them to the client administrator by other secure means:

   - The HSM Server Certificate (**server.pem**) from the Luna Network HSM 7.

- The partition identity public key for each partition the client will access (**154438865304.pid** in the example above).
- [Optional] The partition identity public key hash for each partition the client will access. This is recommended so that the client can verify the key's integrity before using the partition. Do not send the hash by the same means as the certificates.

Preparing the Client to Use STC

To access partitions on the HSM using STC, you must first create an STC token and identity on the client. These operations are performed by the client administrator.

> **CAUTION!** If you already have STC connections to partitions on other HSMs, skip this procedure and use the existing client token/identity. If you re-initialize an existing client token/identity, active STC connections to this client will be broken.

To prepare the client for STC connections

1. Open a command prompt or terminal and navigate to the Luna HSM Client directory.

   > **NOTE** On Windows, ensure that you open a command prompt with Administrator privileges.

   - Windows: **C:\Program Files\SafeNet\LunaClient**
   - Linux/AIX: **/usr/safenet/lunaclient/bin**
   - Solaris: **/opt/safenet/lunaclient/bin**
2. [Optional] Launch LunaCM and verify that the STC client token is uninitialized.

   lunacm:> **stc tokenlist**
3. Initialize the STC client token, specifying a token label.

   lunacm:> **stc tokeninit -label** <token_label>
4. Create a client identity on the token.

   lunacm:> **stc identitycreate -label** <client_identity>

   The STC client identity public key is automatically exported to:

   <client_install_directory>**/data/client_identities/**

Creating a Client-Partition STC Connection

To access STC partitions on the Luna Network HSM 7 appliance, you must first register the HSM Server Certificate. The STC connection is then created by registering one or more partition identity public keys to the client identity and enabling STC on the client. These operations are performed by the client administrator, with **admin** access to the Luna Network HSM 7 appliance. If you do not have **admin** access, or a firewall blocks file transfer over the network, the appliance **admin** must provide these files by other secure means.

To create a Client-Partition STC Connection

1. On the client workstation, use **pscp** or **sftp** to import the HSM Appliance Server Certificate (**server.pem**) from the appliance. You require the appliance's **admin** password to complete this step.

   > **TIP** If you are importing certificates from multiple appliances to this client, rename each certificate during the **pscp**/**sftp** transfer. This will prevent you from accidentally overwriting one **server.pem** certificate with another.

   pscp admin@<host/IP>**:server.pem** <target_filename>

   or

   sftp admin@<host/IP>**:server.pem** <target_filename>

2. Register the HSM Server Certificate with the client, using the **vtl** utility from the command line or shell prompt. If using a host name, ensure the name is reachable over the network (**ping** <hostname>). Thales recommends specifying an IP address to avoid network issues.

   > **vtl addServer -n** <Network_HSM_hostname/IP> **-c** <server_certificate>

3. [Optional] To check that you have successfully registered the appliance with the client, display the list of registered servers.

   > **vtl listServers**

4. Use **pscp** or **sftp** to import the partition identity public keys for all partitions you will access with STC. The files are named with the partition serial number (<partitionSN>**.pid**). You require the appliance's **admin** password to complete this step.

5. Register the partition identity public key to the client. Specify the path to the key file and, optionally, a label for the partition identity.

   lunacm:> **stc partitionregister -file** <partition_identity> [**-label** <partition_label>]

   ```
   lunacm:> stc partitionregister -file /usr/safenet/lunaclient/data/partition_
   identities/154438865304.pid -label app_par1

   Partition identity 154438865305 successfully registered.
   ```

   Repeat this step for each partition identity public key you wish to register to this client.

6. [Optional] If the HSM SO provided the partition identity public key hash, verify that the hashes match.

   lunacm:> **stc identityshow**

   If the hashes do not match, deregister the partition and contact your HSM SO.

   lunacm:> **stc partitionderegister -serial** <partitionSN>

7. Display the list of registered Luna Network HSM 7 servers to find the server ID of the appliance that hosts the partition(s).

   lunacm:> **clientconfig listservers**

8. Enable the STC connection.

> **CAUTION!** This forces the client to use STC for all links to the specified Luna Network HSM 7 appliance. If the server has partitions assigned to this client using NTLS, those connections will be terminated. Ensure you have registered the partition identity for all applicable partitions on this HSM before continuing.

lunacm:> **stc enable -id** <server_ID>

LunaCM restarts. If successful, the partition appears in the list of available slots.

9. [Optional] Set the active slot to the new partition and verify the STC link.

   lunacm:> **slot set -slot** <slot>

   lunacm:> **stc status**

The Partition SO can now initialize the partition (see "Initializing an Application Partition" on page 332). When the partition is initialized, the following actions are performed automatically:

- The client identity public key is registered to the partition.
- Partition policy 37: Force Secure Trusted Channel is enabled on the partition.

Once the partition is initialized, you can allow additional clients to connect to it using STC (see "Connecting an Initialized STC Partition to Multiple Clients" below).

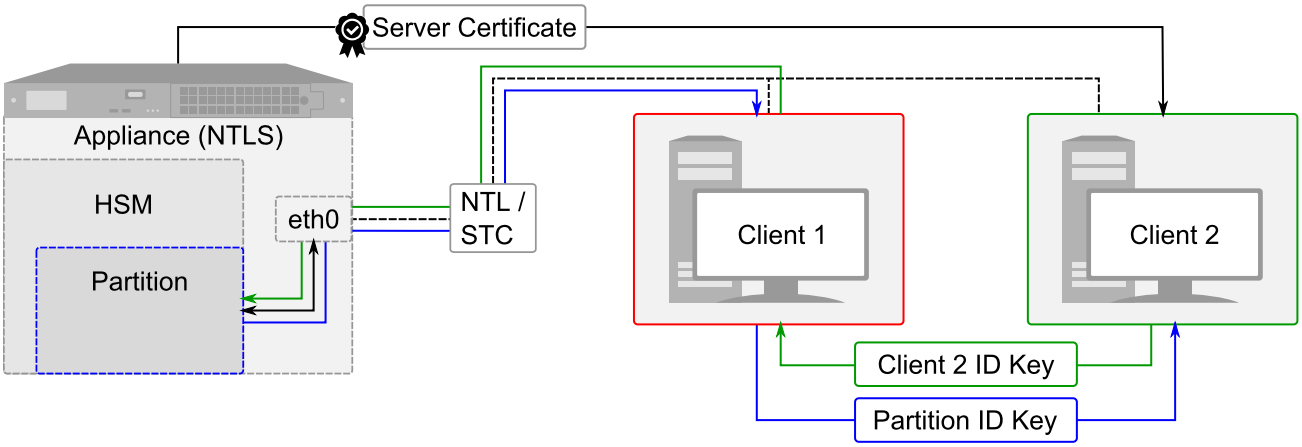
STC provides several configurable options that define the network settings for an STC link, and the security settings for the messages transmitted over the link. Although default values are provided that provide the optimal balance between security and performance, you can override the defaults, if desired. See "Configuring STC Identities and Settings" on page 141 for more information.

Connecting an Initialized STC Partition to Multiple Clients

Once an STC connection has been established between the partition and Client1, and the partition initialized, the Partition SO can allow other clients to access the partition. Since the Partition SO has control of the partition via Client1, they must provide the partition ID key to the Client2 administrator, and register Client2's ID key to the partition.

This procedure is completed by the Partition SO (using Client1) and the Client2 administrator in the following phases:

1. "Preparing the Additional Client to Use STC" on the next page
2. "Connecting an Additional Client to the Initialized STC Partition" on page 135



Preparing the Additional Client to Use STC

To access partitions on the HSM using STC, you must first create an STC token and identity on the client. These operations are performed by the client administrator.

> **CAUTION!** If you already have STC connections to partitions on other HSMs, skip this procedure and use the existing client token/identity. If you re-initialize an existing client token/identity, active STC connections to this client will be broken.

To prepare the client for STC connections

1. Open a command prompt or terminal and navigate to the Luna HSM Client directory.

   > **NOTE** On Windows, ensure that you open a command prompt with Administrator privileges.

   - Windows: **C:\Program Files\SafeNet\LunaClient**
   - Linux/AIX: **/usr/safenet/lunaclient/bin**
   - Solaris: **/opt/safenet/lunaclient/bin**
2. [Optional] Launch LunaCM and verify that the STC client token is uninitialized.

   lunacm:> **stc tokenlist**
3. Initialize the STC client token, specifying a token label.

   lunacm:> **stc tokeninit -label** <token_label>
4. Create a client identity on the token.

   lunacm:> **stc identitycreate -label** <client_identity>

   The STC client identity public key is automatically exported to:

   <client_install_directory>**/data/client_identities/**
5. [Optional] Display the client ID key hash. You can provide this hash to the Partition SO to verify the key's integrity.

   lunacm:> **stc identityshow**

6. Provide the following certificate/information to the Partition SO (Client1) via **pscp**, **sftp**, or other secure means:
   - Client2 identity public key
   - [Optional] Client2 identity public key hash (do not provide the hash by the same means as the key)

Connecting an Additional Client to the Initialized STC Partition

This procedure will allow an additional client (Client2 in the examples below) to access an initialized STC partition. The Partition SO (using Client1) and the Client2 administrator must complete the procedure.

Partition SO (Client1): To allow an additional client access to the STC partition

1. Ensure that you have received the following certificates/information from the Client2 administrator:
   - Client2 identity public key
   - [Optional] Client2 identity public key hash
2. On Client1, launch LunaCM and log in as Partition SO.

   lunacm:> **slot set -slot** <slotnum>

   lunacm:> **role login -role po**
3. Register the Client2 ID key to the partition. Specify a label for Client2 and the path to the key file.

   lunacm:> **stcconfig clientregister -label** <client_label> **-file** <path/client_ID>
4. [Optional] Display the hash for the Client2 identity.

   lunacm:> **stcconfig clientlist**

   If the displayed hash does not match the hash you received from the Client2 administrator, deregister the client identity and contact the Client2 administrator:

   lunacm:>**stcconfig clientderegister -label** <client_label>

   > **NOTE** If the Client2 administrator has **admin** access to the Luna Network HSM 7 appliance, and the partition identity public key is still available in the **admin** user's files on the appliance (lunash:> **my file list**), steps 5-7 are unnecessary.

5. Export a copy of the partition identity public key to the Client1 filesystem.

   lunacm:> **stcconfig partitionidexport**

   The partition ID key is named for the partition serial number (<serialnum>**.pid**) and automatically exported to:

   <Lunaclient_install_directory>**/data/partition_identities/**
6. [Optional] Display the partition ID key hash. You can provide this hash to the Client2 administrator to verify the key's integrity. Do not send the hash by the same means as the key.

   lunacm:> **stc identityshow**
7. Provide the following certificates/information to the Client2 administrator via **sftp**, **pscp**, or other secure means (see SCP and PSCP):
   - Partition identity public key

- [Optional] Partition identity public key hash (do not provide the hash by the same means as the key)
- HSM Server Certificate, located in:

  <Lunaclient_install_directory>**/cert/server/**<hostname/IP>**Cert.pem**

Client2 administrator: To create the client-partition STC connection

1. Ensure that you have received the following certificates/information from the Partition SO:
   - HSM Server Certificate (***.pem**)
   - Partition identity public key (***.pid**)
   - [Optional] Partition identity public key hash

   > **NOTE** If the Client2 administrator has **admin** access to the Luna Network HSM 7 appliance, and the partition identity public key is still available in the **admin** user's files on the appliance (lunash:> **my file list**), you can retrieve the HSM Server Certificate (**server.pem**) and the partition ID key (<partition_serialnum>**.pid**) directly from the appliance using **pscp** or **sftp**.

2. Open a command prompt or terminal window and navigate to the Luna HSM Client installation directory.
3. Register the HSM Server Certificate to the client.

   > **vtl addServer -n** <HSM_hostname/IP> **-c** <server_certificate>

4. Launch LunaCM and register the partition ID key to the client. Specify the path to the key file and an optional label for the partition.

   lunacm:> **stc partitionregister -file** <path/IDfile>**.pid** [**-label** <partition_label>]

5. [Optional] Display the hash for the partition ID key.

   lunacm:> **stc identityshow**

   If the displayed hash does not match the hash you received from the Partition SO, deregister the partition and contact the Partition SO:

   lunacm:> **stc partitionderegister -serial** <partition_serialnum>

6. Display the list of registered Luna Network HSM 7 servers to find the server ID of the appliance that hosts the partition(s).

   lunacm:> **clientconfig listservers**

7. Enable the STC connection.

   > **CAUTION!** This forces the client to use STC for all links to the specified Luna Network HSM 7 appliance. If the server has partitions assigned to this client using NTLS, those connections will be terminated. Ensure you have registered the partition identity for all applicable partitions on this HSM before continuing.

   lunacm:> **stc enable -id** <server_ID>

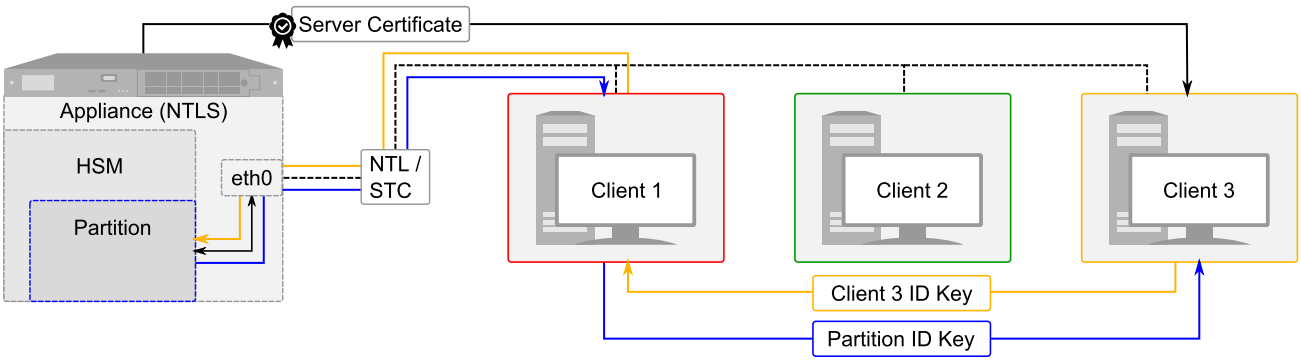
   LunaCM restarts. If successful, the partition appears in the list of available slots.

8. [Optional] Set the active slot to the new partition and verify the STC link.

   lunacm:> **slot set -slot** <slot>

lunacm:> **stc status**

Client2 can now access the partition via an STC connection. You can repeat the procedure to allow more clients to access the partition.



NOTE Each client identity registered to a partition uses 2392 bytes of storage on the partition. Ensure that the partition is large enough to store the identity of every client that will access the partition, in addition to cryptographic objects. If necessary, the HSM SO can re-size an existing partition (see Customizing Partition Sizes).

STC provides several configurable options that define the network settings for an STC link, and the security settings for the messages transmitted over the link. Although default values are provided that provide the optimal balance between security and performance, you can override the defaults, if desired. See "Configuring STC Identities and Settings" on page 141 for more information.

Converting Initialized NTLS Partitions to STC

If you have initialized partitions already assigned to a client using NTLS, you can use the following procedure to switch to a more secure STC connection. All of the client's assigned partitions on the specified Luna Network HSM 7 must be converted. It is not possible for a client to connect to multiple partitions on a single Luna Network HSM 7 using a combination of NTLS and STC.

The Partition SO must complete this procedure on the client workstation.

Prerequisites

> If you are using Luna HSM Firmware 7.4.2 or earlier, the HSM SO must set HSM Policy 39: Allow Secure Trusted Channel to **1** (ON).

To convert an NTLS partition-client connection to STC

1. Launch LunaCM and create the client token and identity.

NOTE This step is not required if you have already created a client token and identity. Verify using **stc identityshow**. If you recreate the client identity, you will have to re-register any existing STC partitions.

lunacm:> **stc tokeninit -label** <token_label>

lunacm:> **stc identitycreate -label** <client_identity>

The STC client identity public key is automatically exported to:

<client_install_directory>**/data/client_identities/**

2. Log in as Partition SO and export the partition ID key.

   lunacm:> **slot set -slot** <slotnum>

   lunacm:> **role login -name po**

   lunacm:> **stcconfig partitionidexport**

   The partition identity public key is named for the partition serial number (<partitionSN>**.pid**) and automatically exported to:

   <client_install_directory>**/data/partition_identities/**

3. Register the partition's public key with the client identity. Specify the path to the key file.

   lunacm:> **stc partitionregister -file** <path/filename>**.pid** [**-label** <partition_label>]

4. Register the client identity to the partition. Specify a label for the client and the path to the client identity file.

   > **NOTE** Each client identity registered to a partition uses 2392 bytes of storage on the partition. Ensure that there is enough free space before registering a client identity.

   lunacm:> **stcconfig clientregister -label** <client_label> **-file** <path/client_identity>

5. Depending on your firmware version, enable partition policy 37: Force STC Connection.

   - **Luna HSM Firmware 7.4.2 or earlier:** You must enable policy 37 to use STC. All clients accessing this partition must perform the STC registration procedure in steps 1-4.
   - **Luna HSM Firmware 7.7.0 or newer:** To enforce STC on all client connections to this partition, enable policy 37. If you want some clients to connect to this partition using NTLS, do not enable this policy.

   > **CAUTION!** Any existing NTLS client connections to this partition will be terminated when you enable policy 37. Ensure that all clients that access this partition have performed the STC registration procedure in steps 1-4 before you enable policy 37.

   lunacm:> **partition changepolicy -slot** <slotnum> **-policy 37 -value 1**

   > **NOTE** When you enable partition policy 37, the client loses contact with the partition until you enable the STC connection in step 7. This is expected behavior.

6. Repeat steps 2-5 for each NTLS partition on the same Luna Network HSM 7 you want to register to this client.
7. Find the server ID for the Luna Network HSM 7 hosting the partition and enable its STC connection. You will be prompted to restart LunaCM and all current sessions will be closed.

   > **CAUTION!** This forces the client to use STC for all links to the specified appliance. Any remaining NTLS links from this client to the appliance will be terminated. Ensure that you have completed steps 2-5 for each of this client's partitions before continuing.

lunacm:> **clientconfig listservers**

lunacm:> **stc enable -id** <server_ID>

If a partition is not visible as a slot when LunaCM restarts, disable STC for the server using lunacm:> **stc disable -id** <server_ID>, and ensure that you have activated partition policy 37.

STC provides several configurable options that define the network settings for an STC link, and the security settings for the messages transmitted over the link. Although default values are provided that provide the optimal balance between security and performance, you can override the defaults, if desired. See "Configuring STC Identities and Settings" on page 141 for more information.

Using the STC Admin Channel

Secure Trusted Channel (STC) can protect all communications to the HSM, including those originating on the Luna Network HSM 7 appliance. The STC admin channel is local to the appliance, and is used to encrypt data transmitted between the HSM and the local services running on the appliance (such as LunaSH, NTLS, and the STC service). The STC admin channel link is configured separately from the client-partition links, and can be enabled or disabled as required by the HSM SO.

> **NOTE** The STC admin channel is configurable using Luna Network HSM 7 Appliance Software 7.4.0 and older, and Luna HSM Firmware 7.4.2 and older. This feature is not available in Luna HSM Firmware 7.7.0 and newer.

Unique STC identities, each defined by a 2048-bit RSA asymmetric public/private key pair, exist on the HSM and the Luna Network HSM 7 appliance operating system. When you enable the STC admin channel, the HSM and the appliance create a trust link by exchanging public keys, and the private keys are used to encrypt all communications between them.

> **NOTE** Enabling the STC admin channel forces all client-partition links (NTLS or STC) to use STC for communications between the appliance and the HSM. This may affect NTLS link performance.

Enabling the STC Admin Channel

When enabled, all communications from the appliance operating system to the HSM are transmitted over the STC admin channel.

> **NOTE** When you enable HSM policy 39: Allow Secure Trusted Channel on Luna HSM Firmware 7.4.2 or earlier, the following LunaSH commands are blocked to protect the integrity of any STC links that are created:
>
> > **hsm stc identity create**
> > **hsm stc identity initialize**
> > **hsm stc identity delete**
> > **hsm stc identity partition deregister**
>
> If you plan to use STC on the admin channel and want to recreate the HSM identity first, see "Configuring STC Identities and Settings" on the next page before continuing.

To enable the STC admin channel

1. Open a LunaSH session on the appliance and log in as the HSM SO.

   lunash:> **hsm login**

2. If you have not already done so, enable HSM Policy 39: Allow Secure Trusted Channel.

   lunash:> **hsm changepolicy -policy 39 -value 1**

3. Enable the STC admin channel.

   > **CAUTION!** Enabling the STC admin channel is service-affecting. It causes an STC service restart, which temporarily terminates all existing STC links to the appliance. It also terminates the existing HSM login session.

   lunash:> **hsm stc enable**

Disabling the STC Admin Channel

When disabled, all communications from the appliance operating system to the HSM are transmitted, unencrypted, over the local bus.

> **NOTE** Disabling the STC admin channel is service affecting. It causes an STC service restart, which temporarily terminates all existing STC links to the appliance. It also terminates the existing HSM login session.

To disable the STC admin channel

1. Open a LunaSH session on the appliance and log in as HSM SO.

   lunash:> **hsm login**

2. Disable the STC admin channel.

   lunash:> **hsm stc disable**

Configuring the STC Admin Channel

STC provides several configurable options that define the network settings for an STC link, and the security settings for the messages transmitted over the link. Although default values are provided that provide the optimal balance between security and performance, you can override the defaults, if desired. See "Configuring STC Identities and Settings" below for more information.

Configuring STC Identities and Settings

Depending on your organization's security needs, you may need to customize some aspects of your Secure Trusted Channel (STC) connections. This can include encryption levels for message verification, request timeouts, periodic replacement of client identities, and more. Luna Network HSM 7 provides configurable options for customizing your STC connections.

> "Configuring STC Settings" below

> "Configuring STC Tokens and Identities" on page 143

Configuring STC Settings

STC provides configurable options that define network settings for an STC link, and security settings for the messages transmitted over that link. Although default values are provided that provide the optimal balance between security and performance, you can override the defaults, if desired.

> "Link Activation Timeout" below

> "Message Encryption" on the next page

> "Message Integrity Verification" on the next page

> "Rekey Threshold" on page 143

For client-partition STC links, these options are set individually for each partition. Using Luna Network HSM 7 Appliance Software 7.4.0 and earlier, they can be set by the HSM SO (using LunaSH) before the STC connection is established, or by the Partition SO (using LunaCM) after the STC partition is initialized. Using Luna Network HSM 7 Appliance Software 7.7.0 and newer, only the Partition SO can configure STC options, after the partition is initialized.

For the STC admin channel, the configuration applies to all communications between the HSM and local services on the appliance, such as LunaSH and NTLS. The STC admin channel options are set by the HSM SO.

> **NOTE** The STC admin channel is configurable using Luna Network HSM 7 Appliance Software 7.4.0 and older, and Luna HSM Firmware 7.4.2 and older. This feature is not available in Luna HSM Firmware 7.7.0 and newer.

Link Activation Timeout

The activation timeout is the maximum time allowed to establish the STC link before the channel request is dropped. You can use the following commands to specify the activation timeout for STC links to this partition.

STC admin channel (HSM SO, Luna HSM Firmware 7.4.2 and earlier)

lunash:> **hsm stc activationtimeout show**

lunash:> **hsm stc activationtimeout set -time** <seconds>

Uninitialized STC Partition (HSM SO, Luna HSM Firmware 7.4.2 and earlier)

lunash:> **stc activationtimeout show**

lunash:> **stc activationtimeout set -partition** <partition> **-time** <seconds>

Initialized STC Partition (Partition SO)

lunacm:> **stcconfig activationtimeoutshow**

lunacm:> **stcconfig activationtimeoutset -time** <seconds>

Message Encryption

By default, all messages traversing an STC link are encrypted. You can use the following commands to specify the level of encryption used (AES 128, AES 192, or AES 256) on all STC links to a partition, or to disable encryption on all STC links to a partition.

STC admin channel (HSM SO, Luna HSM Firmware 7.4.2 and earlier)

lunash:> **hsm stc cipher show**

lunash:> **hsm stc cipher enable** {**-all** | **-id** <cipher_id>}

lunash:> **hsm stc cipher disable** {**-all** | **-id** <cipher_id>}

Uninitialized STC Partition (HSM SO, Luna HSM Firmware 7.4.2 and earlier)

lunash:> **stc cipher show**

lunash:> **stc cipher enable -partition** <partition_name> {**-all** | **-id** <cipher_id>}

lunash:> **stc cipher disable -partition** <partition_name> {**-all** | **-id** <cipher_id>}

Initialized STC Partition (Partition SO)

lunacm:> **stcconfig ciphershow**

lunacm:> **stcconfig cipherenable** {**-id** <cipher_ID> **-all**}

lunacm:> **stcconfig cipherdisable** {**-id** <cipher_ID> **-all**}

Message Integrity Verification

By default, the integrity of all messages traversing an STC link is verified using an HMAC message digest algorithm. You can use the following commands to specify the algorithm used (HMAC with SHA 256, or HMAC with SHA 512).

STC admin channel (HSM SO, Luna HSM Firmware 7.4.2 and earlier)

lunash:> **hsm stc hmac show**

lunash:> **hsm stc hmac enable -id** <hmac_ID>

lunash:> **hsm stc hmac disable -id** <hmac_ID>

Uninitialized STC Partition (HSM SO, Luna HSM Firmware 7.4.2 and earlier)

lunash:> **stc hmac show**

lunash:> **stc hmac enable -partition** <partition_name> **-id** <hmac_ID>

lunash:> **stc hmac disable -partition** <partition_name> **-id** <hmac_ID>

Initialized STC Partition (Partition SO)

lunacm:> **stcconfig hmacshow**

lunacm:> **stcconfig hmacenable -id** <hmac_ID>

lunacm:> **stcconfig hmacdisable -id** <hmac_ID>

Rekey Threshold

The session keys and encryption keys created when an STC tunnel is established are automatically regenerated after the number of messages specified by the rekey threshold have traversed the link. You can use the following commands to specify the key life for the session and encryption keys used on all STC links to a partition. Specify the <threshold> value in millions of messages.

STC admin channel (HSM SO)

lunash:> **hsm stc rekeythreshold show**

lunash:> **hsm stc rekeythreshold set -value** <threshold>

Uninitialized STC Partition (HSM SO)

lunash:> **stc rekeythreshold show**

lunash:> **stc rekeythreshold set -partition** <partition_name> **-value** <threshold>

Initialized STC Partition (Partition SO)

lunacm:> **stcconfig rekeythresholdshow**

lunacm:> **stcconfig rekeythresholdset -value** <threshold>

Configuring STC Tokens and Identities

Each Luna HSM Client and partition that serves as an STC endpoint (including the HSM SO partition and the appliance operating system) has a unique identity, defined by a 2048-bit RSA asymmetric public/private key pair. The STC identity key pair is stored in the STC token associated with the client or partition (or the appliance or HSM). Before STC can create secure tunnels, trust must be established through the exchange of public keys.

Partition and HSM tokens and identities are created automatically and cannot be recreated. Client tokens and identities are created manually using LunaCM. The appliance token and identity is created automatically but can be recreated if necessary using LunaSH.

Under normal operating conditions, you should not need to recreate the STC tokens or identities. If you have operational or security reasons to do so, use the following commands:

Client Tokens and Identities

Use the following LunaCM commands:

Command	Description
stc identitycreate	Create a client identity on the STC client token.
stc identitydelete	Delete a client identity from the STC identity token.
stc identityexport	Export the STC client identify to a file.
stc identityshow	Display the client name, public key hash, and registered partitions for the STC client token.
stc partitionderegister	Remove a partition identity from the STC client token.
stc partitionregister	Register a partition to the STC client token.
stc tokeninit	Initialize a client token.
stc tokenlist	List the available STC client identity tokens.

STC Admin Channel Appliance Identity

> **NOTE** The STC admin channel is configurable using Luna Network HSM 7 Appliance Software 7.4.0 and older, and Luna HSM Firmware 7.4.2 and older. This feature is not available in Luna HSM Firmware 7.7.0 and newer.
>
> To ensure the integrity of existing STC connections, many of the following commands cannot be used when HSM policy 39: Allow Secure Trusted Channel is on. You must disable HSM policy 39 before recreating the admin channel identity.

Use the following LunaSH commands:

Command	Description
hsm stc identity create	Create a STC client identity for the STC admin channel.
hsm stc identity delete	Delete the STC admin channel client identity.
hsm stc identity initialize	Initialize the STC admin channel client token.
hsm stc identity partition deregister	Remove the HSM SO partition identity public key that is currently registered with the STC admin channel client token.
hsm stc identity partition register	Register the HSM SO partition identity public key with the STC admin channel client token.

Command	Description
hsm stc identity show	Display the name, public key hash, and registered partitions for the STC admin channel client token.

Restoring Broken NTLS or STC Connections

If a certificate used to authenticate NTLS or STC connections is deleted, regenerated, or has expired, the TLS handshake fails, and connections must be re-established before cryptographic operations can resume. This can be the result of HSM or partition zeroization (STC), regeneration/expiry of the HSM server certificate (**server.pem**) on the Luna Network HSM 7 appliance, or expiry of a client certificate. The procedures on this page will allow you to restore your broken connections, wherever possible.

> "Restoring NTLS/STC Connections after Regenerating the Server and/or Client Certificates" below

> "Restoring Connections After HSM Zeroization" on the next page

> "Restoring STC Connections After Partition Zeroization" on the next page

Restoring NTLS/STC Connections after Regenerating the Server and/or Client Certificates

If you regenerate the HSM server certificate (**server.pem**) and/or a client certificate, you must restore all NTLS and STC connections using the new certificate(s).

To restore NTLS connections using an HSM server certificate signed by a third-party CA

Restore NTLS connections using the procedure for "Authenticating an Appliance Certificate With a Trusted CA and Registering the CA Chain" on page 122. You do not need to re-install the CA certificate chain, only the new server certificate.

To restore NTLS connections using a client certificate signed by a third-party CA

Restore NTLS connections using the procedures for "Creating an NTLS Connection Using Certificates Signed by a Trusted Certificate Authority" on page 120. You do not need to re-install the CA certificate chain, only the new server certificate.

To restore NTLS or STC connections using a self-signed HSM server certificate

Appliance admin:

1. Using LunaSH, restart the NTLS and STC services.

   lunash:> **service restart ntls**

   lunash:> **service restart stc**

2. Provide the new HSM Server Certificate (**server.pem**) to each client by **pscp**, **sftp**, or other secure means.

Client administrators:

1. If you have access to LunaSH on the Luna Network HSM 7 appliance, you can retrieve the new HSM server certificate (**server.pem**) using **pscp** or **sftp**. Otherwise, the appliance administrator must provide it.

2. Delete the original server identity from the client.

   >**vtl deleteServer -n** <hostname/IP>

3. Register the new HSM server certificate with the client.

   >**vtl addServer -n** <hostname/IP> **-c** <cert_filename>

4. If you are restoring STC connections, launch LunaCM, find the new Server ID, and enable STC for the server.

   lunacm:> **clientconfig listservers**

   lunacm:> **stc enable -id** <server_ID>

Restoring Connections After HSM Zeroization

If the HSM is zeroized, all partitions and their contents are erased. New partitions must be created and assigned to their clients via the usual connection procedure.

NTLS connections

The HSM SO must re-initialize the HSM, create new partitions, and assign them to their respective registered clients (see "Assigning or Revoking NTLS Client Access to a Partition" on page 127). You do not need to register new appliance/client certificates unless they are regenerated.

STC connections

When the HSM is zeroized, the following occurs:

- HSM policy 39: Allow Secure Trusted Channel is turned off.
- The STC application partition identities are deleted along with the partitions.
- If the STC admin channel is enabled, the STC admin partition identity is deleted, breaking the STC admin channel between LunaSH and the HSM.

Create new STC connections using the standard procedure found in "Creating an STC Connection" on page 128. You can use the existing client tokens/identities. You do not need to register a new HSM server certificate unless it was regenerated using lunash:> **sysconf regencert**.

Restoring STC Connections After Partition Zeroization

The registered client identities used to validate STC clients are stored on each partition. Since they are not cryptographic objects, they are not backed up as part of a normal partition backup operation. If the partition is zeroized due to multiple login failures, the registered client identities are erased and regenerated. The HSM SO must provide the new partition identity to the client administrator, who must register the new identity.

To restore an STC connection after partition zeroization

HSM SO:

1. Log in to LunaSH and log in as HSM SO.

   lunash:> **hsm login**

2. Export the new partition identity key to the appliance filesystem.

   lunash:> **stc partition export -partition** <label>

3. Provide the new partition identity key (<partitionSN>**.pem**) to the client by **pscp**, **sftp**, or other secure means.

Client administrator:

1. If you have access to LunaSH on the Luna Network HSM 7 appliance, you can retrieve the new partition identity key (<partitionSN>**.pem**) using **pscp** or **sftp**. Otherwise, the HSM SO must provide it.
2. Launch LunaCM and de-register the original partition identity from the client.

   lunacm:> **stc partitionderegister -serial** <partitionSN>
3. Register the new partition identity key (<partitionSN>**.pem**)to the client.

   lunacm:> **stc partitionregister -file** <path/filename> [**-label** <label>]
4. Restart LunaCM.

   lunacm:> **clientconfig restart**

   You can now re-initialize the STC partition.

CHAPTER 3: V0 and V1 Partitions

Luna HSM Firmware 7.7.0 and newer preserve the traditional keys-in-hardware mode of operation and improve on it with fixes and security updates, while also adding the option to securely store more keys than will fit inside a crypto module. Traditional support of Luna features is retained in what we call version zero (V0) partitions, to distinguish from version one (V1) partitions that do the following:

- Implement "Scalable Key Storage" on page 165 to securely externally store and manage keys in vastly greater quantities than can fit inside a crypto module.
- Conform with FIPS SP 800-131A (revised).
- Comply with current and anticipated Common Criteria and eIDAS requirements (including "Per-Key Authorization" on page 190).

Partition type ==> Firmware 7.7.0 and newer	V0	V1
Description ==>	> Continues the Luna tradition of "keys always in hardware", for ongoing compatibility with existing applications and use-cases	> Adds the ability to store large numbers of keys safely within the security perimeter of the HSM in a database or file system, > Adds Per-Key Authorization capability, > ... both for RSS and other applications.

After updating to Luna HSM Firmware 7.7.0 or newer, any pre-existing partition becomes a V0 partition. If you create new partitions after the firmware update, the default is V0 with a further option to convert to V1 if there is ever a need. V0 partitions preserve your content, allow your applications to continue functioning with those partitions as they always did, and leverage the benefits of new fixes and security updates introduced with the newer firmware. To access the new features mentioned above, V0 partitions must be converted to V1 partitions after the update to Luna HSM Firmware 7.7.0 or newer. A summary of the origin of each partition type is provided in "The Origin of Each Partition Type" on the next page. For more information about converting V0 partitions to V1, refer to "Converting Partitions from V0 to V1 or V1 to V0" on page 161.

> **NOTE** V0 partitions have been designed to preserve as much compatibility as possible with your existing applications, while setting some necessary infrastructure for Luna HSM Firmware 7.7.0 features and future developments. However, the conversion of pre-existing partitions to V0 has implications for the firmware update process that must be considered before updating to Luna HSM Firmware 7.7.0 or newer. If you have not updated your Luna HSM to Luna HSM Firmware 7.7.0 or newer but plan to do so, Thales recommends reading Special Considerations for Luna HSM Firmware 7.7.0 and Newer before updating and reading the information in this section.

This section is of interest

- to customers who already have HSMs in operation and are looking to upgrade to Luna HSM Firmware 7.7.0 or newer, where possible, or
- to anyone wanting to know what differences in behavior to expect if you elect to change from the default V0 partition type to V1.

Version 7.x hardware can accept upgraded firmware, discussed in this section, and works best in conjunction with

- Luna HSM Client 10.3.0 or newer
- Luna Network HSM 7 Appliance Software 7.7.0 or newer

You might be looking to migrate existing keys and objects from their application partitions to updated partitions, both for older HSM generations being supplanted, and for current hardware being updated.

The ability to migrate keys and objects from pre-firmware 7.7 to new partitions is preserved. Some new administrative commands, and new options to pre-existing commands, have been added. Any API changes are as transparent as possible to maintain the function of existing customer and partner applications.

> **TIP** Luna Network HSM 7 Appliance Software 7.8.0 (or newer) and Luna HSM Client 10.5.0 (or newer), added cloning protocol 4 (CPv4) and Extended Domain Management, improving the process of migrations and HA and generally working with mixed groups of HSMs. The former barrier is removed:
>
> - between on-premises password-authenticated and multifactor quorum-authenticated partitions, or
> - between on-premises multifactor quorum-authenticated partitions and Luna Cloud HSM services (password authenticated).
>
> As long as you can authenticate to each of the parties, and the on-premises partition is Luna HSM Firmware 7.8.0 (allowing multiple domains to a partition) you can clone between them, in either direction, subject only to restrictions regarding repositories of greater vs lesser security.

The Origin of Each Partition Type

This section describes the origin of pre-firmware 7.7.0, V0, and V1 partitions.

Pre-Firmware 7.7.0 Partition

This is any partition on a Luna HSM from Luna 5.x and 6.x, and including Luna 7.x from firmware version 7.0 up to (but not including) version 7.7.0.

> **NOTE** When older Luna models like 5.x and 6.x are involved in migration of key material to Luna 7.x, there might be additional restrictions on the older units, such that firmware update paths must be followed before interaction with Luna 7.x HSMs is possible.
>
> Be aware also that applications making use of older HSMs might have created and used outdated key-types and cryptographic mechanisms that are not accepted (or are allowed only restricted use) by newer HSM versions, especially in FIPS-compliant operation.

V0 Partition

Any pre-existing partition, when HSM firmware is updated to version 7.7.0 or newer, becomes a V0 partition.

Any new partition, created using the default partition type (**-version** option) of the command "partition create" on page 1 (in lunash).

Any partition created on a firmware-7.7.0 (or newer) HSM, by an older client that does not know about the V0 / V1 distinction, is always V0.

The V0 status is a partition policy (#41).

V1 Partition

Any partition created with the [non-default] V1 option selected in the command "partition create" on page 1 (in lunash) becomes a V1 partition.

A V0 partition (whether created as V0 or a pre-existing partition that got upgraded with 7.7.0 firmware) can be converted to a V1 partition, without losing any contained objects. Do this by changing Policy 41 to value 1.

The Effects of Each Partition Type on HSM and Partition Functionality

The following sections describe how various aspects of HSM and application-partition functionality are affected by upgrading and by creating new partitions with the relevant creation-time options.

Partition Policy Considerations

Pre-firmware 7.7.0 Partition

There are no special considerations for pre-existing partitions, created with earlier firmware. Behavior with respect to Partition Policies is unchanged.

V0 Partition and V1 Partition

Partition policy 41 - Partition version

> Defaults to version zero (partition V0)
>
- when a new partition is created on a firmware 7.7.0 (or newer) HSM

- when a partition already exists on a pre-7.7.0 HSM that is then updated to firmware 7.7.0 (or newer)

> Becomes version one (partition V1)

- when **-version 1** is specified as a new partition is created on a firmware 7.7.0 (or newer) HSM
- when lunacm command **partition changepolicy -policy 41 -value 1** is issued

> V0 partition contents are preserved if/when policy 41 is set to version 1 (convert V0 partition to V1)

> The V1 status persists for the life of the partition unless command partition **changepolicy -policy 41 -value 0** is issued

- which reverts the partition to V0, and
- all partition contents are erased, and
- Scalable Key Storage and Per-Key Authorization are disabled.

> Some other policies are also interdependent with the partition version status. See "Partition Capabilities and Policies" on page 337 3, 7, 31, 32, and 40.

General HSM Behavior

Pre-firmware 7.7.0 Partition

There are no special considerations for pre-existing partitions, created with earlier firmware. Behavior with respect to cloning, backup, etc. is unchanged.

V0 Partition

Behavior defaults to V0 behavior, where keys reside in hardware.

Keys can be archived to a Backup HSM or shared for purposes of redundancy and load-balancing in an HA environment, but only when securely cloned among HSMs within the same encryption domain.

Your historic applications and integrations are supported.

V1 Partition

V1 option adds key export capability when you need to support larger numbers of keys than will fit inside an HSM, yet they must remain within the secure cryptographic boundary ("Scalable Key Storage" on page 165)

The exported keys are always encrypted by a master key (SMK) that remains within an HSM and can be securely copied only to another HSM that shares the same cryptographic domain.

Admin Partition Behavior with Pre-7.7.0 HSM / Pre-10.3.0 Client

Older client software (example 7.4 or 10.2.0) cannot create a V1 partition on an HSM with firmware 7.7.0 or newer.

Similarly, if a V1 partition is created on a 7.7.0 (or newer) Network HSM appliance and linked to an older Client, the client can see the remote partition, but cannot initialize or use the V1 partition. (Expect error CKR_ACCESS_ID_INVALID)

Client must be version 10.3.0 or newer to create and work with V1 partitions (see ** at the bottom of this page).

Cloning

Pre-firmware 7.7.0 Partition

There are no special considerations for pre-existing partitions, created with earlier firmware. Behavior with respect to cloning, Key Export etc. is unchanged.

V0 Partition

When an HSM's firmware is updated to version 7.7.0 or newer, any existing partitions become V0, and all contents are updated.

Keys/objects that have newer attributes, unknown to the older firmware, can be cloned if the newer attributes are at default value (unset), allowing them to be dropped by the older, receiving HSM. If a newer security-related attribute has been set, then the object is not cloned to an older HSM that is not aware of the attribute.

Keys can be imported from a pre-7.x HSM (5.x or 6.x) into a V0 partition on the firmware 7.7.0 (or newer) HSM, just as you would any object.

When cloning objects from HSM firmware earlier than 7.7.0 into V0 partition, the size of the object increases. As such, synchronization from partition in pre-7.7.0-firmware HSM to V0 partition may fail due to storage limitation.

V1 Partition

When a new V1 partition is created, or when a V0 partition is converted to V1,

- Objects cannot be cloned from a V0 partition or from a pre-7.7.0 partition into a V1 partition, and
- objects cannot be cloned from a V1 partition to a non-V1 partition.

> **NOTE** For older Luna versions, or situations where only cloning protocol version one (CPv1) is available, the library attempts to perform the individual actions of a cloning operation in sequence on the respective partitions, opening and closing a separate session for each object to be copied. If the policies and partition types on the source and target partitions are incompatible, the **partition clone** command (or an attempted HA synchronization) can fail with a message like CKR_DATA_LEN_RANGE while trying to clone. This can occur if a key object from the source partition is a different size than an equivalent object expected by the target.
>
> **UPDATE**: Using Luna HSM Firmware 7.8.0 and newer, when a cloning negotiation agrees on the use of CPv4, a call to clone multiple keys/objects launches a *single session for all the requested objects*, rather than opening and closing individual sessions for each object. The above portion of this note about mismatched sizes remains valid.

Objects can be cloned, unwrapped, or legacy-SKS-inserted, directly to a V1 partition (i.e., SIMinsert) - note that cloning in such case is a one-way operation; V1 partitions perform outbound cloning only for SMKs.

SMK (SKS Master Key)

Pre-firmware 7.7.0 Partition

This is not applicable before firmware 7.7.0, because SKS and therefore the SMK do not exist in Luna HSM version 7 prior to firmware 7.7.0.

V0 Partition

Each V0 partition has a unique Primary SMK generated when the Crypto Officer role logs in for the first time, but it cannot be seen or used while the partition is in V0 state. However, that SMK is in place, in case you ever change partition policy 41 to make the current partition a V1 partition.

V1 Partition

Each V1 partition has a unique Primary SMK generated when the Crypto Officer role logs in for the first time, but it can also accept a replacement Primary SMK via cloning (such as when joining a partition to an existing HA group)

Each V1 partition also has additional SMK slots or holding areas for:

- Rollover SMK,
- SMKs from earlier-model HSMs,
- FM SMK for partitions with Functionality Modules enabled.

The Primary SMK secret is used to extract and to insert keys/objects; all other SMK secrets can be used only to insert keys/objects.

Behavior at Partition Level

Pre-firmware 7.7.0 Partition

There are no special considerations for pre-existing partitions, created with earlier firmware. Behavior with respect to cloning, backup, HA, etc., is unchanged.

V0 Partition

Whether pre-existing and updated, or newly created, V0 partitions should be generally indistinguishable from previous-firmware partitions -- continuing to work with your applications -- with provisos mentioned below. Cloning or Export/Import function as expected:

- from older versions (pre-firmware 7.7),
- to-or-from V0 partitions (firmware 7.7.0 or newer),
- but not back to older-version partitions.

Client versions earlier than 10.3 do not support expression of V0/V1 partition types (policy 41) for Partition Policy Template (PPT)

- **partition showPolicies -exportTemplate** does not report V0/V1 partition policy.
- **partition init -label** <somelabel> **-applytemplate** <template file> supports management of V0/V1 partition template correctly.
- Partition initialization without V0/V1 partition policy succeeds with correct default value (V0).

V1 Partition

Only the SKS Master Key (the SMK) is cloned from partition to partition, or from HSM to HSM for HA or for Backup/Restore. All other objects are encrypted with the SMK and Extracted for external storage or retrieved from external storage and inserted for use within the HSM.

Structure of Partition

Pre-firmware 7.7.0 Partition
There are no special considerations for pre-existing partitions, created with earlier firmware. Structure is unchanged until you update firmware to version 7.7.0 or newer.

V0 Partition
Partition structure is generally as for pre-7.7.0 partition, but with some updated overhead taking up some space; a completely filled pre-7.7.0 partition would need more room for objects after migration/firmware-update, but this is taken care of by partition size increases, as needed, during firmware update. The increase is enabled by an increase in available memory that is also part of the update process (see below).

V1 Partition
When a new partition is created at V1 or a V0 partition is converted to V1, the new structural overhead applies, including the space allotted for Primary and other SMKs.

Also, some keys can have new/additional attributes necessary to satisfy newer crypto and security standards.

Objects in a Partition

Pre-firmware 7.7.0 Partition
Object characteristics and behavior are unchanged until you update firmware to version 7.7.0 or newer.

V0 Partition
Memory allotment is increased (from pre-7.7) to allow increased partition size, all pre-existing keys and all new keys receive new attributes (if applicable to key type) but those attributes are not used for anything in V0 partitions (see * at the bottom of this page).

> **NOTE** The doubling of the memory allotted to partitions does not double the key capacity of partitions. It increases headroom to allow for increases in the data space required by keys, to accommodate new features and abilities, and does not alter published baselines for key capacity.

V1 Partition
Memory allotment is increased (from pre-7.7) to allow increased partition size. There are no pre-existing keys in new V1 partitions, and all new keys receive the new attributes.

> **NOTE** The doubling of the memory allotted to partitions does not double the key capacity of partitions. It increases headroom to allow for increases in the data space required by keys, to accommodate new features and abilities, and does not alter published baselines for key capacity.

Memory

Pre-firmware 7.7.0 Partition

Memory availability and usage are unchanged until you update firmware to version 7.7.0 or newer.

V0 Partition and V1 Partition

Size limit with FW version< 7.7.0	New size limit after upgrading to FW version >= 7.7.0
2 MB	4 MB
16 MB	32 MB
32 MB	64 MB

Example after mid-size update:

```
Partition Storage:
                    Total Storage Space:  3306327
                    Used Storage Space:   0
                    Free Storage Space:   3306327
                    Object Count:         0
                    Overhead:             15560
```

> **NOTE** The doubling of the memory allotted to partitions does not double the key capacity of partitions. It increases headroom to allow for increases in the data space required by keys, to accommodate new features and abilities, and does not alter published baselines for key capacity.

In summary, if you could store X-number of a given size of keys on your partition or HSM, then you can still store them all after 7.7.0 f/w update. The increase, at each allotment level was chosen to accommodate increased partition overhead and object size changes, plus some extra just in case (see * at the bottom of this page).

Behavior at Key Level

Pre-firmware 7.7.0 Partition

Key object characteristics and behavior are unchanged until you update firmware to version 7.7.0 or newer.

V0 Partition and V1 Partition

Some key types and algorithms might have constraints on the allowed uses of some older key and algorithm types and sizes, due to changes in the security and threat environments over time.

Check the latest mechanism summary tables in the SDK.

As with any firmware version update, some key types might be newly added, and some existing key types might have attributes that are used differently from previous. Where this might become apparent is when keys are replicated in an HA group with a mix of HSM firmware versions - for example, an existing application might

attempt to make use of key sizes that are recognized as too small and of low security by group members with newer firmware, or a newer application might make use of newer attributes that are not recognized by older firmware (see Per-key Authorization, below, for example).

Partition Policy Template

Pre-firmware 7.7.0 Partition

partition showPolicies -exportTemplate generates a template file containing current policy settings

partition init -label <label> **-applytemplate** <template file> applies an existing template with the contained policy settings

V0 Partition

Both commands behave the same as in previous versions. V0 partitions have some policies that do not exist in pre-7.7.0 partitions. As long as none of the policies in your template conflict with the state of a new policy, your pre-existing templates should work correctly. Any policy that is not mentioned in a template is set to its default value when the template is applied.

If there is a mismatch between template policies and the default values of newer or dependent policies, then the attempt to apply the old policy would fail with CKR_FAILED_DEPENDENCIES.

You have the option to edit a policy file before applying it, to add newer policies.

V1 Partition

The default for new partition creation with firmware 7.7.0 (or newer) is a V0 partition. You could apply your PPT to creating a V1 partition only by pre-editing the policy template file to include setting policy 41 to a value of 1.

See also the lists of dependencies below the table at "Partition Capabilities and Policies" on page 337.

Per-Key Authorization

Pre-firmware 7.7.0 Partition

This is a new feature with firmware 7.7.0 and has no bearing on partitions at earlier firmware versions.

V0 Partition

Partition Policy 40 Enable Per-key Authorization Data defaults to 0 (zero, or off) for V0 partitions, and cannot be turned on. V0 partitions do not immediately provide per-key authentication data or attributes upon conversion to this partition version and key objects received from partitions on other HSMs (7.x-pre-7.7.0 HSMs, as well as 5.x and 6.x HSMs) come into the partition without auth data.

V1 Partition

Partition Policy 40 Enable Per-key Authorization Data defaults to 1 (one, or on) for V1 partitions, and can be turned off for performance.

Relevant keys have attributes, and are automatically assigned the CKA_AUTH_DATA attribute, that allow HSM owner to provide individual users access to specific keys for Sole Ownership and Control, such as in Remote Signing and Sealing applications.

Special considerations apply when importing key objects without per-key auth data into a Luna 7.7.0+ partition that enforces per-key authentication. For more information, refer to Migration Scenarios for Per-Key Auth.

Multifactor Quorum Authentication

Pre-firmware 7.7.0 Partition

Behavior of partitions at earlier firmware versions continues as-is (except see exceptions in next paragraph, if PEDs are updated).

V0 Partition and V1 Partition

> Luna PEDs with firmware 2.7.4 or 2.9.0 or newer can function with Luna HSM 7.7.0 and newer HSMs that have been updated for compliance with eIDAS/Common Criteria and NIST 800-56A standards.

> When an HSM is at firmware version 7.7.0 or newer, it verifies that any connecting PED is at PED firmware 2.7.4 or firmware 2.9.0, or the HSM refuses the connection and issues an error.

> The RPV of an orange PED key, created with PED firmware 2.7.4 or 2.9.0 against a firmware 7.7.0 HSM has additional features compared to previous RPV, necessary for current authentication standards.

> PEDs with firmware older than 2.7.4 or 2.9.0 can do the following:

- Use a newer RPV, which was created with PED firmware 2.7.4 or 2.9.0 against a Luna HSM with firmware 7.7.0. However, the PED remains unaware of the newer security components.
- Duplicate a newer RPV onto another orange key while only imprinting the older security components; that is, the newer security components are lost. The duplicated RPV can then be used with pre-firmware-7.7.0 HSMs, but since the newer security components are missing, the 'duplicate' orange key (and any copy of it) cannot be used with HSMs at version 7.7.0 or newer.

> For Remote PED operation,

- any blank RPK must first be provisioned with new Critical Security Parameters (CSP) via a local PED connection;
- the content of a previously provisioned orange Remote PED Key (RPK) with old CSP must be migrated to new CSP. The same is true for a newer-style RPV that had the newer security components stripped by copying with a non-updated PED, as described above. For more information, refer to Migrating Existing Orange Remote PED Keys.
- When the ped vector init' command raises the PED prompt about "reuse an existing keyset?", this will lead to RPK migration (old to new).

> For Local PED, the local-connection handshake is now similar to that being used for updated, improved-security connections with Remote PED.

Client Software Interaction

Pre-firmware 7.7.0 Partition

Newer client software can include commands and options that are not applicable to partitions in older-firmware HSMs.

V0 Partition
Older client software (example 7.4) can create only a V0 partition on an HSM with firmware 7.7.0 or newer (see ** at the bottom of this page).

V1 Partition
Older client software (example 7.4 or 10.2.0) cannot create a V1 partition on an HSM with firmware 7.7.0 or newer.

Similarly, if a V1 partition is created on a 7.7.0 (or newer) Network HSM appliance and linked to an older Client, the client can see the remote partition, but cannot initialize or use the V1 partition. (Expect error CKR_ACCESS_ID_INVALID)

Client must be version 10.3.0 or newer to create and work with V1 partitions (see ** at the bottom of this page).

Client-Mediated High Availability

Pre-firmware 7.7.0 Partition
HA behaves as it always has, for pre-7.7.0 HSMs.

V0 Partition
Generally as-is (backward compatible) except for any provisos around permissibility of certain mechanisms and key sizes, such as in FIPS 140 approved configuration (formerly FIPS mode), and the usual considerations where an HA group should have all members at the same firmware .

Migration must be done via Luna Backup HSM while any application partitions on an HSM being updated to firmware 7.7.0 (or newer) must be removed from any HA group, at the time. The partitions can become members of HA groups after all are at the newer version.

Key/object replication among HA group members continues to use cloning.

V0 primary partitions in an HA group cannot synchronize with other members on a pre-7.7.0-firmware HSM.

A V0 partition on an appliance with HSM firmware 7.7.0 (or newer) can be added to an existing HA group that already has HA members made up of partitions from an HSM with pre-version-7.7.0 firmware.

V1 Partition
With V1 partitions, HA must function with SKS as the method of object / key replication among members, rather than cloning. Because this type of HA is client-mediated, you need Luna Client 10.3.0 or newer.

V1 partition can be added to an existing HA group that already has HA members made up of partitions from a pre-7.7.0-firmware HSM. However, when V1 partition becomes the primary member of the HA group, synchronization with remaining member of the HA group will no longer function.

An HA group with V1 partitions must have all members at V1 and all members sharing the same SMK.

A V1 partition cannot be a member of more than one HA group unless both groups have the same SMK. If a member of an existing HA group is added to a different HA group with a different SMK, the new member takes on the SMK of the new HA group and ceases to function properly in its original group (and should be removed).

High Availability Indirect Login

This type of HA is set up and managed by means of the HA Indirect Login API (a.k.a. "roll your own HA"), and does not rely on the Client.

Pre-firmware 7.7.0 Partition

HA behaves as it always has, for pre-7.7.0 HSMs (see High Availability Indirect Login Functions Prior to HSM Firmware 7.7.0).

V0 Partition and V1 Partition

For adjustments to API and behavior, see High Availability Indirect Login For HSM Firmware 7.7.0 and Newer

Functionality Modules

Pre-firmware 7.7.0 Partition

FM behavior is as previously, for pre-7.7.0 HSMs.

V0 Partition and V1 Partition

Backup HSM (G5) at firmware 6.28 - FM-vs-non-FM support

	to HSM FW <= 7.4 FM	**to HSM FW <= 7.4 non-FM**	**to HSM FW 7.7 V0 FM**	**to HSM FW 7.7 V0 non-FM**	**to HSM FW 7.7 V1 FM**	**to HSM FW 7.7 V1 non-FM**
From HSM FW 7.4 FM	**yes**	no	**yes**	**yes**	**yes**	**yes**
From HSM FW 7.4 non-FM	no	**yes**	no	**yes**	no	**yes**

"**yes**" indicates a supported backup/restore path

Partition Roles

Pre-firmware 7.7.0 Partition

Roles and their behavior remain as-is for pre-7.7.0 HSMs.

V0 Partition

As in pre-7.7.0

V1 Partition

V1 partitions add the Limited Crypto Officer role for Per-Key Authorization operations see "Partition Roles" on page 363.

Backup/Restore

Pre-firmware 7.7.0 Partition
Backup and restore remain as-is for pre-7.7.0 HSMs.

V0 Partition and V1 Partition
Use of Luna Backup HSM G5 with V0 or V1 partitions implies Backup HSM firmware 6.28 and Luna Client 10.3 (or newer). Both the Client and the RBS server version must be aligned -- that is, the RBS server must be installed from the 10.3 Client or newer, replacing any previous RBS server.

A Luna Backup HSM G5 with firmware earlier than 6.28.0 can restore onto a partition in a firmware-7.7.0 (or newer) HSM, but the Luna Backup HSM G5 must be at firmware 6.28.0 in order to properly backup from a version 7.7.0 (or newer) application partition. In other words, if your Luna Backup HSM G5 is not updated, then its contents can be considered a backup for key-migration, but not a production backup for firmware 7.7.0 (and newer) HSM partitions.

If there is a need to maintain an older version of client library for your main application and to use Luna Backup HSM G5 firmware 6.28.0 for backup/restore purposes, then you must have a separate workstation dedicated for running the RBS server from Luna Client 10.3.

Even if RBS service is not required, you would still need the separate workstation to run lunacm to take advantage of Luna Backup HSM G5 firmware 6.28.0.

Luna Backup HSM (G5) at firmware 6.28.0 used locally with Luna Network HSM appliance with software <=7.4

	to HSM FW <= 7.4	**to HSM FW 7.7 V0**	**to HSM FW 7.7 V1**
From HSM FW <= 7.4	not recommended	supported	supported

See also the functionality module-related Luna Backup HSM G5 concerns, above on this page.

> **NOTE** To perform backup operations on Luna HSM Firmware 7.7.0 or newer (V0 or V1 partitions) you require at minimum:
>
> > Luna Backup HSM 7 Firmware 7.7.1
>
> > Luna Backup HSM G5 Firmware 6.28.0
>
> You can use a Luna Backup HSM with older firmware to restore objects to a V0 or V1 partition, but this is supported for purposes of getting your objects from the older partitions onto the newer V0 or V1 partitions only. V0 and V1 partitions are considered more secure than partitions at earlier firmware versions - any attempt to restore from a higher-security status to lower-security status fails gracefully.
>
> When the Luna Backup HSM is connected directly to the Luna Network HSM 7 appliance, only the SMK can be backed up from or restored to a V1 partition.

The Limited Crypto Officer role does not do cloning, and therefore cannot transfer the current partition's SMK to a Backup HSM for SKS backup.

Secure Trusted Channel

Pre-firmware 7.7.0 Partition
STC remains as-is for pre-7.7.0 HSMs.

V0 Partition and V1 Partition

> A newer version of STC is supported with cryptography that has been enhanced in the following ways:
  - New FIPS and Common Criteria compliant cipher suites added -- ECDH P-521 + AES-GCM and ECDH P-521 + AES-CTR + HMAC-SHA-512 -- for key derivation, encryption and authentication.
  - Key Derivation – Perfect forward secrecy
    - Each party provides an ephemeral key and a static key
    - Compromising the static key doesn't compromise all past and future communications
  - Bilateral Key Confirmation and Unidirectional AES keys [NIST requirement]
    - Both parties ensure that the other party has derived the same keys
    - 2 AES keys are derived: one for encryption and one for decryption
    - Prevents reverse replay attacks

  For more information see "Creating an STC Connection" on page 128

> To use functionality modules (FMs) with STC client connections, you require the newer version of STC, which is used in Client-V0/V1 partition connections.

> To use the updated STC connections, you require Luna Network HSM 7 Appliance Software 7.7.0 or newer, Luna HSM Firmware 7.7.0 or newer, and Luna HSM Client 10.3.0 or newer.

> If you are using Luna HSM Firmware 7.4.2 or older, STC requires Luna HSM Client 10.2.0 or older.

(* If you had thousands of very small keys, you would notice a definite increase in the partition space taken by those keys after update.
If you had 100 big keys or fewer in the partition, you would barely notice a change in required space, as the overhead [new attributes] per key is proportionately much smaller against an individual large key.)

(** Luna Cloud HSM software has historically been named/numbered for the associated HSM version. The Client numbering has been restarted at 10.x to decouple from specific firmware and software versions.)

Converting Partitions from V0 to V1 or V1 to V0

> **CAUTION!** Back up any important keys and objects before proceeding.

This section describes how to convert partitions from V0 to V1 and from V1 to V0. It assumes that you have updated your Luna HSMs to Luna HSM Firmware 7.7.0 or newer, with all pre-existing partitions automatically converted to V0. For more information about V0 and V1 partitions, refer to "V0 and V1 Partitions" on page 148.

Converting a Partition From V0 to V1

To convert a partition from V0 to V1, the procedure differs, depending on whether the partition is a member partition of an HA group or a non-member partition.

Converting a non-member partition from V0 to V1

To convert your non-member partitions to V1 partitions with minimum application downtime, use the following procedure.

To convert a non-member partition from V0 to V1

1. Have the chosen partition visible in lunaCM.
2. Select that partition with the lunaCM command **slot set -slot** <slot number>
3. [Optional] Show the current partition policy values and verify that policy 41 is set to version 0, **partition showpolicies**
4. Log in to the partition as the Partition Security Officer with **role login -name po**
5. Change the value of **Partition Policy 41: Enable Partition Version** to version 1, with **partition changepolicy -policy 41 -value 1**

Converting an HA group member partition from V0 to V1

To convert your HA group member partitions to V1 partitions with minimum application downtime, use the following procedure. This procedure is performed by the HSM SO for each Luna Network HSM 7, the Partition SO and Crypto Officer for the HA group members.

Prerequisites

> You must be aware of the guidelines for upgrading an HA member partition to *any* firmware version and adhere to them carefully. For more information, read "Guidelines and Recommendations For Updating or Converting HA Member Partitions" on page 465.

> **NOTE** You must update/convert secondary partitions first and the primary partition last. If you do not adhere to this guideline, you may experience issues while updating/converting.

> You require **admin**-level access to the Luna Network HSM 7.

To convert an HA group member partition from V0 to V1

1. Migrate the HA group to V0 by completing the procedures described in Migrate High Availability Groups to Luna HSM Firmware 7.7.0 or Newer for every member partition.
2. On the client workstation that administers the HA group, stop all client applications.
3. Update the Luna HSM Client software to Luna HSM Client 10.3.0 or newer (see "Updating the Luna HSM Client Software" on page 106).

4. [Optional] You may now restart your client applications, or wait until the end of the procedure.
5. Launch LunaCM and use the following procedure to convert each HA member partition to V1. To prevent the HA group serial number from changing and disrupting your client applications, the member originally used to create the group must be the last member still remaining in the group:

   > **NOTE** The member partition that has the same serial number as the HA group, minus the leading **1**, is the original member.

   a. Remove a member partition from the HA group (see "Adding/Removing an HA Group Member" on page 449).

      lunacm:> **hagroup removemember -group** <label> {**-slot** <slotnum> | **-serial** <serialnum>}

   b. Log in as Partition SO.

      lunacm:> **role login -name po**

   c. Convert the partition to V1 by changing partition policy **41: Partition Version**.

      lunacm:> **partition changepolicy -policy 41 -value 1**

   d. Repeat steps **a**-**c** until only the original member remains in the HA group.

   e. When only the original member remains in the group, log in as Partition SO and convert it to V1. This member's SMK will be the one used for the entire HA group (see "Scalable Key Storage" on page 165 for more information).

      lunacm:> **role login -name po**

      lunacm:> **partition changepolicy -policy 41 -value 1**

   f. Add each V1 partition back to the HA group (see "Adding/Removing an HA Group Member" on page 449). The primary member's SMK is automatically cloned to each new member added to the HA group.

      lunacm:> **hagroup addmember -group** <label> {**-slot** <slotnum> | **-serial** <serialnum>}

Converting a Partition From V1 to V0

1. Backup any valuable keys or objects.

   > **CAUTION!** This operation, going from V1 back to V0, is destructive. All objects on the partition are destroyed, as well as the SMK(s). If any valuable objects were created and archived from a version one (V1) partition, then they must have been SKS-stored off the HSM, and the SMK that encrypted those objects must be preserved on a Backup HSM or in another partition (that remains at V1), if those objects might ever be needed in future.
   >
   > If no valuable SKS blobs have been encrypted by the partition's current SMK, then there is no need for backup.

2. Have the chosen partition visible in lunaCM.
3. Select that partition with the lunaCM command **slot set -slot** <slot number>
4. [Optional] Show the current partition policy values and verify that policy 41 is set to version 1, **partition showpolicies**
5. Log into the partition as the Partition Security Officer with **role login -name po**

6. Change the value of policy 41 to version 0, with **partition changepolicy -policy 41 -value 0**

CHAPTER 4: Scalable Key Storage

This section describes the Scalable Key Storage (SKS) feature for Luna HSMs and assumes the following:

- You understand the differences between V0 and V1 partitions and how these partitions affect the operation of Luna HSM. For more information, refer to "V0 and V1 Partitions" on page 148.
- You plan on converting a partition from V0 to V1. For more information, refer to "Converting Partitions from V0 to V1 or V1 to V0" on page 161.

You do not need anything on the pages in this section *until* you convert an existing V0 partition to V1 or create new version one (V1) partitions, which will be using the new cloning protocol and SKS feature.

What is Scalable Key Storage?

SKS is virtually unlimited secure storage and handling of your sensitive keys.

By default, keys have resided in HSM hardware for Luna HSMs. This remains true, by default, with the introduction of Luna HSM Firmware 7.7.0. However, firmware 7.7.0 (and newer) adds key export flexibility to expand the Luna HSM's assurance boundary, to encompass much greater numbers of keys than the internal capacity of an HSM.

Keys secure anywhere, the SKS eIDAS model

Beginning with Luna HSM Firmware 7.7.0, you have the option to use the Scalable Key Storage (SKS) feature by converting or creating a partition as Version 1 (V1). When a partition is created, it is given a unique SKS Master Key (SMK) (See "V0 and V1 Partitions" on page 148).

SKS is based upon a model where keys generated on the HSM:

- are securely extracted as encrypted SKS objects and
- are inserted back into the HSM, to be temporarily decrypted, when cryptographic operations are to be performed with those keys.

Similarly, when a unique key encrypts data, the data and the key can be stored as an encrypted **b**inary **l**arge **o**bject (blob) up to 64KB in size, that can be decrypted only within the HSM.

If the HSM is upgraded from an earlier HSM firmware version to Luna HSM Firmware 7.7.0 or newer, then any existing partitions become version zero (V0). Similarly, if you create a new partition on a Luna HSM Firmware 7.7.0 (or newer) HSM, with the default "-version 0" option, it becomes a V0 partition. A V0 partition retains compatibility with older partitions and applications:

- that rely on cloning (secure copying/moving of objects between HSMs or HSM partitions or Backup HSMs, also known as Keys Always in Hardware)
- while benefiting from fixes and security updates that come with the new firmware, but with no access to the newer eIDAS-mandated features.

If you create a new partition in an HSM with Luna HSM Firmware 7.7.0 or newer, and select the V1 option (**-version 1**), then the new partition is version one (V1) and gets a unique SMK and uses SKS (rather than cloning) to replicate keys for HA or to Backup and Restore. The partition also engages Per Key Authorization and other eIDAS related features, but is incompatible with V0. You can also update a V0 partition to V1 while retaining existing objects (but de-converting, or converting back is not an option).

Here is what you will find in the pages of this section:

When to use SKS

When would it be appropriate to use SKS?

Use SKS when you need to handle greater numbers of keys and objects than can be stored within the HSM, and you want to employ methods more secure than wrap-off / wrap-on. You would also use it when needed to comply with a regulatory regime like eIDAS.

Any application where large numbers of very sensitive keys or records must be protected with the highest possible security, while remaining available and accessible to authorized users and applications, is a candidate for the Luna HSM with Scalable Key Storage. The SKS method - in contrast with merely wrapping-off/unwrapping-on - is needed when the HSM must be the assurance boundary for the keys. If it is permissible for the key material to originate outside the assurance boundary, or to reside outside the assurance boundary, then the extra security of SKS is not required.

A general use case for SKS is storing encrypted keys in external databases.

> Generate keys inside the cryptographic module.
> Using the SIMExtract API, extract the encrypted keys and store them in external databases and delete the original keys inside the cryptographic module.
> Insert individual encrypted keys back into the HSM when you need to use them for cryptographic operations inside the cryptographic module.

One example might be the creation and use of electronic signatures (for natural persons) or electronic seals (for organizations) for remote signing (RSS). The signatures or seal key materials are created within the HSM, extracted (not wrapped) in strongly encrypted form that preserves attributes, and stored in a repository. When they are needed, they are found in the repository by the managing system, inserted into the HSM for decryption by a master key that never resides outside an HSM, then used for signing or sealing respectively, and discarded from the HSM (the encrypted versions remain stored in the repository for the next time they are needed).

Another example might be a database of customers, with their contact and shipping information, credit-card information, history of purchases, current/recent browse interests on your commerce site, etc. All of that is likely to be sensitive information protected by regulations and by your own published privacy policies. In this case, the primary concern is privacy of data.

A third example might be a government database of land ownership, including detailed and official property descriptions, current ownership with identifying details, history of title transfers, subdivisions, legal rulings and encumbrances (such as rights of way and covenants), liens, and so on. In this case, the data is meant to be publicly viewable, but its integrity against unauthorized change is paramount.

Security consideration

Various models exist in the industry for handling of huge numbers of sensitive keys and objects. An important consideration is the manner in which the keys and objects are handled.

Method	Security
Wrap-off/wrap-on	Keys and objects can have unknown, uncontrolled origin, potentially outside the assurance boundary. Keys and objects can potentially be accessed outside the HSM, made available and used externally, in potentially unsafe environments. . Keys can have security attributes stripped.
SKS extract/insert	The history of keys and objects is known, controlled, auditable. Keys and objects remain within the security and access perimeter of the HSM. The master key never exists outside a Luna HSM, and all extracted keys and objects must be inserted back into the HSM at time of decryption and use. Keys retain their attributes.

SKS model

On this page:

In an SKS model, in compliance with relevant standards, an application maintains thousands or millions of encrypted objects as records in a repository (such as a database, file system, cloud storage, etc.). SKS does not impose any restrictions on where or how you store encrypted blobs; although constraints like record-size might come into play depending on the type of storage in your implementation. The repository might have each record/object encrypted with a unique key. Examples of applications might include Remote Signing identities (Common Criteria PP 419221-5 use case). The salient points, for SKS, are that:

> Encryption and decryption of objects must take place in the cryptographic module.

> More individual object-encryption keys are needed by the application than can be accommodated by the internal capacity of any cryptographic module.

> Records or objects, and the keys that encrypt them, do not exist in-the-clear - both the record (data object, ID, etc.) and its encrypting key are stored in encrypted form.

> Keys that encrypt objects or signatures must have originated within the assurance boundary (a.k.a. security perimeter of the cryptographic module (HSM)) and are only ever decrypted within the assurance boundary.

> An object can be inserted into any cryptographic module (HSM) version that supports the same version of SKS and supports the same attributes that the original HSM supports for that object.

> **NOTE** Objects extracted from a current-version HSM can be inserted into older version HSMs with known vulnerabilities -- for security reasons, this should be avoided; as a best practice, keep your appliance software and crypto-module firmware up-to-date.

> Objects extracted from an HSM with Functionality Modules disabled cannot be inserted into an HSM with FMs enabled (including via backup/restore operations) - security rules prevent moving keys from a more-secure to a less-secure environment.

How does SKS work?

A key is created in an HSM partition at the direction of an application.

It might be intended as an ID for purposes of signing documents and verifying by private persons, or for sealing of documents and records by organizations.

It might be intended to encrypt records stored in an external database (perhaps customer-identifying records, perhaps medical records, perhaps supply-chain information, or other uses that require privacy and controlled access).

> Each record-encrypting key, or ID key, when not in immediate use inside a cryptographic module, is itself encrypted for extraction by an extraction/insertion key derived (in compliance with NIST SP800-108) from a master encryption key that never leaves the HSM; this is the SKS Master Key, or SMK.

> From the application's perspective, the data record, or the key/cert-as-ID, emerges from the HSM uniquely encrypted as a secure SKS blob (**b**inary **l**arge **ob**ject up to 64KB) that remains within the HSM's security and access perimeter, so it can be safely stored anywhere.

> The operations that an application might perform are:

- creating an identity or a record or data object
- acquiring a suitable key (SKS key) for encrypting that record or data object by either
  - requesting a new object-encryption key (new SKS key) to be generated by the HSM, or
  - providing an already existing object-encryption-key (SKS key) for the HSM to use
- encrypting that ID or object with the new SKS key, (or with the pre-existing, supplied SKS key, which must first be inserted and decrypted for use by the HSM)
- storing the encrypted record or key within the repository
- retrieving the encrypted record or key at a later time (when called by the application)
- inserting/decrypting the SKS blob into the HSM, using the SKS Master Key (SMK)
- using the decrypted key
  - to sign or seal documents or transactions in the case of RSS, or
  - to further decrypt a database record for reading or editing and then re-encrypting the record if it changed and sending the re-encrypted changed record back to storage
- deleting / destroying the material from the HSM (cryptographic module), once it is not needed (the encrypted SKS blob still exists in the external repository, for the next time it is required)

> The application is responsible for the storage and availability of the SKS object in the repository of choice (database, file system, directory, NAS, cloud, etc.).

It is possible to create data objects to store any kind of data in an HSM partition, SKS blobs included (which is essentially what is done if you choose to archive SKS objects in a Backup HSM). However, a practical workflow involving large numbers of keys or sensitive data objects, is assumed to include backing up SMKs, but not SKS blobs, since the latter are already securely encrypted and can be stored anywhere that is reasonably secure, and in numbers far greater than the capacity of any cryptographic module. However, we cannot anticipate all use-cases, and the onboard storage option exists.

To summarize, the options for SKS blobs (SMK-encrypted keys and objects) include:

> Storage of limited numbers of keys and objects encrypted within the cryptographic module;

- Storage of a larger, but still limited number of keys and objects encrypted within a backup HSM;
- Storage of a still larger number of encrypted keys and objects within the file system of the host -- this might include storing and sharing objects within key rings, shared/duplicated across clusters of Luna Network HSM 7 appliances;
- Storage of a virtually unlimited number of encrypted keys and objects across databases or networked file systems.

Optionally, such as in the case of Trust Service Providers, during the SKS object creation process, authentication can be added such that a password must be provided before the keys in an SKS object can be used.

SKS objects use 256-bit AES-GCM encryption for confidentiality and integrity protection.

Limitation and scalability

Because the SKS objects are stored external to the cryptographic module (but must be individually inserted back into the crypto module before use), there is *no capacity limitation* from an HSM perspective. The only scalability limitation would be based upon the number of SKS operations to be performed simultaneously (that is, SKS object creation/extract/insert, and resulting cryptographic operations).

In a Luna HSM at firmware 7.7.0 or newer, partitions are either V0 or V1.

- **In V0 partitions**, keys and objects are handled generally as in previous versions, depending on the type of crypto module configuration (cloning, key extraction, etc.); cloning of objects, such as for backup, HA, etc., is performed as you and your applications have always done, using Luna HSMs. All copying of keys and objects is done between application partitions or Backup partitions that share a security/cloning domain
- **In V1 partitions**, SKS is used. Other than for some SMK management operations, the copying and moving operations for partition objects/keys/etc., have been kept as similar to historic handling as practical. Thus, while the cloning commands are used, only the SMK is ever cloned in the Luna-traditional sense.

  Cloning operations (like Backup, and like HA-group synchronization) for V1 partitions use cloning for the SMK, but also seamlessly invoke SKS to transfer or extract/insert keys and objects as blobs encrypted by the SMK. Thus, encrypted keys/objects *can* be copied securely to another partition or a Backup HSM, if the quantity/storage usage is within the capacity of the receiving partition.

  Recall that partitions are created with default sizes, but you can specify to use up to all remaining space in the cryptographic module and you could store a number of keys/objects as SKS blobs within such a partition, but for large quantities, it is expected that you would store them encrypted in the HSM appliance's file system or external database, while maintaining them within the cryptographic module's security boundary.

Characteristics and Implementation Notes

On this page:

Characteristics of the SKS Implementation

> The SKS feature is implemented at the application partition level; this differs from the older SKS protocol that was applied HSM-wide.

> The SKS mechanism complies with the per-key authorization requirements of Common Criteria PP 419221-5 ("Per-Key Authorization" on page 190).

> The cryptographic mechanisms employed by SKS comply with the FIPS 140 and PP 419221-5 standards (Secure External Scalable Key Storage Extensions).

> A migration path is available for customers using the Luna HSM firmware 6 SKS protocol, such that existing customer applications continue to work against the SKS implementation.

> For certification requirements and/or security best practices, the following are not allowed:

- The SKS implementation prevents objects that are extracted from an HSM with firmware 7.x being inserted into an older version of the HSM that might have known vulnerabilities.
- The SKS implementation prevents objects that are extracted from an HSM that has Functionality Modules (FMs) *disabled*, from being inserted into an HSM that has FMs *en*abled (or ever had FMs *en*abled) - doing so via a backup HSM is also prevented.

Functional Notes

SKS supports the following functionality:

> You can extract all key objects within a given partition by specifying an empty list on the input. Otherwise, specify only individual objects that you wish to extract at one time.

> You have the option to impose/require two possible authentication methods when extracting key objects:

- None (no extra authentication data)
- Password (MofN supported)

> The AES-GCM algorithm is used for encryption of objects during the creation of SKS blobs (extraction from the partition) or decryption upon insertion of an external SKS blob into the partition.

> You can see, and practice, examples of such usage in the ckdemo utility :

- OFFBOARD KEY STORAGE Menu Functions

SMK Locations in a Partition

Each SKS-capable partition supports several types of SMK, each with its own location and limitations within the partition.

The **Primary SMK**, generated at partition initialization time, or replaced via **partition smkclone** command from another partition, resides in the Primary SMK location in the partition, and is used for extraction and insertion operations. Only the firmware 7.x primary SMK can be used for extraction. No path is allowed for extraction to older-version HSMs and partitions.

The **Rollover SMK** location holds the replaced primary, when **partition smkrollover start** command creates a new primary SMK. The rollover SMK is retained while SMK rollover is taking place, to allow all SKS blobs that were encrypted/extracted with the old SMK to be brought back into the HSM partition so that they can be re-extracted with the new primary SMK. When **partition smkrollover end** signals the completion of the rollover operation, the rollover SMK is deleted and the new Primary SMK remains.

The **FM SMK** location is specifically for transfer from a partition on an HSM where the FM policy is enabled to a partition on an FM-never-enabled HSM. That is, a **partition smkclone** command from a partition on an HSM where the FM policy is enabled places the source Primary SMK into the FM SMK location of the target partition. This ensures that key material can be transferred from partitions on an HSM where the FM policy is enabled to partitions on an FM-never-enabled HSM. It is *not permitted* to move key material from partitions on an FM-never-enabled HSM to a partition on an HSM where the FM policy is enabled, because a Functionality Module could extract in plain text. Similarly, there is no FM SMK rollover location.

The **Firmware 6 SMK** location can contain an SMK from a firmware 6.x HSM (if one has been cloned in) for the purpose of inserting SKS blobs that were extracted from a firmware 6.x partition. After insertion of such older blobs, the key material is extracted as a new firmware 7 SKS blob, encrypted via the partition's primary SMK. Whenever a partition is on the receiving end of a **partition smkclone** operation, any contents of the primary and non-primary locations from the source partition overwrite their equivalent locations in the target partition.

High Availability and SKS

To address performance and availability requirements SKS supports high availability configurations similar to Luna HSM Cloning models, with some minor differences. High availability and load balancing is implemented in the Luna Cloud HSM software and is completely transparent to the application, in that the application is configured to use a virtual slot and not a physical slot on the HSM.

One difference, from cloning HSMs in HA configuration is that, for SKS HA, the **hagroup addmember** command clones the SMK from the initial SKS application partition to all other group member partitions as they are added. Thereafter, your application deals with the HA virtual slot, and HA operation is automatic.

> **NOTE** **V1 partitions:** If you add an application partition with an existing SMK to an HA group, the primary member's SMK overwrites the existing SMK of the joining partition.
>
> If a partition's SMK has ever been used to encrypt important SKS objects, save a backup of the SMK before adding that partition to any HA group.

> **NOTE** If a remote partition is involved (Network HSM) on either side of the SMK cloning operation, the HSM that contains the remote partition must have Network Replication enabled. See HSM Capabilities and Policies "Policy 16 - Allow network replication".

When the application needs to perform a cryptographic operation, it employs the SKS API call, which imports an SKS object into the HSM. In an HA configuration, the Luna HSM Client also replicates the SKS blob to all HSMs that have been included in the defined HA group (by performing sksExtract from the source partition and sksInsert into the target partition as a single, combined operation, repeated for each), unless HA synchronization is turned off. When the application requests the "HSM" to perform a cryptographic operation (sign, encrypt, decrypt, etc.) the Luna HSM Client load balances the request to the application partition that is available, in essence making the SKS operation stateless. The SKS operation succeeds because all partitions in the HA group have a copy of the imported SKS object.

Replication of objects in HA is accomplished by SKS feature's SIM`extract` and SIM`insert` in one call, transparently.

TIP If your primary use-case is to insert a key and use it for one signing operation, then consider using the multisign API for better performance with SKS under HA, since invoking multisign would use the key on just the one physical partition, and would avoid the overhead of having the inserted key replicated unnecessarily to other HA group members.

NOTE HA failover is not supported in the case of member failure during a SIMInsert, SIMExtract, or SIMMultisign operation.

Preparing and Administering SKS Partitions

On this page:

Checklist

The following subsections describe briefly what you need

> to set up one or more SKS partitions ready for use,

> to backup and restore SKS Master Keys (SMK) from-and-to the SKS partition, and

> to directly replicate the SMK from one SKS partition to another for High Availability operation.

Cross-reference links are provided to each topic or section, containing explicit instructions for each task.

Provisioning SKS

> You need at least one Luna Network HSM 7 at Luna Network HSM 7 Appliance Software 7.7.0 or newer and Luna HSM Firmware 7.7.0 or newer, or Luna PCIe HSM 7 at Luna HSM Firmware 7.7.0 or newer.

> If you already have an older 7.x HSM, download and install the updates from the Support Portal.

> Install Luna HSM Client 10.3.0 or newer, which includes a version of the lunacm tool that supports the "partition smkrollover" commands - this ensures that the associated library has the updated SKS capabilities and is also able to handle migration from legacy SKS instances

> Follow the instructions at the beginning of "Preparing to use SKS" on the next page to get the appliance installed and network connected

> If you plan to use an HA group, then repeat the above process with the second HSM, and again with any additional active or standby members.

Replicating the SMK to another SKS Partition

Stand-alone - If you are using a *single HSM* with your application, you should have at least one backup copy of the SMK (for each partition) so that any SKS blobs encrypted by that SMK are recoverable in case of loss or damage to the original HSM or partition.

> proceed to "Backup the SKS Master Key (SMK)" on page 180.

HA group - If you are using an *HA group* with your application, then initially, each member has a unique SMK created when its SKS partition is created. For HA operation, the **hagroup addmember** command replicates the desired SMK from the initial member to all additional members of the group. This means that, in order for a partition to take part in HA operation as the second or later member, its original SMK is overwritten by the SMK of the first member of the group.

> Safeguard the desired SMK by backing it up to a Backup HSM before going further. See "SKS Backup and Restore" on page 179.

> **NOTE** If an SMK, already existing on a partition, has ever been used to encrypt an SKS key or objects, then you must backup the existing SMK before replacing/overwriting it, if you wish to ever retrieve the previously encrypted SKS key and objects.

> Follow the instructions for using the **partition smkclone** command in "High Availability and SKS" on page 172.

Backing up the SMK

Always ensure that you have safeguarded any important SMK (one that has been used to encrypt key material for export from the HSM) by backing it up to a Backup HSM partition before you perform any action that might destroy that SMK (such as cloning a different SMK to the current HSM, or restoring a different SMK from a Backup HSM partition).

> To backup, see "Backup the SKS Master Key (SMK)" on page 180.

Restoring the SMK from Backup

When you wish to use the SKS partition to encrypt objects or decrypt objects with an SMK other than the SMK that resides in the current partition, you must restore from a Backup of the desired SMK to overwrite the current SMK in the current partition. If the current SMK (before restoring from archive) is valuable, then back it up first before restoring a different SMK to overwrite the current one.

> To restore, see "Restore an SKS Master Key (SMK)" on page 182.

Preparing to use SKS

Perform all the steps to install and configure a Luna HSM, as described at Installing and Configuring Your New Luna Network HSM 7.

1. If your HSM is not already at Luna HSM Firmware 7.7.0, follow the instructions in the 7.7.0 Customer Release Notes, to securely copy the Luna Network HSM 7 Appliance Software 7.7.0 Software Update package to the appliance, and perform the software and firmware update.

> **NOTE** To update an HA group to Luna HSM Firmware 7.7.0 or newer, all the *non-primary partitions* must be updated *first*, to ensure that the key objects from the firmware 7.7-or-newer primary can still move to the non-primaries through key cloning. *Then* the primary member can be updated.

2. When you reach the steps to create an application partition, ensure that it is created as the default version one (V1), which is necessary for SKS operation.

> **NOTE** The SKS Master Key (or SMK) is created when the partition Crypto Officer logs in. For security reasons the SMK is not made visible in output of the usual commands that show objects on an HSM partition (lunacm:>**partition contents** and lunash:>**partition showcontents**).

3. Go to "Using SKS" below to continue with SKS.

Using SKS

On this page:

Logistical considerations for SKS operation include:

> you must create a V1 partition before you can use the SKS functionality;

> initializing and logging into an SKS partition (as CO) creates a unique SMK in that partition, at the same time;

> you have the option to use the SMK that is created with the partition, or you can overwrite that SMK with

- an SMK that is restored from an SKS partition on a Backup HSM (with **partition archive restore**), or
- an SMK that is cloned (with **partition smkclone**) from another partition (usually for purposes of HA operation);

> **NOTE** **V1 partitions:** If you add an application partition with an existing SMK to an HA group, the primary member's SMK overwrites the existing SMK of the joining partition.
>
> If a partition's SMK has ever been used to encrypt important SKS objects, save a backup of the SMK before adding that partition to any HA group.

Using SKS - options

Two approaches are available, to use SKS :

> Use the API, where you have the ability to write or modify your applications:
  - Directly access the PKCS#11 C-language extensions that interact with the HSM.

    or

  - Use the provided Java toolkit.

> Use a commercial, off-the-shelf Windows CNG application, mediated by the provided SKS Client Extension for Luna HSM KSP.

API

Authorization forms currently supported are none, and password.

Export a key from a partition as an SMK-encrypted blob, using SIMExtract function

```
SIM_AUTH_FORMS = (CKA_SIM_NO_AUTHORIZATION,
                  CKA_SIM_PASSWORD)
CK_RV CA_SIMExtract(CK_ULONG handleCount, CK_ULONG *handleList,

   CK_ULONG authForm, CK_ULONG authDataCount, CK_ULONG subsetRequired,

   CK_BYTE **authDataList,

   CK_BOOL deleteAfterExtract,

   CK_ULONG *pBlobSize, CK_BYTE *pBlob );
```

Import a previously extracted blob, using the SIMInsert function

```
CK_RV SIMInsert( CK_ULONG blobSize, CK_BYTE *pBlob,

   CK_ULONG authForm, CK_ULONG authDataCount, CK_BYTE **authDataList,

   CK_ULONG *pHandleListSize, CK_ULONG *pHandleList );
```

For further information, refer to the SDK Guide (Secure External Scalable Key Storage Extensions).

ckdemo example

1. Start by running ckdemo and executing **Open Session (1)** to the slot and **Login (3)** as **Crypto Officer**, giving the partition password.
2. Generate an AES key using **Simple Generate Key (45)** and keep track of the object handle for the generated key.
3. Execute **SIMExtract (105)**.

   Enter the object handle for **Enter handle of object to add to blob** and then

   Enter **0** to **end the list**.

   Enter **1** for **Enter authentication form**.

   Enter **1** for **number of authorization secrets (N value)**.

   Enter **1** for **Enter subset size required for key use (M value)**.

Enter a password.

Enter **1** for **Delete after extract.**

The masked key is saved to **blobfile.sim**.

4. List all of the objects in the partition by running **Find object (26)** with option **All Standard Objects (6)**.
5. Execute **SIMInsert (106)**.

   Enter **blobfile.sim** for **Enter filename with object to insert**.

   Enter **1** for **Enter authentication form**.

   Enter **1** for **Enter number of authorization secrets to be provided**.

   Enter the password that was entered in the previous step.
6. List all of the objects in the partition by running **Find object (26)** with option **All Standard Objects (6)**. The key that was extracted should now be present in the partition.

> **NOTE** The example above uses the password authentication form. Other authentication forms can be used.

Java Sample

As a prerequisite, ensure that the **LunaProvider.jar** and **libLunaAPI.so** has been installed to your JDK.

1. Navigate to the directory that contains the java sample:

   cd JavaSample
2. In the **SIMExtractInsert.java,** modify the **slot** and **hsmPass** variables appropriately.
3. Compile the sample using **javac**:

   javac SIMExtractInsert.java
4. Run the sample using java.

High Availability

Replication of objects in HA is accomplished by SIMExtract and SIMInsert in one call, transparently.

> **TIP** Turn off HA synchronization for better performance if the inserted keys are to be used for single operations. An inserted key would be replicated to all HA group members even if it were to be used only by one member for (say) one signing operation.
>
> If inserted keys are likely to be used for multiple load-balanced operations, then the overhead of replicating to all members is unavoidable and would be minimal in that context.

OPTIONS for HA Operation are:

- Use with V0 partitions. The assumption is that you have existing partitions and application(s) that use those partitions in HA. Continue as you already do. Your pre-existing upgraded partitions, as well as any new partitions that you create with the V0 option will work as before.
- Use with V1 partitions - all HA group members must be at V1, and your application must be SKS-aware, because only SMKs are cloned between V1 partitions; all other objects are extracted/inserted as SKS blobs.

Constraints on SKS HA

HA for SKS requires that each member of the HA group must have the same SMK.

> **NOTE** For HA environments, if you perform SMK rollover on a member, then the new SMK must be cloned to all members. However, database / repository update for rollover should be done by directly addressing the primary physical member, and *not* using the virtual slot (to avoid the performance penalty when keys inserted to the virtual slot during rollover would be propagated to all members before the re-extraction).

Replicating the SMK to all group members

> **CAUTION!** Each SKS partition contains a single SMK, and this operation overwrites the SMK in the target partition. Therefore, always ensure that the SMK in the target partition is **not** of any use, or that it has been backed up, before you perform **partition smkclone**.

These steps are performed from the lunacm utility in the client computer.

1. Use **slot set** to select one of the HSM partitions, with the desired SMK, as the current slot.

```
lunacm:> slot set -slot 0

Command Result : No Error
```

2. Log into the current slot as Crypto Officer.

```
lunacm:> role login -name Crypto Officer

        Please attend to the PED.

Command Result : No Error
```

3. Use the **partition smkclone** operation to clone the desired SMK from the current slot SKS partition to another SKS partition in a named slot. The SMK from the source slot overwrites the SMK in the target slot. You are logged into the source partition, where you are launching this command, but not to the target partition; specify the password for the target partition.

```
partition smkclone -slot 1 -password myuserpin

  Logging in to target slot 1

  Cloning the SMK.

    The SMK was cloned successfully.

Command Result : No Error
```

   To verify that the SMKclone operation was successful, see below.

4. Repeat the previous step for any other HA group members, to ensure that all members share the same SMK.
5. Set HA-only mode for the group, so that your application sees only the HA virtual slot, and not any of the physical slots.

```
lunacm:> haGroup HAOnly -enable

Command Result : No Error
```

6. Note the number of the HA virtual slot. Your application will direct all operations to that slot.

Verify SMKclone

If necessary, verify the success of the SMKclone operation as follows.

1. Create an object on the original HSM partition (for example, a keypair).
2. SIMExport the object from that original HSM and SIMInsert it into the second (target) HSM, using any of the methods (API, CKDemo, or Java, earlier on this page). If the importation is successful, then the SMK on the second HSM is the same as the SMK on the first, or source HSM.

When NOT to address the virtual slot
As indicated above, blob insertion results in the inserted key being propagated to the other HA members.

However, key external storage database rollover *should **not*** be done over the virtual slot as this would cause the inserted keys to be propagated *before* the re-extraction, affecting performance. Ideally, the database update should be done by direct communication with the primary HA member.

SKS Backup and Restore

On this page:

As described, the partition permanently stores the SKS Master Key (SMK) from the time of its creation when the partition is created. The application partition is intended :

> to create encryption keys as session objects

> to encrypt those keys with the SMK, for SKS extraction from the HSM

> to extract those encrypted keys, as encrypted SKS blobs, for storage by your application, external to the cryptographic module

> to temporarily SKS insert (decrypt) blobs to make individual keys available for crypto operations within the partition

> to use the decrypted key

- to sign or seal documents and other digital objects, using the inserted key as a personal or organization identity, or
- to encrypt data (records) for your external database, archive, cloud, or other repository, or
- decrypt and modify such records before re-encrypting them to go back into your repository.

Therefore, it is not intended that objects other than the SMK be stored in the SKS partitions; however, you can do so if you wish, up to the limits of the partition capacity.

Using Luna HSM Firmware 7.7.0 and newer, SMKs are replicated (for HA) or are backed up and restored using cloning. All other objects are treated as SKS objects and are encrypted/decrypted by the SMK for extraction and [re-]insertion as needed.

Constraints on SKS Backup and Restore

SKS Backup and Restore are intended to redundantly safeguard the SMK, only. The following conditions and constraints apply :

- The SMK is not visible as a partition object, and does not appear in list output.
- Backing up and restoring the SMK uses the same commands as backing up and restoring ordinary cryptographic objects.
- SKS Backup and Restore require a Backup HSM with Luna Backup HSM G5 Firmware 6.28.0 or Luna Backup HSM 7 Firmware 7.7.1 (or newer).
- SKS Backup and Restore is supported only when the currently selected slot is an SKS partition (V1).
- Individual SKS blobs are limited to 64KB in size. Large groups of keys, or larger data objects might need to be split across multiple blobs for extraction or insertion.
- The **partition archive backup** and **partition archive restore** commands test the currently selected slot to ensure it is an SKS-capable (V1) partition.
- If the current slot is an SKS partition, then the **partition archive** commands perform backup or restore of the SMK, and ignore any other objects.

> **TIP** The assumption is that since any extracted SKS blobs are solidly encrypted and remain within the assurance boundary, they can be safely stored in any repository. There is no need to store such blobs in another crypto-capable HSM partition, nor in a Backup HSM partition (though you could do the latter by storing as data objects, if desired) - they can be decrypted and used only by SKS insertion into an HSM partition that contains the relevant SMK.

If you invoked scalable key storage (SKS) for your applications to create and store large numbers of keys, then the partition is V1. If you perform cloning operations (including HA) or Backup and Restore, see "Cloning or Backup / Restore with SKS" on page 216.

Backup the SKS Master Key (SMK)

The SMK backup operation creates a new partition on the backup HSM, using a partition name that is automatically created at the time of the backup operation. The system ensures that the archive partition name does not already exist on the Backup HSM by creating the target partition with a unique name that combines

- the serial number of the source partition (from your Network HSM) with
- a time stamp.

To back up the SMK, do the following:

1. Have a Backup HSM connected to the client workstation from which you are running the command, or have a Backup HSM connected through a Remote Backup Server (see "Configuring a Remote Backup Server" on page 561). The Backup HSM must be visible as a slot.
2. Launch the lunacm utility.

3. Use **slot list** to determine the slot numbers of the SKS partition and of the Backup HSM.
4. Set the SKS partition as the current slot.

```
lunacm:> slot set -slot 4

Command Result : No Error
```

5. Log into the current slot as Crypto Officer.

```
lunacm:> role login -name Crypto Officer

        Please attend to the PED.

Command Result : No Error
```

6. Use **partition archive backup** to backup the SMK from the current slot to the indicated Backup HSM.

> **NOTE** Do not name the target partition to be created on the Backup HSM, because SKS backup creates the name from the serial number of the source partition, combined with a time-stamp.

```
lunacm:>partition archive backup -slot 5

You are backing up a SKS partition.
 Only the SKS master key (SMK) will be backed up.
 No other objects will be cloned.

Are you sure you wish to continue?

Type 'proceed' to continue, or 'quit' to quit now ->proceed

Logging in as the SO on slot 5.

Please attend to the PED.

Creating partition 358628973182_2017:03:09-16:52:47 on slot 5.

Please attend to the PED.

Logging into the container 358628973182_2017:03:09-16:52:47 on slot 5 as the user.

Please attend to the PED.

Creating Domain for the partition 358628973182_2017:03:09-16:52:47 on slot 5.

Please attend to the PED.

The SMK was cloned successfully.

Command Result : No Error
```

7. You can test the success by
   a. creating and initializing a V1 test partition on any HSM with Luna HSM Firmware 7.7.0 or newer,
   b. restoring the backed-up SMK onto that test partition, and

c. successfully importing an SKS blob (that was previously extracted using the specific SMK) into that partition.

Restore an SKS Master Key (SMK)

To restore the SMK from backup, follow these steps.

> **CAUTION!** When you restore an SMK from a Backup HSM, that restored SMK overwrites (destroys) any SMK that was already present on the partition. If the current SMK has been used to encrypt any important keys, ensure that you have backed it up safely before restoring a different SMK over it.
>
> Also be sure to record the particulars of that backup, including the backup partition name and some notes to identify which keys have been encrypted by the SMK archived in that partition, for future reference.

1. Have a Backup HSM connected to the client workstation from which you are running the command, or have a Backup HSM connected through a Remote Backup Server (see "Configuring a Remote Backup Server" on page 561). The Backup HSM must be visible as a slot.
2. Launch the lunacm utility.
3. Use **slot list** to determine the slot numbers of the SKS partition and of the Backup HSM.
4. Set the SKS partition as the current slot.

```
lunacm:> slot set -slot 4

Command Result : No Error
```

5. Log into the current slot as Crypto Officer.

```
lunacm:> role login -name Crypto Officer

        Please attend to the PED.

Command Result : No Error
```

6. Use **partition archive restore** to restore the SMK from the current slot to the indicated Backup HSM, naming the partition with the desired SMK,on the Backup HSM.

```
lunacm:>partition archive restore -slot 5 -partition 358628973182_2017:03:09-16:52:47

You are restoring a SKS partition.
 Only the SKS master key (SMK) will be restored.
 No other objects will be cloned.

Are you sure you wish to continue?

Type 'proceed' to continue, or 'quit' to quit now ->proceed

Logging in to partition 358628973182_2017:03:09-16:52:47 on slot 5 as the user.

Please attend to the PED.
```

```
The SMK was cloned successfuly.

Command Result : No Error
```

7. You could test the success by restoring the SMK to a test partition, and successfully importing an SKS object that was previously exported, encrypted with that SMK.

Backup objects

In most cases, only the SMK needs preserving, and any crypto objects on the SKS partition are just passing through (as temporary session objects), so there is no provision to backup crypto objects from an SKS partition. It is possible to store SKS blobs, but only as data objects, not as crypto objects. Therefore, to use them in any way, they must be inserted back into a V1 partition that has the correct SMK in either the Primary SMK location or the Rollover SMK location.

The Backup HSM can support a mix of

> SKS-only archive partitions that each can contain a single SMK, and

> ordinary cloning-backup partitions that each can contain multiple crytographic objects for traditional cloning-based (non-SKS) HSM backup and restore operations.

In other words,

> you can use an SKS client to backup crypto objects
  - from a non-SKS partition
  - into a non-SKS archive partition on the Backup HSM)

> you can restore crypto objects
  - from a non-SKS archive partition on the Backup HSM
  - to a regular cloning-based (non-SKS) HSM partition.

> you cannot restore ordinary objects onto an SKS partition; they must be SKS inserted (siminsert API call.)

Troubleshooting SKS Backup and Restore

The following are some examples that highlight incorrect usage, along with the communication from the system.

Not logged into partition at current slot

Here is an example of the output if you attempt to use the partition archive command without logging in as "Crypto Officer" on the SKS partition slot, which must be the current slot.

> **NOTE** "-slot 5" in the example points to the Backup HSM slot, not the current SKS partition slot.

```
lunacm:>partition archive backup -slot 5

You are backing up a SKS partition.
 Only the SKS master key (SMK) will be backed up.
 No other objects will be cloned.

Are you sure you wish to continue?
```

```
Type 'proceed' to continue, or 'quit' to quit now ->proceed

Error: Failed to open session.

Command Result : 0xb0 (CKR_SESSION_CLOSED)
```

An incorrect option is specified for backup

In this example, the command fails because the "-partition" option is not applicable for SKS backup:

```
lunacm:>partition archive backup -slot 5 -partition test

You are backing up a SKS partition.
Only the SKS master key (SMK) will be backed up.
No other objects will be cloned.

Are you sure you wish to continue?
Type 'proceed' to continue, or 'quit' to quit now ->proceed

Syntax Error: Option -partition cannot be used for SKS operation.

Command Result : No Error
```

Archive contains crypto objects

If the backup partition to be restored contains crypto objects and SKS backup is being performed, restore of the SMK proceeds with a warning.

```
lunacm:> par ar r -s 5 -par pre-7-7

You are restoring an SKS partition.
Only the SKS master key (SMK) will be restored.
CAUTION: The existing SMK will be overwritten.


Are you sure you wish to continue?
Type 'proceed' to continue, or 'quit' to quit now -> proceed
Logging in to partition mypar on slot 5 as the user.

Please attend to the PED.

WARNING: Crypto object(s) detected in the backup device container.
Dedicated backup container for SKS Master key is recommended.

The SMK was cloned successfully.

Command Result : No Error
```

SMK Rollover

Mandated rollover schedules might place limits on the allowable lifetimes of important keys, like the SMK. Toward this end, the **partition smkrollover** command generates a new SMK and allows you to re-encrypt your SKS blobs with a new SMK.

However, if a new SMK is created - such as in the SMK rollover operation - then every blob that has been encrypted with the old SMK must be inserted, its contained keys/objects re-encrypted with the new SMK and extracted as a new blob, and this must be accomplished for all such externally stored blobs, *during* the **smkrollover -start** to **smkrollover -end** interval. Any blobs for which the encrypting SMK no longer exists can no longer be decrypted.

To rollover the current SMK

If you wish to perform SMK rollover, please realize that it is a disruptive process and a major one. Hence, plan it appropriately by scheduling a down-time and then follow this three-step procedure:

1. Dismantle the HA group.
2. Perform SMK Rollover.
   a. Begin with **partition smkrollover -start**
      – the original SMK is moved to the Rollover area, and
      – a new SMK is created and placed in the Primary SMK area of the current HSM.
   b. Retrieve every blob from your repository, one at a time, insert it into the HSM partition - the insert action is performed with *SIMInsert API* using the former-primary-now-rollover SMK.
   c. After each key or object is inserted, extract it again to external storage - the extract action is performed with *SIMExtract* using the new Primary SMK.
   d. When *all* blobs have been retrieved (from repository, database, backup HSM, etc), inserted and re-extracted, then perform **partition smkrollover -end** one time, to delete the previous SMK from the rollover slot, to conclude the rollover operation.
3. Re-create the HA group, cloning the *new* SMK to each HSM that will become a member of the recreated HA group, and resume operations.

You can generate a new SMK and immediately discard the old one with **partition smkrollover -start -end** command. Do this *only* if you know that no blobs exist that are encrypted with the old SMK, otherwise they will be orphaned.

> **NOTE** For HA environments, if you perform SMK rollover on a member, then the new SMK must be cloned to all members. However, database / repository update for rollover should be done by directly addressing the primary physical member, and *not* using the virtual slot (to avoid the performance penalty when keys inserted to the virtual slot during rollover would be propagated to all members before the re-extraction).

Migrating Scalable Key Storage (SKS)

On this page:

The SKS feature beginning with Luna HSM Firmware 7.7.0 (and newer) uses the same API as previous SKS versions, to retain backward compatibility; your applications that used older SKS should still work. However, the new structure for SKS was developed in conjunction with an updated cloning protocol and other features of Luna HSM Firmware 7.7.0 associated with V1 partitions, and you (or a regulatory regime under which your organization operates) might see benefit in migrating existing SKS secrets to the newer form.

For purposes of migration, the SKS Master Key (SMK) is cloned to a target partition - this is the only use for the cloning protocol in V1 partitions. *Objects **encrypted** by the SMK (that is, SKS blobs)* are generally expected to be stored external to the crypto module in a repository via SKS extract operation (SIMextract API call) from the partition and later SKS insert operation (SIMinsert API call) when needed. If blobs are small enough or few enough in number, such objects could be replicated to other members in an HA group, or to keyrings in a cluster, or stored on a Backup HSM if desired, but as data objects only.

Off-board storage is assumed to be the primary method of storing such blobs (and *not* storage inside a general purpose crypto module or a backup HSM).

Cloning the SKS Master Key (SMK)

No changes to the existing older SKS host API are necessary for the cloning of the earlier and the newer SMKs. The choice is based on the formatting of the incoming SMK Secret.

> **NOTE** If a remote partition is involved (Network HSM) on either side of the SMK cloning operation, the HSM that contains the remote partition must have Network Replication enabled. See HSM Capabilities and Policies "Policy 16 - Allow network replication".

The following table shows possible migration paths for existing SMKs -- the leftmost column is possible sources, while the heading row across the top lists possible destinations, and the intersecting table cells are the possible result for each source-to-destination scenario.

Destination **Source**	**FM6 SKS appliance**	**FW6 SKS G5 Backup (6.25)**	**FW7.7 eIDAS G5 Backup (6.28)**	**FW<7.7 HSM**	**FW>=7.7**	**FM HSM FW>=7.7 Non-FM HSM**
FW6 SKS appliance	FW6 SMKs	FW6 SMKs	FW6 SMKs	No SMK support on target	Target has FM cert only	FW6 SMKs
FW6 SKS G5 Backup (6.25)	FW6 SMKs	FW6 SMKs	FW6 SMKs	No SMK support on target	Target has FM cert only	FW6 SMKs
FW7.7 eIDAS G5 Backup (6.28)	FW6 SMKs	FW6 SMKs	All SMKs (cloning protocol used by V1 partitions)	No SMK support on source/target	All SMKs (cloning protocol used by V1 partitions)	All SMKs (cloning protocol used by V1 partitions)

Destination Source	FM6 SKS appliance	FW6 SKS G5 Backup (6.25)	FW7.7 eIDAS G5 Backup (6.28)	FW<7.7 HSM	FW>=7.7	FM HSM FW>=7.7 Non-FM HSM
FW<7.7 HSM	No SMK support on source	No SMK support on source	No SMK support on source	No SMK support on target	No SMK support on source	No SMK support on source
FW7.7 FM HSM	Source has FM cert only	Source has FM cert only	All SMKs (cloning protocol used by V1 partitions)	No SMK support on target	All SMKs (cloning protocol used by V1 partitions)	All SMKs (FW7.7-Primary -> FW7.7-FM, FW7.7-Rollover dropped) (V1 partition)
FW7.7 Non-FM SKS HSM	Required cloning protocol not on target	Required cloning protocol not on target	All SMKs (cloning protocol used by V1 partitions)	No SMK support on target	Blocked by V1 cloning protocol	All SMKs (cloning protocol used by V1 partitions)

(**FW>=7.7** means Luna HSM Firmware 7.7.0 or newer)

To migrate an older SMK

To move/copy an SMK from one of the supported source configurations to one of the supported targets

1. If the source and target are crypto partitions, clone the SMK secrets between partitions with the partition smkclone lunacm command.
2. If the target is a Luna Backup HSM clone to the Backup HSM with the partition archive backup lunacm command.
3. If the source is a Luna Backup HSM, clone from the Backup HSM with the partition archive restore lunacm command.

Only some combinations of source and target are supported. Reasons for a path to not be supported are summarized in the table.

SCENARIO 1:

0. You have a pre-existing 7.7.0 SMK, and a bunch of extracted blobs (encrypted with that SMK) in a repository, all of which you want to preserve, and to which you want to add 6.x SMK and SKS blobs after migrating them.

1. Backup the modern 7.7.0 SMK to a partition on a firmware 6.28.0 or 7.7.0 Backup HSM (partition archive backup).
2. Backup the 6.x or 4.x SMK (partition archive backup).
3. Restore the 6.x SMK onto a V1 partition (partition archive restore). This puts it in the Primary SMK slot of that partition, overwriting any SMK that was there.

4. Perform partition smkrollover **-start** on the V1 partition. This moves the 6.x SMK into the Rollover slot of the partition and generates a new Primary SMK.
5. Restore the production 7.7.0 SMK that you backed-up earlier (partition archive restore). This overwrites the newly-generated Primary, but the Rollover partition still has the 6.x SMK.
6. Insert one of your 6.x SKS blobs. The HSM tests it and knows to use the Rollover SMK to perform the SKS insert operation.
7. Extract the key/crypto-object as a new blob - the HSM allows only the Primary (the one you just got back from archive) to perform the extraction.
8. Repeat for all your old-style blobs to get them all encrypted with the new SMK for storage in your repository.
9. Perform partition smkrollover **-end** which deletes the old SMK from the Rollover slot of the V1 partition.

SCENARIO 2:

You have 6.x SMK and SKS blobs. You want to migrate the older blobs to use in a new V1 partition (7.7). So, no 7.7.0 SMK and blobs currently exist that need conserving.

1. 1. Backup the 6.x (to partition on 6.28.0 or 7.7.0 Backup HSM) with partition archive backup command.
2. Create a new V1 partition (partition create), initialize it (partition init), log in as CO (role login **-name co**) etc., but you don't care about the generated 7.7.0 SMK.
3. Restore the 6.x SMK onto the V1 partition (partition archive restore). This puts it in the Primary SMK slot of that partition, overwriting any SMK that was there.
4. Perform partition smkrollover **-start** on the V1 partition. This moves the 6.x SMK into the Rollover slot of the partition and generates a new Primary SMK.
5. Insert one of your 6.x SKS blobs. The HSM tests it and knows to use the Rollover SMK to perform the SIMinsert operation.
6. Extract the blob - the HSM allows only the Primary (recently generated) to do the extraction.
7. Repeat for all your old-style blobs to get them all encrypted with the new SMK for storage in your repository.
8. Perform partition smkrollover **-end** which deletes the old SMK from the Rollover slot of the V1 partition.

SKS Blob Migration

With no modifications to the pre-existing host API for offboard key storage (SKS), the version number for the mechanism used is prepended to the SKS blobs and employed to select the correct mechanism to insert the blob back into the HSM.

HSMs are allowed to extract key blobs using only the latest and greatest SMK secret and mechanism available to them. Incoming older blobs with older SMK secrets and mechanisms are accepted, provided that the HSM f/w supports them.

> **NOTE** Migration from older HSMs, that used possibly outdated encrypting keys/mechanisms should not present a problem, since older blobs would be inserted only. The same material would then be extracted using newer or unrestricted key types or sizes.

The following table shows options for SKS blob insertion into a partition, Protocol and Key Import/Export vs External Storage. The leftmost column is possible sources, while the heading row across the top lists possible destinations, and the intersecting table cells are the possible result for each source-to-destination scenario.

	FW6 SKS appliance	FW7.7.0 Non-FM HSM (V0 Partition)	FW7.7.0 FM HSM (V0 Partition)	FW7.7.0 Non-FM HSM (V1 Partition)	FW7.7.0 FM HSM (V1 Partition)
FW6 SKS appliance	Using FW6 SMK (V1 partition cloning protocol) Import/Export	No SKS Support on source/target	No SKS Support on source/target	Using FW6 SMK (V0 partition cloning protocol) Import/Export	No FW6 SMKs on Target
FW7.7.0 Non-FM HSM (V0 Partition)	No FW7.7.0-Primary SMK on Target	No SKS Support on source/target	No SKS Support on source/target	No SKS Support on source	No identical FW7.7.0-Primary SMK on Target
FW7.7.0 FM HSM (V0 Partition)	No FW7.7.0-FM SMK on Target	No SKS Support on source/target	No SKS Support on source/target	No SKS Support on source	No SKS Support on source
FW7.7.0 Non-FM HSM (V1 Partition)	No FW7.7.0-Primary SMK on Target	No SKS Support on source/target	No SKS Support on source/target	Using FW7.7.0-Primary SMK (V1 partition cloning protocol) External Storage	No identical FW7.7.0-Primary SMK on Target
FW7.7.0 FM HSM (V1 Partition)	No FW7.7.0-FM SMK on Target	No SKS Support on source/target	No SKS Support on source/target	Using FW7.7.0-FM SMK (V1 partition cloning protocol) External Storage	Using FW7.7.0-Primary SMK (V1 partition cloning protocol) External Storage

To migrate an older SKS blob:

To insert an SKS blob, for any of the supported scenarios (table above),

1. Insert with the SIMinsert operation as you always have.

   The CKDemo Utility, command 106, demonstrates the action - see OFFBOARD KEY STORAGE Menu Functions.

CHAPTER 5: Per-Key Authorization

Per-key authorization or authentication (PKA) is a feature introduced with Luna HSM Firmware 7.7.0 to support the eIDAS use case of Remote Signing and Sealing (RSS) and the relevant Protection Profile (PP 419-221.5). PKA introduces data structures to keys that are created and manipulated in the HSM such that keys can be handled in the ways that applications normally handle key material, but under the sole ownership and control of an end-user natural person or legal entity. The attributes are applied to keys that are created with Luna HSM Firmware 7.7.0 (or newer). In V0 partitions those attributes are simply ignored. In V1 partitions the attributes are actively used. See more at "V0 and V1 Partitions" on page 148.

Keys for use in eIDAS schemas:

- have authentication data structures that allow the possibility for an entity to have sole ownership and control
- can be unassigned, waiting for distribution to eventual owners/Users, or
- can be assigned to the control of a specific owner/User.

When a key has auth code data attached, then by definition anyone who holds the auth code is a/the key owner. But before it is assigned, the key does not have an owner/User, and might be part of a pool of unassigned keys, waiting for distribution to users. Keys do take some time to generate, so in times of high demand, it could be practical and convenient to have some ready-to-go.

Keys are intended to be used, but they must also be administered. That is, an individual natural person (or a non-natural legal entity) authorizes cryptographic usage of a key, perhaps to sign forms or documents. At the same time, the HSM has roles that perform actions within the HSM, either:

- *generally* - the eIDAS Administrator role represented by the Crypto Officer (CO) role in the HSM) or
- *specifically/individually* - the eIDAS User role represented by the Limited Crypto Officer (LCO) role in the HSM.

So, a citizen might log into a service and perform an action that directs the application to retrieve their existing personal key from a database/repository and insert/decrypt the key into an HSM partition, where the citizen authorizes a signing or other operation, and then the copy of the key is deleted from the HSM partition, but the archived, encrypted copy resides safely in the database for future retrieval and use. Alternatively, a citizen might request a single-use key, that is generated 'on-the-spot' in the HSM partition (by the LCO role) and is authorized by that citizen to perform one action (like signing) and then the key is permanently deleted, with no copy existing anywhere.

Generally, these and other operations related to keys are not performed by administrative commands (tools like lunacm); rather, they are performed via the PKA API or REST, while the performing application is logged in as one or the other of the partition roles.

Example Use Case

For example, an application might be instructed to retrieve a certain key and use it to sign a document on behalf of a citizen. The application acquires the key from a database (in the form of an encrypted blob) and inserts it into an HSM where it is decrypted to reveal the key that is to be used. But the application is able to actually use that key only when the owner/citizen presents her/his unique authentication data, which is part of the key attributes.

New Role and Handling

In order to manage this service, the individual application partition's Crypto Officer role and a new role called Limited Crypto User handle the actions of creating, modifying, and using keys containing auth data.

A key can be created in an assigned state, where it is immediately associated with an entity, or a key can be created in unassigned state and only later assigned to an owner, when convenient.

No New Administrative Commands

Because the operations around PKA and RSS are handled programmatically, no particular administrative commands are introduced - only a **-version** option for partition creation and a partition policy 40, which is off for V0 partitions, and which defaults to on for V1 partitions, but can be turned off if desired. Everything else about PKA is handled by the "PKA API" on page 1.

Dependencies and Interactions with Other Features

PKA generally requires Luna HSM Firmware 7.7.0 or newer and Luna HSM Client 10.3.0 or newer, and Luna Network HSM 7 Appliance Software 7.7.0 or newer. The feature is ignored by older clients and applications that do not know how to make use of it. Active use of PKA requires a V1 partition, which means that cloning is used for:

- incoming keys and objects from older firmware, but not outgoing (that is, on V1 partitions, cloning of keys is inbound migration, only)
- copying (such as for HA), or backing-up/restoring, of the SMK

All other objects are stored, encrypted by the SMK, in external storage using the Scalable Key Storage (SKS) feature.

Stored Data Integrity (SDI) is also mandated by eIDAS and is therefore applied by Luna HSM Firmware 7.7.0 and newer.

HA Indirect Login support is constrained differently for V0 and V1 partitions - see section "V0 Partitions" in PKA API.

CHAPTER 6: Key Cloning

You can clone key material between partitions to back up the keys, or to migrate the keys from one HSM to another. The rules, prerequisites, and procedures for migrating your key material are described in the following topics:

Overview and Key Concepts

A Crypto Officer can clone the cryptographic objects (keys) from one user partition to another user partition provided that:

> The user partitions share the same cloning/security domain. See "Domain Planning" on the next page. For firmware newer than 7.8.0 and client newer than Luna HSM Client 10.5.0 see "Enabling and Using CPv4" on page 227 and "Universal Cloning" on page 198.
> The user partitions use the same authentication method (muiltifactor or password).
> The CO has the required credentials on both user partitions.
> The capabilities and policies set on the source and target HSM and user partitions allow cloning. See HSM Capabilities and Policies and "Partition Capabilities and Policies" on page 337.

Enhanced utility with Universal Cloning
Using Luna HSM Firmware 7.8.0 or newer, you can engage Extended Domain Management and Cloning Protocol Version 4 (CPv4) as described at "Universal Cloning" on page 198 to improve the versatility of key cloning.

Changes introduced with Luna HSM Firmware 7.7.0 and newer
You can update Luna HSM Client software, Luna Network HSM 7 appliance software, and Luna HSM firmware at different times, according to your needs.

When the HSM is updated to Luna HSM Firmware 7.7.0 or newer, some changes take place in the partitions and their contents, such that updated Client software is needed to make full use of the updated partitions and their contents. See "V0 and V1 Partitions" on page 148 for more detail on behaviors and constraints of the partition types.

- In HA groups, update the secondary members first, and then the primary member last.
- Older clients will continue to work with V0 partition for Luna Network HSM 7.
- For Luna PCIe HSM 7, must use Luna HSM Client 10.3.0 or newer
- Need newer client for V1 partitions when you want to use SKS or PKA.
- Client software must be Luna HSM Client 10.3.0 or newer to work with V1 partitions to support SKS and/or PKA, and HA. See "V0 and V1 Partitions" on page 148 for more detail.

> **NOTE** For older Luna versions, or situations where only cloning protocol version one (CPv1) is available, the library attempts to perform the individual actions of a cloning operation in sequence on the respective partitions, opening and closing a separate session for each object to be copied. If the policies and partition types on the source and target partitions are incompatible, the **partition clone** command (or an attempted HA synchronization) can fail with a message like CKR_DATA_LEN_RANGE while trying to clone. This can occur if a key object from the source partition is a different size than an equivalent object expected by the target.
>
> **UPDATE**: Using Luna HSM Firmware 7.8.0 and newer, when a cloning negotiation agrees on the use of CPv4, a call to clone multiple keys/objects launches a *single session for all the requested objects*, rather than opening and closing individual sessions for each object. The above portion of this note about mismatched sizes remains valid.

Further information
For an overview of cloning protocols from earlier versions to current, their respective capabilities and applicability, see "Cloning Protocols and Cipher Suite Selection" on page 217.

Considerations when Performing Cloning and Backup-Restore Operations, when SKS is Involved

If you invoked scalable key storage (SKS) for your applications to create and store large numbers of keys, then the partition is V1. If you perform cloning operations (including HA) or Backup and Restore, see "Cloning or Backup / Restore with SKS" on page 216.

Domain Planning

The cloning or security domain is an element of Layered Encryption.

A security domain or cloning domain is a layer of encryption that is created, during initialization, on an HSM or HSM partition that you control. The domain determines whether a crypto object can leave the HSM, and where it can go if it is allowed to leave. That is, a security or cloning domain is a method for administrators to limit key migration such that your keys will only exist in specifically your Thales HSMs and not in simply any Thales HSM.

Cloning is a secure-copy operation by which sensitive HSM objects are copied, while strongly encrypted, from one HSM to another HSM. The security domain, or cloning domain, is a special-purpose secret that is attached to a partition on an HSM. It determines *to* which, and *from* which, other partitions (on the same HSM or on other HSMs) the current partition can clone objects. Partitions that send or receive partition objects by means of the cloning protocol must share identical cloning domain secrets. That is, the protocol verifies that the destination domain matches the source domain; otherwise an error is displayed and the attempted operation fails. This is important for:

- Cloning in backup and restore operations
- Synchronization in HA groups.

Characteristics of Cloning Domains

Password-authenticated HSMs have text-string cloning domains for the HSM admin partition and for any partitions that are created on the HSM. HSM and Partition domains are typed at the command line of the host computer, when required. Password authentication cloning domains are created by you.

Multifactor Quorum-authenticated cloning domains are created by a Luna HSM, which could be the current HSM, or it could be a previously initialized HSM that you wish to include in a cloning group with the current HSM. Multifactor Quorum-authenticated HSMs have cloning domains in the form of encrypted secrets on red PED keys, for the admin partition and for any partitions that are created on the HSM.

The following characteristics are common to security (cloning) domains on all Luna HSMs.

- The unique *admin partition* security domain can be created in the HSM at initialization time, or it can be imported, meaning that it is shared with one-or-more other HSMs.
- The *application partition* security domain can be created by the current HSM when the partition is initialized, or it can be imported, meaning that it is shared with one-or-more other HSM partitions, and therefore direct cloning, backup/restore, and HA sync operations can be performed among the partitions that share a given domain.
- The application partition security domain is usually distinct from the HSM domain, as they are controlled by different people; on multipartition HSMs, the PSO is usually not the same person as the HSM SO, but on a single-partition HSM the two SOs might be the same person.
- The application partition security domain can be the same as the domain of another partition on the same HSM (for HSMs that support multiple partitions).

For multifactor quorum-authenticated HSMs, the domain secret for the admin partition or for an application partition can be a single red PED key, or it can be split (by the MofN quorum feature) over several red keys, which are then distributed among trusted personnel such that no single person is able to provide the cloning domain without oversight from other trusted personnel.

In scenarios where multiple HSM partitions are in use, it can be useful to segregate those partitions according to department or business unit, or according to function groups within your organization. This ensures that personnel in a given group are able to clone or backup/restore only the contents of partitions sharing the domain for which they are responsible. The segregation is maintained by physical and procedural control of the relevant PED keys that each group is allowed to handle.

For password-authenticated HSMs, that sort of segregation is maintained entirely by procedure and by trust, as you rely on personnel not to share the domain text strings, just as you rely on them not to share other passwords.

Have your naming conventions and allotments planned out ahead of HSM initialization and partition creation, including a well-thought-out map of who should control cloning domain access for admin partitions and for application partitions. These decisions must be made before you create the partitions.

Cloning Domains Using Luna HSM Firmware 7.8.0 and CPv4

Luna HSM Firmware 7.8.0 introduced Extended Domain Management, the ability for an application partition to support up to three KCVs (Key Cloning Vectors) or cloning/security domains. The original domain is retained as a PSK (pre-shared secret or pre-shared key) for CPv4 negotiation, so that existing workflows are not affected. Two additional domains can optionally be added, for greater operational flexibility, if needed. See partition domainadd.

Additional domains are optional, and can be typed (as for password-authenticated HSM partitions) or can be input from a PED, if the **-dped** option is used with the **partition domainadd** command.

Extended Domain Management permits:

- rotation of domain secrets (similar to password rotation)
- shifting of domains
- splitting of domains
- joining of domains.

As indicated elsewhere, it is possible to affect the availability of cloning protocols

- by turning CPv1 on with partition policy 42 "Allow CPv1" setting (which disables the use of CPv3 and CPv4, and must be decided in advance because it is destructive), or
- by using the partition cipherenable and partition cipherdisable commands to control the availability of cipher suites.

If the use of CPv1 or CPv3 is forced (by disabling all the CPv4 cipher-suite choices), then only the use of a single domain is supported. If more than one exists in a partition, the primary is used. By default, the primary domain is the one that was imprinted when the partition was initialized, but you can make a different domain the primary by specifying the **-primary** option when using the **partition domainadd** command.

> **NOTE** CPv1 UPDATE: In Luna Cloud HSM firmware 3.0 and newer, CPv1 is removed from FIPS firmware support as it is no longer compliant with 140-3. As this only affects FIPS 140 approved configuration (formerly FIPS mode), all affected users should use CPv4 or transition service to non-FIPS 140 approved configuration. If Luna Network HSM users want to clone to Luna Cloud HSM they will have to use Luna 7.8 or higher. See Universal Cloning for more information.

Domains per partition - (copying across domains)

Copying of keys and objects, which is controlled and constrained by domains assigned to a partition, has generally been consistent over many Luna releases. Luna HSM Firmware 7.8.0 expands the historic capability, while retaining integration with existing customer applications. That is, with respect to domains, Luna HSM Firmware 7.8.0 (and newer)

- **defaults to pre-existing domain-related behavior**,

- An application partition can have one cloning domain, which is either a password-like typed string, or a randomly generated PED key domain string.
- A cloning domain is not labeled, because there is no use for a label when only one is possible.
- It is not possible to clone objects from two or more different cloning domains to a single partition.
- By design, there is no provision to change the cloning domain of a partition without initializing it, which destroys any objects in that partition.

> **but expands the domain-related behavior, if desired**

- An application partition is initialized with a domain that can optionally have a label, or remain unlabeled.
- Up to two additional domains can be added to a partition
  - a label allows you to visually distinguish which domain is which
  - the original domain is designated "primary", but you have the option to declare a new domain to be the primary
- A new domain can be a text string (for password-authenticated domains) or a domain secret stored on a PED key
  - if you include a string, that is used
  - if you do not include a string, when adding a domain, the dialog prompts for PED input.

No common domains across password-authenticated and multifactor quorum-authenticated HSMs (superseded restriction)

> **NOTE** The information in this section concerns Luna HSM firmware versions older than Luna HSM Firmware 7.8.0.

Password-authenticated application partitions, with identical security domains, can clone partition contents one to the other, if the HSM type supports cloning. Multifactor Quorum-authenticated application partitions, with identical security domains, can clone partition contents one to the other, if the HSM type supports cloning.

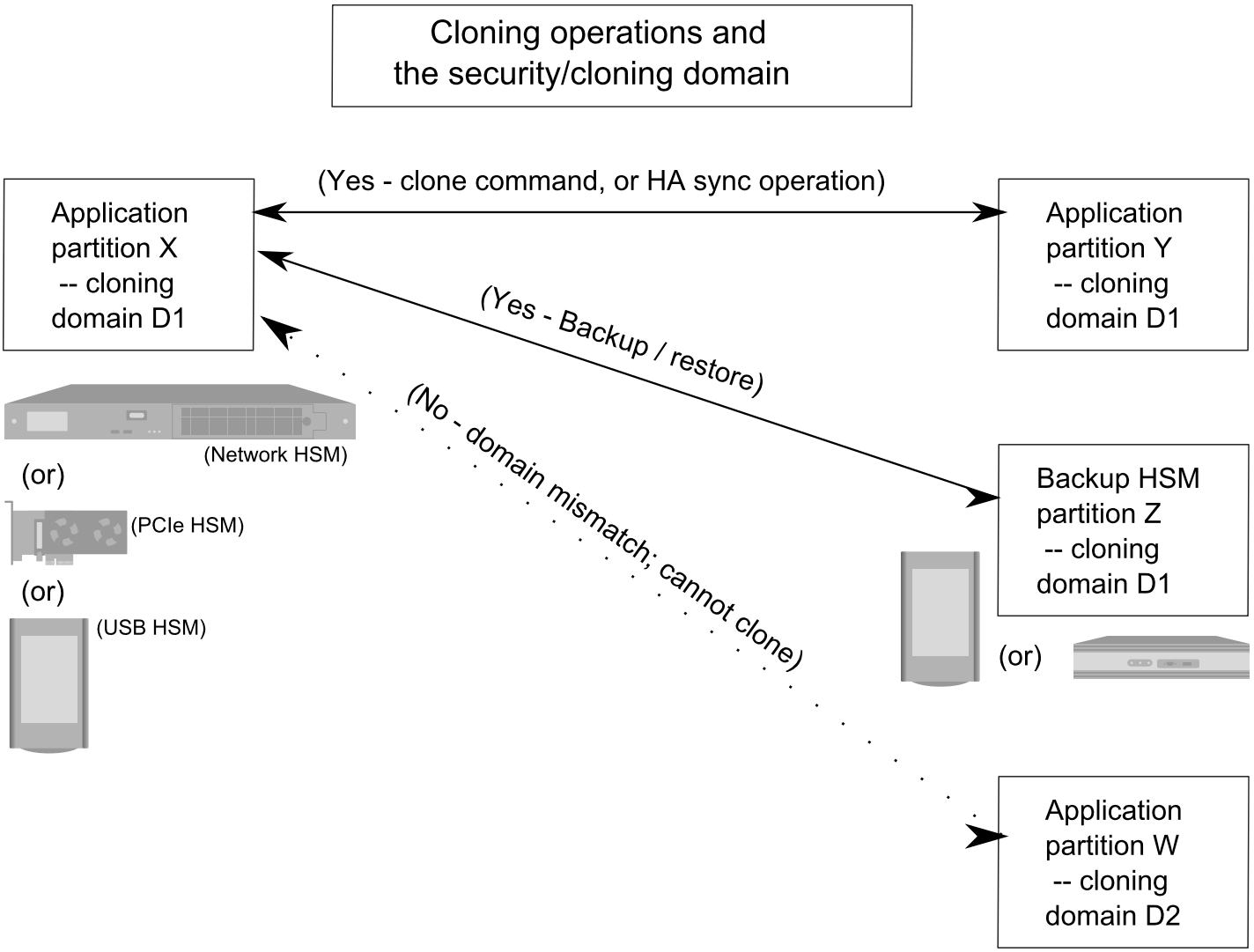
Prior to firmware version 7.8.0, password-authenticated HSM partitions cannot perform cloning with multifactor quorum-authenticated HSM partitions.

The security design consideration is that, if you have a key or object stored in a multifactor quorum-authenticated partition:

> It cannot be altered to a less-secure state and moved outside the protection of its original security/cloning domain.
> [The firmware 7.8.0 and newer Extended Domain Management allows you to clone keys and objects across domains, as long as you have control of the domain(s) on the source partition and on the destination partition - nobody can transfer any partition content, or invoke any other domain, without your authorization]

> You are assured that the key or object has never been outside its original security/cloning domain, or in any less-secure state.
> [Firmware 7.8.0 and newer Extended Domain Management modifies this proviso to "has never been outside a domain under your control".]

Using Luna HSM Firmware 7.7.0 or newer, any V1 (non-backward-compatible) partitions, backup and HA replication of crypto objects are accomplished with SKS encrypted blobs, and the cloning protocol is reserved for the SMK - the key that encrypts the SKS blobs.



Universal Cloning

At the highest level, Key Migration is a facility to move keys securely between our various forms of HSMs. The list of HSMs currently includes the Luna Network HSM 7 appliance, Luna PCIe HSM 7, Luna Backup HSM G5), and Luna USB HSM 7 form factors, as well as Luna Cloud HSM services.

The Luna HSM product satisfies two main requirements on where and how keys can be shared:

> I want my keys to exist only in Thales Luna HSMs.

> I want my keys to exist only in Thales Luna HSMs/Partitions that I control.

What is universal cloning?

Universal cloning is the combination of two features added by Luna HSM Firmware 7.8.0 and client Luna HSM Client 10.5.0:

- **Cloning protocol version 4 (CPv4)**, which makes new cipher suites available to the cloning process and adds flexibility in handling and choosing among options
- **Extended Domain Management**, which adds the ability for a partition to have up to three cloning domains, any of which can be text-strings (Password Authentication) or can be multifactor quorum secrets, regardless of the nominal authentication mode of the current partition.

Prior to Luna HSM Firmware 7.8.0, Luna HSM partitions were generally limited to cloning objects between partitions that had the same single cloning security domain. As of firmware 7.8.0 and newer, application partitions can each optionally have as many as three cloning domains, and these can be added and deleted without risk to keys and other objects contained within the partition. With firmware 7.8.0 and client Luna HSM Client 10.5.0 the introduction of Extended Domain Management along with cloning protocol version 4 (CPv4) - increases your flexibility in choice of form factor:

- Originally chose on-premises HSM, but want to move to the cloud.
- Originally chose the cloud but have a need to repatriate back to on-premises HSM.
- Want to use both types seamlessly, such as in hybrid HA groups.

Extended Domain Management allows you to specify the source of a partition's domain, and provides the ability to have more than one domain in a partition, permitting:

- changing/rollover of cloning domains similar to what is done for passwords when your security regime mandates a refresh/rollover/password-change interval for all forms of authentication,
- migration of keys and objects between multifactor quorum authenticated partitions using authentication secrets on ikeys and
- migration of keys and objects between password authenticated partitions (on-premises HSMs) and services (Luna Cloud HSM) using text strings,
- migration of keys and objects between on-premises multifactor quorum authenticated HSM partitions and password authenticated partitions that share a common domain.
- migration of keys and objects between on-premises multifactor quorum authenticated HSM partitions and Luna Cloud HSM services that share a common domain.

In other words, if at least one of [as many as] three domain secrets on one partition is a match for at least one domain on the other partition, you can migrate objects between them. If more than one domain matches, then if one has been set as primary, that one is selected for the operation.

Enabling before using CPv4

To enable the use of CPv4, the HSM Security Officer MUST set the clock either before or immediately after the firmware has been updated. This is needed by the timing requirements inherent in the protocol.

No additional configuration steps are required as the Luna Cloud HSM service is compatible.

> **NOTE** There is no plan to update the older Luna Backup HSM G5 (nor any version of firmware 6) to support CPv4.

CPv4 is handled like the prior CPv1 cloning protocol.

- The default cloning configuration displays both CPv1 and CPv4. (However UC 10.4.x users and below can clone using only CPv1).
- Any scenario where CPv1 can be used, CPv4 can be used.
- Any scenario where CPv1 cannot be used, CPv4 cannot be used.
- CPv4 can be used to clone objects in V0 partitions, and clone the SMK in V1 partitions.
- The same set of roles that can use CPv1 can use CPv4. This includes the allowance for the Crypto-User to clone public objects.
- In Luna Cloud HSM service, CPv4 is the default protocol; however if only one side has CPv4 available the cloning process reverts to CPv1.
- Partition Policy 8 in Luna Cloud HSM displays both CPv1 and CPv4; however when paired with UC 10.4.x and lower only CPv1 is permitted.

> **NOTE** The Partition Policy difference between Luna Network HSM 7 and Luna Cloud HSM is as follows:
> - **Luna Network HSM 7 slot** Policy “42: Allow CPv1 : 1” means to force CPv1 and disable all other cloning protocols.
> - **Luna Cloud HSM slot** Policy “8: Allow CPv1 (Cryptovisor Only) : 1” means to allow CPv1 but you can *also* clone using CPv4.

Preconditions for Universal Cloning with Cloud or On-premises HSM Partitions

In order to use the Universal Cloning feature, the following must be true:

- you have Luna HSM Client 10.5.0 or newer
- you have either
  - Luna Cloud HSM, and on-premises HSM at firmware version 7.8.0 or newer
    OR
  - an on premises Luna HSM at firmware 7.8.0
    and
    another on-premises Luna HSM at any firmware version suitable for "Migrating Keys to Your New HSM" on page 384

    or

    another on-premises Luna HSM at any firmware 7.8.0 and onward for the widest range of domain management and key-cloning cipher options
- the source partition's security policy allows cloning of private and secret keys
- Partition policy **44**: "Allow Extended Domain Management" on page 351 is set to ON for at least one firmware 7.8.0 or newer partition.
- you have an uninitialized destination Luna Cloud HSM service (if cloning to Luna Cloud HSM rather than on-premises HSM)
- if your client host is a Windows device, you have installed pscp (PuTTY Secure Copy Protocol).

To clone cryptographic keys from one HSM to another, the HSMs must share the same cloning domain. As a result, you must initialize the destination partition with the source partition's cloning domain. The cloning domain was specified as a string when the source partition was initialized. However, with the advent of Extended Domain Management, you can add up to two additional security/cloning domains to any on-premises HSM partition (where the firmware is at version 7.8.0 or later).

The key migration process does not impose that both platforms use the same authentication mechanism. Key migrations are possible between HSMs that use Password or multifactor quorum authentication mechanisms. The only requirements on authorization, at either end, are that the user must be authenticated AND the user must be authorized to perform the migration.

To move objects between HSMs/Partitions that have different default security/cloning domain types (whether multifactor quorum/multifactor quorum,or password/password, or multifactor quorum/password) ensure that both source and target have at least one common domain among the three that each can have with Extended Domain Management (as of firmware 7.8.0).

> **CAUTION!** Domain secret strings for password-authenticated HSMs and Luna Cloud HSMs are used to generate the secret key for cloning, and are as cryptographically sensitive as a user password. The domain label associated with a domain string is not sensitive, and is used only to distinguish the domain from others assigned to the same partition. ***Never use the same string for the domain* label *and for the domain* secret.**

The commands to manage domains are:

- partition domainadd
- partition domainchangelabel
- partition domaindelete
- partition domainlist

For more information about CPv4, see "Enabling and Using CPv4" on page 227 and "Cloning Protocols and Cipher Suite Selection" on page 217.

Next steps

When you have arranged for two application partitions to have at least one domain in common, and have ensured that they support a suitable cloning protocol and available cipher suites in common (see "Cloning Protocols and Cipher Suite Selection" on page 217), you are ready to perform direct partition-to-partition cloning, or to backup from one and restore to the other, or to join them to an HA group.

Cloning Objects to Another Application Partition

You can back up partition objects from an application partition to any other partition that shares its cloning domain. The Crypto Officer of both partitions can perform this operation using LunaCM.

TIP ***The various ways you might use cloning***

- **Basic cloning partition-to-partition** This page talks about generically performing cloning procedures with the explicit cloning commands **partition clone**.
- **Key migration** In cases where you are looking to bring your important keys and objects from application partitions on older Luna HSMs to more modern HSMs or to HSMs with equivalent hardware, but with more recent firmware versions, then you might prefer to refer to the page: "Migrating Keys to Your New HSM" on page 384. The underlying cloning operations are the same, but the emphasis and discussion are more oriented to the migration task, and cover some activities and caveats not addressed here.
- **Backup and Restore** Similarly, the backup and restore operations (to and from dedicated backup HSMs), for offline storage, employ dedicated commands that nevertheless invoke cloning operations and protocols. See "Partition Backup and Restore" on page 467.
- **High Availability (HA)** Finally, the Luna High Availability (HA) feature permits you to dedicate two or more application partitions (usually on separate hosts) to processing your applications' cryptographic calls by sharing the workload across HA group-member partitions whose content is synchronized by means of cloning operations. Concepts and instructions are here "High-Availability Groups" on page 413.

Considerations when Performing Cloning and Backup-Restore Operations, when SKS is Involved

If you invoked scalable key storage (SKS) for your applications to create and store large numbers of keys, then the partition is V1. If you perform cloning operations (including HA) or Backup and Restore, see "Cloning or Backup / Restore with SKS" on page 216.

Prerequisites

- **Partition policy 0:** "Allow private key cloning" on page 338 must be set to **1** (ON) on both the source and target partitions.
- The target partition must be initialized with the same cloning domain as the source partition.
- You require the Crypto Officer credential for both the source and the target partition.
- Both partitions must be visible as slots in LunaCM.
- [Remote PED] This procedure is simpler when both partitions are activated (see "Activation on Multifactor Quorum-Authenticated Partitions" on page 373). If the partitions are not activated, you must connect the source partition to PEDserver before logging in, disconnect it, and then connect the target partition to PEDserver by specifying its slot.

  lunacm:> **ped connect** [**-ip** <IP>] [**-port** <port>]

  lunacm:> **ped disconnect**

  lunacm:> **ped connect -slot** <target_slot> [**-ip** <IP>] [**-port** <port>]

NOTE For older Luna versions, or situations where only cloning protocol version one (CPv1) is available, the library attempts to perform the individual actions of a cloning operation in sequence on the respective partitions, opening and closing a separate session for each object to be copied. If the policies and partition types on the source and target partitions are incompatible, the **partition clone** command (or an attempted HA synchronization) can fail with a message like CKR_DATA_LEN_RANGE while trying to clone. This can occur if a key object from the source partition is a different size than an equivalent object expected by the target.

UPDATE: Using Luna HSM Firmware 7.8.0 and newer, when a cloning negotiation agrees on the use of CPv4, a call to clone multiple keys/objects launches a *single session for all the requested objects*, rather than opening and closing individual sessions for each object. The above portion of this note about mismatched sizes remains valid.

NOTE CPv1 UPDATE: In FW 3.0 for Luna Cloud HSM, CPv1 is removed from FIPS firmware support because it is not compliant with 140-3. As this only affects FIPS mode, all affected users should use CPv4, or else transition service to non-FIPS mode. If Luna Network HSM 7 users want to clone to Luna Cloud HSM the use of Luna HSM Firmware 7.8.0 or newer is required. See Universal Cloning for more information.

NOTE Use of CPv4 requires that the HSM time be set before any cloning operation is attempted that invokes the protocol.

It is recommended to have the host synchronized to a secure ntp or nts server, before synchronizing the HSM time to host time.

Use hsm time commands.

To clone partition objects to another application partition

This is the simplest case, where source and target are similarly configured. For more detailed procedures, when cipher-suite availability and partition authorization method might differ, and additional choices are needed, see below.

1. In LunaCM, set the active slot to the source partition and log in as Crypto Officer.

   lunacm:> **slot set -slot** <slotnum>

   lunacm:> **role login -name co**

2. [Optional] View the partition objects and their object handles.

   lunacm:> **partition contents**

3. Clone objects on the partition to the target partition by specifying the target slot. You can choose which objects to clone by specifying a comma-separated list of object handles, or specify **all** to clone all objects on the partition. Present the target partition's Crypto Officer credential when prompted.

   lunacm:> **partition clone -slot** <slotnum> **-objects** <comma-separated_list/**all**>

   The specified objects are cloned to the target partition. Any objects that already exist on the target are not cloned.

NOTE When a password-authenticated HSM partition is acquiring a domain from a multifactor quorum-authenticated partition, the password-authenticated HSM must have a Luna PED locally connected to it to facilitate the operation.

A remote PED connection is not effective in this scenario, and any pedserver activity on the client should be halted for this cloning operation with "pedserver -mode stop" on page 318.

This is not necessary if you instead choose to add a text domain to your multifactor quorum-authenticated HSM partition (requires Luna HSM Firmware 7.8.0 or newer and Luna HSM Client 10.5.0 or newer) with partition domainadd. However, you might choose to import an authentication secret (PED key) to a password-authenticated partition if there is no room for another domain on your multifactor quorum-authenticated partition, and you have strong reason to preserve all domains already there.

Copying Keys and Objects with Universal Cloning

Luna HSM Firmware 7.8.0 and Luna HSM Client 10.5.0 introduce universal cloning, which rounds out the ability to clone objects between differing HSM partitions and also with Cloud crypto services (now running Luna Cloud HSM firmware 2.0 or newer). Universal cloning makes use of cloning protocol version 4 (CPv4) along with Extended Domain Management features that allow each HSM partition to have, and select among, up to three different cloning/security domains. That is, a single partition can contain, and perform crypto, with keys and objects that are protected under as many as three different domains, whether password-authenticated or multifactor quorum-authenticated.

If your on-premises HSMs are at an earlier firmware version, you can still use the older protocol versions among on-premises partitions, and cloud crypto services can still negotiate back to CPv3 or CPv1 with the inherent abilities and limitations of those earlier versions. See "Cloning Protocols and Cipher Suite Selection" on page 217 for more details on the specifics of each generation of cloning protocol, including which ones were introduced with which HSM firmware versions.

NOTE **Partition policy 44:** "Allow Extended Domain Management" on page 351 must be set to 1 or ON, in order for your partition to have more than one cloning/security domain, and to allow either adding new domain strings (password-authenticated), or adding-by-importing existing multifactor quorum domains (PED keys).

NOTE CPv1 UPDATE: In FW 3.0 for Luna Cloud HSM, CPv1 is removed from FIPS firmware support because it is not compliant with 140-3. As this only affects FIPS mode, all affected users should use CPv4, or else transition service to non-FIPS mode. If Luna Network HSM 7 users want to clone to Luna Cloud HSM the use of Luna HSM Firmware 7.8.0 or newer is required. See Universal Cloning for more information.

NOTE Use of CPv4 requires that the HSM time be set before any cloning operation is attempted that invokes the protocol.

It is recommended to have the host synchronized to a secure ntp or nts server, before synchronizing the HSM time to host time.

Use hsm time commands.

To clone partition objects from on-premises password-authenticated partition to on-premises multifactor quorum-authenticated partition using Luna HSM Firmware 7.8.0 or newer

Requires Luna HSM Client 10.5.0 or newer.

This procedure is for:

> an on-premises password-authenticated Luna Network HSM 7 partition as the source, which could be for:

- a routine cloning between two HSM partitions that are at Luna HSM Firmware 7.8.0 or newer,
- *migration cloning of keys and objects* from a legacy HSM partition (firmware 5.x, 6.x), or from firmware older than Luna HSM Firmware 7.8.0.

> an on-premises multifactor quorum-authenticated Luna Network HSM 7 partition as the target (at Luna HSM Firmware 7.8.0 or newer).

1. Ensure that the two partitions can both use a common cloning protocol.
   a. for HSMs (both legacy and 7.x) before Luna HSM Firmware 7.7.0, only protocol CPv1 is available
   b. for Luna HSM Firmware 7.7.1 and newer, if partition policy 42 - Enable CPv1 is ON, then that protocol is chosen and others are disabled

      lunacm:> **slot set -slot** <slotnum>

      lunacm:> **partition showpolicies**
   c. if partition policy 42 - Enable CPv1 is OFF, then negotiation of common cipher suites is attempted between partitions; this is preferred when available.
   d. if CPv1 has not been forced, and all cipher suites for CPv4 have been disabled on one of the participating partitions, then only CPv3 remains and a common CPv4 cipher suite cannot be negotiated.
2. Ensure that the source and target partitions have a cloning domain in common.
   a. In LunaCM, set the active slot to the target multifactor quorum-authenticated partition and log in as Partition SO (po).

      lunacm:> **slot set -slot** <slotnum>

      lunacm:> **role login -name po**
   b. View the partition domains and note their labels.

      lunacm:> **partition domainlist**
   c. If the two partitions share a common domain, proceed to cloning.
   d. If the two partitions do not share a common domain, then make room, if necessary, by deleting one domain you can do without on the target partition.

      lunacm:> partition domaindelete
   e. Add a domain that matches one from the source partition

      lunacm:> **partition domainadd -domain** <text domain secret> **-domainlabel** <label of the text domain being duplicated>
3. In LunaCM, set the active slot to the source partition and log in as Crypto Officer.

   lunacm:> **slot set -slot** <slotnum>

   lunacm:> **role login -name co**

4. [Optional] View the partition objects and their object handles.

   lunacm:> **partition contents**

5. Clone objects on the current partition to the target partition by specifying the target slot. You can choose which objects to clone by specifying a comma-separated list of object handles, or specify **all** to clone all objects on the partition. Present the target partition's Crypto Officer credential when prompted.

   lunacm:> **partition clone -slot** <slotnum> **-objects** <comma-separated_list/**all**>

   The specified objects are cloned to the target partition. Any objects that already exist on the target are not cloned.

6. [OPTIONAL] You can retain an added domain on a partition as long as it remains useful
   - as long as the partition contains objects encrypted under that particular domain, or
   - while you think the current partition might clone (as source or as target) objects with a partition or service using that domain.

   Or you can delete a domain using partition domaindelete if it is no longer needed.

To clone partition objects from on-premises multifactor quorum-authenticated partition to on-premises password-authenticated partition using Luna HSM Firmware 7.8.0 or newer

Requires Luna HSM Client 10.5.0 or newer.

This procedure is for :

> an on-premises multifactor quorum-authenticated Luna Network HSM 7 partition as the source, which could be for:
   - a routine cloning between two HSM partitions that are at Luna HSM Firmware 7.8.0 or newer,
   - *migration cloning of keys and objects* from a legacy HSM partition (firmware 5.x, 6.x), or from firmware older than Luna HSM Firmware 7.8.0.

> an on-premises password-authenticated Luna Network HSM 7 partition as the target (at Luna HSM Firmware 7.8.0 or newer).

1. Ensure that the two partitions can both use a common cloning protocol
   a. if the source has partition policy 42 - Enable CPv1 on , then that protocol is chosen and others are disabled (or if the source has firmware earlier than Luna HSM Firmware 7.7.0, meaning that CPv1 is the only protocol); this imposes restrictions on operations, see "Cloning Protocols and Cipher Suite Selection" on page 217

      lunacm:> **slot set -slot** <slotnum>

      lunacm:> **partition showpolicies**

   b. if partition policy 42 - Enable CPv1 is OFF, then negotiation of common cipher suites is attempted between partitions; this is preferred when available.

   c. if CPv1 has not been forced, and all cipher suites for CPv4 have been disabled on one of the participating partitions, then only CPv3 remains and a common CPv4 cipher suite cannot be negotiated.

2. Ensure that the source and target partitions have a cloning domain in common.

a. If the source is a Luna HSM Firmware 7.8.0 or newer partition, then it can accept the target's domain string (password-authenticated) into the multifactor quorum-authenticated source partition, avoiding the need to connect a Luna PED to the target, in which case, skip to step **d.**; otherwise, go to step **b.**

b. If the source multifactor quorum-authenticated partition is at any firmware version older than Luna HSM Firmware 7.8.0, it cannot have more than one domain, so its PED key secret must be brought to the target; connect a Luna PED locally to the password-authenticated target.

c. In LunaCM, set the active slot to the target partition and log in as Partition SO.

lunacm:> **slot set -slot** <slotnum>

lunacm:> **role login -name po**

d. View the partition domains and note their labels.

lunacm:> **partition domainlist**

e. If the two partitions share a common domain, proceed to cloning.

f. If the two partitions do not share a common domain, then make room, if necessary, by deleting one domain you can do without.

lunacm:>partition domaindelete

g. Add a domain that matches one from the other partition.

lunacm:> **partition domainadd -domain** <text domain secret> **-domainlabel** <label of the text domain being duplicated>

3. In LunaCM, set the active slot to the source partition and log in as Crypto Officer.

lunacm:> **slot set -slot** <slotnum>

lunacm:> **role login -name co**

4. [Optional] View the partition objects and their object handles.

lunacm:> **partition contents**

5. Clone objects on the partition to the target partition by specifying the target slot. You can choose which objects to clone by specifying a comma-separated list of object handles, or specify **all** to clone all objects on the partition. Present the target partition's Crypto Officer credential when prompted.

lunacm:> **partition clone -slot** <slotnum> **-objects** <comma-separated_list/**all**>

The specified objects are cloned to the target partition. Any objects that already exist on the target are not cloned.

6. [OPTIONAL] You can retain an added domain on a partition as long as it remains useful

- as long as the partition contains objects encrypted under that particular domain, or
- while you think the current partition might clone (as source or as target) objects with a partition or service using that domain.

Or you can delete a domain using partition domaindelete if it is no longer needed.

To clone keys and objects from a Luna Cloud HSM Service slot to an on-premises multifactor quorum-authenticated partition

1. Ensure that the two partitions can both use a common cloning protocol

a. the source, as an Luna Cloud HSM service, is already at firmware 2.0 or a later level that supports CPv4

b. if partition policy **42 - Allow CPv1** is ON, for the target partition, then that protocol is chosen and others are disabled

lunacm:> **slot set -slot** <slotnum>

lunacm:> **partition showpolicies**

> **NOTE** CPv1 UPDATE: In FW 3.0 for Luna Cloud HSM, CPv1 is removed from FIPS firmware support because it is not compliant with 140-3. As this only affects FIPS mode, all affected users should use CPv4, or else transition service to non-FIPS mode. If Luna Network HSM 7 users want to clone to Luna Cloud HSM the use of Luna HSM Firmware 7.8.0 or newer is required. See Universal Cloning for more information.

c. if partition policy **42 - Allow CPv1** is OFF, then the cloning protocols available will sort themselves and negotiate common cipher-suites between source and target; this is preferred.

d. if all cipher suites for CPv4 have been disabled on the on-premises HSM partition, then only CPv3 remains and a common CPv4 cipher suite cannot be negotiated; this imposes restrictions on operation see "Cloning Protocols and Cipher Suite Selection" on page 217 for more detail.

2. Ensure that the source and target partitions have a cloning domain in common.

Just as you must know the password (text string) for a Luna Cloud HSM service, you must also know the domain secret (text string) to proceed with this cloning operation.

a. In LunaCM, set the active slot to the on-premises target partition and log in as Partition SO.

lunacm:> **slot set -slot** <slotnum>

lunacm:> **role login -name po**

b. View the partition domains and note their labels.

lunacm:> **partition domainlist**

c. If the Luna Cloud service and the target partition share a common domain, proceed to cloning.

d. If the two do not share a common domain, then make room on the on-premises target, if necessary, by deleting one domain you can do without.

lunacm:>partition domaindelete

e. Add a domain that matches one from the source Luna Cloud HSM service

lunacm:> **partition domainadd -domain** <text domain secret> **-domainlabel** <label of the text domain being duplicated>

3. In LunaCM, set the active slot to the source partition and log in as Crypto Officer.

lunacm:> **slot set -slot** <slotnum>

lunacm:> **role login -name co**

4. [Optional] View the partition objects and their object handles.

lunacm:> **partition contents**

5. Clone objects on the cloud service to the target partition by specifying the target slot. You can choose which objects to clone by specifying a comma-separated list of object handles, or specify **all** to clone all objects on the partition. Present the target partition's Crypto Officer credential when prompted (in this case, the appropriate black PED key).

   lunacm:> **partition clone -slot** <slotnum> **-objects** <comma-separated_list/**all**>

   The specified objects are cloned to the target partition. Any objects that already exist on the target are not cloned.

6. [OPTIONAL] You can retain an added domain on a partition as long as it remains useful
   - as long as the partition contains objects encrypted under that particular domain, or
   - while you think the current partition might clone (as source or as target) objects with a partition or service using that domain.

   Or you can delete a domain using partition domaindelete if it is no longer needed.

To clone keys and objects from an on-premises multifactor quorum or password-authenticated partition to a Luna Cloud HSM service

This includes

> an on-premises Luna Network HSM 7 multifactor quorum-authenticated or password-authenticated partition as the source, and

> a Luna Cloud HSM service (password-auth) as the target.

1. Ensure that the two partitions can both use a common cloning protocol
   a. if the target partition is a Luna Cloud HSM service, then it is at a firmware level that supports CPv4
   b. if the source has partition policy 42 - Enable CPv1 on, then that protocol is chosen and others are disabled

      lunacm:> **slot set -slot** <slotnum>

      lunacm:> **partition showpolicies**

   c. if partition policy 42 - Enable CPv1 is off for the on-premises HSM partition, then the cloning protocols available will sort themselves and negotiate common cipher-suites between source and target; this is preferred.
   d. if all cipher suites for CPv4 have been disabled on the on-premises HSM partition, then only CPv3 remains and a common CPv4 cipher suite cannot be negotiated; this imposes restrictions on operation see "Cloning Protocols and Cipher Suite Selection" on page 217 for more detail.
2. Ensure that the source and target partitions have a cloning domain in common.
   a. In LunaCM, set the active slot to the source partition and log in as Partition SO (po).

      lunacm:> **slot set -slot** <slotnum>

      lunacm:> **role login -name po**

   b. View the partition domains and note their labels.

      lunacm:> **partition domainlist**

   c. If the source partition and the target cloud service share a common domain, proceed to cloning.

d. If the source partition and the target cloud service do not share a common domain, then make room, if necessary, by deleting from the source partition one domain you can do without.

lunacm:>partition domaindelete

e. Add a domain to the source partition that matches the domain from the Luna Cloud HSM Service target.

lunacm:> **partition domainadd -domain** <text domain secret> **-domainlabel** <label of the text domain being duplicated>

3. In LunaCM, set the active slot to the source partition and log in as Crypto Officer.

lunacm:> **slot set -slot** <slotnum>

lunacm:> **role login -name co**

4. [Optional] View the partition objects and their object handles.

lunacm:> **partition contents**

5. Clone objects on the partition to the target Luna Cloud HSM Service by specifying the target slot. You can choose which objects to clone by specifying a comma-separated list of object handles, or specify **all** to clone all objects on the partition. Present the target partition's Crypto Officer credential when prompted.

lunacm:> **partition clone -slot** <slotnum> **-objects** <comma-separated_list/**all**>

The specified objects are cloned to the target Luna Cloud HSM Service. Any objects that already exist on the target are not cloned.

6. [OPTIONAL] You can retain an added domain on a partition as long as it remains useful

- as long as the partition contains objects encrypted under that particular domain, or
- while you think the current partition might clone (as source or as target) objects with a partition or service using that domain.

Or you can delete a domain using partition domaindelete if it is no longer needed.

> **NOTE** CPv1 UPDATE: In FW 3.0 for Luna Cloud HSM, CPv1 is removed from FIPS firmware support because it is not compliant with 140-3. As this only affects FIPS mode, all affected users should use CPv4, or else transition service to non-FIPS mode. If Luna Network HSM 7 users want to clone to Luna Cloud HSM the use of Luna HSM Firmware 7.8.0 or newer is required. See Universal Cloning for more information.

Cloning Keys Between Luna 6, Luna 7, and Luna Cloud HSM, Password or Multifactor Quorum

Luna HSM Client allows you to clone keys between Luna 6 partitions, Luna 7 partitions, and Thales Data Protection on Demand (DPoD) Luna Cloud HSM services. This includes creating HA groups made up of different HSM versions. This configuration is useful for:

> migrating your keys directly from Luna 6 to your new Luna 7 HSMs

> migrating your keys from Luna Network HSM 7 to the cloud, or vice-versa

> gradually upgrading your on-premises production environment from Luna 6 to Luna 7 HSMs

> maintaining a real-time, cloud-based backup of your cryptographic objects

This page contains guidelines and general considerations for cloning keys between the different HSMs, or using mixed-version HA groups. Mixed-version HA groups have all the same requirements of standard HA groups (see "Planning Your HA Group Deployment" on page 426), in addition to the considerations listed below.

Luna On-Premises/Luna Cloud HSM Cloning

Cloning between Luna partitions and Luna Cloud HSM services require the following special considerations, in addition to the general considerations below.

> **NOTE** This feature requires minimum Luna HSM Client 10.2.0 for password-authenticated partitions, and minimum Luna HSM Client 10.4.1 for multifactor quorum-authenticated partitions. The scope is expanded with Universal Cloning in Luna HSM Firmware 7.8.0 and Luna HSM Client 10.5.1.

Authentication

Luna Cloud HSM services use password authentication, but you can use them with multifactor quorum-authenticated Luna HSM partitions by importing a domain PED key secret during initialization. This feature requires minimum Luna HSM Client 10.4.1. Refer to "Initializing a Luna Cloud HSM Service" on page 73 for the procedure.

> **NOTE** All PED FW versions are currently compatible with Luna Cloud HSM.

Network Latency and Luna Cloud HSM as Active HA Member

Requests performed by cloud services like Luna Cloud HSM may experience greater network latency than those sent to on-premise HSMs. Thales recommends using a Luna Cloud HSM service as a standby HA member to achieve the best performance. By default, you can add a Luna Cloud HSM service as a standby HA member only. If all other HA members fail and the Luna Cloud HSM service becomes active, it will revert to standby when another member recovers.

If you prefer to use the Luna Cloud HSM service as an active HA member, you must first edit the following toggle in the **Chrystoki.conf/crystoki.ini** configuration file (see "Configuration File Summary" on page 76):

```
[Toggles]
lunacm_cv_ha_ui = 0
```

> **CAUTION!** HA failover from multifactor quorum-authenticated Luna partitions to Luna Cloud HSM requires minimum Luna HSM Client 10.5.0. Refer to known issue LUNA-23945.

Cloning Capacity Limitations

The following limitations apply to clients accessing a Luna Cloud HSM service:

- 100 token objects (or 50 RSA-2048 key pairs) per service.
- 100 session objects (or 50 RSA-2048 key pairs) per application.
- 100 simultaneous sessions per application.

Clients that exceed the token object and session object limits can experience slow or failed request responses. The session limit is enforced, and the client receives the error CKR_MAX_SESSION_COUNT when the application reaches the limit.

If you exceed the recommended maximum number of objects cloned to/from a Luna Cloud HSM service in a single cloning operation, the operation sometimes fails with CKR_DEVICE_ERROR. In the case of HA groups, this could include key creation operations, since objects are then cloned to the Luna Cloud HSM service.

Supported Software/Firmware Versions

Thales supports cloning between Luna 6/7 partitions and Luna Cloud HSM services using combinations of appliance software/firmware as outlined in the table below.

> **NOTE** Luna HSM Firmware 7.7.0 is not compatible with older Luna versions or Luna Cloud HSM.

Client Software	Luna Appliance Software	Luna HSM Firmware
Luna only: 10.3.0 or higher	7.7.0 or higher	7.7.0 or higher
Luna Cloud HSM with password-authenticated Luna 6/7: 10.2 or higher **Luna Cloud HSM with multifactor quorum-authenticated Luna 6/7:** 10.4.1 or higher **Luna 6/7:** 7.2 or higher	6.2.1 or higher	6.10.9 or higher

Mismatched Partition Policies and FIPS 140 Approved Configuration

Partitions in an HA group, and the HSMs on which they reside, must be configured with the same policy settings (see "HSM and Partition Prerequisites" on page 426). For example, Luna 6 HSMs have certain policies that have been removed from Luna 7 and Luna Cloud HSM, and new policies have been introduced.

Ensure that policies common to Luna 6/7/Luna Cloud HSM members have the same settings, according to your deployment requirements.

lunacm:> **partition showpolicies**

When setting up HA groups, note the following:

- If you are using Luna HSM Client 10.4.0 or newer, you *can* set up an HA group with a mix of FIPS and non-FIPS partitions as members. However, some limitations must be considered. For more information, refer to "Key Replication" on page 419.
- If you are using Luna HSM Client 10.3.0 or older, you *cannot* set up an HA group with a mix of FIPS and non-FIPS partitions as members.

Mismatched Key Types/Cryptographic Mechanisms

Cloning is limited to key types that are recognized by the firmware on both HSMs. If an HSM does not recognize the type of key being cloned to it, the cloning operation may fail. Ensure that the firmware on the destination HSM is capable of recognizing all cryptographic objects that are to be cloned to the source HSM.

> **NOTE** Luna HSMs comply closely with the relevant FIPS standards and their generally accepted interpretations. These are moving targets, as the crypto and security climate continues to evolve. It is possible for a validated HSM version (firmware) to be fully compliant when its NIST certificate is issued, and for same-model HSMs with newer firmware and more stringent restrictions to refuse to accept "less secure" objects.
>
> Alternatively, the more up-to-date HSM might accept an object from an earlier-firmware HSM, but permit only limited uses of such an object. This can affect the operation of HA groups, and other situations, where applications attempt operations against old keys, or with the use of antiquated mechanisms.
>
> If you are cloning between HSMs operating in FIPS 140 approved configuration (formerly FIPS mode), please consult Supported Mechanisms, for the destination HSM's version, to determine if all key types can be cloned.

Mixed-version HA groups are limited to functions that are common to all member partitions. Mechanisms are added to/removed from new firmware releases, to provide new functionality and fix vulnerabilities. Operations assigned by load-balancing to a member lacking the correct mechanism will fail. Keys created on one member may fail to replicate to the other group members.

Ensure that your applications use only mechanisms that are available on all HA group members. Use LunaCM to see a list of mechanisms available on each partition/service.

lunacm:> **partition showmechanism**

Minimum Key Sizes

Minimum key sizes are enforced when using certain cryptographic algorithms. These minimums may differ between versions. If a Luna 6 partition creates a key that is smaller than the minimum size required by Luna 7 or Luna Cloud HSM, the key will not be replicated to the other partitions in the HA group.

> **NOTE** Minimum key sizes for many mechanisms are larger in FIPS 140 approved configuration (formerly FIPS mode), and FIPS minimums may vary among firmware releases.

To avoid this, use LunaCM to check a mechanism's minimum key size. Check the same mechanism on each HA member slot, and always use the highest minimum reported in the HA group.

lunacm:> **partition showmechanism -m** <mechanism_ID>

SafeXcel 1746 Co-Processor

Luna 6 HSMs include the SafeXcel 1746 security co-processor, which is used to offload packet processing and cryptographic computations from the host processor. Applications using this co-processor are not compatible with mixed-version HA groups.

The co-processor is not enabled by default. If you have previously enabled it on your Luna 6 HSMs, you can disable it by editing the **Chrystoki.conf/crystoki.ini** configuration file as follows:

```
[Misc]
PE1746Enabled=0
```

RSA-186 Mechanism Remapping for FIPS Compliance

Under FIPS 186-3/4, the only RSA methods permitted for generating keys are 186-3 with primes and 186-3 with aux primes. RSA PKCS and X9.31 key generation is not approved in a FIPS-compliant HSM. While Luna 6.10.9 firmware allows these older mechanisms, later firmware does not (and keys created using these mechanisms cannot be replicated to Luna 7 HSMs or Luna Cloud HSM services).

If you have older applications that use RSA PKCS and X9.31 key generation, you can remap these calls to use the newer, secure mechanisms. Add a line to the **Chrystoki.conf/crystoki.ini** configuration file as follows:

```
[Misc]
RSAKeyGenMechRemap=1
```

> **NOTE** This setting is intended for older applications that call outdated mechanisms, to redirect calls to FIPS-approved mechanisms. The ideal solution is to update your applications to call the approved mechanisms.
>
> Mechanism remapping is automatic, and ignores the configuration file entry if:
>
> - you are using Luna HSM Client 10.1.0 or newer, and
> - HSM firmware is older than Luna HSM Firmware 7.7.1 (which introduced FIPS mode on individual partitions; clients up to and including Luna HSM Client 10.3.0 are unaware of the independent partition setting and do not remap mechanisms).
>
> Luna HSM Client 10.4.0 and newer are aware of the change in Luna HSM Firmware 7.7.1 and perform the mechanism remapping as expected when the current partition is in FIPS mode.

HA Performance Optimization

Luna Network HSM 7 provides significant (10x) performance improvements over Luna 6 HSMs. In a mixed-version HA group, operations assigned to Luna 6 member partitions will take longer than those assigned to Luna 7 members. The HA logic does not compensate for these performance differences, and schedules operations on the partition with the shortest queue. Since Luna 7 partitions complete operations more quickly, they will naturally be assigned more operations, but a mixed-version HA group generally does not perform as well as an HA group made up entirely of Luna 7 partitions.

The performance of Luna Cloud HSM services may be limited by network latency, compared to on-premises Luna HSMs. See "Luna On-Premises/Luna Cloud HSM Cloning" on page 211.

Thales recommends that you set a Luna 7 partition as the primary HA member (the first member specified when creating the HA group). All key generation takes place on the primary HA member, so this allows you to take advantage of the Luna Network HSM 7's vastly improved performance for:

- key generation

> random number generation

The load-balancing logic is determined by the Luna HSM Client software, so the Luna 7 behavior applies to mixed-version HA (see "Load Balancing" on page 417).

> **NOTE** The primary HA member may not remain the same over time. If the primary member fails, another member takes over all key generation operations. If you notice a significant drop in performance for key generation operations, it could mean that a Luna 6 partition or Luna Cloud HSM service has become the primary member. By default, a Luna Cloud HSM service will revert to standby once another HA member recovers.

Cloning between multifactor quorum and password-authenticated HSM partitions

Beginning with Luna HSM Firmware 7.8.0 and Luna HSM Client 10.5.0, Extended Domain Management allows you to clone keys and objects between multifactor quorum-authenticated and password-authenticated HSM partitions, including Luna Cloud HSM services, in either direction.

> Luna HSM Firmware 7.8.0 introduced the ability for any partition to have up to three cloning/security domains,
  - the original domain, created when the partition is first initialized and,
  - one or two more that can, optionally, be added later by command.

> Luna HSM Client 10.5.0 introduced the client-side commands to manage the additional domains.

Older HSMs to pre-Firmware 7.8.1 Luna 7 must already have the same Domain on both

Prior to Luna HSM Firmware 7.8.0, each application partition had a single domain that was created at partition initialization and remained until the partition was reinitialized or deleted. Cloning operations (including HA and backup) could take place between a partition with a given domain and any other partition (or service) with the same domain.

HSMs with Firmware 7.8.0 onward can import the Domain (PW or PED) of an older HSM or partition

With firmware Luna HSM Firmware 7.8.0 onward, a partition can have up to three domains, which can be created in place or imported from other Luna HSMs. Historically, it is generally possible to migrate keys *from* a given authentication type to the *same* type (PW or PED).

For migration from older HSMs to Luna HSM Firmware 7.8.0 onward it is possible to migrate keys from Multifactor Quorum (PED) authenticated or Password authenticated source partition to either Multifactor Quorum (PED) authenticated or a password-authenticated target HSM provided that the domain secret from the source partition is set as the Prime domain (of the three possible domains) on the target partition. See "Universal Cloning" on page 198.

> **NOTE** Cloning Protocol version 1 (CPv1 - the protocol available for historic HSMs) is not available in FIPS 140 approved configuration (formerly FIPS mode) from HSM firmware version 7.8.4 onward.

Cloning from a password authenticated partition is trivial, because the password text string from the source HSM needs merely to be typed in as the prime domain on the target. The only requirement for the target HSM is that it must be at a firmware version earlier than 7.8.4, if the target is in FIPS 140 approved configuration, so that it has CPv1 available to match with CPv1 from the source HSM.

Cloning from a Multifactor Quorum (PED) authenticated source to either type of target requires a PED. That is not an issue when your HSMs are deployed at physically remote locations, since you already would be using Remote PED. However, PED-authenticated source to a Password-authenticated target partition, is more involved, because Password authenticated HSMs do not have the capability to use Remote PED; therefore a PED must be connected directly to the target HSM at its remote location

> **NOTE** Password based HSMs do not have the capability to use Remote PED. The PED must be connected directly to the HSM card of a password HSM when it is to receive keys/objects cloned from a Multifactor Quorum (PED) authenticated HSM. Only then can the PED prompt for the domain key when the HSM needs it. Migration from end-of-life HSMs to current HSM partitions is expected to be a one-time event.

For a Luna SA 5/6 PED-auth migrating to a Luna 7 HSM password-auth partition, either your Luna 7 HSM is new from the factory, meaning that it would be shipped at firmware version 7.0.3 (as that was the first FIPS approved version and is still in use by many customers and requested for further purchases so as to fit into existing stables), or it has been updated to a firmware that still supports CPv1.

What is needed is to:

- clone your keys over to Luna 7 while that target is at a firmware that supports CPv1, and then
- update the firmware to one that supports Extended Domain Management
- create a partition

Cloning or Backup / Restore with SKS

Primary use-case

Huge numbers of keys can be safely off-boarded from the crypto module by scalable key storage (SKS) and stored, securely encrypted, in databases or file systems (which normally have their own backup regimes), or in keyrings within the Luna Network HSM 7 file system.

Only the SKS Master Key (SMK) needs to be

- backed-up to a Backup HSM with **partition archive backup**, for storage only (not used for any encryption/decryption while it is archived), or
- cloned to another HSM partition with **partition smkclone** (such as in an HA group), where it can insert/decrypt externally-stored keys and objects as needed when they are retrieved from your database by your application.

However, for smaller numbers of keys that fit inside a partition...

If an application creates smaller quantities of keys that can all fit inside a partition of a crypto module, then it can be convenient to keep the SMK *and* those keys together within the crypto module.

Backup/Restore

If a V1 partition containing some number of keys is backed-up (**partition archive backup**. command) to a Backup HSM, then that target Backup HSM sees

> one SMK cloned in from the source V1 partition,

> along with several SMK-encrypted opaque blobs having only their OUIDs and labels visible.

The blobs are not decrypted in the Backup HSM, but they can be identified for retrieval.

Cloning from General Purpose HSM source to General Purpose HSM target

Where you are cloning such a mixed partition (SMK plus multiple keys) from a V1 source partition to a V1 target (non-Backup),

> the SMK must be cloned over first, (with **partition smkclone**),

> then the **partition clone** command is invoked against the keys,

- where it actually launches SKS* to extract the keys (encrypted by the original SMK) from the source partition and,
- inserts the keys in the target partition, using the SMK copy that was just cloned over.

(*In this case, you use the **partition clone** command that quietly invokes SKS, so that your action to launch the process has the same "look and feel" as key cloning operations in V0 and pre-firmware-version 7.7.0 application partitions.)

For non-V1 partitions, the partition clone command behaves as previously, with no SKS involvement..

Cloning Protocols and Cipher Suite Selection

Cloning protocols for Luna HSMs have evolved along with abilities of HSMs and the complexities of inter-operation. Cloning protocols are not actions that you perform directly; rather, they are previously-agreed-upon methods used when you request cloning operations for:

> direct partition-to-partition cloning of single objects or bulk copying,

> synchronization between/among HA group member partitions,

> cloning between on-premises HSM partitions and Luna Cloud HSM services, or

> backup/restore operations with dedicated Backup HSMs.

The most straightforward circumstances for those operations are situations where all members (or all source and target combinations):

> are at the same firmware version,

> have the same partition type

- pre-7.7.0, or
- V0 , or
- V1,

> have the same settings to allow or disallow non-FIPS-approved cryptographic mechanisms, and

> have the same FM-enabled status, either:

- never been enabled (dual HOC -- regular HOC is primary and is the only one used, while FM-HOC is on standby in case FMs are enabled)
- have been enabled at some point (single HOC -- the original primary HOC was deleted, leaving only FM-HOC).

Changing any of those variables adds considerations that affect migrating/HA/cloning, where some differences might disallow the ability to clone from a source partition in one condition to a target in a different condition. When combinations of the variables coexist in the cloning scenario, that scope expands. This page sorts out who can clone (or HA) with whom, and what versions of firmware and client are needed.

In general, the cloning operation involves three parties:

> the source,

> the target, and

> the host application (examples include lunacm, Luna shell (lunash), ckdemo, or possibly your application) that invokes the client library.

Each party has a different responsibility. The host application acts as a middle-man, and the cloning rules are enforced by the general-purpose HSMs and Backup HSMs. Source HSMs use only their highest-available protocol to clone objects out, and *do not allow* an object to be cloned to an HSM (or, more specifically, to a partition / slot of an HSM, or a Luna Cloud HSM service)

> that does not share the same cloning / security domain, or

> that has a lower security profile than the source.

Additional information regarding the security aspects of cloning is available publicly from NIST Computer Security Resource Center.

Protocol	When?	What for?
CPv0	Legacy products, f/w 4.8.5 through end of f/w 5.x	End of life. No longer used.
CPv1	Luna products before release 7.7.0	The Luna HSM cloning work-horse protocol for many years. (Uses token wrapping certificate TWC3 based on RSA 2048-bit key-pair) For Luna Cloud and Luna on-premises HSMs, as long as both devices supported the same object type algorithms, then interoperability and hybrid operation were available both • *in* FIPS 140 approved configuration and • *not in* FIPS 140 approved configuration. At firmware release 7.7.0, protocol used to clone keys to Luna Cloud HSM was CPv3, and CPv1 was disabled for that purpose
CPv2	-- briefly	For Small Form-factor backup (SFF) - not used by any other product, and no longer supported.

Protocol	When?	What for?
CPv3	7.7.0	**Advantages** Addresses certification issues. (Uses a token wrapping certificate based on RSA 4096-bit key-pair - can clone objects *out* only using CPv3; can clone objects *in* via CPv0, CPv1, or CPv3, but permits only the highest supported by the sending HSM) > Compliance with FIPS SP800-131Ar2, SP800-56Br2 and current interpretations of FIPS in general > Common Criteria certifications > Uses a 4096-bit certificate, signed by the primary HOK of the HSM > Backup HSM for FM-Enabled HSMs > SKS Blob Based Backup instead of cloning based backup > Support for old client applications and old client libraries > General security improvements > Functionality Module-Enabled HSM can clone out to an FM-Disabled HSM > FM-Disabled HSM prevented from cloning out to an FM-Enabled HSM(*) **Constraints** Satisfying all the wide-ranging requirements met by CPv3 resulted in CPv1 support being dropped, which limited the ability of Luna on-premises HSMs to share keys with Luna Cloud HSM. Keys could flow from Cloud to on-premises, but not the other way. CPv1 is enabled or disabled via partition policy **42** setting (introduced with firmware 7.7.1). When CPv1 is enabled, CPv3/CPv4 are disabled.

Protocol	When?	What for?
CPv1 bi-directional	7.7.1	Re-added support for CPv1 when not in FIPS 140 approved configuration (formerly FIPS mode). However, many use-cases require/mandate FIPS 140 approved configuration. Having policy 42 **on** - Allows interoperability between CPv1 HSMs (pre-7.7.0 Luna Network HSM 7 and Luna PCIe HSM 7, and Luna Cloud HSM) on the one hand, and CPv3 (7.7.x) HSMs on the other > Cloning objects in both directions (to/from) the CPv1-only and CPv3- or CPv4-capable HSMs > Mixed-mode HA, including Luna Network HSM 7 partitions with Luna Cloud services ("partitions") > does not deal with incompatibility where multifactor quorum authentication was not supported by Luna Cloud HSM at the time this was released > when policy 42 is **on** for a partition, CPv3 and CPv4 options are overridden and the use of CPv1 is forced for both incoming and outgoing cloning operations, so the operations can proceed only if *both* source and target partitions have CPv1 available. • pre-7.7.1 firmware, CPv1 is always available • Luna HSM Firmware 7.7.1 or newer, CPv1 can be disabled/enabled by policy See "CPv3 characteristics" on the next page below for additional information regarding CPv3. CPv1 is enabled or disabled via partition policy **42** setting (introduced with firmware 7.7.1). When CPv1 is enabled, CPv3/CPv4 are disabled.

Protocol	When?	What for?
CPv4	7.8	CPv4, along with Extended Domain Management, implements universal cloning. Multiple strong cipher-suites are available to choose from and they can be activated or deactivated at will. Each partition can have up to three cloning/security domains, allowing you to clone keys and objects, in both directions, when in FIPS 140 approved configuration (formerly FIPS mode) or not in FIPS 140 approved configuration, between: > on-premises multifactor quorum-authenticated partitions and either • on-premises password-authenticated partitions or • Luna Cloud HSM services. Extended Domain Management permits: > rotation of domain secrets (similar to password rotation) > shifting of domains > splitting of domains > joining of domains. Having CPv4 available requires Luna HSM Firmware 7.8.0 or later. *Using* CPv4 requires that the other HSM(s) involved in cloning, HA, etc., also have Luna HSM Firmware 7.8.0 or later; no additional configuration is needed to make CPv4 available (except CPv1, policy **42** must be off). Luna Cloud HSM is already updated to interact with on-premises HSMs at Luna HSM Firmware 7.8.0 > CPv4 is configured by commands to display and select cipher suites that can potentially be used by CPv4, as well as the suite used by CPv3, and allows enabling or disabling of any, or all, of the individual supported suites. > Sessions in CPv4 do not persist indefinitely. CPv4 times-out and re-establishes sessions for Perfect Forward Secrecy. This is done transparently to the user and requires no overt action from you. > When CPv4 is negotiated between two partitions (or a partition and a cloud service), the protocol clones all objects specified by the command during a single negotiated session. This contrasts with CPv1 and CPv3, which open and close a session for each individual object. See "Universal cloning (CPv4) characteristics" on page 223 below for additional information regarding CPv4. CPv1 is enabled or disabled via partition policy **42** setting (introduced with firmware 7.7.1). When CPv1 is enabled, CPv3/CPv4 are disabled.

(* An FM-Enabled target is not trusted compared to one that has never had external/foreign code introduced. Enabling FMs on an HSM is a one-way operation, because it deletes the standard Luna hardware origin key [HOK] and asserts an FM-HOK.)

CPv1 partition policy **42** can never be ON for V1 partitions.
For V0 partitions, whether created new, or updated from pre-7.7.x by firmware update, HSM policy **12** is OFF; therefore CPv1 partition policy **42** is OFF initially.

CPv3 characteristics

1. The Backup HSM prevents objects, cloned in by CPv3, from being cloned back out with CPv1.

2. The Backup HSM prevents cloning objects from a pre-FM HSM (firmware older than Luna HSM Firmware 7.4.0) to an HSM with FMs enabled.
3. The Backup HSM prevents cloning objects from an FM-never-enabled HSM (any firmware) to an FM-Enabled HSM.
4. A single Backup HSM can be used with most Luna HSMs, while preventing the Backup from moving keys to an HSM with known vulnerabilities. The Backup HSM must use CPv3 and be FM-Disabled, but it is allowed to move keys
   - to a CPv1 HSM or
   - to an FM-Enabled HSM

   if the object or SIM secret came from a CPv1 or FM-Enabled HSM.
5. Luna HSM Firmware 7.7.0 clones objects *out* only using the CPv3, and clones *in* via CPv3, CPv1 or CPv0, depending on the cloning source.
   - V0 partitions use CPv3 to clone cryptoki object or SMK to a Luna Backup HSM G5 or an HSM with Luna HSM Firmware 7.7.0 or newer.
   - V1 partitions use CPv3 to clone SMK to a Luna Backup HSM G5 or to an HSM with Luna HSM Firmware 7.7.0 or newer. Cryptoki objects can be extracted in a SIM4 SKS blob structure only.
6. The Backup HSM can clone objects *out* using CPV3 or CPV1 depending on the target HSM to which it attempts to clone.
7. For HA groups that include members with firmware versions 5.x, 6.x, 7.x (older than Luna HSM Firmware 7.7.0), HSM migration to an HA group of Luna HSM Firmware 7.7.0 HSMs requires that each 7.x HSM member be upgraded to Luna HSM Firmware 7.7.0 firmware. The master must be the last updated to allow cloning from protocol v1 to protocol v 3 as the reverse is not possible.
8. The firmware no longer drops unrecognized attributes in incoming flattened objects, and generally refuses to create the object if any are encountered during the un-flattening process. To ensure maximum backward compatibility while maintaining security, certain newer attributes (like CKA_ASSIGNED) *that are in the false/unset state* are excluded from this behavior (so the attribute is, in fact stripped), allowing an object to be cloned to an HSM that does not support the attribute. This is possible because both copies of the object would remain functionally equivalent. By contrast, an attribute like CKA_PER_KEY_AUTH_DATA, if present on an object, has no default and is mandatory, so if such an attribute is set it is always included when the object is cloned (if the cloning-target partition supports that attribute to allow cloning of the object).

   To summarize: former behavior was to strip/drop unknown attributes and permit the reduced object to be cloned, while the new behavior is to simply refuse to clone a source object (that is set) if the destination is not configured to support the attribute.
9. If Luna Backup HSM G5 is used, full backup support for CPv3 HSMs requires Luna Backup HSM G5 Firmware 6.28.0.
10. Cloning with CPv3 is more complex than CPv1, and therefore might take slightly longer per individual operation.

Backup

The Backup HSM and the general-purpose HSMs (at whatever firmware versions) generally determine what is properly allowed to be backed-up and restored. The Backup HSM prevents itself from being used as a gateway device to allow keys to be cloned in from what we consider a more secure environment and be cloned out to a

less secure environment. For example, it is not permissible to clone a key from an FM-Disabled HSM to a Backup HSM, and then clone that key from the backup HSM to an FM-Enabled HSM.

Universal cloning (CPv4) characteristics

Application is required to drive cloning-session renegotiation - Universal cloning session negotiation is a multi-step process, and if the CPv4 configuration changes while the negotiation is in progress - perhaps by making a negotiated algorithm unavailable - the negotiation might fail. Initial negotiation is part of the protocol, but re-negotiation would need the driving application to perform the re-negotiation. This is likely a rare combination, but possible. However, once a negotiation is complete, the CPv4 session is self-contained, and can be used until it expires or is closed.

Validity timestamps have some forgiveness - The protocol messages, the negotiated session keys, and the extracted key blobs, have timestamps that define for how long they are valid. In order to avoid the chance of clock-related problems, or network latency and delays, during negotiation, the protocol is forgiving of a delta up to 5 minutes out-of-sync between members - any greater, and the universal cloning negotiation fails with CKR_CLOCK_NOT_IN_SYNC.

HSM clock management by SO - The Audit role has always been able to set time, and beginning with Luna HSM Firmware 7.8.0 and newer, clock management can be performed by the HSM SO using lunash hsm time get and hsm time sync commands. These should be run to initialize the HSM clock time, then HSM Policy **57 - Allow sync with host time** should be set (ON) so that the one-time manual sync operation becomes a daily, automatic event to prevent HSM clock drift outside of parameters; note that it is OFF by default, for backward compatibility.

> **NOTE** You can encounter the error CKR_TIME_NOT_INITIALIZED if lunash hsm time get and hsm time sync commands have not been employed to set the time. As well, you could encounter CKR_CLOCK_NOT_IN_SYNC if the clocks on source and target HSMs are not within time tolerance for CPv4 cloning operations.
> Additionally, other operations need HSM time properly set and synchronized - remote Audit logging, for example, expects tight drift control, to prevent log messages appearing out of order.
>
> Clock synchronization, leading back to trusted time source, is needed on both the source HSM and the target.

HSM clock sync with host time - The HSM clock syncs against host time every 24 hours. If time has become so far out-of-sync that the HSM clock will not accept automatic update, then an Alarm is logged and attempts to synchronize are stopped until an administrator (SO or Audit roles) updates by command -- after which, automatic synchronization resumes.

Functionality Module cloning restrictions - Universal cloning continues to enforce the Functionality Module (FM) restrictions, where objects from a non-FM-Enabled HSM are not permitted to be transferred to an HSM that is FM-enabled. That is, keys can flow only from an FM-Enabled HSM

- to an FM-never-Enabled HSM or
- to an FM-Disabled or never-enabled HSM, and
- never in the other direction.

Unknowns are rejected - Universal cloning rejects

- any object that it does not recognize,
  as well as
- any object with an *attribute* that it does not recognize.

The HSM returns OBJECT_TYPE_UNKNOWN and ATTRIBUTE_TYPE_UNKNOWN.

Non-extractable must remain non-extractable - Keys that are in a Luna HSM where they are not extractable cannot be moved to an environment where they could be extracted.

Interoperability of universal cloning - For improved compatibility when sharing keys with other HSMs, non-standard PKCS11 attributes are stripped when they have values that are safe to remove (that is, if they have the permissive default values).

- PKA attributes (Per Key Authorization) are removed if PKA has not been set for the key; CPv4 is allowed for objects* only in V0 partitions (cloning partitions), so the PKA attributes should not be set.
- CKA_ASSIGNED is removed if it is false, which it should be for V0 partitions.
- DES3 counters are removed.
- Key Ring related attributes are removed.

> **NOTE** * In V1 partitions, CPv4 is used to clone the SKS master key (SMK) and blobs encrypted by it (for HA and backup), but not individual keys and objects.

No Network Replication - CPv4 is always allowed when the corresponding cloning policies are enabled; the Network Replication policy 16 (which, in previous HSM versions could have overridden the cloning policies) does not apply to CPv4. Network Replication is considered almost irrelevant when most applications use HA, and many also use STC. But if your application requires it, then simply disable CPv4 by disabling all its cipher suites.

Enabling Extended Domain Management - Extended Domain Management is enabled with Partition Policy **44 - Enable/Allow Extended Domain Management** (where the capability is always ON (following firmware update) and the policy is initially OFF by default, for backward compatibility. Optionally add up to *two* additional security/cloning domains to a partition, in addition to the original domain that is imprinted when the partition is initialized. This permits previously disallowed or inconvenient key-cloning actions like:

- cloning keys across domains that differ from the original, as long as source and target partitions share *at least one domain in common*,
- cloning in both directions between password-authenticated partitions (including Luna Cloud HSM services) and multifactor quorum-authenticated partitions.

As a general rule, to establish a security/cloning domain in common between a password-authenticated partition, and a multifactor quorum-authenticated partition,

- you require a locally-connected Luna PED at the password-authenticated HSM, if you choose to import a domain secret PED key into a password-authenticated partition, but
- you do not need a Luna PED at the password-authenticated partition location if you are adding its text domain into a multifactor quorum-authenticated partition.

Domain labeling - when and how - Extended Domain Management implements the naming / labeling of domains in order to distinguish them - the initial domain can be labeled, but is initially unlabeled, for backward compatibility (where only one domain was ever used), whereas if more domains are added, then *all* must have

different labels. The feature is managed by lunacm commands:

> partition domainlist and

> partition domainadd and

> partition domainchangelabel and

> partition domaindelete.

Which domain is primary and how to change - All partitions, after initialization, have the current or original security/cloning domain marked as the primary, the domain that is chosen by default for cloning. For a partition with more than one domain, either of the others can be designated as primary, instead, using the **partition domainadd** and **partition domainchangelabel** commands, by invoking their **-primary** option.

Partition PO role login is required, to create or change a domain (after the first domain created by partition initialization). This command requires that **partition policy 44:** "Allow Extended Domain Management" on page 351 is set to ON.

Domain ordering unaltered by deleting - If three domains exist, and you delete the middle one, that does *not* alter the domain ordering, nor does it alter which one is the primary. You cannot delete the primary domain from a partition at Luna HSM Firmware 7.8.0, or newer, even if other non-primary domains are set; you *must set a different primary first*.

Increase of existing partition size - At firmware update, from pre-7.8.0 versions, all pre-existing partitions are given a little more space to store domain labels and universal cloning configuration. This would generally not be noticed, as it is negligible against the size increase afforded by the prior Luna HSM Firmware 7.7.0 update (with SKS, PKA, etc.).

Primary domain - On pre-firmware 7.8.0 HSM partitions the single possible domain is effectively the primary domain. For firmware 7.8.0 and newer, partitions can have as many as three domains. Of the three possible, one domain is always primary, but the status of primary can be moved to another domain if needed. "Primary" in this context means "the one that is tried first". If there is no match for the primary domain on the source partition, the systems goes on to try for other matching domains.

[**Summary**]

When cloning from a partition of an HSM with firmware version lower than 7.8.0 to a version 7.8.0 or higher with multiple domains, the primary domain is used.

[**Explanation**]

On firmware version 7.8.0-or-newer HSM partitions, the partition always has at least one domain, and can have as many as three, any of which can be a password-style text domain, or a multi-factor quorum type (PED key-secret domain. One of the three possible domains is designated primary, and is the first one looked at when a cloning/migration operation is attempted.

If a firmware version 7.8.0-or-newer target is already a member of the same domain as a pre-7.8.0 firmware source partition, and that domain is primary on the v7.8.0-or-newer partition, then cloning/migration can proceed straightaway.

If the target HSM partition is at firmware 7.8.0 or newer, then if its partition initially has a different domain from the source partition, the target partition can:

- use Extended Domain Management to add the source partition's domain as one of the three domains that the target can support and

- make the domain that was obtained from the source become the primary domain on the target by using the **-primary** option when adding a domain with partition domainadd, and
- cloning/migration can proceed (includes backup, HA, etc.).

CHAPTER 6: Enabling and Using CPv4

Enabling and Using CPv4

The ability to employ cloning protocol version 4 (CPv4) becomes available when using Luna HSM Client 10.5.0 and newer, with HSMs at Luna HSM Firmware 7.8.0 or newer. (See "Universal Cloning" on page 198)

In order to enable the use of CPv4, the HSM Security Officer *must* set the clock either before or immediately after the firmware has been updated.

No additional configuration steps are required; the Luna Cloud HSM service is compatible.

> **NOTE** There is no plan to update the Luna Backup HSM G5 (nor any version of firmware 6) to support CPv4.

CPv4 is handled like the prior CPv1 cloning protocol.

- The default cloning configuration displays both CPv1 and CPv4. (However, Luna HSM Client 10.4.1 and older can clone using only CPv1).
- Any scenario where CPv1 can be used, CPv4 can be used.
- Any scenario where CPv1 cannot be used, CPv4 cannot be used.
- CPv4 can be used to clone objects in V0 partitions, and clone the SMK in V1 partitions.
- The same set of roles that can use CPv1 can use CPv4. This includes the allowance for the Crypto-User to clone public objects.
- Prior rules about Functionality Modules are preserved where the current partition has FMs allowed:
  - can clone out to non-FM partitions (allowed to clone from lesser-security to greater-security environment)
  - non-FM partitions cannot clone in (not allowed to clone from greater-security to lesser-security environment)
- In Cloud HSM service, CPv4 is the default protocol; however if only one side has CPv4 available, it reverts to CPv1.
- Partition Policy 8 in Luna Cloud HSM fdisplays both CPv1 and CPv4, however when paired with Luna HSM Client 10.4.1 and earlier clients, only CPv1 is permitted.

> **NOTE** The Partition Policy difference between Luna Network HSM 7 and Luna Cloud HSM is as follows:
>
> - Luna HSM slot “42: Allow CPv1 : 1” means to force CPv1 and disable all other cloning protocols
> - Luna Cloud HSM slot “8: Allow CPv1 (Cryptovisor Only) : 1” means to allow CPv1 but you can *also* clone using CPv4

Where is copying, sharing, migration of keys possible - from what source to what destination?

This section summarizes paths and limitations for getting keys and objects from one HSM to another, starting with the most general case and proceeding through more particular and sheltered use cases.

Moving or copying objects from any HSM to any Luna HSM or any Luna HSM to any HSM

Only objects whose properties allow them to be wrapped-off of an HSM and wrapped-onto another HSM can be transferred or migrated(*) in this fashion between Luna HSMs and non-Luna HSMs.

> **NOTE** SKS blobs (objects encrypted by the Scalable Key Storage Master Key [SMK]) can be considered wrapped-off/wrapped-on, but the SMK that can decrypt them is shared only among Thales Luna HSMs with a security/cloning domain in common.

By manipulating HSM or partition policies, it is possible to modify the handling options for keys/objects having secret and private attributes. This requires pre-planning (before creating and using such objects) for these reasons:

- because the changes to such policies tend to be destructive of partition contents;
- because, while in many cases it is possible to override policy-change destructiveness defaults by using a Partition Policy Template that was adjusted for the purpose, when creating a partition, this same action is refused by the HSM if the HSM detects an attempted modification that would compromise security.

Moving or copying objects from any Luna HSM to any other Luna HSM

Luna HSMs have more options for sharing/migrating keys and objects with other Luna HSMs, although specific cooperation and prearrangement are needed.
That is, you can share keys with virtually any Luna HSM, but only if you want to allow it. Your HSM and its application partitions and their contents are not open to other Luna HSMs without your permission and action.
When testing requests for copying/migrating keys and objects, a Luna HSM needs to recognize the other HSM by its Hardware Origin Certificate, derived from its Hardware Origin Key.

Moving or copying objects from any Luna HSM that you control to another Luna HSM that you control

This is the situation where you own/control both HSMs that are to be exchanging key material ("control", in addition to outright ownership of on-premises HSMs, could refer to a Luna Cloud HSM service subscription). The parameters of such a situation might be either:

- before firmware 7.8.0, any other HSM of the same authentication type, Password-auth vs Multifactor Quorum, can clone objects with each other of the same authentication type (as pre-configured before being populated), but not with the other type.

  or

- starting with firmware 7.8.0,
  - source can be either Password-authenticated or Multifactor Quorum authenticated, and
  - destination can be either Password-authenticated or Multifactor Quorum authenticated
  - as long as

  - at least one of the HSMs is at firmware version 7.8.0 or newer and
  - the affected partition has Partition Policy 44 Allow Extended Domain Management set to ON, and
- as long as the intended source and destination partitions have at least one security/cloning domain in common between them and
- as long as both have matching cloning cipher suites available (in other words, not disabled).

Two arrangements

That is, both parties to the transaction can have firmware 7.8.0 or newer and have Policy 44 set to ON, which offers maximum flexibility, as either participant could import a domain from the other (depending on which selection was more convenient).

Or just one party could have the newer firmware and policy setting, while the other (perhaps with older firmware) could have only its original default domain, and in that case, the partition with up to three domains must import the domain from the partition with only the single domain it acquired at initialization.

Across authentication types

In general, if cloning is desired between a password-authenticated partition and a multifactor quorum-authenticated partition, it is more convenient to share the password-auth domain string to become one of the domains on the multifactor quorum-auth partition, since the latter would already have a PED connection, whereas to share an iKey secret from a multifactor quorum partition to a password-auth partition requires that a Remote PED connection be established to the password-auth partition's HSM for just that purpose. However, both options are open, if your situation imposes constraints.

Restriction to any Luna HSM that you control that has certain configurations

Configuration settings that differ between participating source-vs-target HSM partitions (or at the level of the containing HSM) might imply constraints that could affect whether an object is acceptable to clone, or if cloned-in or wrapped-in, whether it can be used in the same ways as on the source partition. Some such configuration differences might include:

> If the source is set to allow use of non-FIPS algorithms and key sizes and curve types, etc., while the destination is restricted to FIPS-only crypto mechanisms, or perhaps the same mechanisms are available on both, but with restricted operations on the target, and so on;

> Similarly, if the firmware versions of source and destination differ, then so might available mechanisms, key sizes, etc.;

> **Functionality Modules**

An HSM might have the option to invoke Functionality Modules, meaning it has firmware version 7.4.0 or newer and was manufactured with both the regular HOK / HOC that all Luna HSMs have, and also the FM-HOK and FM-HOC included, on standby, and might never have FMs activated (by never having HSM Policy 50 Allow Functionality Modules set to ON

- cloning is permitted from an HSM that has FMs allowed to another HSM that has had FMs allowed (FM license is installed and HSM Policy 50 is ON)
- cloning is permitted from an FM-allowed HSM to a never-FM-allowed HSM
- cloning is not permitted from an HSM where FMs have never been allowed (the HSM retains its HOK/HOC) to an FM-allowed HSM (has only FM-HOK/FM-HOC) - this is considered an attempt to clone from a higher-security environment to a lower-security environment.

> **NOTE** Use of CPv4 requires that the HSM time be set before any cloning operation is attempted that invokes the protocol.
>
> It is recommended to have the host synchronized to a secure ntp or nts server, before synchronizing the HSM time to host time.
>
> Use hsm time commands.

Enabling and Disabling CPv4 Cipher Suites

Cipher suites for Cloning Protocol version 4 (CPv4) are used for cloning to-and-from partitions, if the individual suites are enabled for a partition, and the use of CPv4 is *not* prevented by CPv1 being active (see "Allow CPv1" on page 349).

> **NOTE** Use of CPv4 requires that the HSM time be set before any cloning operation is attempted that invokes the protocol.
>
> It is recommended to have the host synchronized to a secure ntp or nts server, before synchronizing the HSM time to host time.
>
> Use hsm time commands.

By default, eight CPv4 cipher suites are available and active, and the system negotiates the best/most secure suite for the current cloning operation, based on which suites are available to both the source and target partitions. If you have reason to do so, you can disable some cipher suites, which reduces negotiation time among those that remain enabled. You can also enable desired cloning cipher suites that have been disabled.

To show the current status of enabled and disabled cipher suites

1. Run **partition ciphershow** command.

```
lunacm:>partition ciphershow

 Cipher ID    Cipher Suite                                                    Enabled
_____________________________________________________________________________________

 0            CPv3 RSA-4096-PKCS-SHA2-384 AES-256-GCM                         Yes

 1            CPv4 ECDSA-P521-SHA2-512 ECDH-P521-SHA2-512 AES-256-GCM         No

 2            CPv4 ECDSA-P521-SHA2-512 ECDH-P521-SHA2-512                     No
              AES-256-CTR-HMAC-SHA2-512

 3            CPv4 ECDSA-BP512-SHA2-512 ECDH-BP512-SHA2-512                   No
              AES-256-GCM

 4            CPv4 ECDSA-BP512-SHA2-512 ECDH-BP512-SHA2-512                   No
              AES-256-CTR-HMAC-SHA2-512

 5            CPv4 ECDSA-P521-SHA3-512 ECDH-P521-SHA3-512 AES-256-GCM         No

 6            CPv4 ECDSA-P521-SHA3-512 ECDH-P521-SHA3-512                     No
              AES-256-CTR-HMAC-SHA3-512

 7            CPv4 ECDSA-BP512-SHA3-512 ECDH-BP512-SHA3-512                   No
```

```
              AES-256-GCM

8             CPv4 ECDSA-BP512-SHA3-512 ECDH-BP512-SHA3-512           No
              AES-256-CTR-HMAC-SHA3-512

9             CPv4 ECDSA-P521-SHA2-512 ECDH-P521-ML-KEM1024-SHA2-512  Yes
              AES-256-GCM

10            CPv4 ECDSA-P521-SHA2-512 ECDH-P521-ML-KEM1024-SHA2-512  Yes
              AES-256-CTR-HMAC-SHA2-512

11            CPv4 ECDSA-BP512-SHA2-512                               Yes
              ECDH-BP512-ML-KEM1024-SHA2-512 AES-256-GCM

12            CPv4 ECDSA-BP512-SHA2-512                               Yes
              ECDH-BP512-ML-KEM1024-SHA2-512
              AES-256-CTR-HMAC-SHA2-512

13            CPv4 ECDSA-P521-SHA3-512 ECDH-P521-ML-KEM1024-SHA3-512  Yes
              AES-256-GCM

14            CPv4 ECDSA-P521-SHA3-512 ECDH-P521-ML-KEM1024-SHA3-512  Yes
              AES-256-CTR-HMAC-SHA3-512

15            CPv4 ECDSA-BP512-SHA3-512                               Yes
              ECDH-BP512-ML-KEM1024-SHA3-512 AES-256-GCM

16            CPv4 ECDSA-BP512-SHA3-512                               Yes
              ECDH-BP512-ML-KEM1024-SHA3-512
              AES-256-CTR-HMAC-SHA3-512
```

The output shown for your partition might vary from the example above. If the output from the command shows only 8 ciphers, you have an older firmware version - update your HSM firmware and client for support of the additional ciphers.

To enable a cipher suite

1. Run the **partition cipherenable** command with the ID of the cloning cipher suite you want to enable.

```
lunacm:>partition cipherenable -id 1
CPv4 ECDSA-P521-SHA-512 ECDH-P521-SHA512 AES-256-GCM is now enabled

Command Result : No Error
```

2. Run **partition ciphershow** command to verify the result.

```
lunacm:>partition ciphershow

 Cipher ID    Cipher Suite                                              Enabled
_____________________________________________________________________________

 0            CPv3 RSA-4096-PKCS-SHA2-384 AES-256-GCM                    Yes

 1            CPv4 ECDSA-P521-SHA2-512 ECDH-P521-SHA2-512 AES-256-GCM   Yes

 2            CPv4 ECDSA-P521-SHA2-512 ECDH-P521-SHA2-512                 No
              AES-256-CTR-HMAC-SHA2-512
```

```
3            CPv4 ECDSA-BP512-SHA2-512 ECDH-BP512-SHA2-512            No
             AES-256-GCM

4            CPv4 ECDSA-BP512-SHA2-512 ECDH-BP512-SHA2-512            No
             AES-256-CTR-HMAC-SHA2-512

5            CPv4 ECDSA-P521-SHA3-512 ECDH-P521-SHA3-512 AES-256-GCM  No

6            CPv4 ECDSA-P521-SHA3-512 ECDH-P521-SHA3-512              No
             AES-256-CTR-HMAC-SHA3-512

7            CPv4 ECDSA-BP512-SHA3-512 ECDH-BP512-SHA3-512            No
             AES-256-GCM

8            CPv4 ECDSA-BP512-SHA3-512 ECDH-BP512-SHA3-512            No
             AES-256-CTR-HMAC-SHA3-512

9            CPv4 ECDSA-P521-SHA2-512 ECDH-P521-ML-KEM1024-SHA2-512   Yes
             AES-256-GCM

10           CPv4 ECDSA-P521-SHA2-512 ECDH-P521-ML-KEM1024-SHA2-512   Yes
             AES-256-CTR-HMAC-SHA2-512

11           CPv4 ECDSA-BP512-SHA2-512                                Yes
             ECDH-BP512-ML-KEM1024-SHA2-512 AES-256-GCM

12           CPv4 ECDSA-BP512-SHA2-512                                Yes
             ECDH-BP512-ML-KEM1024-SHA2-512
             AES-256-CTR-HMAC-SHA2-512

13           CPv4 ECDSA-P521-SHA3-512 ECDH-P521-ML-KEM1024-SHA3-512   Yes
             AES-256-GCM

14           CPv4 ECDSA-P521-SHA3-512 ECDH-P521-ML-KEM1024-SHA3-512   Yes
             AES-256-CTR-HMAC-SHA3-512

15           CPv4 ECDSA-BP512-SHA3-512                                Yes
             ECDH-BP512-ML-KEM1024-SHA3-512 AES-256-GCM

16           CPv4 ECDSA-BP512-SHA3-512                                Yes
             ECDH-BP512-ML-KEM1024-SHA3-512
             AES-256-CTR-HMAC-SHA3-512
```

To disable a cipher suite

1. Run the **partition cipherdisable** command with the ID of the cloning cipher suite you want to disable.

```
lunacm:>partition cipherdisable -id 0
CPv3 RSA-4096-PKCS-SHA-384 AES-256-GCM is now disabled

Command Result : No Error
```

2. Run the **partition ciphershow** command to verify the result.

```
lunacm:>partition ciphershow

 Cipher ID    Cipher Suite                                          Enabled
```

```
0           CPv3 RSA-4096-PKCS-SHA2-384 AES-256-GCM                  No

1           CPv4 ECDSA-P521-SHA2-512 ECDH-P521-SHA2-512 AES-256-GCM  Yes

2           CPv4 ECDSA-P521-SHA2-512 ECDH-P521-SHA2-512               No
            AES-256-CTR-HMAC-SHA2-512

3           CPv4 ECDSA-BP512-SHA2-512 ECDH-BP512-SHA2-512             No
            AES-256-GCM

4           CPv4 ECDSA-BP512-SHA2-512 ECDH-BP512-SHA2-512             No
            AES-256-CTR-HMAC-SHA2-512

5           CPv4 ECDSA-P521-SHA3-512 ECDH-P521-SHA3-512 AES-256-GCM   No

6           CPv4 ECDSA-P521-SHA3-512 ECDH-P521-SHA3-512               No
            AES-256-CTR-HMAC-SHA3-512

7           CPv4 ECDSA-BP512-SHA3-512 ECDH-BP512-SHA3-512             No
            AES-256-GCM

8           CPv4 ECDSA-BP512-SHA3-512 ECDH-BP512-SHA3-512             No
            AES-256-CTR-HMAC-SHA3-512

9           CPv4 ECDSA-P521-SHA2-512 ECDH-P521-ML-KEM1024-SHA2-512   Yes
            AES-256-GCM

10          CPv4 ECDSA-P521-SHA2-512 ECDH-P521-ML-KEM1024-SHA2-512   Yes
            AES-256-CTR-HMAC-SHA2-512

11          CPv4 ECDSA-BP512-SHA2-512                                 Yes
            ECDH-BP512-ML-KEM1024-SHA2-512 AES-256-GCM

12          CPv4 ECDSA-BP512-SHA2-512                                 Yes
            ECDH-BP512-ML-KEM1024-SHA2-512
            AES-256-CTR-HMAC-SHA2-512

13          CPv4 ECDSA-P521-SHA3-512 ECDH-P521-ML-KEM1024-SHA3-512   Yes
            AES-256-GCM

14          CPv4 ECDSA-P521-SHA3-512 ECDH-P521-ML-KEM1024-SHA3-512   Yes
            AES-256-CTR-HMAC-SHA3-512

15          CPv4 ECDSA-BP512-SHA3-512                                 Yes
            ECDH-BP512-ML-KEM1024-SHA3-512 AES-256-GCM

16          CPv4 ECDSA-BP512-SHA3-512                                 Yes
            ECDH-BP512-ML-KEM1024-SHA3-512
            AES-256-CTR-HMAC-SHA3-512
```


Updating or rotating cloning domain secrets

From time to time, it might be necessary to change the secret associated with a role on an HSM appliance, a role on a cryptographic module (HSM) or a partition of an HSM, or a cloning domain secret. Reasons for changing credentials include:

> Regular credential rotation as part of your organization's security policy

> Compromise of a role or secret due to loss or theft of a PED key

> Personnel changes in your organization or changes to individual security clearances

> Changes to your security scheme (implementing/revoking M of N, PINs, or shared secrets)

Changing/rotating domains was not possible when each partition was allowed only one cloning domain (prior to HSM firmware 7.8.0), as changing the domain required initializing the partition, which destroyed all data. It also meant that once a partition had a new domain, it could not receive objects restored from backup HSM/partition that had only the old cloning domain.

This changed with the introduction of Extended Domain Management in Luna HSM Client 10.5.0 and Luna HSM Firmware 7.8.0 or newer, where each partition can have up to three different cloning domains, and those domains can be added or deleted as needed.

How to change or rotate the cloning domain

Prerequisites

> The partition must belong to an HSM at firmware 7.8.0 or newer.

> Partition Policy 44: Allow Extended Domain Management must be set to ON (partition showpolicies and partition changepolicy if necessary).

> The partition must have an available space for the new domain

- either it currently has just one or two domains (partition domainlist),
- or it has three domains (the maximum) but one can be deleted to make room for a new domain (with partition domaindelete) if necessary.

To rotate the cloning domain of an application partition,

1. Create the new domain, assigning it primary status and a label that will be used for that domain in any other partition or backup HSM with which it will need to clone objects.

   partition domainadd -domain <string for PW-auth> **-domainlabel** <unique label for new domain> **-primary**

   (provide a suitable text string as the domain secret

   or

   partition domainadd -domainped -domainlabel<unique label for new domain> **-primary**

   (attend to the PED with an appropriate iKey [blank if creating a new domain, or else containing an existing domain to be reused here], since you have elected to add a PED-mediated domain secret)

2. If the next partition you are rotating already has three domains (partition domainlist) delete one of them that is no longer needed, to make room for the new domain secret.

   partition domaindelete **-domainlabel**<label of domain to delete>

Otherwise, skip to the next step.

3. Continue to create the same domain, as in step 1, on each other partition/HSM that is expected to participate in cloning with the first partition

   (includes any of another partition that needs to contain the same key material, such as

   - a backup HSM that already contains relevant backed-up material ,
   - other members of any HA group where the current partition is a member,
   - etc.)

   partition domainadd -domain <string for PW-auth> **-domainlabel** <same new label as used in step 1, above> **-primary**

   or

   partition domainadd -domainped -domainlabel<same new label as used in step 1, above> **-primary**

4. Delete the old, rotated-out domain from all affected partitions/HSMs, after all other intended cloning partners possess the new domain that is being rotated in

   partition domaindelete **-domainlabel**<label of domain to delete>

5. All the affected partitions should now have the new domain, properly labeled and identified as "primary" and should no longer have the old, superseded domain secret. You have satisfied the rotation requirement (with respect to domain secrets) for the current cycle. Carry on with normal operation.

6. [Optional] If you wish to verify before deleting all copies of the old pre-rotation domain secret, that the new, post-rotation domain is functional, and is now the domain being used for cloning,

   a. Delete the old domain secret from all but one of the affected partitions.

   b. Leave the old domain on just that one (source) partition (that also has the new domain) for now, so you still have a copy of it.

   c. Perform a cloning operation between the partition that has both the old and new domain secrets, and any of the other partitions that have the new domain, but no longer have the old domain. Successful cloning indicates that the new domain secret is being used, as it should be the only domain in common among all the affected partitions/HSMs.

   d. Delete the last remaining copy of the old domain.

> **NOTE** For multi-factor quorum-authenticated partitions when all partitions have been imprinted with new role credentials and new domain secrets, then it no longer matters if any iKeys retain the old secrets, as they can no longer be used to access or clone respectively, and can be re-used/overwritten if desired.

CHAPTER 7: Multifactor Quorum Authentication

The Luna PIN Entry Device (Luna PED) provides PIN entry and secret authentication to a Luna HSM that requires trusted-path multifactor quorum authentication. The requirement for multifactor quorum or password authentication is configured at the factory, according to the HSM model you selected at time of purchase.

The Luna PED and PED keys are the only means of accessing the multifactor quorum-authenticated HSM's administrative functions. They prevent key-logging exploits on workstations connected to the host HSM, because authentication is delivered directly from the hand-held Luna PED to the HSM via the independent, trusted-path interface. No password is entered via computer keyboard.

> **NOTE** If you are updating or have already updated to Luna HSM Firmware 7.7.0 or newer, refer to Special Considerations for Luna HSM Firmware 7.7.0 and Newer for more information about multifactor quorum authentication.
>
> Luna Network HSM 7 7.x requires Luna PED Firmware 2.7.1 or newer. This firmware is backward-compatible with Luna Network HSM 7 6.x.

This chapter contains the following sections about multifactor quorum authentication:

Multifactor Quorum Authentication Architecture

The multifactor quorum authentication architecture consists of the following components:

- **Luna PED:** a PIN Entry Device with a local or remote connection to the HSM. The PED reads authentication secrets from PED keys on behalf of an HSM or partition (see "Luna PED Hardware Functions" on page 246).
- **Authentication secrets:** Cryptographic secrets generated by the HSM and stored on PED keys. These secrets serve as login credentials for the various roles on the HSM. They can be shared among roles, HSMs, and partitions according to your security scheme.
- **PED keys:** physical USB-connected devices that contain authentication secrets, created by the HSM (see "PED keys" on the next page). PED keys have the following custom authentication features:
  - **Shared Secrets:** PED keys of the same type can be reused or shared among HSMs or partitions, allowing domain sharing (necessary for HA and backup configurations), legacy-style Security Officer authentication, and other custom configurations. See "Shared PED key Secrets" on page 240.
  - **PINs:** optional PINs associated with specific PED keys, set by the owner of the PED key at the time of creation. PINs offer an extra layer of security for PED keys which could be lost or stolen. See "PINs" on page 241.
  - **M of N Split Key Scheme:** optional configuration which allows a role to split its authentication secret across multiple PED keys, and require a minimum number of those keys for authentication. This scheme can be customized to be as simple or complex as your organization's security policy dictates. See "Quorum Split Secrets (M of N)" on page 241.

Comparing Password and Multifactor Quorum Authentication

The following table describes key differences between password- and multifactor quorum-authenticated HSMs.

	Password authentication	Multifactor Quorum authentication
Ability to restrict access to cryptographic keys	> Knowledge of role password is sufficient > For backup/restore, knowledge of partition domain password is sufficient	> Ownership of the black Crypto Officer PED key is mandatory > For backup/restore, ownership of both black CO and red domain PED keys is mandatory > The Crypto User role is available to restrict access to read-only, with no key management authority > Option to associate a PIN with any PED key, imposing a two-factor authentication requirement on any role
Dual Control	> Not available	> Quorum (also called MofN split-knowledge secret sharing) requires "M" different holders of portions of the role secret (a quorum) in order to authenticate to an HSM role - can be applied to any, all, or none of the administrative and management operations required on the HSM

	Password authentication	Multifactor Quorum authentication
Key-custodian responsibility	> Password knowledge only	> Linked to partition password knowledge > Linked to black PED key(s) ownership and optional PIN knowledge
Two-factor authentication for remote access	> Not available	> Remote PED and orange (Remote PED Vector) PED key deliver highly secure remote management of HSM, including remote backup

PED keys

PED keys are USB authentication devices, embedded in a molded plastic body. Each contains a secret, generated by the HSM, that authenticates a role, cloning domain, or remote PED server. This secret is retained until deliberately changed by an authorized user.

The Luna PED does not hold the authentication secrets. They reside only on the portable PED keys.

PED keys are created when an HSM, partition, role, or Remote PED vector is initialized. Each PED key can contain only one authentication secret at a time, but it can be overwritten with a new authentication secret. See "Multifactor Quorum PED key Management" on page 287.

> **CAUTION!** Do not subject PED keys to extremes of temperature, humidity, dust, or vibration. Use the included key cap to protect the USB connector.

PED key Types and Roles

The Luna PED uses PED keys for all credentials. You can apply the appropriate labels included with your PED keys, according to the table below, as you create them.

The PED key colors correspond with the HSM roles described in HSM Roles. The following table describes the keys associated with the various roles:

Lifecycle	PED key	Authentication Secret	Function
HSM Administration	**Blue**	HSM Security Officer (HSM SO) secret	Authenticates the HSM SO role. The HSM SO manages provisioning functions and security policies for the HSM. **Mandatory**
	Red 	HSM Domain or Key Cloning Vector	Cryptographically defines the set of HSMs that can participate in cloning for backup. See "Domain PED keys" on page 241. **Mandatory**
	Orange 	Remote PED Vector	Establishes a connection to a Remote PED server. See * below table. **Optional**
HSM Auditing	**White** 	Auditor (AU) secret	Authenticates the Auditor role, responsible for audit log management. This role has no access to other HSM services. **Optional**
Partition Administration	**Blue**	Partition Security Officer (PO) secret	Authenticates the Partition SO role. The PO manages provisioning activities and security policies for the partition. **NOTE:** If you want the HSM SO to also perform Partition SO duties, you can use the same blue key to initialize both roles. **Mandatory**
	Red 	Partition Domain or Key Cloning Vector	Cryptographically defines the set of partitions that can participate in cloning for backup or high-availability. See "Domain PED keys" on page 241. **Mandatory**

Lifecycle	PED key	Authentication Secret	Function
Partition Operation	**Black** 	Crypto Officer (CO) secret	Authenticates the Crypto Officer role. The CO can perform both cryptographic services and key management functions on keys within the partition. **Mandatory**
	Gray 	Limited Crypto Officer (LCO) secret **	Authenticates the Limited Crypto Officer role. The LCO can perform a subset of the actions available to the Crypto Officer. **Optional (used in eIDAS-compliant schemes)**
	Gray 	Crypto User (CU) secret	Authenticates the Crypto User role. The CU can perform cryptographic services using keys already existing within the partition. It can create and back up public objects only. **NOTE:** If administrative separation is not important, you can use a single black key to initialize the Crypto Officer and Crypto User roles and still have two separate challenge secrets to distinguish read-write and read-only role privileges. **Optional**

* Orange PED keys (RPK) for use with Luna HSM Firmware 7.7.0 or newer, with enhanced security to address modern threat environments and to comply with updated standards, have increased infrastructure onboard the key. If such an initialized RPK is overwritten to become a different role PED key (example SO), this process that formerly would take about six seconds now takes about 36 seconds.

** No use-case is anticipated that requires both the LCO and the CU roles at the same time (Crypto User for Luna use-cases and Limited Crypto Officer for eIDAS use-cases), so the gray Crypto User stickers should be adequate to identify either role as you manage and distribute PED keys.

Shared PED key Secrets

The Luna PED identifies the type of authentication secret on an inserted PED key, and secrets of the same type (color designation) can be used interchangeably. During the key creation process, you have the option of reusing an authentication secret from an existing key rather than have the HSM create a new one. This means that you can use the same PED key(s) to authenticate multiple HSMs or partitions. This is useful for:

- legacy-style authentication schemes, where the HSM SO also functions as the owner of application partitions. This is achieved by using the same blue PED key to initialize the HSM and some or all of the partitions on the HSM.
- allowing a single HSM SO to manage multiple HSMs, or a single Partition SO to manage multiple partitions
- ensuring that HSMs/partitions share a cloning domain (see "Domain PED keys" on the next page)
- allowing a read-write Crypto Officer role and a read-only Crypto User role to be managed by the same user

It is not necessary for partitions in an HA group to share the same blue Partition SO key. Only the red cloning domain key must be identical between HA group members.

> **NOTE** Using a single PED key secret to authenticate multiple roles, HSMs, or partitions is less secure than giving each its own PED key. Refer to your organization's security policy for guidance.

Domain PED keys

A red domain PED key holds the key-cloning vector (the domain identifier) that allows key cloning between HSMs and partitions, and is therefore the PED key most commonly shared between HSMs or partitions. Cloning is a secure method of copying cryptographic objects between HSMs and partitions, required for backup/restore and within HA groups. It ensures that keys copied between HSMs or partitions are:

- strongly encrypted
- copied only between HSMs and partitions that share a cloning domain.

For more information about cloning domains, see "Domain Planning" on page 194.

> **NOTE** An HSM or partition can be a member of only one domain, decided at initialization. A domain can only be changed by re-initializing the HSM. Partition domains may not be changed after initialization.

PINs

The Luna PED allows the holder of a PED key to set a numeric PIN, 4-48 characters long, to be associated with that PED key. This PIN must then be entered on the Luna PED keypad for all future authentication. The PIN provides two-factor authentication and ensures security in case a key is lost or stolen.

PINs can be set only at the time of key creation, and can be changed only by changing the secret on the PED key. Duplicate keys made at the time of creation can have different PINs, allowing multiple people access to the role (see "Creating PED keys" on page 287). Copies made later are true copies with the same PIN, intended as backups for one person (see "Duplicating Existing PED keys" on page 298). Duplicates of the PED key all have the same PIN.

If you are using an M of N configuration, each member of the M of N keyset may set a different PIN.

> **CAUTION!** Forgetting a PIN is equivalent to losing the key entirely; you can no longer authenticate the role, domain, or RPV. See "Consequences of Losing PED keys" on page 295.

Quorum Split Secrets (M of N)

The Luna PED can split an authentication secret among multiple PED key iKeys (up to 16), and require a minimum number of the split keys (a quorum of key-holders) to authenticate the role. This provides a customizable layer of security by requiring multiple trusted people (sometimes called the quorum) to be present for authentication to the role. The key splits are presented to the Luna PED and combined inside the Luna HSM firmware to authenticate the role.

This can be likened to a club, or a board of directors, or a legislature, with some arbitrary number of members. You don't need all members present, to make a decision or perform an action, but you do not want a single person to be able to arbitrarily make decisions or take action affecting everyone. So your security rules set out a number of participants - a quorum - who must be assembled in order to perform certain actions

For example, you could decide (or your security policy could dictate) that at least three trusted people must be present for changes to the HSM policies or for client partition assignments. To accommodate illness, vacations, business travel, or any other reasons that a key-holder might not be present at the HSM site, it is advisable to split the authentication secret among more than three people. If you decide on a five-key split, you would specify M of N for the HSM SO role, or for the cloning domain, or any other role/function, to be 3 of 5. That is, the pool of individual holders of splits of that role secret is five persons, and from among them, a quorum of three must be available to achieve authentication (any three in this 3 of 5 scenario, but cannot be the same key presented more than once during an authentication attempt).

In this example scenario, the HSM SO authentication secret is split among five blue PED key iKeys, and at least three of those keys must be presented to the Luna PED to log in as HSM SO. The PED enforces your quorum rule.

This feature can be used to customize the level of security and oversight for all actions requiring multifactor quorum authentication. You can elect to apply a quorum (M of N split-secret) scheme to all roles and secrets, to some of them, or to none of them. If you do choose to use M of N, you can set different M and N values for each role or secret. Please note the following recommendations:

- M = N is not recommended; if one of the key holders is unavailable, you cannot authenticate the role.
- M = 1 is not recommended; it is no more secure than if there were no splits of the secret - a single person can unlock the role without oversight. If you want multiple people to have access to the role, it is simpler to create multiple copies of the PED key.

> **NOTE** Using an M of N split secret can greatly increase the number of PED keys you require. Ensure that you have enough blank or rewritable PED keys on hand before you begin backing up your M of N scheme.
>
> More keys means that some actions:
>
> - initialization
> - organization-mandated "password" change or roll-over
>
> can require longer to complete and might risk Luna PED timeout. In that case, the options are to increase the PED timeout value in the config file, or become very organized and adept at getting all participants to quickly perform their pin-change tasks.

Activated Partitions and Quorum (M of N)

For security reasons, the HSM and its servers are often kept in a locked facility, and accessed under specific circumstances, directly or by secure remote channel. To accommodate these security requirements, the Crypto Officer and Crypto User roles can be Activated (to use a secondary, alpha-numeric login credential to authenticate - Partition Policy 22), allowing applications to perform cryptographic functions without having to present a black or gray PED key (see "Activation on Multifactor Quorum-Authenticated Partitions" on page 373). In this case, if the HSM is rebooted for maintenance or loses power due to an outage, the cached authentication secret is erased and the role must be reactivated (by logging in the role via LunaCM and presenting the requisite

M number, or quorum, of PED keys) before normal operations can resume. A further measure called Auto-Activation (Partition Policy 23) can cache the authenticated state as long as two hours, allowing automatic, hands-off resumption of operation.

If Auto-Activation is not allowed, or if it is common for your devices to experience outages greater than two hours in duration, you can invoke Remote PED operation and perform PED operations from a location that is distant from the HSM, and possibly more convenient for your authentication secret key holders to convene. See "About Remote PED" on page 252.

Updated Luna PED Behavior Notes

USB-powered and DC-powered Luna PEDs can be updated to Luna PED Firmware 2.9.0 and Luna PED Firmware 2.7.4 respectively.

> Updated Luna PEDs support new communications security protocols for compliance with evolving standards. For more information about these changes, refer to the following sections:

- "Secure Communication Between the Local PED and Luna Network HSM 7s With Firmware 7.7.0 and Newer" on page 252
- "Secure Communication Between the Remote PED and Luna Network HSM 7s With Firmware 7.7.0 and Newer" on page 256

> A Luna HSM at Luna HSM Firmware 7.7.0 or newer requires connection with an updated Luna PED.

- If you are updating to Luna HSM Firmware 7.7.0 or newer, refer to Special Considerations for Luna HSM Firmware 7.7.0 and Newer, Luna PED Firmware 2.9.0, and Luna PED Firmware 2.7.4 *before* proceeding with the update.
- For more information about PED behavior on Luna HSMs that contain V0 or V1 partitions, refer to "Multifactor Quorum Authentication" on page 157.

> PEDs with Luna PED Firmware 2.7.4 or Luna PED Firmware 2.9.0 or newer can also function with the following HSMs:

- HSMs with firmware 5.x and 6.x that will not be updated with the new PED communication protocols.
- HSMs with firmware 7.x that have yet to be updated for compliance with current eIDAS/Common Criteria and NIST standards.

New-series Luna PED Behavior Notes

All of the following points apply to the newer-series PED (firmware versions 2.8.0, 2.8.1, or 2.9.0).

> If a PED is connected via USB to a version 7.x HSM (whether that HSM is installed in a host computer or is embedded in a Luna Network HSM 7 appliance), if the server housing the HSM is booted from a power-off condition, the PED display might come up blank. The PED must be reset.

> If a new-series PED is powered via USB from a 7.x HSM, and the HSM is reset, the PED will become unresponsive. The PED must be reset.

> If a PED is connected via USB to a PED server (for Remote PED), if the server is booted from a power-off condition, the PED display might come up blank OR the PED might be unresponsive to the PED server. The PED must be reset.

> A new-series PED will be unresponsive after a 7.x HMS firmware update or rollback, and/or the display might come up blank. The PED must be reset.

> In environments where the user is switching RPED connections to the same PED between a Luna Network HSM 7 with Luna HSM Firmware 7.7.0 and one with firmware older than 7.7.0, the following error may occur after receiving a prompt to present the orange PED key:
>
> - PED_ERROR when running Luna HSM Client 10.3.0 or newer.
> - DEVICE_ERROR when running Luna HSM Client 10.2.0 or older.
>
> The user may need to clear the RPK from the PED's cache before attempting to switch connection to the non-7.7.0 HSM by pressing the **<** key, or repeat the command after encountering the error.

References to resetting the PED mean cycling the power. This can be done by disconnecting and reconnecting the USB cable.

A new-series PED, powered by a 7.x HSM over USB retains the AC power socket of the older-series model. If an AC power block is plugged into the power socket of the PED, this will reset the PED.

Updating or Rolling Back Multifactor Quorum-Authenticated HSM Firmware

After a version 7.x HSM is updated to Luna HSM Firmware 7.7.0 (or newer), or rolled back to an earlier firmware version, a USB-connected PED should be power cycled. Also restart the PED after any appliance reboot. Without this action, attempted operations against the HSM can result in "device error".

Luna PED Received Items

This chapter describes the items you received with your Luna PED device. For instructions on setting up the PED, see "Multifactor Quorum Authentication" on page 236.

Basic Luna PED Order Items

The following items are included with your Luna PED. All are required for a successful installation.

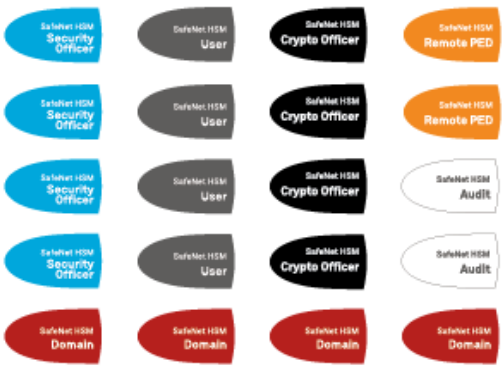
Qty	Item
1	**Luna PED** (with Luna PED Firmware 2.7.1 or newer) 

Qty	Item
1	**Power Supply** kit with replaceable mains plug modules for international use (employed when the PED is operated in Remote PED mode) **NOTE** If your PED has Luna PED Firmware 2.8.0 or newer, it contains refreshed internal hardware and is powered by USB connection. Refreshed PEDs are not shipped with the external power supply, as they do not need it. 
1	**Cable, USB 2.0, Type A to Mini B connectors** (for Remote PED operation). 

Qty	Item
1	**Cable, Data, 9-pin, Micro-D to Micro-D connectors** (for local PED operation prior to HSM firmware versions 7.x.). 

Other Required Multifactor Quorum-Authentication Items

The following required items may be shipped with your Luna PED, or ordered separately.

Qty	Item
1	**Set of PED keys and Labels**   Your order may include a set of PED keys and peel-and-stick labels.

Luna PED Hardware Functions

The Luna PED reads authentication secrets from PED keys on behalf of an HSM or partition. This section contains the following information about the Luna PED device:

> "Physical Features" on the next page

> "Keypad Functions" on page 248

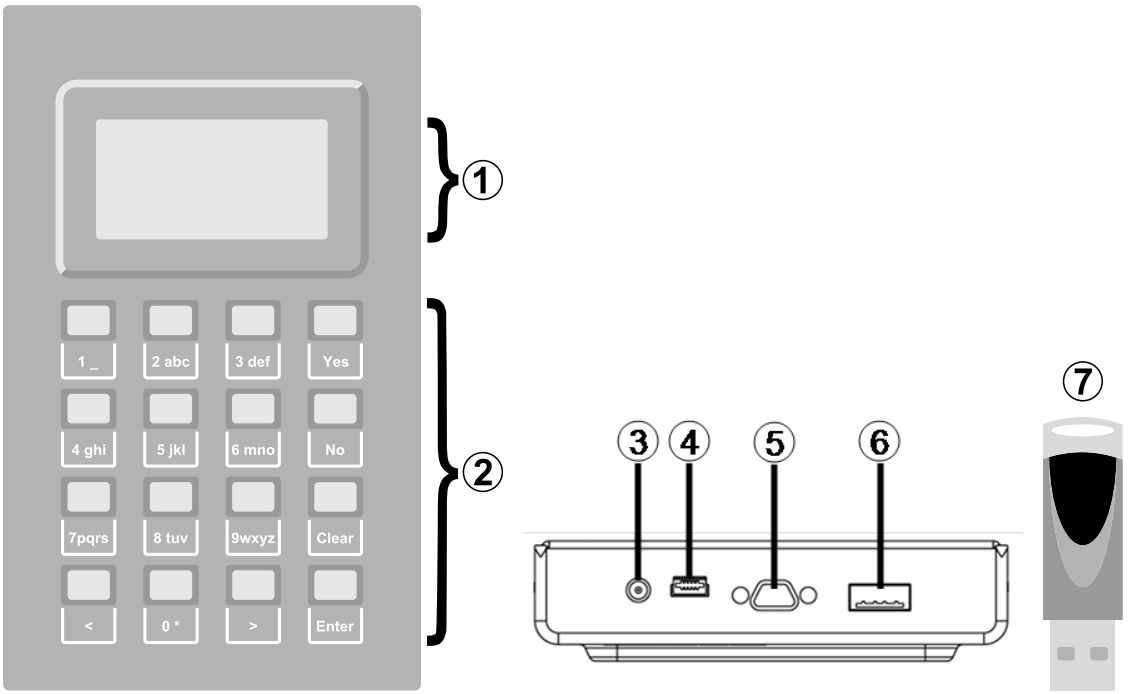
> "Modes of Operation" on page 248

> "Admin Mode Functions" on page 249

> "Luna PED with Newer CPU (External Power Supply Now Optional)" on page 249

Physical Features

The Luna PED is illustrated below, with important features labeled.



1	Liquid Crystal Display (LCD), 8 lines.
2	Keypad for command and data entry. See "Keypad Functions" on the next page.
3	DC power connector. Not used for PED version 2.8 and above. *
4	USB mini-B connector. Used for connecting to the HSM and for file transfer to or from the PED. Luna PED Firmware 2.8.0 and above is powered by this USB connection.
5	Micro-D subminiature (MDSM) connector. Not used for Luna release 7.x.
6	USB A-type connector for PED keys.
7	PED key. Keys are inserted in the PED key connector (item 6).

* Luna PEDs with Luna PED Firmware 2.8.0 and newer are powered by any USB 2.x or 3.x connection, and do not have an external DC power supply. The PED driver must be installed on the connected computer. If the Luna PED is connected to a hub or to a computer without the driver, then the PED display backlight illuminates, but no PED menu is presented.)

Keypad Functions

The Luna PED keypad functions are as follows:

Key	Function
Clear	> Clear the current entry, such as when entering a PIN > Hold the key down for five seconds to reset the PED during an operation. This applies only if the PED is engaged in an operation or is prompting for action. There is no effect when no command has been issued or when a menu is open
<	> **Backspace:** clear the most recent digit you typed on the PED > **Exit:** return to the previous PED menu
>	> **Log:** displays the most recent PED actions (since entering Local or Remote Mode)
Numeric keys	> Select numbered menu items > Input PINs
Yes and No	> Respond to Yes or No questions from the PED
Enter	> Confirm an action or entry

Modes of Operation

The Luna PED can operate in four different modes, depending on the type of HSM connection you want to use:

> **Local PED-SCP:** This mode is reserved for legacy Luna 6.x HSMs that use an MDSM connector between the PED and the HSM. It does not apply to Luna 7.x. Initial HSM configuration must be done in Local PED mode. See "Local PED Setup" on page 250 for instructions.

> **Admin:** This mode is for upgrading the Luna PED device firmware, diagnostic tests, and PED key duplication. See "Admin Mode Functions" on the next page for the functions available in this mode.

> **Remote PED:** In this mode, the PED is connected to a remote workstation and authenticated to the HSM with an orange PED key containing a Remote PED Vector (RPV) secret. This mode allows the Luna Network HSM 7 to be located in a data center or other location restricting physical access. See "About Remote PED" on page 252 for more information.

> **Local PED-USB:** In this mode, the PED is connected directly to the HSM card with a USB mini-B to USB-A connector cable. Initial HSM configuration must be done in Local PED mode.

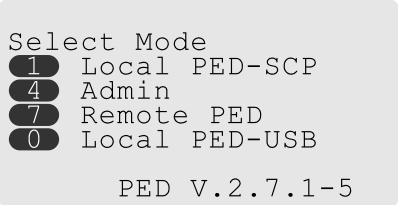
If the Luna PED is connected to an interface when it is powered up, it automatically detects the type of connection being used and switches to the appropriate mode upon receiving the first command from the HSM.

Changing Modes

If you change your PED configuration without disconnecting the PED from power, you must select the correct mode from the main menu.

To change the Luna PED's active mode

1. Press the **<** key to navigate to the main menu.



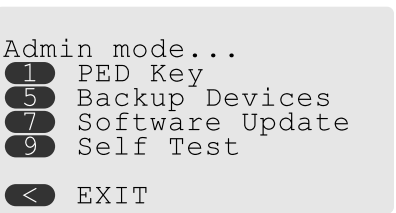
   The main menu displays all the available modes, as well as the PED's current firmware version.

2. Press the corresponding number on the keypad for the desired mode.

> **NOTE** The Luna PED must be in **Local PED-USB** mode when connected to a Luna Network HSM 7 7 card, or LunaSH/LunaCM will return an error (CKR_DEVICE_ERROR) when you attempt authentication.

Admin Mode Functions

In this mode, you can upgrade the Luna PED device software, run diagnostic tests, and duplicate PED keys without having the Luna PED connected to an HSM. Press the corresponding number key to select the desired function.



- **PED Key:** allows you to identify the secret on an inserted PED key, or duplicate the key, without having the Luna PED connected to an HSM.
- **Backup Devices:** Not applicable to Luna 7.x.
- **Software Update:** requires a PED software file and instructions sent from Thales.
- **Self Test:** test the PED's functionality. Follow the on-screen instructions to test button functions, display, cable connections, and the ability to read PED keys. The PED returns a PASS/FAIL report once it concludes the test.

Luna PED with Newer CPU (External Power Supply Now Optional)

A refresh of PED hardware (December 2017) was made necessary by suppliers discontinuing some original components. One of the replaced parts was the CPU, which necessitated a new line of PED firmware, incompatible with the previous versions.

The older PED was shipped with an AC adapter.

The newer PED has the same socket, for connection to an AC adapter, but an adapter/power-block is not shipped with the PED. You can purchase one locally if desired, but the new-CPU PED is reliably powered via USB.

The following points apply to the new-CPU PED - versions 2.8, 2.8.1, 2.9.0 - (that is, any released new CPU PED firmware version)

- when connected over USB to a Luna PCIe HSM 7 or Luna Network HSM 7, if the server housing the HSM card is booted from power off - the PED display might come up blank. The PED must be reset. Reset = power cycle
- when connected via USB to a server (but not directly to the HSM card), if the server is booted from power off - the PED display may come up blank OR unresponsive to PED server; the PED must be reset.
- when powered by the HSM over USB, if an AC power block is then connected, the PED resets.
- when powered by an AC power block, and also plugged into the HSM's USB port ,then if the AC power block is disconnected, the PED will power off.
- the new-CPU PED will be unresponsive after HSM firmware update or rollback, and the display might come up blank; the PED must be reset.
- if the new-CPU PED is powered via the USB connection on the HSM, and the HSM is reset, the PED becomes unresponsive; the PED must be reset.
- if the new-CPU PED is connected to AC and to the HSM's USB connector, if the server housing the HSM is power cycled (not the PED), the PED will not be unresponsive when the server and the HSM are back online; nevertheless, the PED must be reset.

"The PED must be reset" means that the PED must be power cycled by unplugging/replugging the USB cable, or by removing/reinserting the cord from the AC power block (if it is in use).

Local PED Setup

A Local PED connection is the simplest way to set up the Luna PED. In this configuration, the PED is connected directly to the HSM card. It is best suited for situations where all parties who need to authenticate credentials have convenient physical access to the HSM. When the HSM is stored in a secure data center and accessed remotely, you must use a Remote PED setup.

Setting Up a Local PED Connection

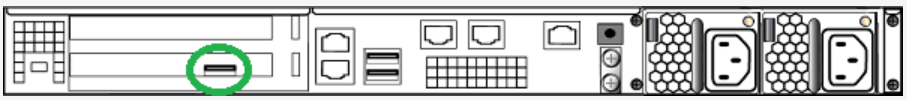
The Luna Network HSM 7 administrator can use these directions to set up a Local PED connection. You require:

- Luna PED with Luna PED Firmware 2.7.1 or newer
- USB mini-B to USB-A connector cable
- Luna PED DC power supply (if included with your Luna PED)

To set up a Local PED connection

1. Connect the Luna PED to the HSM using the supplied USB mini-B to USB-A connector cable.

NOTE To operate in Local PED-USB mode, the Luna PED must be connected directly to the HSM card's USB port, and not one of the other USB connection ports on the appliance.



2. Luna PED Firmware 2.8.0 and newer is powered via the USB connection. If you are using Luna PED Firmware 2.7.1, connect it to power using the Luna PED DC power supply.

   As soon as the PED receives power, it performs start-up and self-test routines. It verifies the connection type and automatically switches to the appropriate operation mode when it receives the first command from the HSM.

3. If you prefer to set the operation mode to **Local PED-USB** manually, see "Changing Modes" on page 248.

The Luna PED is now ready to perform authentication for the HSM. You may proceed with setting up or deploying your Luna Network HSM 7. All commands requiring authentication (HSM/partition initialization, login, etc.) will now prompt the user for action on the locally-connected Luna PED.

PED Actions

There are several things that you can do with the Luna PED at this point:

> Wait for a PED authentication prompt in response to a LunaSH or LunaCM command (see "Performing Multifactor Quorum Authentication" on page 293)

> Create copies of your PED keys (see "Duplicating Existing PED keys" on page 298)

> Change to the Admin Mode to run tests or update PED software (see "Changing Modes" on page 248)

> Prepare to set up a Remote PED server (see "About Remote PED" on the next page)

Secure Local PED

PED firmware can be updated to Luna PED Firmware 2.7.4 or newer on a PED with older CPU, and to Luna PED Firmware 2.9.0 or newer on a PED with new CPU.

> The firmware update is optional for multifactor quorum-authenticated HSMs with firmware versions older than Luna HSM Firmware 7.7.0, and *required* to work with HSMs at Luna HSM Firmware 7.7.0 and newer. This combination complies with an eIDAS-related requirement for an updated secure channel.

> The updated secure channel for Remote PED operation is now also replicated in the local channel, but because it is local it does not need to be mediated via an orange PED key. The Luna PED, however, sees both local and remote connections as equivalent.

NOTE Pressing the "**<**" key on the Luna PED, to change menus, now warns that the RPV will be invalidated, even though the local connection does not use an orange PED Key. Simply ignore the message.

Secure Communication Between the Local PED and Luna Network HSM 7s With Firmware 7.7.0 and Newer

Luna HSM Firmware 7.7.0 introduces a level of protection for data that is exchanged between the Local PED (running Luna PED Firmware 2.7.4 or Luna PED Firmware 2.9.0) and Luna Network HSM 7. All exchanged data is protected in the following way:

- All CSPs exchanged between the Local PED and the Luna Network HSM 7 are protected using an AES-256-KWP CSP wrapping key (CWK).
- The CWK is established using the One-pass Diffie-Hellman key agreement scheme C(1e, 1s ECDH CDH) with unilateral key confirmation, as defined in *NIST Special Publication 800-56A Revision 3*.The key agreement scheme requires the following:
  - The Luna Network HSM 7 uses a static ECDH key pair. In this case, the HSM generates its own static P-521 ECDH key on startup and the key is assigned a certificate which chains back to the HSM's ECC HOC.
  - The Local PED uses an ephemeral ECDH key pair. In this case, the Local PED generates its ephemeral P-521 ECDH key pair during the key agreement.
- The SHA-512 based Single-step key derivation function defined in *NIST Special Publication 800-56C Revision 1* is used to derive the CWKs from the shared secret. The derivation function derives separate CWKs for HSM-to-Local PED and Local PED-to-HSM communication.

About Remote PED

A Remote PED connection allows you to access multifactor quorum-authenticated HSMs that are kept in a secure data center or other remote location where physical access is restricted or inconvenient. This section provides descriptions of the following aspects of Remote PED connections:

- "Remote PED Architecture" on the next page
- "Remote PED Connections" on page 254
- "Secure Communication Between the Remote PED and Luna Network HSM 7s With Firmware 7.7.0 and Newer" on page 256
- "Initializing the Remote PED Vector and Creating an Orange Remote PED key" on page 259
- "Installing PEDserver and Setting Up the Remote Luna PED" on page 263
- "Opening a Remote PED Connection" on page 265
- "Ending or Switching the Remote PED Connection" on page 274
- Configuring PED Timeout Settings
- "Remote PED Troubleshooting" on page 275

> **NOTE** From Luna Network HSM 7 appliance software version 7.7.0 through 7.9.0, remote PED operation is limited to a 600-second (10 minute) timeout on PED operations, which might affect use-cases with a large number of M-of-N splits. This is not adjustable. The workaround is to use a pre-7.7.0 or post-7.9.0 software version.

Remote PED Architecture

The Remote PED architecture consists of the following components:

> **Remote PED:** a Luna PED with Luna PED Firmware 2.7.1, connected to a network-connected workstation, powered on, and set to Remote PED mode.

> **NOTE** For the enhanced connection security and NIST SP 800-131A Rev.1 compliance implemented with Luna HSM Firmware 7.7.0 and newer, the following Luna PED firmware versions are required:
>
> > Luna PED Firmware 2.7.4 for PEDs that require the external power block
>
> > Luna PED Firmware 2.9.0 for USB-powered PEDs

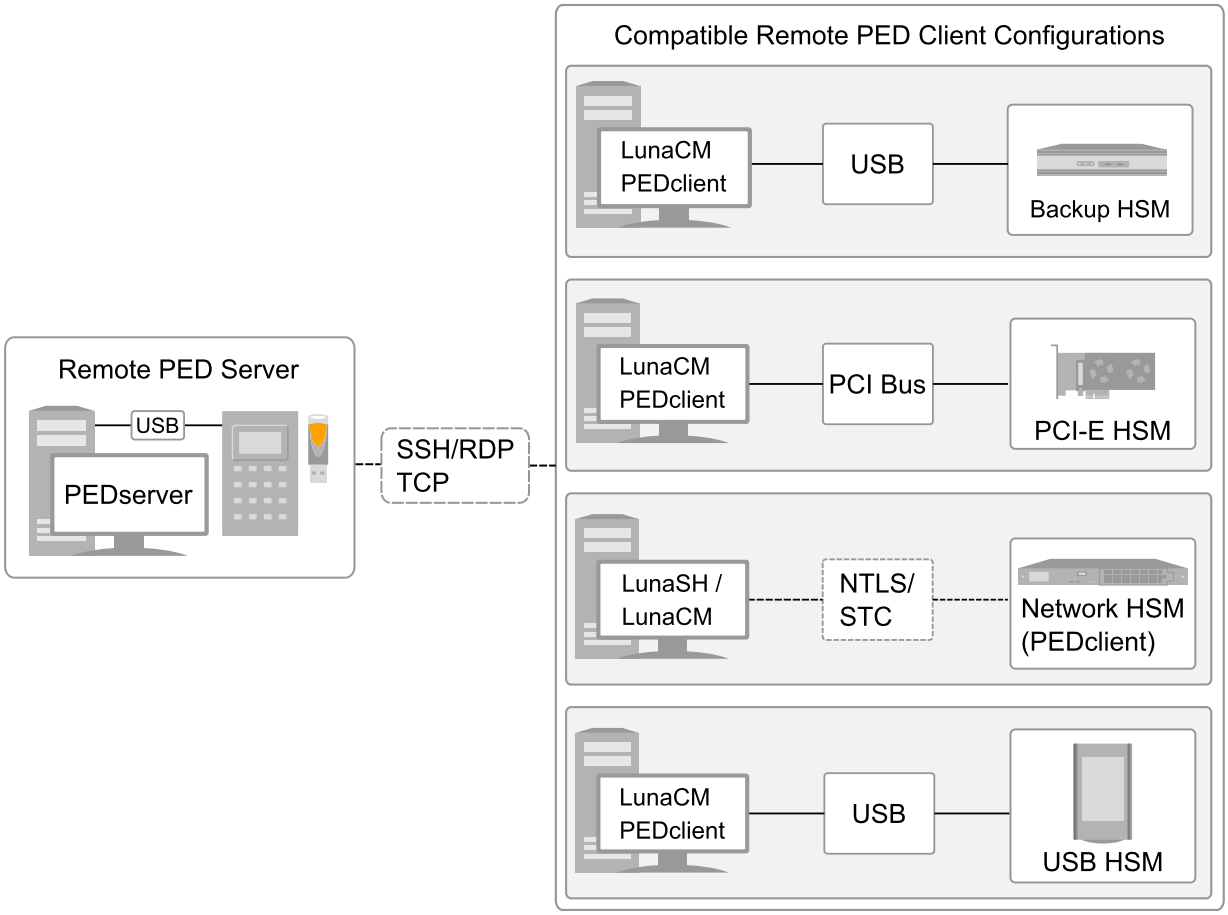
> **Remote PED Vector (RPV):** a randomly generated, encrypted value used to authenticate between a Remote PED (via PEDserver) and a Luna HSM (via PEDclient).

> **Remote PED Key (RPK):** an orange PED key containing an RPV (or multiple PED keys with a split RPV in an M of N quorum implementation).

> **PEDserver:** software that runs on the remote workstation with a USB-connected Luna PED. PEDserver accepts requests from and serves PED actions and data to PEDclient.

> **PEDclient:** software that requests remote PED services from PEDserver. PEDclient runs on the network-connected system hosting the HSM, which can be one of the following:

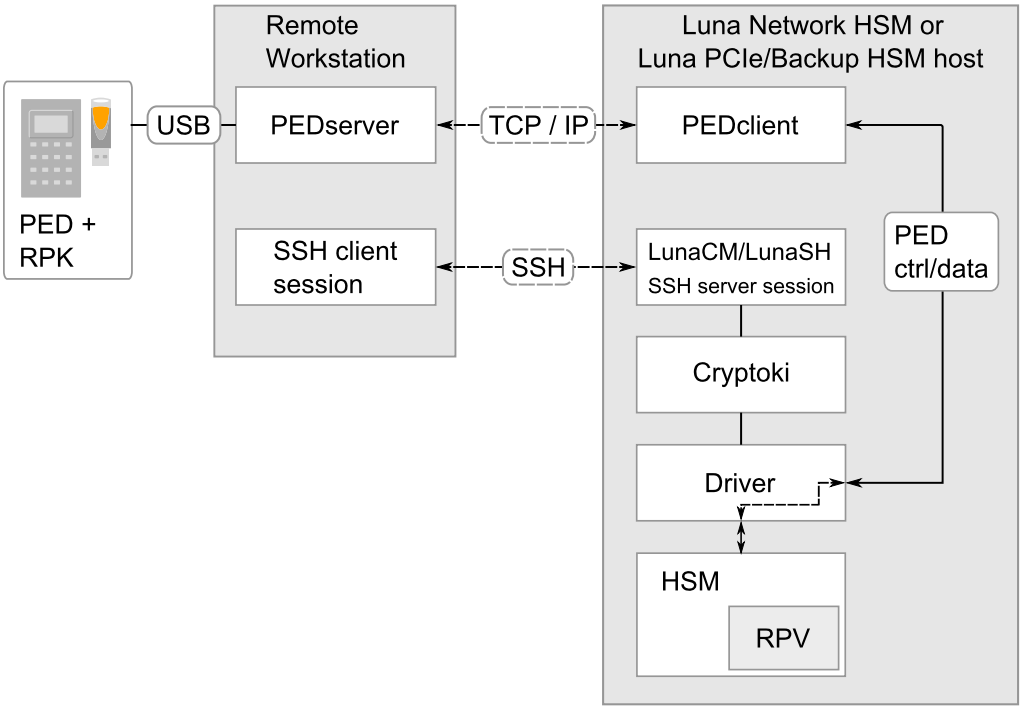
- Host computer with USB-connected Luna Backup HSM, configured for remote backup
- Host computer with Luna PCIe HSM 7 installed
- Luna Network HSM 7
- Host computer with Luna USB HSM 7 connected



Remote PED Connections

A Luna Network HSM 7 can establish a Remote PED connection with any workstation that meets the following criteria:

- PEDServer is running
- a Luna PED with Luna PED Firmware 2.7.1 or newer is connected
- The orange PED key containing the Remote PED Vector (RPV) for that HSM is available



Bi-directionality

There are two methods of establishing a Remote PED connection to the HSM:

- **HSM-initiated:** When the HSM requires authentication, it sends (via PEDclient) a request for PED services to the Remote PED host (which receives the request via PEDserver). This requires that the Luna Network HSM 7 be allowed to initiate external connections, and that the PEDserver IP port remains open. If the Luna Network HSM 7 resides behind a firewall with rules prohibiting these connections, or if your IT policy prohibits opening a port on the Remote PED host, use a PED-initiated connection. See "HSM-Initiated Remote PED" on page 265.
- **PED-initiated:** The HSM and Remote PED host exchange and register certificates, creating a trusted connection. This allows the Remote PED host (via PEDserver) to initiate the connection to the Luna Network HSM 7. If you have firewall or other constraints that prevent your HSM from initiating a connection to a Remote PED in the external network, use this connection method. See "PED-Initiated Remote PED" on page 270.

The following constraints apply to PED-initiated connections:

- A maximum of 20 Remote PED servers can be registered in PEDclient.
- A maximum of 80 Luna Network HSM 7 appliances can be registered in PEDserver.
- If the connection is terminated abnormally (for example, a router switch died), there is no auto-reconnection. PEDserver automatically restarts and runs in HSM-initiated connection mode.
- When running in PED-initiated connection mode, PEDserver does not listen for new HSM-initiated connections, for security and to simplify usability.

Priority and Lockout

If a Local PED connection is active and an operation is in progress, a Remote PED connection cannot be initiated until the active Local PED operation is completed. If the Local PED operation takes too long, the Remote PED command may time out.

When a Remote PED connection is active, the Local PED connection is ignored, and all authentication requests are routed to the Remote PED. Attempts to connect to a different Remote PED server are refused until the current connection times out or is deliberately ended. See "Ending or Switching the Remote PED Connection" on page 274.

One Connection at a Time

Remote PED can provide PED services to only one HSM at a time. To provide PED service to another HSM, you must first end the original Remote PED connection. See "Ending or Switching the Remote PED Connection" on page 274.

Timeout

Remote PED connections have configurable timeout settings. For more information, refer to Configuring PED Timeout Settings.

Once a partition has been activated and cached the primary authentication (PED key) credential, the Crypto Officer or Crypto User can log in using only the secondary (alphanumeric) credentials and the Remote PED connection can be safely ended until the Partition SO needs to log in again.

Broken Connections

A Remote PED connection is broken if any of the following events occur:

- The connection is deliberately ended by the user
- The connection times out (default: 1800 seconds)
- Luna PED is physically disconnected from its host
- VPN or network connection is disrupted
- You exit Remote PED mode on the Luna PED. If you attempt to change menus, the PED warns:

```
   ** WARNING **
Exiting now will
invalidate the RPK.
Confirm? YES/NO
```

If the link is broken, as long as the network connection is intact (or is resumed), you can restart PEDserver on the Remote PED host and run **hsm ped connect** in LunaSH or **ped connect** in LunaCM to re-establish the Remote PED link. In a stable network situation, the link will remain available until timeout.

Secure Communication Between the Remote PED and Luna Network HSM 7s With Firmware 7.7.0 and Newer

Communication between the Remote PED and Luna Network HSM 7 is kept secure before and after RPV initialization.

Secure Communication Before RPV Initialization

Before the RPV is initialized, data exchanged between the Remote PED and Luna Network HSM 7 is protected using the same method that is used to secure communication between a Local PED and Luna Network HSM 7 (see "Secure Communication Between the Local PED and Luna Network HSM 7s With Firmware 7.7.0 and Newer" on page 252). However, the SHA-512 based single-step key derivation function defined in *NIST Special Publication 800-56C Revision 1* is used to derive the CWKs from the shared secret, with a password used instead of an RPV.

After the RPV is initialized, the Luna Network HSM 7 and Remote PED re-establish a full secure channel. For more information, see the section below.

Secure Communication After RPV Initialization

After the RPV is initialized, all communication between the Remote PED and the Luna Network HSM 7 is transmitted within a secure channel that is protected using an AES-256-CTR data encryption key (DEK) and a SHA-512-HMAC data MAC key (DMK). CSPs transmitted within the secure channel are additionally encrypted using an AES-256-KWP CSP wrapping key (CWK). The secure channel is established using the Full Unified C (2e, 2s ECDH CDH) key agreement scheme with bilateral key confirmation, as defined in *NIST Special Publication 800-56A Revision 3*. The key agreement scheme requires each party to use a static ECDH key pair and an ephemeral ECDH key pair. The key pairs are generated in the following ways:

> The HSM generates its own static P-521 ECDH key on startup. During RPV initialization, the HSM generates a static P-521 ECDH key and loads it onto the RPK along with the RPV (or RPV split is MofN is used). Both static keys are assigned certificates which chains back to the HSM's ECC HOC.

> During the key agreement, the Luna Network HSM 7 and Remote PED both generate their ephemeral P-521 ECDH key pair.

As part of the key agreement, the SHA-512 based single-step key derivation function defined in *NIST Special Publication 800-56C Revision 1* is used to combine the shared secret, the RPV, and the derived the secure channel protection keys; that is, the DEK, DMK, and CWK. The derivation function derives separate keys (DEK, DMK, and CWK) for HSM-to-Remote PED and Remote PED-to-HSM communication.

The RPV ensures mutual authentication of both end points and the HSM's static ECDH key ensures additional authentication of the HSM.

Secure Communication Between the Remote PED and Luna Network HSM 7s With Firmware 7.4.2 and Older

All communication between the Remote PED and the HSM is transmitted within an AES-256 encrypted channel, using session keys based on secrets shared out-of-band. This is considered a very secure query/response mechanism. The authentication conversation is between the HSM and the PED. Authentication data retrieved from the PED keys never exists unencrypted outside of the PED or the HSM. PEDclient and PEDserver provide the communication pathway between the PED and the HSM, and the data remains encrypted along that path.

Once the PED and HSM are communicating, they establish a common Data Encryption Key (DEK). DEK establishment is based on the Diffie-Hellman key establishment algorithm and a Remote PED Vector (RPV), shared between the HSM and the PED via the orange Remote PED Key (RPK). Once a common Diffie-Hellman value is established between the parties via the Diffie-Hellman handshake, the RPV is mixed into the value to create a 256-bit AES DEK on each side. If the PED and the HSM do not hold the same RPV, the resulting DEKs are different and communication is blocked.

Mutual authentication is achieved by exchanging random nonces, encrypted using the derived data encryption key. The authentication scheme operates as follows:

HSM	_	Remote PED
Send 8 bytes random nonce, R1, encrypted using the derived encryption key.	$\{R1 \mid\mid padding\}_{Ke}$ ->	
	<- $\{R2 \mid\mid R1\}_{Ke}$	Decrypt R1. Generate an 8 byte random nonce, R2. Concatenate R2 || R1 and encrypt the result using the derived encryption key.
Decrypt R2 || R1. Verify that received R1 value is the same as the originally generated value. Re-encrypt R2 and return it to Remote PED.	$\{padding \mid\mid R2\}_{Ke}$ ->	Verify that received R2 value is the same as the originally generated value.

Following successful authentication, the random nonce values are used to initialize the feedback buffers needed to support AES-OFB mode encryption of the two communications streams (one in each direction).

Sensitive data in transition between a Luna PED and an HSM is end-to-end encrypted: plaintext security-relevant data is never exposed beyond the HSM and the PED boundaries at any time. The sensitive data is also hashed, using a SHA-256 digest, to protect its integrity during transmission.

PEDServer Configuration File

PED-initiated Remote PED introduces a pedServer.ini/pedServer.conf file. The **Appliances** section manages registered appliances.

> **CAUTION!** Do not edit the pedServer.ini/pedServer.conf file. If you have any issues, contact Thales Technical Support.

```
[Appliances]
ServerCAFile=C:\Program Files\SafeNet\LunaClient\cert\PedServerCAFile.pem
SSLConfigFile=C:\Program Files\SafeNet\LunaClient\openssl.cnf
ServerName00=myHSM
ServerIP00=192.20.11.78
ServerPort00=9697
CommonCertName00=66331
[RemotePed]
AdminPort=1502
BGProcessShutdownTimeoutSeconds=25
BGProcessStartupTimeoutSeconds=10
ExternalAdminIF=0
ExternalServerIF=1
IdleConnectionTimeoutSeconds=1800
InternalShutdownTimeoutSeconds=10
LogFileError=1
LogFileInfo=1
LogFileName=C:\Program Files\SafeNet\LunaClient\remotePedServerLog.log
LogFileTrace=0
LogFileWarning=1
```



```
MaxLogFileSize=4194304
PingInterval=1
PongTimeout=5

RpkSerialNumberQueryTimeout=15
ServerPortValue=1503
SocketReadRspTimeoutSeconds=60
SocketReadTimeoutSeconds=60
SocketWriteTimeoutSeconds=15
```

A new entry in the main Crystoki.ini/Chrystoki.conf file points to the location of the pedServer.ini/pedServer.conf file.

```
[Ped Server]
PedConfigFile = /usr/safenet/lunaclient/data/ped/config
```

Initializing the Remote PED Vector and Creating an Orange Remote PED key

The Remote PED (via PEDserver) authenticates itself to the Luna Network HSM 7 with a randomly-generated encrypted value stored on an orange PED key. That secret originates in an HSM, and can be carried to other HSMs via the orange key. An newly-configured HSM either:

- generates its own RPV secret to imprint on an orange PED key,
- accepts a pre-existing RPV from a previously imprinted orange PED key, at your discretion.

The orange key proves to the HSM that the Remote PED is authorized to provide authentication for HSM roles. A Luna Network HSM 7 administrator can create this key using one of the following two methods:

- "Local RPV Initialization" below**:** The RPV is initialized using a Luna PED connected to the USB port on the HSM card. This is the standard method of initializing the RPV.
- "Remote RPV Initialization" on the next page**:** The RPV is initialized using a Luna PED connected to a remote workstation running PEDserver. A one-time numeric password is used to authenticate the Remote PED to the HSM before initializing the RPV. This optional method is useful if the HSM SO has only remote SSH access to the appliance. It is available only if the HSM is in a zeroized state (uninitialized) and your firewall settings allow an HSM-initiated Remote PED connection. If you choose this method, you will set up Remote PED before initializing the RPV.

> **NOTE** Generally, the HSM SO creates an orange PED key (and backups), makes a copy for each valid Remote PED server, and distributes them to the Remote PED administrators.

See also "Rotating or Re-Initializing the Orange Remote PED key" on page 262.

Local RPV Initialization

If the HSM is already initialized, the HSM SO must log in to complete this procedure. You require:

- Luna PED with Luna PED Firmware 2.7.1 or newer
- USB mini-B to USB-A connector cable
- Luna PED DC power supply (if included with your Luna PED)

> Blank or reusable orange PED key (or multiple keys, if you plan to make extra copies or use an M of N security scheme). See "Creating PED keys" on page 287 for more information.

> **NOTE** Orange PED keys (RPK) for use with Luna HSM Firmware 7.7.0 or newer, with enhanced security to address modern threat environments and to comply with updated standards, have increased infrastructure onboard the key. If such an initialized RPK is overwritten to become a different role PED key (example SO), this process that formerly would take about six seconds now takes about 36 seconds.

To initialize the RPV and create the orange PED key locally

1. If you have not already done so, set up a Local PED connection (see "Local PED Setup" on page 250).
2. Using a serial or SSH connection, log in to the Luna Network HSM 7 appliance as **admin**.
3. If the HSM is initialized, log in as HSM SO (see Logging In as HSM Security Officer). If not, skip to the next step.

   lunash:> **hsm login**
4. Ensure that you have the orange PED key(s) ready. Initialize the RPV.

   lunash:> **hsm ped vector init**
5. Attend to the Luna PED and respond to the on-screen prompts. See "Creating PED keys" on page 287 for a full description of the key-creation process.

   ```
   SLOT
   SETTING RPV...
   Would you like to
   reuse an existing
   keyset?(Y/N)
   ```

   - If you have an orange PED key with an existing RPV that you wish to use for this HSM, press **Yes**.
   - If you are creating a new RPV, press **No**.

   ```
   SLOT
   SETTING RPV...
   Insert a
   RPK / Remote
   PED Key (ORANGE)
   Press ENTER.
   ```

   Continue following the prompts for PIN, M of N, and duplication options.

To continue setting up a Remote PED server, see "Installing PEDserver and Setting Up the Remote Luna PED" on page 263.

Remote RPV Initialization

When you initialize an RPV with the PED connected locally, you have direct physical control of the operation and its security.

When you initialize an RPV remotely, you must secure the link and the operation with a one-time password. The HSM must be *uninitialized* for this operation.

> **NOTE** This feature requires minimum Luna Network HSM 7 Appliance Software 7.2.0 and Luna HSM Client 7.2.0.
>
> The maximum timeout for all operations when a PED-invoking HSM command is launched is 600 seconds.
>
> To meet evolving FIPS security requirements the protocol to communicate with the PED Key has been enhanced with firmware 7.7.0 and above thus causing additional delay during the initialization of those PED keys. For larger quantities of MofN splits (the maximum is 16) the 600 seconds is not sufficient to complete a Remote PED key initialization.

Use the following procedure to initialize the RPV. You require:

> A blank or reusable orange PED key (or multiple keys, if you plan to make extra copies or use an M of N security scheme). See "Creating PED keys" on page 287 for more information.

> The HSM must be in a zeroized state and the RPV uninitialized.

To initialize the RPV and create the orange key remotely

1. Open an HSM-initiated Remote PED connection. Using Luna HSM Firmware 7.7.0 or newer, the **-password** option is mandatory; you can include an 8-digit numeric PIN, or specify **-password** alone to have one randomly generated.

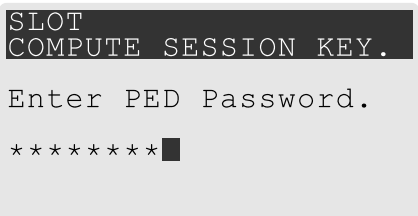
   lunash:> **hsm ped connect -ip** <PEDserver_IP> **-password** <optional_PIN>

   The Remote PED connection command prepares to secure the connection and LunaSH does one of the following:

   - If you are using Luna HSM Firmware 7.7.0 or newer and Luna HSM Client 10.3.0 or newer, and did not specify a PIN in the command line, LunaSH presents a randomly-generated 8-digit numeric one-time password that the HSM will use to identify the Remote PED server.

     ```
     Please attend to the PED and enter following password: 18246843

     Command Result : No Error
     ```

     The remote Luna PED prompts you for the one-time password:

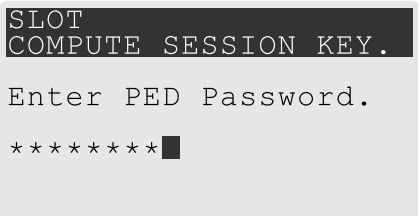
     

   - If you are using Luna HSM Firmware 7.4.2 or older and Luna HSM Client 10.2.0 or older, LunaSH returns the following message:

     ```
     Luna PED operation required to connect to Remote PED - use orange PED key(s).


     Enter PED Password:
     ```

In LunaSH, when prompted to "Enter PED Password" set any 8-digit numeric one-time password that the HSM will use to identify the Remote PED server. The following message is displayed in LunaSH, and the Luna PED prompts you for the password:

```
Luna PED operation required to connect to remote PED - Enter PED password.
```



2. Enter the numeric password on the PIN pad, exactly as you entered it/it was displayed in LunaSH, and press **Enter**.
3. Ensure that you have the orange PED key(s) ready. Initialize the RPV.

   lunash:> **hsm ped vector init**
4. Attend to the Luna PED and respond to the on-screen prompts. See "Creating PED keys" on page 287 for a full description of the key-creation process.

   When the initialization is complete, the HSM launches PEDclient and establishes a Remote PED connection using the newly-created RPV.

```
Ped Client Version 2.0.1 (20001)
Ped Client launched in "Release ID" mode.
Callback Server is running..
ReleaseID command passed.
"Release ID" command passed.
Ped Client Version 2.0.1 (20001)
Ped Client launched in "Delete ID" mode.
Callback Server is running..
DeleteID command passed.
"Delete ID" command passed.

Command Result : 0 (Success)
```

   You may now initialize the HSM. See Initializing the HSM for more information.

> **NOTE** After creating the orange (Remote PED Vector) key for an HSM using the single-session, one-time password-authenticated PED connection that is used to create the key, the Luna PED prompts for the one-time password when you end the session using **ped disconnect**.
>
> This prompt can be safely ignored. The PED session is disconnected properly by pressing the Enter key on the Luna PED, without entering the password.

Rotating or Re-Initializing the Orange Remote PED key

You can rotate the RPV at any time, using either a local or remote Luna PED. This might be necessary if an orange PED key is lost, or as part of scheduled security measures. If the original orange PED key is lost, or you do not have enough M of N splits to reach a quorum, you must use a local PED. You require:

> [Remote PED only] The original orange PED key or enough M of N splits for a quorum.

> A blank or reusable orange PED key (or multiple keys, if you plan to make extra copies or use an M of N security scheme). See "Creating PED keys" on page 287 for more information.

To rotate or re-initialize the orange remote PED key

1. If you have not already done so, set up a Local or Remote PED connection (see "Local PED Setup" on page 250 or "Opening a Remote PED Connection" on page 265).
2. Using a serial or SSH connection, log in to the Luna Network HSM 7 appliance as **admin**.
3. [Remote PED only] Open a Remote PED connection using the original orange PED key(s).

   lunash:> **hsm ped connect -ip** <PEDserver_IP>
4. Log in as HSM SO (see Logging In as HSM Security Officer).

   lunash:> **hsm login**
5. Ensure that you have the blank orange PED key(s) ready. Initialize the RPV.

   CAUTION! Do not overwrite your original orange PED key(s) unless you have a backup copy. The RPV is not rotated until the entire operation is complete. If you encounter network connectivity or PED timeout issues, particularly when presenting multiple M of N splits, you might not have enough splits of the original RPV left for quorum. In this case, you must re-initialize the orange RPK using a Local PED connection.

   lunash:> **hsm ped vector init**
6. Attend to the Luna PED and respond to the on-screen prompts. See "Creating PED keys" on page 287 for a full description of the key-creation process.

Installing PEDserver and Setting Up the Remote Luna PED

The PEDserver software, installed on the Remote PED host workstation, allows the USB-connected Luna PED to communicate with remotely-located HSMs. The Remote PED administrator can install PEDserver using the Luna HSM Client installer. You require:

> Network-connected workstation with compatible operating system (refer to the release notes)
> Luna HSM Client installer
> Luna PED with Luna PED Firmware 2.7.1 or newer
> USB mini-B to USB-A connector cable
> Luna PED DC power supply if required by your Luna PED hardware

NOTE To set up a Remote PED Server on Linux, you require Luna HSM Client 10.1.0 or newer.

To install PEDserver and the PED driver, and set up the Luna PED

1. Run the Luna HSM Client installer and follow the on-screen instructions, as detailed in "Luna HSM Client Software Installation" on page 20, and select the **Luna Remote PED** option. Any additional installation choices are optional, for the purpose of this procedure.

2. On Windows, if you are prompted to install the driver, accept the installation.
3. On Windows, reboot the computer to ensure that the Luna PED driver is accepted by Windows. This step is not required for Linux or Windows Server operating systems.
4. Connect the Luna PED to a USB port on the host system using the supplied USB mini-B to USB-A connector cable.

   Luna PED with Luna PED Firmware 2.8.0 and above is powered via the USB connection. If you are using a Luna PED with Luna PED Firmware 2.7.1, connect it to power using the Luna PED DC power supply.

   As soon as the PED receives power, it performs start-up and self-test routines (for PED v2.8 and later, the PED driver must be installed on the connected computer, or the display remains blank). It verifies the connection type and automatically switches to the appropriate operation mode when it receives the first command from the HSM.

   To manually set the operation mode to **Remote PED**, see "Changing Modes" on page 248.
5. On Windows, open the Windows **Device Manager** to confirm that the Luna PED is recognized as **PED2**. If it appears as an unrecognized USB device:
   a. Disconnect the Luna PED from the host USB port.
   b. Reboot the computer to ensure that the Luna PED driver is accepted by Windows.
   c. Reconnect the Luna PED.

To continue setting up a Remote PED connection, see "Opening a Remote PED Connection" on the next page.

PED Utilities Run by Non-root Users

The default location of the PED utility log is the current directory where the PED utility command has executed, like ./remotePedServerLog.log. Non-root users, even members of the hsmusers group, do not have write permission to the bin directory, or any directory in /usr/safenet, so the PED utility PedServer or PedClient started by a non-root user fails to start.

PED Server

Without root access (or workaround... see below), the utility fails to launch, displaying the following error message:

```
[bin]$ ./PedServer -m start
Ped Server Version 1.0.6 (10006)
Ped Server launched in startup mode.
Connecting to PED. Please wait.........InternalRead: 10 seconds timeout
Failed to recv query response command: RC_OPERATION_TIMED_OUT c0000303
Failed to connect to PED. Please see logs for further details.
Ped Server Process created, exiting this process.
-------------------------------------------------------------------------------
```

The service needs to log all its actions, including the action of making a connection to the PED, so after failing to create the log (no write permission), it aborts the action of connecting to the PED.

The **workaround** is to set the PED server LogFileName to a location where the current user has read and write access, such as the user's home directory.

Examples:

```
$ ./PedServer -mode config -set LogFileName  $HOME/remotePedServerLog.log
```

OR

```
$./PedServer -mode config -set LogFileName  /tmp/remotePedServerLog.log
```

Then run $./PedServer -mode start

OR

start the PedServer with log file option: -logfilename /dev/null

```
$bin/PedServer -m start -logfilename /dev/null
$bin/PedServer -m start -logfilename  $HOME/remotePedServerLog.log
```

PEDClient
PedClient has some similar requirements.

Have the user in an appropriate user group, and they can then launch with systemctl

Opening a Remote PED Connection

There are two methods of establishing a Remote PED connection to the HSM:

- **HSM-initiated:** When the HSM requires authentication, it sends (via PEDclient) a request for PED services to the Remote PED host (which receives the request via PEDserver). This requires that the Luna Network HSM 7 be allowed to initiate external connections, and that the PEDserver IP port remains open. If the Luna Network HSM 7 resides behind a firewall with rules prohibiting these connections, or if your IT policy prohibits opening a port on the Remote PED host, use a PED-initiated connection instead.

  See "HSM-Initiated Remote PED" below.

- **PED-initiated:** The HSM and Remote PED host exchange and register certificates, creating a trusted connection. This allows the Remote PED host (via PEDserver) to initiate the connection to the Luna Network HSM 7. If you have firewall or other constraints that prevent your HSM from initiating a connection to a Remote PED in the external network, use this connection method.

  See "PED-Initiated Remote PED" on page 270.

> **NOTE** For the Luna Network HSM 7, only Luna Shell commands can be used with a *PED-initiated Remote PED connection*. Client-side LunaCM commands such as **partition init** cannot be executed. This means that only administrative personnel, logging in via Luna Shell (lunash:>) can authenticate to the HSM using a PED-initiated Remote PED connection.
>
> To perform actions requiring authentication on Luna Network HSM 7 partitions (that is, from the client side) any Remote PED connection must be launched by the HSM, and the data-center firewall rules must permit such outward initiation of contact.

HSM-Initiated Remote PED

The HSM/client administrator can use this procedure to establish an HSM-initiated Remote PED connection. The procedure is different depending on whether you are setting up Remote PED for the Luna Network HSM 7 appliance or a client. You require:

- Administrative access to a network-connected workstation with PEDserver installed and Luna PED connected (see "Installing PEDserver and Setting Up the Remote Luna PED" on page 263)
- Administrative access to the Luna Network HSM 7 via SSH (if using Remote PED for HSM-level authentication)

> Administrative access to a Luna HSM Client workstation with an assigned user partition (if using Remote PED for partition-level authentication)

> One of the following:

- Orange PED key with the HSM's RPV (see "Initializing the Remote PED Vector and Creating an Orange Remote PED key" on page 259)
- Blank orange PED key (or multiple keys, if you plan to use an M of N scheme)

If you encounter issues, see "Remote PED Troubleshooting" on page 275.

To launch PEDserver

1. On Windows, open an Administrator command prompt by right-clicking the Command Prompt icon and selecting **Run as administrator**. This step is not necessary if you are running Windows Server 20xx, as the Administrator prompt is launched by default.
2. Navigate to the Luna HSM Client install directory.

   Windows default: **cd C:\Program Files\SafeNet\LunaClient**

   Linux/UNIX default: **cd /usr/safenet/lunaclient**
3. Launch PEDserver. If you are launching PEDserver on an IPv6 network, you must include the **-ip** option.

   > **"pedserver -mode start" on page 316** [**-ip** <PEDserver_IP>]

```
C:\Program Files\SafeNet\LunaClient>pedserver mode start
Ped Server Version 1.0.6 (10006)
Ped Server launched in startup mode.
Starting background process
Background process started
Ped Server Process created, exiting this process.
```

4. Verify that the service has launched successfully.

   > **"pedserver -mode show" on page 314**

   Note the **Ped2 Connection Status**. If it says **Connected**, PEDserver is able to communicate with the Luna PED.

   Note also the server port number (default: **1503**). You must specify this port along with the PEDserver host IP when you open a connection.

```
c:\Program Files\SafeNet\LunaClient>pedserver mode show
Ped Server Version 1.0.6 (10006)
Ped Server launched in status mode.


   Server Information:
      Hostname:                         DWG9999
      IP:                               0.0.0.0
      Firmware Version:                 2.7.1-5
      PedII Protocol Version:           1.0.1-0
      Software Version:                 1.0.6 (10006)

      Ped2 Connection Status:           Connected
      Ped2 RPK Count                    0
      Ped2 RPK Serial Numbers           (none)

   Client Information:                  Not Available
```

```
Operating Information:
   Server Port:                         1503
   External Server Interface:           Yes
   Admin Port:                          1502
   External Admin Interface:            No

   Server Up Time:                      190 (secs)
   Server Idle Time:                    0 (secs) (0%)
   Idle Timeout Value:                  1800 (secs)

   Current Connection Time:             0 (secs)
   Current Connection Idle Time:        0 (secs)
   Current Connection Total Idle Time: 0 (secs) (100%)
   Total Connection Time:               0 (secs)
   Total Connection Idle Time:          0 (secs) (100%)

Show command passed.
```

5. Use **ipconfig** (Windows) or **ifconfig** (Linux) to determine the PEDserver host IP. A static IP is recommended, but if you are connecting over a VPN, you may need to determine the current IP each time you connect to the VPN server.

If you are setting up Remote PED with a Luna Network HSM 7 appliance, see "To open a Remote PED connection from the Luna Network HSM 7 appliance" below.

If you are setting up Remote PED with a client, see "To open a Remote PED connection from a client workstation" on the next page.

To open a Remote PED connection from the Luna Network HSM 7 appliance

1. Open an SSH session to the Luna Network HSM 7 and log in to LunaSH as **admin**.
2. Initiate the Remote PED connection from the Luna Network HSM 7.

   lunash:> **hsm ped connect -ip** <PEDserver_IP> **-port** <PEDserver_port> [**-serial** <serial#>]

   > **NOTE** The **-serial** option is required only if you are using Remote PED to authenticate a Luna Backup HSM connected to one of the Luna Network HSM 7's USB ports. If a serial number is not specified, the appliance's internal HSM is used.

```
lunash:>hsm ped connect -ip 192.124.106.100 -port 1503

Luna PED operation required to connect to Remote PED - use orange PED key(s).
```

- If you have not yet initialized the RPV, and the HSM is not in initialized state, LunaSH prompts you to enter a password.

  ```
  Enter PED Password:
  ```

  See "Remote RPV Initialization" on page 260 for this procedure.
- If you already initialized the RPV, the Luna PED prompts for the orange PED key.

```
SLOT
COMPUTE SESSION KEY.
Insert a
RPK / Remote
PED Key (ORANGE)
Press ENTER.
```

Present the orange PED key with the correct RPV. The HSM authenticates the RPV, and control is returned to the LunaSH prompt.

```
Command Result : 0 (Success)
```

The HSM-initiated Remote PED connection is now open.

3. Verify the Remote PED connection by entering a command that requires multifactor quorum authentication.
   - If the HSM is already initialized and you have the blue HSM SO PED key, you can use lunash:> **hsm login**.
   - If the HSM is uninitialized, you can initialize it now with lunash:> **hsm init -label** <label>. Have blank or reusable blue and red PED keys ready (or multiple blue and red keys for M of N or to make multiple copies). See "Creating PED keys" on page 287 for more information.

   > **NOTE** The HSM-initiated Remote PED connection eventually times out (default: 1800 seconds), and must be re-initiated each time authentication is required. To simplify this process, you can set a default IP address and/or port for LunaSH to use each time you connect. To drop the Remote PED connection manually, see "Ending or Switching the Remote PED Connection" on page 274.

4. [OPTIONAL] Set a default IP address and/or port for the Luna Network HSM 7 to look for a configured Remote PED.

   lunash:> **hsm ped set -ip** <PEDserver_IP> **-port** <PEDserver_port>

   ```
   lunash:>hsm ped set -ip 192.124.106.100 -port 1503

   Command Result : 0 (Success)
   ```

   With this default address set, the HSM administrator can use lunash:> **hsm ped connect** (without specifying the IP/port) to initiate the Remote PED connection. The orange Luna PED will be required each time.

   > **NOTE** If you want to use the Remote PED to authenticate a different HSM, you must first drop the current connection. See "Ending or Switching the Remote PED Connection" on page 274.

To open a Remote PED connection from a client workstation

1. Launch LunaCM on the client.
2. Initiate the Remote PED connection.

   lunacm:> **ped connect -ip** <PEDserver_IP> **-port** <PEDserver_port>

   ```
   lunacm:>ped connect -ip 192.124.106.100 -port 1503

   Command Result : No Error
   ```

3. Issue the first command that requires authentication.
   - If the partition is already initialized and you have the blue Partition SO key, log in.

     lunacm:> **role login -name po**
   - If the partition is uninitialized, you can initialize it now. Have blank or reusable blue and red PED keys ready (or multiple blue and red keys for MofN or for multiple copies). See "Creating PED keys" on page 287 for more information on creating PED keys.

     lunacm:> **partition init -label** <label>
4. The Luna PED prompts for an orange PED key. Present the orange PED key with the correct RPK.

```
SLOT
COMPUTE SESSION KEY.
Insert a
RPK / Remote
PED Key (ORANGE)
Press ENTER.
```

5. The Luna PED prompts for the key associated with the command you issued. Follow the on-screen directions to complete the authentication process.

```
SLOT
SO LOGIN...
Insert a
SO / HSM Admin
PED Key (BLUE)
Press ENTER.
```

> **NOTE** The HSM-initiated Remote PED connection eventually times out (default: 1800 seconds), and must be re-initiated each time authentication is required. To simplify this process, you can set a default IP address and/or port for LunaCM to use each time you connect. To drop the Remote PED connection manually, see "Ending or Switching the Remote PED Connection" on page 274.

6. [OPTIONAL] Set a default IP address and/or port for the Luna Network HSM 7 to look for a configured Remote PED.

   lunacm:> **ped set -ip** <PEDserver_IP> **-port** <PEDserver_port>

```
lunacm:>ped set -ip 192.124.106.100 -port 1503

Command Result : 0 (Success)
```

   With this default address set, the HSM administrator can use lunacm:> **ped connect** (without specifying the IP/port) to initiate the Remote PED connection. The orange PED key may be required if the RPK has been invalidated on the PED since you last used it.

> **NOTE** If you want to use the Remote PED to authenticate a different HSM, you must first drop the current connection. See "Ending or Switching the Remote PED Connection" on page 274.

PED-Initiated Remote PED

A PED-initiated connection requires the HSM and Remote PED host to exchange and register certificates, creating a trusted connection. This allows the Remote PED host (via PEDserver) to initiate the connection to the Luna Network HSM 7. If you have firewall or other constraints that prevent your HSM from initiating a connection to a Remote PED in the external network, use this connection method. The HSM administrator can use this procedure to set up the connection. You require:

- Administrative access to a network-connected workstation with PEDserver installed and Luna PED connected (see "Installing PEDserver and Setting Up the Remote Luna PED" on page 263)
- Orange PED key with the HSM's RPV (see "Initializing the Remote PED Vector and Creating an Orange Remote PED key" on page 259)
- Administrative access to the Luna Network HSM 7 via SSH

> **NOTE** The PED-initiated Remote PED connection procedure requires **admin** access to the appliance via LunaSH, and therefore this method cannot directly provide authentication services for client partitions.

- Only self-signed certificates are supported for this procedure.

To open a PED-initiated Remote PED connection

1. On Windows, open an Administrator command prompt on the Remote PED host. (If you are running Windows Server 20xx, the Administrator prompt is launched by default. For any other supported Windows version, right-click the Command Prompt icon and select **Run as administrator**.)
2. Navigate to the Luna HSM Client install directory (**C:\Program Files\SafeNet\LunaClient** or **/usr/safenet/lunaclient**)
3. You will need the Remote PED host's NTLS certificate. If you have already set up an NTLS client connection to the appliance using LunaCM, you can find the certificate in **C:\Program Files\SafeNet\LunaClient\cert\client** or **/usr/safenet/lunaclient/cert/client**. If the certificate is not available, you can generate it with the PEDserver utility.

> **CAUTION!** If the Remote PED host has registered NTLS partitions on any HSM, regenerating the certificate will cause you to lose contact with your registered NTLS partitions. Use the existing certificate instead.

> **"pedserver -regen" on page 319 -commonname** <name>

```
c:\Program Files\SafeNet\LunaClient>pedserver -regen -commonname RemotePED1
Ped Server Version 1.0.6 (10006)

Are you sure you wish to regenerate the client certificate?
All registered partitions may disappear.

Are you sure you wish to continue?

Type 'proceed' to continue, or 'quit' to quit now -> proceed

Private Key created and written to: C:\Program
Files\SafeNet\LunaClient\cert\client\RemotePED1Key.pem
```

```
Certificate created and written to: C:\Program
Files\SafeNet\LunaClient\cert\client\RemotePED1.pem

Successfully regenerated the client certificate.
```

4. Use **pscp** or **sftp** to securely retrieve the Luna Network HSM 7's NTLS certificate. Enter the appliance's admin account password when prompted. Note the period at the end of the command.

   >**pscp admin@**<appliance_IP>**:server.pem .**

   ```
   c:\Program Files\SafeNet\LunaClient>pscp admin@192.20.11.78:server.pem .
   admin@192.20.11.78's password:

   server.pem                | 1 kB |   1.1 kB/s | ETA: 00:00:00 | 100%
   ```

   > **TIP** SCP is deprecated and SFTP is enabled by default for file transfer operations with Luna HSMs and clients. While you can continue using scp with Luna products, for the time being, eventually openSSL might discontinue scp support, and we recommend that you "future-proof" your operations by updating scripts and procedures to call sftp by preference.

5. Use **pscp** or **sftp** to securely transfer the Remote PED host's NTLS certificate to the Luna Network HSM 7's **admin** account.

   >**pscp .\cert\client**<certname> **admin@**<appliance_IP>**:**

   ```
   c:\Program Files\SafeNet\LunaClient>pscp .\cert\client\RemotePED1.pem admin@192.20.11.78:
   admin@192.20.11.78's password:

   RemotePED1.pem              | 1 kB |   1.1 kB/s | ETA: 00:00:00 | 100%
   ```

6. Register the Luna Network HSM 7 certificate with PEDserver. Use the mandatory **-name** argument to set a unique name for the appliance. The appliance listens for the SSL connection from PEDserver at the default port **9697**.

   >**"pedserver -appliance register" on page 308 -name** <appliance_name> **-certificate** <cert_filename> **-ip** <appliance_IP> **-port** <port>

7. Open an SSH session to the Luna Network HSM 7 and log in to LunaSH as **admin**.

8. Register the PEDserver host certificate.

   lunash:> **hsm ped server register -certificate** <certname>

   ```
   lunash:>hsm ped server register -certificate RemotePED1.pem

   'hsm ped server register' successful.

   Command Result : 0 (Success)
   ```

9. Initiate the connection between PEDserver and the Luna Network HSM 7.

   >**"pedserver -mode connect" on page 312 -name** <appliance_name>

   ```
   c:\Program Files\SafeNet\LunaClient>pedserver mode connect -name myLunaHSM
   Ped Server Version 1.0.6 (10006)

   Connecting to myLunaHSM. Please wait..

   Successfully connected to myLunaHSM.
   ```

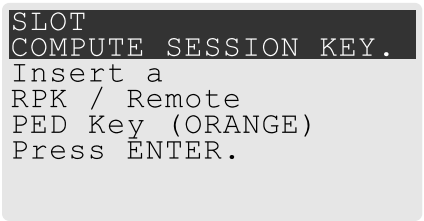
10. Using LunaSH, list the available registered Remote PED servers to find the server name (taken from the certificate filename during registration). Select the server you want to use to authenticate credentials for the appliance.

    lunash:> **hsm ped server list**

    lunash:> **hsm ped select -host** <server_name>

```
lunash:>hsm ped server list

   Number of Registered PED Server :  1

      PED Server  1 : CN = RemotePED1

Command Result : 0 (Success)


lunash:>hsm ped select -host RemotePED1



Luna PED operation required to connect to Remote PED - use orange PED key(s).
```

11. The Luna PED prompts for an orange PED key. Present the orange PED key with the correct RPK for the HSM.



    The secure network connection is now in place between PEDserver and the appliance. You may now perform any actions that require Remote PED authentication, from lunash. The PED-initiated Remote PED connection does not time out as long as PEDserver is running. If you wish to end the connection in order to connect to a different instance of PEDserver, see "Ending or Switching the Remote PED Connection" on page 274.

PED-initiated Remote PED for Client (lunacm)

LunaCM, which is a client-side tool, is not able to launch a PED-initiated Remote PED connection if the firewall blocks the initial attempt. LunaCM does not have administrative access to the HSM appliance and is not aware of PED-client settings on the HSM side (such as the port at which the HSM will look for the PED.

If you control two roles, if you are both the HSM SO and the owner/user/PSO of the application partition that is assigned for crypto operations, then you can coordinate actions in Luna Shell (LunaSH command line) and in LunaCM at the client end, to establish a Remote PED connection.

Or, you can do the same if you are the partition owner and are also able to coordinate closely with a person who has administrative access to LunaSH on the HSM appliance.

> Setup PED-initiated Remote PED connection (refer to the steps above in "To open a PED-initiated Remote PED connection" on page 270 section).

> On the Remote PED host, use the lunacm **ped** commands to set the identity of the PedServer to match what you have told the HSM to expect

- Use **ped set** to provide the IP address and the port number that you determined (or that your colleague determined) in the LunaSH session.

> **NOTE** IP address and port number are found in the "Connected PED Server Table:" section of lunash hsm ped show command output.
>
> The port number will need to be opened for inbound traffic on the host with that IP address.

> On the Client (which could also be the Remote PED host, or could be a separate computer/application server), run a command that invokes PED operation, like the **role login** command.

> The HSM receives the command and looks to the PED (in this case the Remote PED) that has been previously specified in LunaSH.

Example:

Person with access to **admin** account on the Luna Network HSM 7 verfies that the HSM is expecting a Remote PED connection on a specific port, from a specific IP address -

```
lunash:>hsm ped show

 <snip>

   Connected PED Server Table:      PED ID:                              4
      Server Hostname:                    192.168.0.178
      Server Port:                        49982
      Status:                             Selected
      Server Information:
         IP:                              192.168.0.178
         Firmware Version:                2.9.0-2
         PedII Protocol Version:          1.0.1-0
         Software Version:                1.0.6 (10006)
         Ped2 Connection Status:          Connected
         Ped2 Connection Type:            Inbound Connection
         Ped2 RPK Count                   0
         Ped2 RPK Serial Numbers          (none)


Show command passed.

Command Result : 0 (Success)
lunash:>
```

If not, see earlier on this page to set up Remote PED.

Person at the PEDserver (which could be the same computer as the partition client, or could be a separate computer, dedicated to being PED server) uses LunaCM to ensure that the PEDserver is using the correct port and IP that the HSM (above) is expecting.

> **NOTE** **pedserver_ip** and **pedserver_port** below are respectively "IP:" and "Server Port:" fields from the "Connected PED Server Table" section.

```
lunacm:> ped set -ip pedserver_ip -port pedserver_port
lunacm:> ped connect
```

Person who is the PSO of the current slot (which is the desired application partition on the distant Luna Network HSM 7) runs the LunaCM commands that will require the HSM to look for PED interaction.

```
lunacm:> partition init -label 550097_par1 -f
lunacm:> ped connect
lunacm:> role login -n po
lunacm:> ped connect
lunacm:> role init -n co
```

> **NOTE** The use of lunacm:> **ped connect** before every partition administrative command is not always necessary, but is a best-practice in unstable network conditions or in situations where network/firewall rules might drop the PEDclient-PEDserver connection frequently or unexpectedly.
>
> If the (re-)connection fails, have the person with "admin" access on the Luna Network HSM 7 re-establish the HSM side of the connection to the PEDserver (expected port and IP) before you issue any more client-side commands that need multifactor quorum authentication.

Ending or Switching the Remote PED Connection

PEDserver runs on the Remote PED host until explicitly stopped. PEDclient (running on the Luna Network HSM 7) behaves differently depending on the type of Remote PED connection. If you want to connect to a different Remote PED server, or allow another HSM to use the current server, you must manually break the Remote PED connection.

To end or switch an HSM-initiated Remote PED connection using LunaSH

1. End the Remote PED connection.

   lunash:> **hsm ped disconnect**

2. You are now able to initiate a connection to a different Remote PED host running PEDserver. You will need to present the orange PED key.

   lunash:> **hsm ped connect -ip** <PEDserver_IP> **-port** <port>

   > **NOTE** Running this command does not change the default Remote PED IP/port you may have previously set. If you want this new Remote PED server to be the default, set it using lunash:> **hsm ped set -ip** <PEDserver_IP> **-port** <port>.

To end or switch an HSM-initiated connection using LunaCM

1. End the Remote PED connection.

   lunacm:> **ped disconnect**

2. You are now able to initiate a connection to a different Remote PED host running PEDserver. You will need to present the orange PED key.

   lunacm:> **ped connect -ip** <PEDserver_IP> **-port** <port>

> **NOTE** Running this command does not change the default Remote PED IP/port you may have previously set. If you want this new Remote PED server to be the default, set it using lunacm:> **ped set -ip** <PEDserver_IP> **-port** <port>.

To end or switch a PED-initiated Remote PED connection

1. End the Remote PED connection with the current host ().

   lunash:> **hsm ped deselect -host** <server_name>

2. Check the available list of Remote PED servers.

   lunash:> **hsm ped server list**

   If the Remote PED you want to use is not in the list, see "PED-Initiated Remote PED" on page 270.

3. The new Remote PED server must initiate the connection to the appliance.

   > **"pedserver -mode connect" on page 312 -name** <appliance_name>

4. In LunaSH, you are now able to select the new Remote PED server from the available list.

   lunash:> **hsm ped select -host** <server_name>

Remote PED Troubleshooting

If you encounter problems at any stage of the Remote PED connection process, the following troubleshooting tips may help resolve the problem:

Luna PED Not Detected if Connected While PEDserver is Stopped

When the Luna PED is connected to the host machine while PEDserver is stopped, it may not be detected even after PEDserver starts up. The output of "pedserver -mode show" on page 314 displays:

```
Ped2 Connection Status:             Disconnected
```

With PEDserver running, disconnect and reconnect the Luna PED to the host machine and wait for it to boot. Run "pedserver -mode show" on page 314 again and ensure the following is displayed:

```
Ped2 Connection Status:             Connected
```

Cryptographic Operations Blocked During Remote PED Operations When Audit Logging Is Enabled

With audit logging enabled on the HSM, crypto operations are blocked on all application partitions during Remote PED operations. During this time, requests sent to HA member partitions on this HSM will not fail over to other members. When the Remote PED operation is complete, all crypto operations resume normally. If your application has its own timeout programmed, it may incorrectly conclude that the entire HA group has failed.

Using Luna HSM Client 10.7.2 or newer, you can configure the "ProbeTimeout" on page 89 setting in the **Chrystoki.conf/crystoki.ini** file to trigger an HA failover after a specified time. This allows operations to continue normally during Remote PED operations.

Intermittent CKR_CALLBACK_ERROR: PED Cannot Service its USB Data Channel Fast Enough to Communicate with PEDserver

> **NOTE** This issue might occur during Remote PED connections between
>
> > A Luna Network HSM 7 with Luna HSM Firmware 7.7.0 or newer and a remote workstation with Luna HSM Client 10.3.0 or newer.
>
> > A Luna Backup HSM 7 with firmware 7.7.1 or newer and a remote workstation with Luna HSM Client 10.3.0 or newer.

The PED might not be able service its USB data channel fast enough to communicate with PEDserver and you will intermittently receive CKR_CALLBACK_ERROR.

The following error appears in the PEDserver log file:

```
* ERROR ** 32725 : pedsock_rmtped_write_1_waitack_write_n_waitack failed: 0xffffffff (-1)*
```

If driver log messages are available on your system, the following message may appear where driver logs are reported:

```
kernel: lunaped: read: usb_bulk_msg: rc = -110
```

To avoid this error, throttle communication between the PED and PEDserver by running the following command from a command prompt:

pedserver -mode config -set -pedwritedelay <int>

> **NOTE** To resolve this error in most cases, Thales recommends setting the value of **-pedwritedelay** to **50**. If you still experience this issue, set **-pedwritedelay** to a value higher than **50**. For more information about this option, refer to "pedserver -mode config" on page 310.

No Menu Appears on Luna PED Display: Ensure Driver is Properly Installed

If the PED driver is not properly installed before connecting the PED to the workstation's USB port, the PED screen does not display the menu. If you encounter this problem, ensure that you have followed the entire procedure at "Installing PEDserver and Setting Up the Remote Luna PED" on page 263.

RC_SOCKET_ERROR: PEDserver Requires Administrator Privileges

If PEDserver is installed in the default Windows directory, it requires Administrator privileges to make changes. if you run PEDserver as an ordinary user, you may receive an error like the following:

```
c:\Program Files\SafeNet\LunaClient>pedserver mode start
Ped Server Version 1.0.6 (10006)
Ped Server launched in startup mode.
Starting background process
Failed to recv query response command: RC_SOCKET_ERROR c0000500
Background process failed to start : 0xc0000500 RC_SOCKET_ERROR
Startup failed. : 0xc0000500 RC_SOCKET_ERROR
```

To avoid this error, when opening a command line for PEDserver operations, right-click the Command Prompt icon and select **Run as Administrator**. Windows Server 20xx opens the Command Prompt as Administrator by default.

> **NOTE** If you do not have Administrator permissions on the Remote PED host, contact your IT department or install Luna HSM Client in a non-default directory (outside the **Program Files** directory) that is not subject to permission restrictions.

LUNA_RET_PED_UNPLUGGED: Reconnect HSM-initiated Remote PED Before Issuing Commands

As described in the connection procedures, HSM-initiated Remote PED connections time out after a default period of 1800 seconds (30 minutes). If you attempt authentication via PED after timeout or after the connection has been broken for another reason, the Luna PED will not respond and you will receive an error like this:

```
lunash:>hsm login

Luna PED operation required to login as HSM Administrator - use Security Officer (blue) PED
key.

Error:  'hsm login' failed. (300142 : LUNA_RET_PED_UNPLUGGED)


Command Result : 65535 (Luna Shell execution)
```

To avoid this error, re-initiate the connection before issuing any commands requiring authentication via PED:

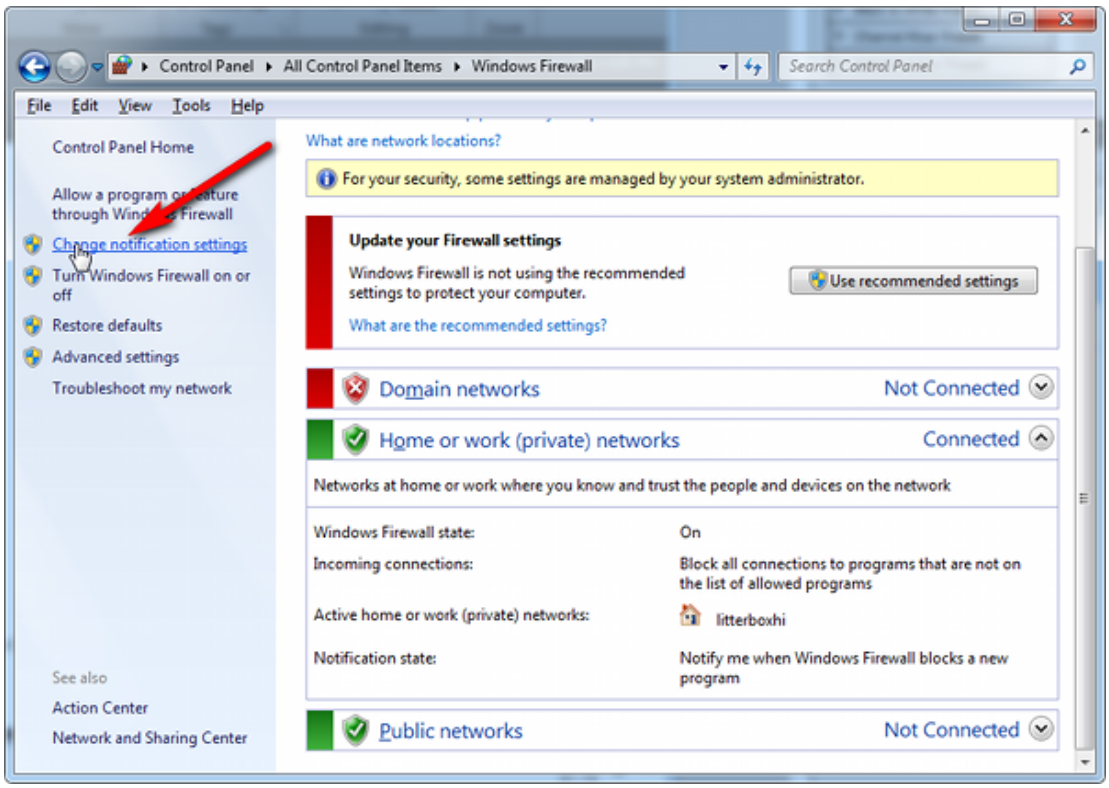
lunash:> **hsm ped connect -ip** <PEDserver_IP> **-port** <PEDserver_port>

lunacm:> **ped connect -ip** <PEDserver_IP> **-port** <PEDserver_port>

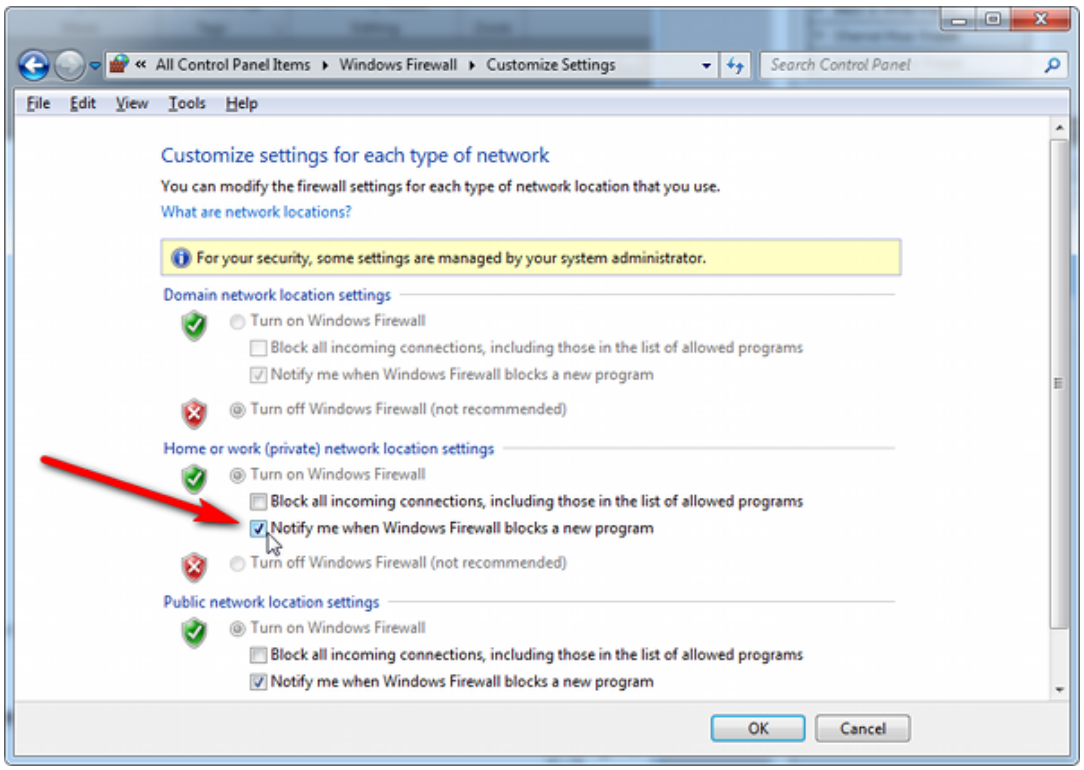
Remote PED Firewall Blocking

If you experience problems while attempting to configure a Luna Remote PED session over VPN, you might need to adjust Windows Firewall settings. If your security policy prohibits changes to Windows Firewall, you can use a PED-initiated connection for HSM SO-level operations. See "PED-Initiated Remote PED" on page 270.

1. From the Windows Start Menu, select **Control Panel**.
2. Select **Windows Firewall**.
3. Select **Change notification settings**.



4. In the dialog **Customize settings for each type of network**, go to the appropriate section and activate **Notify me when Windows Firewall blocks a new program**.



With notifications turned on, a dialog box appears whenever Windows Firewall blocks a program, allowing you to override the block as Administrator. This allows PEDserver to successfully listen for PEDclient connections.

Remote PED Blocked Port Access

The network might be configured to block access to certain ports. If ports 1503 (the default PEDserver listening port) and 1502 (the administrative port) are blocked on your network, choose a different port when starting PEDserver, and when using lunacm:> **ped connect** or lunash:> **hsm ped connect** to initiate the Remote PED connection. Contact your network administrator for help.

You might choose to use a port-forwarding jump server, co-located with the Luna Network HSM 7(s) on the datacenter side of the firewall. This can be a low-cost solution for port-blocking issues. It can also be used to implement a PKI authentication layer for Remote PED or other SSH access, by setting up smart-card access control to the jump server.

For example, you can use a standard Ubuntu Server distribution with OpenSSH installed and no other changes made to the standard installation with the following procedure:

1. Connect the Luna PED to a Windows host with Luna HSM Client installed and PEDserver running.
2. Open an Administrator command prompt on the Remote PED host and start the port-forwarding service.

   >**plink -ssh -N -T -R 1600:localhost:1503 <user>@<Ubuntu_server_IP>**.
3. Login to the appliance as **admin** and open the HSM-initiated connection.

   lunash:> **hsm ped connect -ip <Ubuntu_server_IP> -port 1600**

The Remote PED host initiates the SSH session, via the Ubuntu jump server, which returns to the Remote PED host running PEDserver.

A variant of this arrangement also routes port 22 through the jump server, which allows administrative access to the Luna Network HSM 7 under the PKI access-control scheme.

ped connect Fails if IP is Not Accessible

On a system with two network connections, if PEDserver attempts to use an IP address that is not externally accessible, lunacm:>**ped connect** can fail. To resolve this:

1. Ensure that PEDserver is listening on the IP address that is accessible from outside.
2. If not, disable the network connection on which PEDserver is listening.
3. Restart PEDserver and confirm that it is listening on the IP address that is accessible from outside.

PEDserver on VPN fails

If PEDserver is running on a laptop that changes location, the active network address changes even though the laptop is not shutdown. If you unplugged from working at home, over the corporate VPN, commuted to the office, and reconnected the laptop there, PEDserver is still configured with the address you had while using the VPN. Running **pedserver -mode stop** does not completely clear all settings, so running **pedserver -mode start** again fails with a message like "Startup failed. : 0x0000303 RC_OPERATION_TIMED_OUT". To resolve this problem:

1. Close the current command prompt window.
2. Open a new Administrator command prompt.

3. Verify the current IP address.

   >**ipconfig**

4. Start PEDserver, specifying the new IP and port number ().

   > **"pedserver -mode start" on page 316** **-ip** <new_IP> **-port** <port>

PED Utilities Run by Non-root Users

The default location of the PED utility log is the current directory where the PED utility command has executed, like ./remotePedServerLog.log. Non-root users, even members of the hsmusers group, do not have write permission to the bin directory, or any directory in /usr/safenet, so the PED utility PedServer or PedClient started by a non-root user fails to start.

PED Server

Without root access (or workaround... see below), the utility fails to launch, displaying the following error message:

```
[bin]$ ./PedServer -m start
Ped Server Version 1.0.6 (10006)
Ped Server launched in startup mode.
Connecting to PED. Please wait.........InternalRead: 10 seconds timeout
Failed to recv query response command: RC_OPERATION_TIMED_OUT c0000303
Failed to connect to PED. Please see logs for further details.
Ped Server Process created, exiting this process.
--------------------------------------------------------------------------------
```

The service needs to log all its actions, including the action of making a connection to the PED, so after failing to create the log (no write permission), it aborts the action of connecting to the PED.

The **workaround** is to set the PED server LogFileName to a location where the current user has read and write access, such as the user's home directory.

Examples:

```
$ ./PedServer -mode config -set LogFileName  $HOME/remotePedServerLog.log
```

OR

```
$./PedServer -mode config -set LogFileName  /tmp/remotePedServerLog.log
```

Then run $./PedServer -mode start

OR

start the PedServer with log file option: -logfilename /dev/null

```
$bin/PedServer -m start -logfilename /dev/null
$bin/PedServer -m start -logfilename  $HOME/remotePedServerLog.log
```

PEDClient

PedClient has some similar requirements.

Have the user in an appropriate user group, and they can then launch with systemctl

PED connection Fails with Error: pedClient is not currently running

It can happen that the callback server gets shut down, which prevents connections that use it, like Remote PED and remote backup. To resolve this:

1. On the appliance, restart the callback service.

lunash:> **service restart cbs**

2. Start the Remote PED connection again (initiated at the PED side or at the HSM side, as appropriate to your network and firewall protocols).

The callback service also restarts when the appliance is rebooted.

Updating External Supply-Powered Luna PED Firmware

This section describes how to update the firmware on your Luna PED that is powered by a power-block. Refer to Customer Release Notes for valid update paths.

> **NOTE** If your Luna PED is the newer model that is powered by a USB connection (and is not shipped with a power-block), see "Updating USB-Powered Luna PED Firmware" on page 284.

Files Included in the Upgrade Package

The update package includes the following files. Both files are required to successfully perform the update:

> Firmware update file for the desired version (<PED_firmware_file_name>**.bin**, where the version is in the range 2.7.x)

> if the package contains **LunaPED_Update.exe** use that; otherwise, download KB0015846 from the Support Portal for a copy of LunaPED_Update.exe that works with PEDs powered by power block.

Preparing for the Update

Before you can install the new firmware, you must download the update package to the Windows Luna HSM Client workstation you will use to perform the update, and configure the Luna PED to accept the update.

> **CAUTION!** It is strongly recommended that you protect both your computer and Luna PED with an uninterruptible power supply during the upgrade operation. A power failure while any of the file images are being applied to the PED can result in loss of function that might require RMA.

To prepare your computer for the update

1. Ensure that Luna HSM Client software, including the Remote PED option, is installed on the Windows PC you will use to update the PED. To verify, ensure that the following files/directories are installed:
   - **C:\Program Files\SafeNet\LunaClient\RemotePEDDriver**
   - **C:\Program Files\SafeNet\LunaClient\pedserver.exe**
2. The update files are provided in an archive file named for the PED upgrade part number. Extract the files to the Windows Luna HSM Client workstation connected to the Luna PED you are updating.
3. On your Luna HSM Client workstation, where the PED is physically connected, stop the pedserver and pedclient services before starting the PED update.

> **NOTE** If you are updating the PED firmware from version 2.4.x to 2.5.0 or to 2.6.0 on a Windows 10 workstation (recall that the upgrade path is 2.4.0=>2.4.1=>2.5.0=>2.6.0=>2.7.0=>2.7.1=>2.7.4), then use the PEDupdate.exe that is included with the 2.7.x or 2.9.0 PED firmware update and *not* the firmware update package that was included with the Luna Client 6.2.2 package.

4. On your Luna HSM Client workstation, open a command prompt window and move to the directory where you copied the files in the update package.

To prepare the Luna PED for the firmware update

1. Connect the Luna PED to power (if you have an older PED that is not powered by the USB connection) and connect the USB cable between the Luna PED and your Luna HSM Client workstation.
2. Allow the PED to boot normally until it reaches the default **Local PED mode Awaiting command…**.
3. Press the **<** key to display the **Mode** menu.
4. Verify the currently-installed PED firmware version.

> **CAUTION!** If you are updating an older PED (not powered by the USB connection), this procedure requires starting from version **2.6.0-6** or newer. If your PED displays an earlier version, the update will fail and the PED will require RMA. If you have an older version, update the PED to 2.6.0-6 before continuing with this procedure.

5. Select **4** to display the **Admin** menu.
6. Select **7** for **Software Update**.
7. Select **0** to reset the PED and immediately press and hold the **<** key while the PED is resetting. Continue to hold the **<** key until the **Select Mode** menu is displayed.
8. Select **USB Mode** (4) when prompted to **Select Mode**. The PED displays **USB Mode**.

Updating the Luna PED Firmware

During this procedure, each of the **.bin** files is individually uploaded from your computer to the Luna PED, and then saved into permanent memory as the new version of that component. Individual responses are required at the PED to accept and load each file.

> **CAUTION!** Complete the following instructions in the order provided, or the PED could be left in an unusable state.

Once you start transferring / uploading a file to the PED, pay attention and promptly respond to the PED messages to acknowledge the upload and then to confirm installation of that new file. The individual PED operations do impose a timeout. However, you can pause before the next file transfer step, as there is no time restriction from one file upload to the next.

To update the Luna PED firmware

1. In the command prompt window on the Windows Luna HSM Client workstation you prepared to perform the update, execute the following command:

   > **LunaPED_Update.exe** <PED_firmware_file_name>**.bin**

   > **NOTE** If you have both older Luna PEDs (that are powered by a power block), and the newer Luna PEDs (powered by USB connection and addressed in "Updating USB-Powered Luna PED Firmware" on the next page), then the LunaPED_Update.exe files for each are different and not interchangeable.

2. On the Luna PED, select **Yes** in response to the prompt: **Software update. Upload Image? YES/NO**.

   Wait approximately six minutes. While transfer is in progress, the command line shows a progress indicator (remaining bytes to transfer), and the PED displays the following message:

   ```
   USB Mode
   Software update
   Uploading image
   ```

3. The output of the update command in the Windows command prompt should be similar to the following:

   ```
   LunaPED_Update v2.1.0-1 Nov 25 2013 12:44:48
   PED operation is required (to upload image)...
   (Sent 3199130 bytes in 327977000 microsecs).
   PED operation is required (to save image)...
   ```

4. If the image has been sent correctly, the PED displays the following message:

   ```
   USB Mode
   Software update
   ** WARNING **
   A power failure during save is unsupported!
   Save Image? YES/NO
   ```

   Select **Yes** to save the new image.

5. Wait for 20-30 seconds. When the PED displays the following message, press the **Enter** key on the PED keypad to return to USB mode:

   ```
   Software update
   Success
   Press ENTER
   ```

6. Unplug all cables from the PED and then reconnect to restart the PED. As the PED starts booting, it should display the following messages:

   ```
   BOOT  V.1.0.6-2,
   loading PED…
   Local PED Mode   Awaiting command…
   ```

7. Press **<** to exit to the **Select Mode** menu. If the update was successful, the new PED version is displayed at the bottom of the PED screen.

8. Your Luna PED is now updated and ready to use. Repeat the procedure for each Luna PEDLuna PED that you own.

Troubleshooting

This section provides guidance for resolving problems you may encounter when updating the PED firmware.

If your update attempt fails with a Receive error (rx error), check if you have Remote PED services running on the computer to which the PED is connected. Issue the command **PedServer –m stop** and restart the update to resolve the problem.

No Luna PED Prompts

You must attend to the PED when image files are being applied. If no prompts appear on the PED shortly after you issue the **LunaPED_Update.exe** command, re-check your connections, as follows:

- The PED power block must be connected to AC power and to the power socket on the PED.
- A USB connection must exist between a USB port on the sending computer and the USB-mini port on the PED (immediately beside the power socket).
- The PED must be powered on, and in USB mode.

Files Uploaded in the Wrong Order

If you attempt to upload the files in the wrong order, the PED performs some verification at the end of a file upload. If the PED displays a message similar to the following, it is a good indication that you uploaded the wrong file first:

```
Failure (VERIFY) (7)
Press Enter
```

You are not given an opportunity to attempt to install/confirm the file if the upload does not verify.

To resolve the issue, restart the process from the beginning of these instructions, ensuring that you follow the sequence in these instructions, taking the upgrade files in the order specified. If that does not correct the problem, contact Technical Support.

Upgrade Failed Message (or Similar)

If the Luna PED displays an **Upgrade Failed** message, or any message that does not say **Upgrade in Progress** followed by **Upgrade Complete**, before the **Admin** menu appears, stop the upgrade process immediately.

To resolve the issue, you can take the following actions:

- Reboot the PED by disconnecting and then re-connecting the PED cables. This might clear the problem. If the problem clears, the PED displays a **Nothing to Upgrade** message. In this case, try the update again.
- If the PED shows **Upgrade in Progress** followed by **Upgrade Failed!** every time you reboot it, contact Customer Support.
- You can re-upload the file and try again if the upload action failed to complete, or if you failed to acknowledge it on the PED.

Updating USB-Powered Luna PED Firmware

This section describes how to update the firmware on your Luna PED that is powered by USB connection. Refer to Customer Release Notes for valid update paths.

To update the Luna PED from Luna PED Firmware 2.8.0 a newer version, follow the steps below.

If your Luna PED is the older type, that was shipped with a power-block, then do not use these instructions; see "Updating External Supply-Powered Luna PED Firmware" on page 281 instead.

Preparing for the Upgrade

> **CAUTION!** It is strongly recommended that both your computer and Luna PED be protected by an uninterruptible power supply during the upgrade operation. A power failure while any of the file images is being applied to the PED can result in loss of function that might require repair at a Thales facility.

Prepare your computer for the upgrade

The needed upgrade files are provided in an archive file named for the PED upgrade part number. At time of writing this instruction, KB0023048 from the Support Portal contained the appropriate firmware and updater files.

1. Extract the files like *ped-2.9.1-0-x-production-itb-real.bin* (or newer if available) and *LunaPED_Update.exe* contained in the zip file, to the Windows PC that is connected to the Luna PED that you are upgrading.

   > **NOTE** If you have both older Luna PEDs (that are powered by a power block and addressed on "Updating External Supply-Powered Luna PED Firmware" on page 281), and the newer Luna PEDs (powered by USB connection and addressed on this page), then the LunaPED_Update.exe files for each are different and *not interchangeable*.

2. On your Windows PC, open a command prompt window and move to the directory where you copied the files in the upgrade package.

Prepare the Luna PED for the firmware upgrade

1. Ensure that the Luna HSM Client, including the Remote PED option, is installed on your Windows PC. To verify, ensure that the following files / directories are installed:
   - C:\Program Files\SafeNet\LunaClient\RemotePEDDriver
   - C:\Program Files\SafeNet\LunaClient\pedserver.exe
2. Connect *the USB data cable between the USB-mini port* on top of the Luna PED and a USB port on your computer.

   > **NOTE** Luna PED Firmware 2.8.0 and newer is powered by the USB port; a separate power supply to the Luna PED is not provided nor required.

3. Allow the PED to boot normally until it reaches the default “Local PED mode Awaiting command…”
4. Press the **<** key to display the **Mode** menu.
5. Verify the PED version – the bottom line of the PED display should say “PED V.2.8.0”

> **CAUTION!** If any other version is shown, stop, acquire a factory shipped Luna PED with Luna PED Firmware 2.8.0, and then return and resume these instructions. If your Luna PED firmware version is older than 2.8.0 (such as 2.6.x) it can only ever be updated to version 2.7.x - see "Updating External Supply-Powered Luna PED Firmware" on page 281 for the relevant update instructions.

6. Select **4** to display the **Admin** menu.
7. Select **7** for **Software Update**.

Upgrading the Luna PED Firmware to Version 2.9.0 (or newer)

During this procedure, the .bin file is individually uploaded from your computer to the Luna PED, and then saved into permanent memory as the new version. Individual responses are required at the PED to accept and load the file.

> **CAUTION!** Complete the instructions in the order provided, otherwise the Luna PED could be left in an unusable state.

Once you start transferring / uploading a file to the PED, pay attention and promptly respond to the PED messages to acknowledge the upload and then to confirm installation of that new file. *The individual PED operations **do** impose a timeout*. However, you can pause before the next file transfer step, as there is no time restriction from one file upload to the next.

Transfer and confirm the Luna PED firmware update

1. In a command prompt window, on your Windows PC, from the directory where you copied the files in the upgrade package, execute the following command:

   Prompt > **LunaPED_Update.exe ped-2.9.x-y-z-production-itb-real.bin** (where x-y-z are numbers specific to the released build of the firmware)
2. At the Luna PED keypad, select **Yes** in response to the prompt.
3. The output of the update command in the Windows command prompt should be similar to the following:

```
LunaPED_Update v3.0.0-1 May 10 2017 22:52:25
PED operation is required (to upload image)...
(Sent xxxxxxx bytes in xxxxxxxxx microsecs).
PED operation is required (to save image)...
```

4. If the image has transferred correctly, Luna PED displays the following message:

```
USB Mode Software update
** WARNING **
A power failure during save is unsupported!
Save Image? YES/NO"
```

5. Select **Yes** to save the new image.
6. Wait approximately 20 seconds. The PED displays the following message:

```
USB Mode Software update Success Press ENTER
```

   Press the **Enter** key on the PED to continue.

7. Unplug all cables from the PED and then reconnect to restart the PED.
8. As the PED starts booting, it should show “BOOT V.1.1.0-1”, then “loading PED…,” and then should finish in “Local PED Mode awaiting command…”

   If you press “<” to exit to “Select Mode” menu, the bottom of the PED screen should now show "PED V.2.8.1-0" (or "PED V.2.9.0" or a newer version, as one becomes available).

Done
Luna PED is now updated and ready to use. Repeat the above sequence for each USB-powered Luna PED that you want to upgrade.

Multifactor Quorum PED key Management

Once you have established a Local or Remote PED connection, you can proceed with initializing roles on the HSM that require multifactor quorum authentication. The procedures in this section will guide you through the Luna PED prompts at each stage of PED key creation, multifactor quorum authentication, and other operations with the Luna PED.

Creating PED keys

When you initialize an HSM, partition, or role, the Luna PED issues a series of prompts for you to follow to create your PED keys. PED key actions have a timeout setting (default: 200 seconds); ensure that you have everything you need before issuing an initialization command. The requirements for the operation depend on the PED key scheme you have chosen in advance, based on your organization's security policy. Consider these guidelines before you begin:

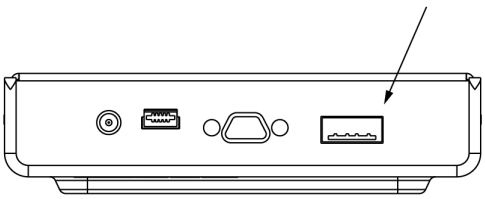
> If you are reusing an existing PED key or keyset, the owners of those keys must be present with their keys and PINs ready.

> If you plan to use an M of N authentication scheme (quorum, or split-secret), all the parties involved must be present and ready to create their authentication split (the initial setup of the quorum and spares). It is advisable for each key holder to create backup duplicates, so you must have a sufficient number of blank or rewritable PED keys ready before you begin.

> If you plan to make backup duplicates of PED keys, you must have a sufficient number of blank or rewritable PED keys ready.

> If you plan to use PINs, ensure that they can be privately entered on the Luna PED and memorized, or written down and securely stored.

Whenever the Luna PED prompts you to insert a PED key, use the USB port on the top of the PED:



To initiate PED key creation

1. Issue one of the following LunaSH or LunaCM commands to initialize the applicable role, domain, or vector.
   - **Blue HSM SO and Red HSM Domain PED key**:

     lunash:> **hsm init**
   - **Orange Remote PED Vector PED key**:

     lunash:> **hsm ped vector init**
   - **Blue Partition SO and Red Partition Domain PED keys**:

     lunacm:> **partition init**
   - **Black Crypto Officer PED key**:

     lunacm:> **role init -name co**
   - **Gray Crypto User PED key**:

     lunacm:> **role init -name cu**
   - **White Audit User PED key**:

     lunash:> **audit init**

   The Luna PED responds, displaying:

```
Remote PED mode
Token found
```

> **NOTE** The Luna PED screen prompts for a black PED key for any of
> > "User",
> > "Crypto Officer",
> > "Limited Crypto Officer",
> > "Crypto User".
>
> The Luna PED is not aware that the key you present has a black or a gray sticker on it. The colored stickers are visual identifiers for your convenience in keeping track of your PED keys. You differentiate by how you label, and how you use, a given physical key that the Luna PED sees as "black" (once it has been imprinted with a secret).

2. Follow the PED prompts in the following four stages.

Stage 1: Reusing Existing PED keys

If you want to use a PED key with an existing authentication secret, have the key ready to present to the Luna PED. Reasons for reusing keys may include:

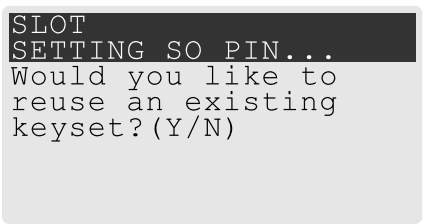
> You want to use the same blue SO key to authenticate multiple HSMs/partitions

> You want to initialize a partition in an already-existing cloning domain (to be part of an HA group)

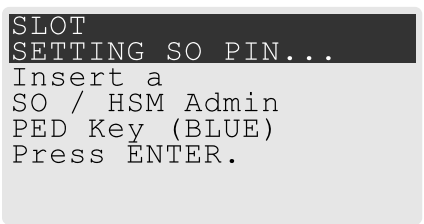
> **CAUTION!** The initialization procedure is the only opportunity to set the HSM/partition's cloning domain. It cannot be changed later without reinitializing the HSM, or deleting and recreating the partition. Ensure that you have the correct red key(s) ready.

See "Shared PED key Secrets" on page 240 and "Domain PED keys" on page 241 for more information.

1. The first Luna PED prompt asks if you want to reuse an existing PED key. Press **Yes** or **No** on the keypad to continue.



   - If you select **No**, skip to "Stage 2: Defining M of N" on the next page.
   - If you select **Yes**, the PED prompts you for a key. Insert the key you want to reuse and press **Enter**.



2. If the key has a PIN, the PED prompts you to enter it now. Enter the PIN on the keypad and press **Enter**.

```
SLOT
READING SO PIN...
Enter PED PIN:
*********
```

3. If the key is part of an M of N scheme, the PED prompts you for the next key. You must present enough key splits (M, a.k.a. the quorum) to reconstitute the entire authentication secret.

```
SLOT
READING SO PIN...
Keys read: 01 of 03
Insert another
SO / HSM Admin
PED Key (BLUE)
Press ENTER.
```

4. The PED asks if you want to create a duplicate set of keys. If you are duplicating an M of N keyset, you need a number of blank or rewritable keys equal to N.

```
SLOT
READING SO PIN...
Are you duplicating
this keyset?(Y/N)
Warning: You will
need all N keys!
```

- If you select **No**, the process is complete.
- If you select **Yes**, complete "Stage 3: Setting a PIN" on the next page for all the duplicate keys you want.

Stage 2: Defining M of N

If you chose to create a new keyset, the Luna PED prompts you to define the M of N scheme (quorum and pool of splits) for the role, domain, or vector. See "Quorum Split Secrets (M of N)" on page 241 for more information. If you do not want to use M of N (authentication by one PED key), enter a value of **1** for both M and N. Effectively, you have set a "quorum" of one key-holder.

1. The PED prompts you to enter a value for M (the minimum number of split-secret keys required to authenticate the role, domain, or vector - the quorum). Set a value for M by entering it on the keypad and pressing **Enter**. If you are not using an M of N scheme, enter "**1**".

```
SLOT
SETTING SO PIN...
M value? (1-16)

>03
```

2. The PED prompts you to enter a value for N -- the total number of split-secret keys you want to create (the pool of splits from which a quorum will be drawn). Set a value for N by entering it on the keypad and pressing **Enter**. If you are not using an M of N scheme, enter "**1**".


```
SLOT
SETTING SO PIN...
N value? (M-16)

>05
```

3. Continue to "Stage 3: Setting a PIN" below. You must complete stage 3 for each key in the M of N scheme.

Stage 3: Setting a PIN

If you are creating a new key or M of N split, you have the option of setting a PIN that must be entered by the key owner during authentication. PINs must be 4-48 digits long. Do not use 0 for the first digit. See "PINs" on page 241 for more information.

> **CAUTION!** If you forget your PIN, it is the same as losing the PED key entirely; you cannot authenticate the role. See "Consequences of Losing PED keys" on page 295.

1. The PED prompts you to insert a blank or reusable PED key. If you are creating an M of N split, the number of already-created splits is displayed.

```
SLOT
SETTING SO PIN...
Insert a
SO / HSM Admin
PED Key (BLUE)
Press ENTER.
```

```
SLOT
SETTING SO PIN...
Keys write: 03 of 05
Insert another
SO / HSM Admin
PED Key (BLUE)
Press ENTER.
```

2. Insert the PED key and press **Enter**. The PED prompts for confirmation.

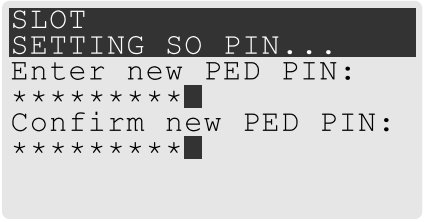
```
SLOT
SETTING SO PIN...
   ** WARNING **
This PED Key is
blank.
Overwrite? YES/NO
```

If the PED key you inserted is not blank, you must confirm twice that you want to overwrite it.

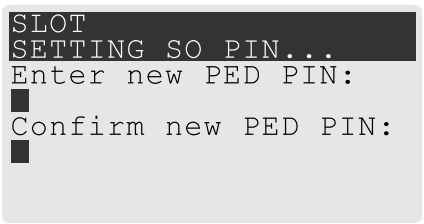
```
SLOT
SETTING SO PIN...
   ** WARNING **
This PED Key is for
Domain.
Overwrite? YES/NO
```

```
SLOT
SETTING SO PIN...
   ** WARNING **
Are you sure you
want to overwrite
this PED key? YES/NO
```

3. The PED prompts you for a PIN.
   - If you want to set a PIN, enter it on the keypad and press **Enter**. Enter the PIN again to confirm it.



- If you do not want to set a PIN, press **Enter** twice without entering anything on the keypad. You will not be asked to enter a PIN for this key in the future.



4. If there are more keys in the M of N scheme, repeat this stage. Otherwise, continue to "Stage 4: Duplicating New PED keys" below.

Stage 4: Duplicating New PED keys

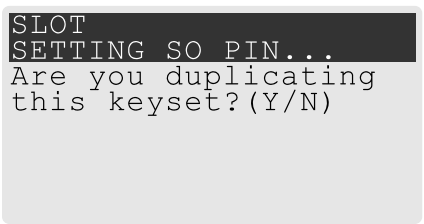
You now have the option to create duplicates of your newly-created PED key(s). There are two reasons to do this now:

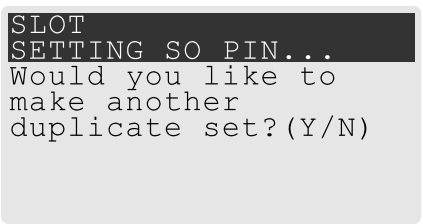
> If you want more than one person to be able to authenticate a role, you can create multiple keys for that role now, with each person being able to set their own PIN. Duplicates you create later are intended as backups, and will have the same PIN (or none) as the key they are copied from.

> In case of key loss or theft.

You can make backups now or later. See also "Duplicating Existing PED keys" on page 298.

1. The next PED prompt asks if you want to create a duplicate keyset (or another duplicate). Press **Yes** or **No** on the keypad to continue.





- If you select **No**, the key creation process is complete.
- If you select **Yes**, complete "Stage 3: Setting a PIN" on the previous page for the duplicate keyset. You can set the same PIN to create a true copy, or set a different PIN for each duplicate.

2. If you specified an M of N scheme, you are prompted to repeat "Stage 3: Setting a PIN" on the previous page for each M of N split. Otherwise, the key creation process is complete.

Performing Multifactor Quorum Authentication

When connected, the Luna PED responds to authentication commands in LunaSH or LunaCM. Commands that require PED actions include:

- Role login commands (blue, black, gray, or white PED keys)
- Backup/restore commands (red PED keys)
- Remote PED connection commands (orange PED key)

> **NOTE** The Luna PED screen prompts for a black PED key for any of
> - "User",
> - "Crypto Officer",
> - "Limited Crypto Officer",
> - "Crypto User".
>
> The Luna PED is not aware that the key you present has a black or a gray sticker on it. The colored stickers are visual identifiers for your convenience in keeping track of your PED keys. You differentiate by how you label, and how you use, a given physical key that the Luna PED sees as "black" (once it has been imprinted with a secret).

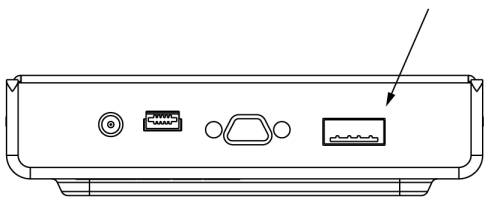
When you issue a command that requires Luna PED interaction, the interface returns a message like the following:

```
lunash:>hsm login

Luna PED operation required to login as HSM Administrator - use Security Officer (blue) PED key.
```

The PED briefly displays the following message before prompting you for the appropriate PED key:

```
Remote PED mode
Token found
```

Whenever the Luna PED prompts you to insert a PED key, use the USB port on the top of the PED:

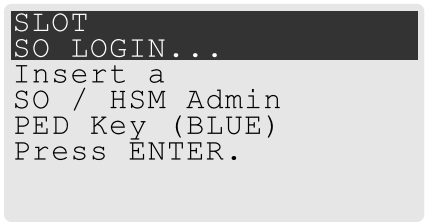


> **CAUTION!** Multiple failed authentication attempts result in zeroization of the HSM or partition, or role lockout, depending on the role. This is a security measure designed to thwart repeated, unauthorized attempts to access cryptographic material. For details, see Logging In as HSM Security Officer or "Logging In to the Application Partition" on page 366.

To perform multifactor quorum authentication with the Luna PED

1. The PED prompts for the corresponding PED key. Insert the PED key (or the first M of N split-secret key) and press **Enter**.

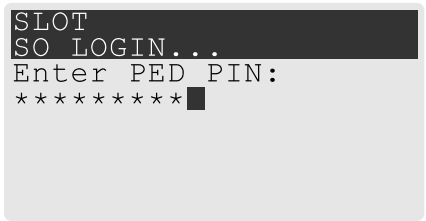
```
lunacm:>role login -name po

        Please attend to the PED.
```



   - If the key you inserted has an associated PIN, continue to step 2.
   - If the key you inserted has no PIN, but it is an M of N split, skip to step 3.
   - Otherwise, authentication is complete and the Luna PED returns control to the command interface.

     ```
     Command Result : No Error
     ```

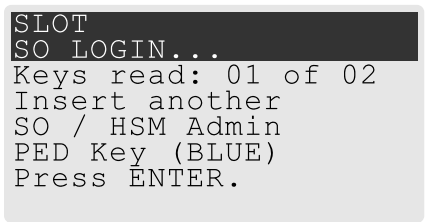
2. The PED prompts for the PIN. Enter the PIN on the keypad and press **Enter**.



   - If the key you inserted is an M of N split, continue to step 3.
   - Otherwise, authentication is complete and the PED returns control to the command interface.

     ```
     Command Result : No Error
     ```

3. The PED prompts for the next M of N split-secret key. Insert the next PED key and press **Enter**.



   - If the key you inserted has an associated PIN, return to step 2.
   - Repeat steps 2 and/or 3 until the requisite M number of keys have been presented to the Luna PED. At this point, authentication is complete and the PED returns control to the command interface.

     ```
     Command Result : No Error
     ```

> **NOTE** When authenticating an M of N split secret, the Luna USB HSM 7 cannot tell if an PED key PIN is entered incorrectly until the whole secret is reassembled. Therefore, PIN entry will appear to succeed and the authentication operation will only fail when all M PED keys have been presented.

Consequences of Losing PED keys

PED keys are the only means of authenticating roles, domains, and RPVs on the multifactor quorum-authenticated Luna Network HSM 7. Losing a keyset effectively locks the user out of that role. Always keep secure backups of your PED keys, including quorum (M of N) split secrets. Forgetting the PIN associated with a key is equivalent to losing the key entirely. Losing a split-secret key is less serious, unless enough splits are lost so that M cannot be satisfied.

If a PED key is lost or stolen, log in with one of your backup keys and change the existing PED key secret immediately, to prevent unauthorized HSM access.

The consequences of a lost PED key with no backup vary depending on the type of secret:

> "Blue HSM SO PED key" below
> "Red HSM Domain PED key" on the next page
> "Orange Remote PED key" on the next page
> "Blue Partition SO PED key" on the next page
> "Red Partition Domain PED key" on the next page
> "Black Crypto Officer PED key" on the next page
> "Gray Crypto User PED key" on page 297
> "White Audit User PED key" on page 297

Blue HSM SO PED key

If the HSM SO secret is lost, you can no longer perform administrative tasks on the HSM, including partition creation and client assignment. If you use the same blue SO key for your HSM backup partitions, the contents of the HSM SO space are unrecoverable. Take the following steps:

1. Contact all Crypto Officers and have them immediately make backups of their existing partitions at the client.
2. When all important partitions are backed up, execute a factory reset of the HSM.
3. Initialize the HSM and create a new HSM SO secret. Use the original red HSM cloning domain key.
4. Restore the HSM SO space contents from a recent backup, if you have one.
5. If you are using Remote PED, you must recreate the Remote PED Vector (RPV). Reuse the original orange key.
6. Recreate the partitions and reassign them to their respective clients.
7. Partition SOs must initialize the new partitions using their original blue and red key(s), and initialize the Crypto Officer role (and Activation secret, if applicable). Supply the new black CO keys to the Crypto Officers.
8. Crypto Officers must change the login credentials from the new black CO key to their original black keys (and reset the Activation secret password, if applicable).
9. Crypto Officers can now restore all partition contents from backup.

Red HSM Domain PED key

If the HSM Key Cloning Vector is lost, you can no longer perform backup/restore operations on the HSM SO space(s). If the HSM is factory-reset, the contents of the HSM SO space are unrecoverable. Follow the same procedure as you would if you lost the blue HSM SO key, but you cannot restore the HSM SO space from backup.

Orange Remote PED key

If the Remote PED Vector is lost, create a new one and distribute a copy to the administrator of each Remote PED server. See "Rotating or Re-Initializing the Orange Remote PED key" on page 262.

Blue Partition SO PED key

If the Partition SO secret is lost, you can no longer perform administrative tasks on the partition. Take the following steps:

1. Have the Crypto Officer immediately make a backup of the partition objects.
2. Have the HSM SO delete the partition, create a new one, and assign it to the same client.
3. Initialize the new partition with a new blue Partition SO key and the original red cloning domain key(s).
4. Initialize the Crypto Officer role (and Activation secret, if applicable). Supply the new black CO key to the Crypto Officer.
5. The Crypto Officer must change the login credentials from the new black CO key to their original black key (and reset the Activation secret password, if applicable).
6. The Crypto Officer can now restore all partition contents from backup.

Red Partition Domain PED key

If the Partition Key Cloning Vector is lost, you can no longer perform backup/restore operations on the partition (s), or make changes to HA groups in that cloning domain. You can still perform all other operations on the partition. Take the following steps:

1. Have the HSM SO create a new partition (or multiple partitions, to replace the entire HA group) and assign it to the same client(s).
2. Initialize the partition(s)with a new cloning domain.
3. Initialize the Crypto Officer role with the original black Crypto Officer key (and Activation password, if applicable).
4. Create objects on the new partition to replace those on the original partition.
5. As soon as possible, change all applications to use the objects on the new partition.
6. When objects on the original partition are no longer in production use, the HSM SO can delete the original partition.

Black Crypto Officer PED key

If the Crypto Officer secret is lost, you can no longer create objects on the partition, or perform backup/restore operations. You might still be able to use the partition, depending on the following criteria:

> **PIN reset by Partition SO:**

- If HSM policy **15: Enable SO reset of partition PIN** is set to **1**, the Partition SO can reset the Crypto Officer secret and create a new black CO key.

  lunacm:>**role resetpw -name co**

- If this policy is set to **0** (default), the CO is locked out unless other criteria in this list apply.

> **Partition Activation:**

- If the partition is Activated, you can still access it for production using the CO challenge secret. Change your applications to use objects on a new partition as soon as possible.
- If the partition is not Activated, read-only access of essential objects might still be available via the Crypto User role.

> **Crypto User**

- If the Crypto User is initialized, you can use the CU role for read-only access to essential partition objects while you change your applications to use objects on a new partition.

If none of these criteria apply, the contents of the partition are unrecoverable.

Gray Crypto User PED key

If the Crypto User secret is lost, the Crypto Officer can reset the CU secret and create a new gray key:

lunacm:>**role resetpw -name cu**

White Audit User PED key

If the Audit User secret is lost, you can no longer cryptographically verify existing audit logs or make changes to the audit configuration. The existing logs can still be viewed. Re-initialize the Audit User role on the affected HSMs, using the same white key for HSMs that will verify each other's logs.

Identifying the PED key Secret

You can use this procedure to identify the type of secret (role, domain, or RPV) stored on an unidentified PED key. This procedure will not tell you:

> identifying information about the HSM the key is associated with

> whether the key is part of an M of N scheme, or how many keys are in the set

> whether the key has a PIN assigned

> who the key belongs to

You require:

> Luna PED in Admin Mode (see "Changing Modes" on page 248)

> the key you want to identify

To identify the type of secret stored on an existing PED key

1. Insert the PED key you want to identify.
2. From the Admin mode menu, press **1** on the keypad to select the **PED Key** option.

```
Admin mode...
1  PED Key
5  Backup Devices
7  Software Update
9  Self Test

<  EXIT
```

3. From the PED Key mode menu, press **3** on the keypad to select the **List types** option.

```
PED Key mode
1  Login
3  List types



<  EXIT
```

The secret type is identified on-screen.

```
PED Key mode
Found keys:
Domain



Press ENTER.
```

Duplicating Existing PED keys

During the key creation process, you have the option to create multiple copies of PED keys. If you want to make backups of your keys later, you can use this procedure to copy PED keys. You require:

> Luna PED in Admin Mode (see "Changing Modes" on page 248)

> Enough blank or rewritable keys to make your copies

The PED key is duplicated exactly by this process. If there is a PIN assigned, the same PIN is assigned to the duplicate key. If the key is part of a quorum (M of N) scheme, the duplicates may not be used in the same login process to satisfy the M of N requirements. You must also have copies of the other keys in the quorum (M of N) keyset. See "Quorum Split Secrets (M of N)" on page 241.

To duplicate an existing PED key

1. Insert the PED key you want to duplicate. Have a blank or rewritable PED key ready.
2. From the Admin mode menu, press **1** on the keypad to login to the PED key.

```
PED Key mode
1  Login
3  List types



<  EXIT
```

3. Press **7** on the keypad and follow the on-screen instructions.

```
PED Key mode
          2 Logout
3 List types
7 Duplicate
< EXIT
```

Changing the PED key Secret

Use the instructions on this page to change/rotate the secrets on any of the indicated PED iKeys.

From time to time, it might be necessary to change the secret associated with a role on an HSM appliance, a role on a cryptographic module (HSM) or a partition of an HSM, or a cloning domain secret. Reasons for changing credentials include:

- Regular credential rotation as part of your organization's security policy
- Compromise of a role or secret due to loss or theft of a PED key
- Personnel changes in your organization or changes to individual security clearances
- Changes to your security scheme (implementing/revoking M of N, PINs, or shared secrets)

The procedure for changing a PED key credential depends on the type of key. Procedures for each type are provided below.

> **CAUTION!** If you are changing a multifactor quorum credential that is shared among multiple HSMs/partitions/roles, always keep at least one copy of the old keyset until the affected HSMs/partitions/roles are all changed to the new credential. When changing multifactor quorum credentials, you must always present the old keyset first; do not overwrite your old PED keys until you have no further need for them.

- "Blue HSM SO PED key" below
- "Red HSM Domain PED key" on the next page
- "Orange Remote PED Vector PED key" on the next page
- "Blue Partition SO PED key" on the next page
- "Red Partition Domain PED key" on the next page
- "Black Crypto Officer PED key" on page 301
- "Gray Crypto User PED key" on page 301
- "White Audit User PED key" on page 302

Blue HSM SO PED key

The HSM SO can use this procedure to change the HSM SO credential.

To change the blue HSM SO PED key credential

1. In LunaSH, log in as HSM SO.

   lunash:> **hsm login**

2. Initiate the PED key change.

   lunash:> **hsm changepw**

3. You are prompted to present the original blue PED key(s) and then to create a new HSM SO keyset. See "Creating PED keys" on page 287.

Red HSM Domain PED key

It is not possible to change an **HSM's cloning domain** without factory-resetting the HSM and setting the new cloning domain as part of the standard initialization procedure.

> **CAUTION!** If you set a different cloning domain for the HSM, you cannot restore the HSM SO space from backup.

Orange Remote PED Vector PED key

The HSM SO can use this procedure to change the Remote PED Vector (RPV) for the HSM.

To change the RPV/orange key credential

1. In LunaSH, log in as HSM SO.

   lunash:> **hsm login**

2. Initialize the RPV.

   lunash:> **hsm ped vector init**

   You are prompted to create a new Remote PED key. See "Creating PED keys" on page 287.

3. Distribute a copy of the new orange key to the administrator of each Remote PED server.

Blue Partition SO PED key

The Partition SO can use this procedure to change the Partition SO credential.

To change a blue Partition SO PED key credential

1. In LunaCM, log in as Partition SO.

   lunacm:> **role login -name po**

2. Initiate the PED key change.

   lunacm:> **role changepw -name po**

3. You are prompted to present the original blue key(s) and then to create a new Partition SO keyset. See "Creating PED keys" on page 287.

Red Partition Domain PED key

If you are using Luna HSM Firmware 7.7.2 and older, it is not possible to change a partition's cloning domain. A new partition must be created and initialized with the desired domain. The new partition will not have access to any of the original partition's backups. It cannot be made a member of the same HA group as the original.

Using Luna HSM Firmware 7.8.0 and newer, each partition can support up to three different cloning domains, allowing your sensitive keys and objects to remain within the cryptographic perimeter of the HSM while:

- migrating objects from one domain to another
- splitting domains
- rotating or rolling-over or refreshing your partition domain secrets as part of mandated periodic changes of credential/authentication, just as you would with passwords for
  - appliance administration (including network, logging, ntp. tamper response, etc.)
  - HSM or partition roles
    - container/partition administrative access
    - client access for crypto operations on keys and objects
  - etc.

To change the domain secret

See "Updating or rotating cloning domain secrets" on page 234.

Black Crypto Officer PED key

The Crypto Officer can use this procedure to change the Crypto Officer credential.

To change a black Crypto Officer PED key credential

1. In LunaCM, log in as Crypto Officer.

   lunacm:> **role login -name co**

2. Initiate the PED key change.

   lunacm:> **role changepw -name co**

3. You are prompted to present the original black key(s) and then to create a new Crypto Officer keyset. See "Creating PED keys" on page 287.

Gray Crypto User PED key

The Crypto User can use this procedure to change the Crypto User credential.

To change a gray Crypto User PED key credential

1. In LunaCM, log in as Crypto User.

   lunacm:> **role login-name cu**

2. Initiate the PED key change.

   lunacm:> **role changepw -name cu**

3. You are prompted to present the original gray key(s) and then to create a new Crypto User keyset. See "Creating PED keys" on page 287.

NOTE The Luna PED screen prompts for a black PED key for any of

- "User",
- "Crypto Officer",
- "Limited Crypto Officer",
- "Crypto User".

The Luna PED is not aware that the key you present has a black or a gray sticker on it. The colored stickers are visual identifiers for your convenience in keeping track of your PED keys. You differentiate by how you label, and how you use, a given physical key that the Luna PED sees as "black" (once it has been imprinted with a secret).

White Audit User PED key

The Audit User can use this procedure to change the Audit User credential.

To change the white Audit User PED key credential

1. Log into LunaSH as **audit**.
2. Log in as the Audit User.

   lunash:> **audit login**
3. Initiate the PED key change.

   lunash:> **audit changepwd**
4. You are prompted to present the original white key(s) and then to create a new Audit User keyset. See "Creating PED keys" on page 287.

PEDserver and PEDclient

You can use the **PEDserver** and **PEDclient** utilities to manage your remote PED devices.

The PEDserver Utility

PEDserver is required to run on any computer that has a Luna Remote PED attached, and is providing PED services.

The PEDserver utility has one function. It resides on a computer with an attached Luna PED (in Remote Mode), and it serves PED operations to an instance of PEDclient that operates on behalf of an HSM. The HSM could be local to the computer that has PEDserver running, or it could be on another HSM host computer at some distant location.

PEDserver can also run in peer-to-peer mode, where the server initiates the connection to the Client. This is needed when the Client (usually Luna Network HSM 7) is behind a firewall that forbids outgoing initiation of connections.

See "pedserver" on page 304.

The PEDclient Utility

PEDclient is required to run on any host of an HSM that needs to be served by a Remote Luna PED. PEDclient must also run on any host of a Remote Backup HSM that will be serving remote primary HSMs.

The PEDclient utility performs the following functions:

- It mediates between the HSM where it is installed and the Luna PED where PEDserver is installed, to provide PED services to the requesting HSM(s).
- It resides on a computer with RBS and an attached Luna Backup HSM, and it connects with another instance of PEDclient on a distant host of an HSM, to provide the link component for Remote Backup Service. See "Configuring a Remote Backup Server" on page 561 for more information.
- It acts as the logging daemon for HSM audit logs.

> **NOTE** PEDclient exists on the Luna Network HSM 7 appliance, but is not directly exposed. Instead, the relevant features are accessed via LunaSH **hsm ped** commands.

Thus, for example, in the case where an administrative workstation or laptop has both a Remote PED and a Remote Backup HSM attached, PEDclient would perform double duty. It would link with a locally-running instance of PEDserver, to convey HSM requests from the locally-connected Backup HSM to the locally-connected PED, and return the PED responses. As well, it would link a locally-running instance of RBS and a distant PEDclient instance to mediate Remote Backup function for that distant HSM's partitions.

See "pedclient" on page 319.

pedserver

Use the **pedserver** commands to manage certificates in PEDserver and the appliance, initiate connections between the Luna PEDand HSM, and select the PED for HSM operation.

To run PEDserver from the command line, you must specify one of the following three options.

Syntax

pedserver

-appliance
-mode
-regen

Option	Description
-appliance	Registers or deregisters an appliance, or lists the registered appliances. Applies to server-initiated (peer-to-peer) mode only. See "pedserver -appliance" on the next page.
-mode	Specifies the mode that the PED Server will be executed in. See "pedserver mode" on page 309.
-regen	Regenerates the client certificate. Applies to server-initiated (peer-to-peer) mode only. See "pedserver -regen" on page 319.

pedserver -appliance

Registers or deregisters an appliance, or lists the registered appliances. These commands apply to PED-initiated mode only.

Syntax

pedserver -appliance

delete
list
register

Option	Description
delete	Deregisters an appliance. See "pedserver -appliance delete" on the next page.
list	Lists the registered appliances. See "pedserver -appliance list" on page 307.
register	Registers an appliance. See "pedserver -appliance register" on page 308

pedserver -appliance delete

Deregister an appliance certificate from PEDserver.

Syntax

pedserver -appliance delete -name <unique name> [**-force**]

Option	Description
-name <unique name>	Specifies the name of the appliance to be deregistered from PEDserver.
-force	Optional parameter. Suppresses any prompts.

Example

```
C:\Program Files\Safenet\LunaClient>pedServer –appliance delete –name hello –force
```

pedserver -appliance list

Displays a list of appliances registered with PEDserver.

Syntax

pedserver -appliance list

Example

```
C:\Program Files\Safenet\LunaClient>pedServer -appliance list
>
Server Name          IP Address        Port Number     Certificate Common
                                                              Name
_____________________________________________________________________________
abox               192.20.1.23       9697              test2
bbox               192.20.12.34      9696              test1
hello              192.20.1.34       9876              hellocert
```

pedserver -appliance register

Register an appliance certificate with PEDserver.

Syntax

pedserver -appliance register -name <unique name> **-certificate** <appliance certificate file> **-ip** <appliance server IP address> [**-port** <port number>]

Option	Description
-name <unique name>	Specifies the name of the appliance to be registered to PEDserver.
-certificate <appliance certificate file>	Specifies the full path and filename of the certificate that was retrieved from the appliance.
-ip <appliance server IP address>	Specifies the IP address of the appliance server.
-port <port number>	Optional field. Specifies the port number used to connect to the appliance (directly or indirectly according to network configuration). **Range:** 0-65525

Example

```
C:\Program Files\Safenet\LunaClient>pedServer –appliance register –name hello -certificate the-best-appliance.pem -ip 123.321.123.321 –port 9697
```

pedserver mode

Specifies the mode that PEDserver will be executed in.

Syntax

pedserver -mode

config
connect
disconnect
show
start
stop

Option	Description
config	Modifies or shows existing configuration file settings. See "pedserver -mode config" on the next page.
connect	Connects to the appliance. See "pedserver -mode connect" on page 312.
disconnect	Disconnects from the appliance. See "pedserver -mode disconnect" on page 313.
show	Queries if PEDserver is currently running, and gets details about PEDserver. See "pedserver -mode show" on page 314.
start	Starts PEDserver. See "pedserver -mode start" on page 316.
stop	Shuts down PEDserver. See "pedserver -mode stop" on page 318

pedserver -mode config

Shows and modifies internal PEDserver configuration file settings.

Syntax

pedserver -mode config -name <registered appliance name> **-show -set** [**-port** <server port>] [**-set**][**-configfile** <filename>] [**-admin** <admin port number>] [**-eserverport** <0 or 1>] [**-eadmin** <0 or 1>] [**-idletimeout** <int>] [**-socketreadtimeout** <int>] [**-socketwritetimeout** <int>] [**-internalshutdowntimeout** <int>] [**-bgprocessstartuptimeout** <int>] [**-bgprocessshutdowntimeout** <int>] [**-logfilename** <filename>] [**-loginfo** <0 or 1>] [**-logwarning** <0 or 1>] [**-logerror** <0 or 1>] [**-logtrace** <0 or 1>] [**-maxlogfilesize** <size>] [**-pedwritedelay** <int>] [**-pinginterval** <int>] [**-pingtimeout** <int>]

Option	Description
-name <registered appliance name>	Specifies the name of the registered appliance to be configured.
-show	Displays the contents of the PEDserver configuration file.
-set	Updates the PEDserver configuration file to be up to date with other supplied options.
-port <server port>	Optional. Specifies the server port number.
-configfile <filename>	Optional. Specifies which PEDserver configuration file to use.
-admin <admin port number>	Optional. Specifies the administration port number.
-eserverport <0 or 1>	Optional. Specifies if the server port is on "localhost" or listening on the external host name.
-eadmin <0 or 1>	Optional. Specifies if the administration is on "localhost" or listening on the external host name.
-idletimeout <int>	Optional. Specifies the idle connection timeout, in seconds.
-socketreadtimeout <int>	Optional. Specifies the socket read timeout, in seconds.
-socketwritetimeout <int>	Optional. Specifies socket write timeout, in seconds.
-internalshutdowntimeout <int>	Optional. Specifies the shutdown timeout for internal services, in seconds.
-bgprocessstartuptimeout <int>	Optional. Specifies the startup timeout for the detached process, in seconds.
-bgprocessshutdowntimeout <int>	Optional. Specifes the shutdown timeout for the detached process, in seconds.

Option	Description
-logfilename <filename>	Optional. Specifies the log file name to which the logger should log messages.
-loginfo <0 or 1>	Optional. Specifies if the logger should log "info" messages. Set to 0 for no, 1 for yes.
-logwarning <0 or 1>	Optional. Specifies if the logger should log "warning" messages. Set to 0 for no, 1 for yes.
-logerror <0 or 1>	Optional. Specifies if the logger should log "error" messages. Set to 0 for no, 1 for yes.
-logtrace <0 or 1>	Optional. Specifies if the logger should log "trace" messages. Set to 0 for no, 1 for yes.
-maxlogfilesize <size>	Optional. Specifies the maximum log file size in KB.
-pedwritedelay <int>	Optional. Specifies the communications delay over USB between the Luna PED and PEDserver, in microseconds. Default value is 0 and maximum value is 10000. This option is available only if you are using a multifactor quorum-authenticated Luna HSM with Luna HSM Firmware 7.7.0 or newer.
-pinginterval <int>	Optional. Specifies the time interval between pink commands, in seconds.
-pingtimeout <int>	Optional. Specifies timeout of the ping response, in seconds.

Example

```
C:\Program Files\Safenet\LunaClient>pedServer -mode config -name hellohi -show
```

pedserver -mode connect

Connects to the appliance by retrieving information (IP address, port, PEDserver certificate) from the PEDserver configuration file.

If the running mode is legacy, an error is returned. **pedserver -mode connect** is not a valid command for legacy connections.

The **connect** command will try connecting to PEDclient 20 times before giving up.

Syntax

pedserver -mode connect -name <registered appliance name> [**-configfile** <filename>] [**-logfilename** <filename>] [**-loginfo** <0 or 1>] [**-logwarning** <0 or 1>] [**-logerror** <0 or 1>] [**-logtrace** <0 or 1>] [**-maxlogfilesize** <size>]

Option	Description
-name <registered appliance name>	Specifies the name of the registered appliance to be connected to PEDserver.
-configfile <filename>	Optional. Specifies which PEDserver configuration file to use.
-logfilename <filename>	Optional. Specifies the log file name to which the logger should log messages.
-loginfo <0 or 1>	Optional. Specifies if the logger should log "info" messages. Set to 0 for no, 1 for yes.
-logwarning <0 or 1>	Optional. Specifies if the logger should log "warning" messages. Set to 0 for no, 1 for yes.
-logerror <0 or 1>	Optional. Specifies if the logger should log "error" messages. Set to 0 for no, 1 for yes.
-logtrace <0 or 1>	Optional. Specifies if the logger should log "trace" messages. Set to 0 for no, 1 for yes.
-maxlogfilesize <size>	Optional. Specifies the maximum log file size in KB.

Example

```
C:\Program Files\Safenet\LunaClient>pedServer -mode connect -name hellohi
>Connecting to Luna SA. Please wait....
>Successfully connected to Luna SA.
```

pedserver -mode disconnect

Disconnects PEDserver from the appliance.

If the running mode is legacy, an error is returned. **pedserver -mode disconnect** is not a valid command for legacy connections.

Termination of the connection may take a few minutes.

Syntax

pedserver -mode disconnect -name <registered appliance name> [**-configfile** <filename>] [**-logfilename** <filename>] [**-loginfo** <0 or 1>] [**-logwarning** <0 or 1>] [**-logerror** <0 or 1>] [**-logtrace** <0 or 1>] [**-maxlogfilesize** <size>]

Option	Description
-name <registered appliance name>	Specifies the name of the registered appliance to be disconnected from PEDserver.
-configfile <filename>	Optional. Specifies which PEDserver configuration file to use.
-logfilename <filename>	Optional. Specifies the log file name to which the logger should log messages.
-loginfo <0 or 1>	Optional. Specifies if the logger should log "info" messages. Set to 0 for no, 1 for yes.
-logwarning <0 or 1>	Optional. Specifies if the logger should log "warning" messages. Set to 0 for no, 1 for yes.
-logerror <0 or 1>	Optional. Specifies if the logger should log "error" messages. Set to 0 for no, 1 for yes.
-logtrace <0 or 1>	Optional. Specifies if the logger should log "trace" messages. Set to 0 for no, 1 for yes.
-maxlogfilesize <size>	Optional. Specifies the maximum log file size in KB.

Example

```
C:\Program Files\Safenet\LunaClient>pedServer –mode disconnect -name hellohi
>Connection to Luna SA terminated.
```

pedserver -mode show

Queries if PEDserver is currently running, and gets details about PEDserver.

Syntax

pedserver -mode show [**-name** <registered appliance name>] [**-configfile** <filename>] [**-logfilename** <filename>] [**-loginfo** <0 or 1>] [**-logwarning** <0 or 1>] [**-logerror** <0 or 1>] [**-logtrace** <0 or 1>] [**-maxlogfilesize** <size>]

Option	Description
-name <registered appliance name>	Specifies the name of the registered appliance to be queried. Applies to server-initiated (peer-to-peer) mode only.
-configfile <filename>	Optional. Specifies which PEDserver configuration file to use.
-logfilename <filename>	Optional. Specifies the log file name to which the logger should log messages.
-loginfo <0 or 1>	Optional. Specifies if the logger should log "info" messages. Set to 0 for no, 1 for yes.
-logwarning <0 or 1>	Optional. Specifies if the logger should log "warning" messages. Set to 0 for no, 1 for yes.
-logerror <0 or 1>	Optional. Specifies if the logger should log "error" messages. Set to 0 for no, 1 for yes.
-logtrace <0 or 1>	Optional. Specifies if the logger should log "trace" messages. Set to 0 for no, 1 for yes.
-maxlogfilesize <size>	Optional. Specifies the maximum log file size in KB.

Example

```
C:\Program Files\Safenet\LunaClient>pedServer –mode show -name hellohi
>Ped Server launched in status mode.
   Server Information:
      Hostname:                          ABC1-123123
      IP:                                192.10.10.123
      Firmware Version:                  2.5.0-1
      PedII Protocol Version:            1.0.1-0
      Software Version:                  1.0.5 (10005)
      Ped2 Connection Status:            Connected
      Ped2 RPK Count                     1
      Ped2 RPK Serial Numbers            (1a123456789a1234)
   Client Information:                   Not Available
   Operating Information:
      Server Port:                       1234
      External Server Interface:         Yes
      Admin Port:                        1235
```

```
      External Admin Interface:           No
      Server Up Time:                     8 (secs)
      Server Idle Time:                   8 (secs) (100%)
      Idle Timeout Value:                 1800 (secs)
      Current Connection Time:            0 (secs)
      Current Connection Idle Time:       0 (secs)
      Current Connection Total Idle Time: 0 (secs) (100%)
      Total Connection Time:              0 (secs)
      Total Connection Idle Time:         0 (secs) (100%)
>Show command passed.
```

pedserver -mode start

Starts up PEDserver.

Syntax

pedserver -mode start [**-name** <registered appliance name>] [**-ip** <server_IP>] [**-port** <server port>] [**-configfile** <filename>] [**-admin** <admin port number>] [**-eserverport** <0 or 1>] [**-eadmin** <0 or 1>] [**-idletimeout** <int>] [**-socketreadtimeout** <int>] [**-socketwritetimeout** <int>] [**-internalshutdowntimeout** <int>] [**-bgprocessstartuptimeout** <int>] [**-bgprocessshutdowntimeout** <int>] [**-logfilename** <filename>] [**-loginfo** <0 or 1>] [**-logwarning** <0 or 1>] [**-logerror** <0 or 1>] [**-logtrace** <0 or 1>] [**-maxlogfilesize** <size>] [**-pinginterval** <int>] [**-pingtimeout** <int>] [**-force**]

Option	Description
-admin <admin port number>	Optional. Specifies the administration port number.
-bgprocessshutdowntimeout <int>	Optional. Specifes the shutdown timeout for the detached process, in seconds.
-bgprocessstartuptimeout <int>	Optional. Specifies the startup timeout for the detached process, in seconds.
-configfile <filename>	Optional. Specifies which PED Server configuration file to use.
-eadmin <0 or 1>	Optional. Specifies if the administration is on "localhost" or listening on the external host name.
-eserverport <0 or 1>	Optional. Specifies if the server port is on "localhost" or listening on the external host name.
-force	Optional parameter. Suppresses any prompts.
-idletimeout <int>	Optional. Specifies the idle connection timeout, in seconds.
-internalshutdowntimeout <int>	Optional. Specifies the shutdown timeout for internal services, in seconds.
-ip <server_IP>	Optional. Specifies the server listening IP address. When **running pedserver -mode start** on an IPv6 network, you must include this option.
-logerror <0 or 1>	Optional. Specifies if the logger should log "error" messages. Set to 0 for no, 1 for yes.
-logfilename <filename>	Optional. Specifies the log file name to which the logger should log messages.
-loginfo <0 or 1>	Optional. Specifies if the logger should log "info" messages. Set to 0 for no, 1 for yes.

Option	Description
-logtrace <0 or 1>	Optional. Specifies if the logger should log "trace" messages. Set to 0 for no, 1 for yes.
-logwarning <0 or 1>	Optional. Specifies if the logger should log "warning" messages. Set to 0 for no, 1 for yes.
-maxlogfilesize <size>	Optional. Specifies the maximum log file size in KB.
-name <registered appliance name>	
-pinginterval <int>	Optional. Specifies the time interval between pink commands, in seconds.
-pingtimeout <int>	Optional. Specifies timeout of the ping response, in seconds.
-port <server port>	Optional. Specifies the server port number.
-socketreadtimeout <int>	Optional. Specifies the socket read timeout, in seconds.
-socketwritetimeout <int>	Optional. Specifies socket write timeout, in seconds.

Example

```
C:\Program Files\Safenet\LunaClient>pedServer –mode start -name hellohi -force
>Ped Server launched in startup mode.
>Starting background process
>Background process started
>Ped Server Process created, exiting this process.
```

pedserver -mode stop

Stops PEDserver.

Syntax

pedserver -mode stop [**-name** <registered appliance name>] [**-configfile** <filename>] [**-socketwritetimeout** <int>] [**-internalshutdowntimeout** <int>] [**-bgprocessstartuptimeout** <int>] [**-bgprocessshutdowntimeout** <int>] [**-logfilename** <filename>] [**-loginfo** <0 or 1>] [**-logwarning** <0 or 1>] [**-logerror** <0 or 1>] [**-logtrace** <0 or 1>] [**-maxlogfilesize** <size>]

Option	Description
-name <registered appliance name>	Specifies the name of the registered appliance on which PEDserver will be stopped. Applies to server-initiated (peer-to-peer) mode only.
-configfile <filename>	Optional. Specifies which PEDserver configuration file to use.
-socketreadtimeout <int>	Optional. Specifies the socket read timeout, in seconds.
-socketwritetimeout <int>	Optional. Specifies socket write timeout, in seconds.
-internalshutdowntimeout <int>	Optional. Specifies the shutdown timeout for internal services, in seconds.
-bgprocessstartuptimeout <int>	Optional. Specifies the startup timeout for the detached process, in seconds.
-bgprocessshutdowntimeout <int>	Optional. Specifes the shutdown timeout for the detached process, in seconds.
-logfilename <filename>	Optional. Specifies the log file name to which the logger should log messages.
-loginfo <0 or 1>	Optional. Specifies if the logger should log "info" messages. Set to 0 for no, 1 for yes.
-logwarning <0 or 1>	Optional. Specifies if the logger should log "warning" messages. Set to 0 for no, 1 for yes.
-logerror <0 or 1>	Optional. Specifies if the logger should log "error" messages. Set to 0 for no, 1 for yes.
-logtrace <0 or 1>	Optional. Specifies if the logger should log "trace" messages. Set to 0 for no, 1 for yes.
-maxlogfilesize <size>	Optional. Specifies the maximum log file size in KB.

Example

```
C:\Program Files\Safenet\LunaClient>pedServer -mode stop -name hellohi
```

pedserver -regen

Regenerates the client certificate. This command is available in server-initiated (peer-to-peer) mode only.

Existing links (PEDserver, NTLS or STC) will not be affected until they are terminated. Afterward, the user is required to re-register the client certificate to NTLS and PEDserver.

> **NOTE** The **pedserver -regen** command should be used only when there is no Luna HSM Client installed. When Luna HSM Client is installed on the host computer, use the LunaCM command **clientconfig deploy** with the **-regen** option or, if necessary, **vtl createCert**.

Syntax

pedserver -regen -commonname <commonname> [**-force**]

Option	Description
-commonname <commonname>	The client's common name (CN).
-force	Optional parameter. Suppresses any prompts.

Example

```
C:\Program Files\SafeNet\LunaClient>pedServer -regen -commonname win2016_server -force
Ped Server Version 1.0.6 (10006)

Private Key created and written to: C:\Program Files\SafeNet\LunaClient\cert\client\win2016_
serverKey.pem
Certificate created and written to: C:\Program Files\SafeNet\LunaClient\cert\client\win2016_
server.pem

Successfully regenerated the client certificate.
```

pedclient

Use the **pedclient** commands to start, stop, and configure the PEDclient service.

Syntax

pedclient -mode

assignid
config
deleteid
releaseid
setid
show

start
stop
testid

Option	Description
assignid	Assigns a PED ID mapping to an HSM. See "pedclient -mode assignid" on the next page.
config	Modifies or shows existing configuration file settings. See "pedclient -mode config" on page 322.
deleteid	Deletes a PED ID mapping. See "pedclient -mode deleteid" on page 324.
releaseid	Releases a PED ID mapping from an HSM. See "pedclient -mode releaseid" on page 325.
setid	Creates a PED ID mapping. See "pedclient -mode setid" on page 326.
show	Queries if PEDclient is currently running and gets details about PEDclient. See "pedclient -mode show" on page 327.
start	Starts up PEDclient. See "pedclient -mode start" on page 328.
stop	Shuts down PEDclient. See "pedclient -mode stop" on page 330.
testid	Tests a PED ID mapping. See "pedclient -mode testid" on page 331.

pedclient -mode assignid

Assigns a PED ID mapping to a specified HSM.

Syntax

pedclient -mode assignid -id <pedid> **-id_serialnumber** <serial> [**-logfilename** <filename>] [**-loginfo** <0 or 1>] [**-logwarning** <0 or 1>] [**-logerror** <0 or 1>] [**-logtrace** <0 or 1>] [**-maxlogfilesize** <size>] [**-locallogger**]

Option	Description
-id <pedid>	Specifies the ID of the PED to be assigned.
-id_serialnumber <serial>	Specifies the serial number of the HSM to be linked to the specified PED ID.
-logfilename <filename>	Optional. Specifies the log file name to which the logger should log messages.
-loginfo <0 or 1>	Optional. Specifies if the logger should log "info" messages. Set to 0 for no, 1 for yes.
-logwarning <0 or 1>	Optional. Specifies if the logger should log "warning" messages. Set to 0 for no, 1 for yes.
-logerror <0 or 1>	Optional. Specifies if the logger should log "error" messages. Set to 0 for no, 1 for yes.
-logtrace <0 or 1>	Optional. Specifies if the logger should log "trace" messages. Set to 0 for no, 1 for yes.
-maxlogfilesize <size>	Optional. Specifies the maximum log file size in KB.
-locallogger	Optional. Specifies that the Remote PED logger should be used, not the IS logging system.

Example

```
C:\Program Files\Safenet\LunaClient>pedClient –mode assignid -id 1234 -id_serialnumber
123456789
```

pedclient -mode config

Modifies or shows existing configuration file settings.

Syntax

pedclient -mode config -show -set [**-eadmin** <0 or 1>] [**-idletimeout** <int>] **[-ignoreideltimeout]** [**-socketreadtimeout** <int>] [**-socketwritetimeout** <int>] [**-shutdowntimeout** <int>] [**-pstartuptimeout** <int>][**-pshutdowntimeout** <int>] [**-logfilename** <filename>] [**-loginfo** <0 or 1>] [**-logwarning** <0 or 1>] [**-logtrace** <0 or 1>] [**-maxlogfilesize** <size>] [**-locallogger**]

Option	Description
-show	Displays the contents of the configuration file.
-set	Updates the configuration file to be up to date with other supplied options.
-eadmin <0 or 1>	Optional. Specifies if the administration port is on "localhost" or on the external host name.
-idletimeout <int>	Optional. Specifies the idle connection timeout, in seconds.
-ignoreidletimeout	Optional. Specifies that the idle connection timeout should not apply to the connection established between the PED and HSM during their assignment.
-socketreadtimeout <int>	Optional. Specifies the socket read timeout, in seconds.
-socketwritetimeout <int>	Optional. Specifies the socket write timeout, in seconds.
-shutdowntimeout <int>	Optional. Specifies the shutdown timeout for internal services, in seconds.
-pstartuptimeout <int>	Optional. Specifies the startup timeout for the detached process, in seconds.
-pshutdowntimeout <int>	Optional. Specifies the shutdown timeout for the detached process, in seconds.
-logfilename <filename>	Optional. Specifies the log file name to which the logger should log messages.
-loginfo <0 or 1>	Optional. Specifies if the logger should log "info" messages. Set to 0 for no, 1 for yes.
-logwarning <0 or 1>	Optional. Specifies if the logger should log "warning" messages. Set to 0 for no, 1 for yes.
-logerror <0 or 1>	Optional. Specifies if the logger should log "error" messages. Set to 0 for no, 1 for yes.
-logtrace <0 or 1>	Optional. Specifies if the logger should log "trace" messages. Set to 0 for no, 1 for yes.

Option	Description
-maxlogfilesize <size>	Optional. Specifies the maximum log file size in KB.
-locallogger	Optional. Specifies that the Remote PED logger should be used, not the IS logging system.

Example

```
C:\Program Files\Safenet\LunaClient>pedClient –mode config -show
```

pedclient -mode deleteid

Deletes a PED ID mapping between a specified Luna PED and PEDserver.

Syntax

pedclient -mode deleteid -id <PED_ID> [**-logfilename** <filename>] [**-loginfo** <0 or 1>] [**-logwarning** <0 or 1>] [**-logerror** <0 or 1>] [**-logtrace** <0 or 1>] [**-maxlogfilesize** <size>] [**-locallogger**]

Option	Description
-id <PED_ID>	Specifies the ID of the PED to be deleted from the map.
-logfilename <filename>	Optional. Specifies the log file name to which the logger should log messages.
-loginfo <0 or 1>	Optional. Specifies if the logger should log "info" messages. Set to 0 for no, 1 for yes.
-logwarning <0 or 1>	Optional. Specifies if the logger should log "warning" messages. Set to 0 for no, 1 for yes.
-logerror <0 or 1>	Optional. Specifies if the logger should log "error" messages. Set to 0 for no, 1 for yes.
-logtrace <0 or 1>	Optional. Specifies if the logger should log "trace" messages. Set to 0 for no, 1 for yes.
-maxlogfilesize <size>	Optional. Specifies the maximum log file size in KB.
-locallogger	Optional. Specifies that the Remote PED logger should be used, not the IS logging system.

Example

```
C:\Program Files\Safenet\LunaClient>pedClient –mode deleteid -id 1234
```

pedclient -mode releaseid

Releases a PED ID mapping from the HSM it was assigned to.

Syntax

pedclient -mode releaseid -id <PED_ID> [**-logfilename** <filename>] [**-loginfo** <0 or 1>] [**-logwarning** <0 or 1>] [**-logerror** <0 or 1>] [**-logtrace** <0 or 1>] [**-maxlogfilesize** <size>] [**-locallogger**]

Option	Description
-id <PED_ID>	Specifies the ID of the PED to be released.
-logfilename <filename>	Optional. Specifies the log file name to which the logger should log messages.
-loginfo <0 or 1>	Optional. Specifies if the logger should log "info" messages. Set to 0 for no, 1 for yes.
-logwarning <0 or 1>	Optional. Specifies if the logger should log "warning" messages. Set to 0 for no, 1 for yes.
-logerror <0 or 1>	Optional. Specifies if the logger should log "error" messages. Set to 0 for no, 1 for yes.
-logtrace <0 or 1>	Optional. Specifies if the logger should log "trace" messages. Set to 0 for no, 1 for yes.
-maxlogfilesize <size>	Optional. Specifies the maximum log file size in KB.
-locallogger	Optional. Specifies that the Remote PED logger should be used, not the IS logging system.

Example

```
C:\Program Files\Safenet\LunaClient>pedClient –mode releaseid -id 1234
```

pedclient -mode setid

Creates a PED ID mapping between a specified Luna PED and PEDserver.

Syntax

pedclient -mode setid -id <PED_ID> **-id_ip** <hostname> **-id_port** <port> [**-logfilename** <filename>] [**-loginfo** <0 or 1>] [**-logwarning** <0 or 1>] [**-logerror** <0 or 1>] [**-logtrace** <0 or 1>] [**-maxlogfilesize** <size>] [**-locallogger**]

Option	Description
-id <PED_ID>	Specifies the ID of the PED to be mapped.
-id_ip <hostname>	Specifies the IP address or hostname of the PEDserver to be linked with the PED ID.
-id_port <port>	Specifies the PED Server port to be linked with the PED ID.
-logfilename <filename>	Optional. Specifies the log file name to which the logger should log messages.
-loginfo <0 or 1>	Optional. Specifies if the logger should log "info" messages. Set to 0 for no, 1 for yes.
-logwarning <0 or 1>	Optional. Specifies if the logger should log "warning" messages. Set to 0 for no, 1 for yes.
-logerror <0 or 1>	Optional. Specifies if the logger should log "error" messages. Set to 0 for no, 1 for yes.
-logtrace <0 or 1>	Optional. Specifies if the logger should log "trace" messages. Set to 0 for no, 1 for yes.
-maxlogfilesize <size>	Optional. Specifies the maximum log file size in KB.
-locallogger	Optional. Specifies that the Remote PED logger should be used, not the IS logging system.

Example

```
C:\Program Files\Safenet\LunaClient>pedClient –mode setid -id 1234 -id_ip myhostname -id_port 3456
```

pedclient -mode show

Queries if PEDclient is currently running and gets details about PEDclient.

Syntax

pedclient -mode show [**-admin** <admin port number>] [**-eadmin** <0 or 1>] [**-socketreadtimeout** <int>] [**-socketwritetimeout** <int>] [**-logfilename** <filename>] [**-loginfo** <0 or 1>] [**-logwarning** <0 or 1>] [**-logtrace** <0 or 1>] [**-maxlogfilesize** <size>] [**-locallogger**]

Option	Description
-admin <admin port number>	Optional. Specifies the administration port number to use.
-eadmin <0 or 1>	Optional. Specifies if the administration port is on "localhost" or on the external host name.
-socketreadtimeout <int>	Optional. Specifies the socket read timeout, in seconds.
-socketwritetimeout <int>	Optional. Specifies the socket write timeout, in seconds.
-logfilename <filename>	Optional. Specifies the log file name to which the logger should log messages.
-loginfo <0 or 1>	Optional. Specifies if the logger should log "info" messages. Set to 0 for no, 1 for yes.
-logwarning <0 or 1>	Optional. Specifies if the logger should log "warning" messages. Set to 0 for no, 1 for yes.
-logerror <0 or 1>	Optional. Specifies if the logger should log "error" messages. Set to 0 for no, 1 for yes.
-logtrace <0 or 1>	Optional. Specifies if the logger should log "trace" messages. Set to 0 for no, 1 for yes.
-maxlogfilesize <size>	Optional. Specifies the maximum log file size in KB.
-locallogger	Optional. Specifies that the Remote PED logger should be used, not the IS logging system.

Example

```
C:\Program Files\Safenet\LunaClient>pedClient –mode show
```

pedclient -mode start

Starts up the PEDclient.

Syntax

pedclient -mode start [**-winservice**] [**-eadmin** <0 or 1>] [**-idletimeout** <int>] [**-socketreadtimeout** <int>] [**-socketwritetimeout** <int>] [**-shutdowntimeout** <int>] [**-pstartuptimeout** <int>][**-pshutdowntimeout** <int>] [**-logfilename** <filename>] [**-loginfo** <0 or 1>] [**-logwarning** <0 or 1>] [**-logtrace** <0 or 1>] [**-maxlogfilesize** <size>] [**-locallogger**]

Option	Description
-winservice	Starts PEDclient for Windows service. The standard parameters used for **pedclient mode start** can be used for **pedclient mode start -winservice** as well.
-eadmin <0 or 1>	Optional. Specifies if the administration port is on "localhost" or on the external host name.
-idletimeout <int>	Optional. Specifies the idle connection timeout, in seconds.
-socketreadtimeout <int>	Optional. Specifies the socket read timeout, in seconds.
-socketwritetimeout <int>	Optional. Specifies the socket write timeout, in seconds.
-shutdowntimeout <int>	Optional. Specifies the shutdown timeout for internal services, in seconds.
-pstartuptimeout <int>	Optional. Specifies the startup timeout for the detached process, in seconds.
-pshutdowntimeout <int>	Optional. Specifies the shutdown timeout for the detached process, in seconds.
-logfilename <filename>	Optional. Specifies the log file name to which the logger should log messages.
-loginfo <0 or 1>	Optional. Specifies if the logger should log "info" messages. Set to 0 for no, 1 for yes.
-logwarning <0 or 1>	Optional. Specifies if the logger should log "warning" messages. Set to 0 for no, 1 for yes.
-logerror <0 or 1>	Optional. Specifies if the logger should log "error" messages. Set to 0 for no, 1 for yes.
-logtrace <0 or 1>	Optional. Specifies if the logger should log "trace" messages. Set to 0 for no, 1 for yes.
-maxlogfilesize <size>	Optional. Specifies the maximum log file size in KB.

Option	Description
-locallogger	Optional. Specifies that the Remote PED logger should be used, not the IS logging system.

Example

```
C:\Program Files\Safenet\LunaClient>pedClient –mode start
```

pedclient -mode stop

Shuts down PEDclient.

Syntax

pedclient -mode stop [**-eadmin** <0 or 1>] [**-socketreadtimeout** <int>] [**-socketwritetimeout** <int>] [**-shutdowntimeout** <int>] [**-pstartuptimeout** <int>][**-pshutdowntimeout** <int>] [**-logfilename** <filename>] [**-loginfo** <0 or 1>] [**-logwarning** <0 or 1>] [**-logtrace** <0 or 1>] [**-maxlogfilesize** <size>] [**-locallogger**]

Option	Description
-eadmin <0 or 1>	Optional. Specifies if the administration port is on "localhost" or on the external host name.
-socketreadtimeout <int>	Optional. Specifies the socket read timeout, in seconds.
-socketwritetimeout <int>	Optional. Specifies the socket write timeout, in seconds.
-shutdowntimeout <int>	Optional. Specifies the shutdown timeout for internal services, in seconds.
-pstartuptimeout <int>	Optional. Specifies the startup timeout for the detached process, in seconds.
-pshutdowntimeout <int>	Optional. Specifies the shutdown timeout for the detached process, in seconds.
-logfilename <filename>	Optional. Specifies the log file name to which the logger should log messages.
-loginfo <0 or 1>	Optional. Specifies if the logger should log "info" messages. Set to 0 for no, 1 for yes.
-logwarning <0 or 1>	Optional. Specifies if the logger should log "warning" messages. Set to 0 for no, 1 for yes.
-logerror <0 or 1>	Optional. Specifies if the logger should log "error" messages. Set to 0 for no, 1 for yes.
-logtrace <0 or 1>	Optional. Specifies if the logger should log "trace" messages. Set to 0 for no, 1 for yes.
-maxlogfilesize <size>	Optional. Specifies the maximum log file size in KB.
-locallogger	Optional. Specifies that the Remote PED logger should be used, not the IS logging system.

Example

```
C:\Program Files\Safenet\LunaClient>pedClient -mode stop
```

pedclient -mode testid

Tests a PED ID mapping between a specified Luna PED and PEDserver.

Syntax

pedclient -mode testid -id <PED_ID> [**-logfilename** <filename>] [**-loginfo** <0 or 1>] [**-logwarning** <0 or 1>] [**-logerror** <0 or 1>] [**-logtrace** <0 or 1>] [**-maxlogfilesize** <size>] [**-locallogger**]

Option	Description
-id <PED_ID>	Specifies the ID of the PED to be tested.
-logfilename <filename>	Optional. Specifies the log file name to which the logger should log messages.
-loginfo <0 or 1>	Optional. Specifies if the logger should log "info" messages. Set to 0 for no, 1 for yes.
-logwarning <0 or 1>	Optional. Specifies if the logger should log "warning" messages. Set to 0 for no, 1 for yes.
-logerror <0 or 1>	Optional. Specifies if the logger should log "error" messages. Set to 0 for no, 1 for yes.
-logtrace <0 or 1>	Optional. Specifies if the logger should log "trace" messages. Set to 0 for no, 1 for yes.
-maxlogfilesize <size>	Optional. Specifies the maximum log file size in KB.
-locallogger	Optional. Specifies that the Remote PED logger should be used, not the IS logging system.

Example

```
C:\Program Files\Safenet\LunaClient>pedClient –mode testid -id 1234
```


CHAPTER 8: Initializing an Application Partition

Before it can be used to store cryptographic objects or perform operations, an application partition must be initialized. Initialization is performed by the Partition Security Officer and sets the authentication credential. There are two scenarios where the Partition SO would initialize the partition:

> **Preparing a new partition:** On a new partition, initialization sets the Partition SO authentication credential, an identifying label for the partition, and the partition's cloning domain (see "Initializing a New Partition" below).

> **Erasing an existing partition:** The Partition SO can re-initialize a partition to erase all cryptographic objects and the Crypto Officer/Crypto User roles, and select a new partition label. The Partition SO credential and the cloning domain remain the same (see "Re-initializing an Existing Partition" on page 336).

Initializing a New Partition

Initializing an application partition for the first time establishes you as the Partition SO and sets a cloning domain for the partition. This procedure can be performed

> from an administrative connection to the network HSM appliance (via SSH) using Luna Shell (LunaSH) commands
  - using Luna Network HSM 7 Appliance Software 7.7.1 or newer, the administrator (HSM SO) can initialize the newly created partition, creating the PSO role
  - and then use the new PSO credential on that partition to initialize the Crypto Officer role), or

> from a registered client, with an NTLS or STC connection, using LunaCM commands.

Any subsequent *re*-initialization of an application partition is performed from the client.

The following attributes are set during a new partition initialization:

Partition Label	The label is a string that uniquely identifies this partition.
	In LunaSH, the partition label created during initialization must be 1-32 characters in length. The following characters are allowed:
	`abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789 !@#$%^*()-_=+ []{}/:',.~`
	Spaces are allowed; enclose the label in double quotation marks if it includes spaces.
	In LunaCM, the partition label created during initialization must be 1-32 characters in length. If you specify a longer label, it will automatically be truncated to 32 characters. The following characters are allowed:
	`abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789 !@#$%^&*()-_=+ []{}|/;:',.<>`~`
	Spaces are allowed; enclose the label in double quotation marks if it includes spaces.
	For more information, refer to "Name, Label, and Password Requirements" on page 379.

Partition SO credentials	For multifactor quorum-authenticated HSMs, create a new Partition SO (blue) PED key(set) or re-use an existing PED key(set) from a partition you want to share credentials with. If you are using multifactor quorum authentication, ensure that you have an authentication strategy before beginning. See "Multifactor Quorum Authentication" on page 236. For password-authenticated HSMs, specify the Partition SO password. Passwords and activation challenge secrets must be 8-255 characters in length. Spaces are allowed; to specify a password with spaces using command-line options, enclose the password in double quotation marks. The space character may not be used as the first character in a password. The following characters are allowed: `!#$%'()*+,-./0123456789:=? @ABCDEFGHIJKLMNOPQRSTUVWXYZ[]^_ abcdefghijklmnopqrstuvwxyz{}~` This character set is enforced when using Luna Appliance Software 7.9.0 or Luna HSM Client 10.8.0 or newer, and recommended for all previous versions. Previously-set passwords and challenge secrets are unaffected, but the new character set is enforced when these passwords are changed.

Cloning domain for the partition	The cloning domain is a shared identifier that makes cloning possible among a group of HSM partitions. The domain secret allows for two layers of cloning security:
	> The Partition SO determines which partitions can clone objects to each other by setting the same domain on the source and destination partitions.
	> The Crypto Officer for the partition must authorize the cloning operation.
	See "Domain Planning" on page 194 for more information.
	For multifactor quorum-authenticated HSMs, create a new Domain (red) PED key(set) or re-use an existing PED key(set) from another partition that this partition will clone objects with.
	For password-authenticated HSMs, create a new domain string or re-use an existing string from another partition that this partition will clone objects with.
	The domain string must be 1-128 characters in length. The following characters are allowed:
	`abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789 !@#$%^*-_=+[]{}()/:',.~`
	The following characters are problematic or invalid and must not be used in a domain string:
	`"&;<>?\`|`
	Spaces are allowed, as long as the leading character is not a space; to specify a domain string with spaces using the **-domain** option, enclose the string in double quotation marks.
	For password-authenticated HSMs, the domain string should match the complexity of the partition password.

Prerequisites

> The new partition must be created and visible in LunaSH if it is to be initialized on the Luna Network HSM 7 appliance, (Luna Network HSM 7 Appliance Software 7.7.1 and newer - see partition init).

> The new partition must be assigned to the client and visible in LunaCM if it is to be initialized from that client (see "Client-Partition Connections" on page 107).

> If you want to configure the partition's policies with a policy template using LunaCM, the template file must be available on the client (see "Setting Partition Policies Using a Template" on page 355).

> If you want to configure the partition's policies with a policy template using LunaSH on the appliance, the pre-edited template file must be uploaded to the appliance.

> Multifactor Quorum authentication: A local or remote Luna PED connection must be established (see "Local PED Setup" on page 250 or "About Remote PED" on page 252). Ensure that you have enough blue (Partition SO) and red (Domain) PED keys for your planned authentication scheme (see "Creating PED keys" on page 287).

To initialize a new application partition in LunaSH on the Luna Network HSM 7 appliance

The following steps assume that the Luna Network HSM 7 **admin** has created the partition (**partition create**).

> **CAUTION!** This command requires Luna Network HSM 7 Appliance Software 7.8.1 or newer. Do not attempt to use it to initialize an STC partition, or assigned clients will lose contact with the partition. The Partition SO must use LunaCM at the client for partition management.

1. In LunaSH, log in to the HSM as SO if you are not already logged in.

   lunash:> **hsm login**

2. Create the partition, if it has not already been created.

   lunash:> **partition create -partition** <partition name>

   Partition names created in LunaSH must be 1-32 characters in length. The following characters are allowed:
   `abcdefghijklmnopqurstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ 0123456789!@#$%^*()-_=+{}[]:',./~`
   Spaces are allowed; enclose the partition name in double quotes if it includes spaces.
   The following characters are not allowed: `&\|;<>`"?`
   No two partitions can have the same name.

3. Initialize the partition by specifying its partition name. You can specify an optional label for the initialized partition; if this is not specified, the label assigned will be the same as the partition name. To initialize the partition using a policy template, specify the template filename.

   - **Password authentication:** You can specify a Partition SO password and/or a domain string with the initialization command, or enter them when prompted.

     lunash:> **partition init -partition** <name> [**-label** <label>] [**-applytemplate** <template_file>] [**-password** <password>] [**-domain** <domain_string>]

   - **Multifactor Quorum authentication:**

     lunash:> **partition init -partition** <name> [**-label** <label>] [**-applytemplate** <template_file>]

     Respond to the Luna PED prompts to create the blue Partition SO key and the red domain key (see "Creating PED keys" on page 287).

4. After the partition is initialized and the PSO created, you can create the Crypto Officer role via lunash on the appliance or with lunacm on a registered client see "Initializing Crypto Officer and Crypto User Roles for an Application Partition" on page 368.

To initialize a new application partition using LunaCM on the Client

1. Launch LunaCM on the client workstation.
2. Set the active slot to the partition you want to initialize.

   lunacm:> **slot set -slot** <slot_number>

3. Initialize the partition by specifying an identifying label. To initialize the partition using a policy template, specify the path to the template file.

   - **Password authentication:** You can specify a Partition SO password and/or a domain string with the initialization command, or enter them when prompted.

     lunacm:> **partition init -label** <label> [**-applytemplate** <template_file>] [**-password** <password>] [**-domain** <domain_string>]

   - **Multifactor Quorum authentication:**

     lunacm:> **partition init -label** <label> [**-applytemplate** <template_file>]

     Respond to the Luna PED prompts to create the blue Partition SO key and the red domain key (see "Creating PED keys" on page 287).

Re-initializing an Existing Partition

The Partition SO can re-initialize an existing partition at any time. Re-initialization erases all cryptographic objects on the partition, and the login credentials for the Crypto Officer and Limited Crypto Officer and Crypto User roles. The Partition SO login credential and cloning domain are retained.

Prerequisites

- The partition must be already initialized.
- Back up any important cryptographic objects stored on the partition.
- [Multifactor Quorum authentication] A local or remote PED connection must be established (see "Local PED Setup" on page 250 or "About Remote PED" on page 252).

To re-initialize an existing application partition

1. Launch LunaCM on the client workstation.
2. Set the active slot to the partition you want to re-initialize.

   lunacm:> **slot set -slot** <slot_number>
3. Initialize the partition by specifying an identifying label. You must specify a label for the partition (the same label or a new one). You are prompted for the current Partition SO credential.

   lunacm:> **partition init -label** <label>

CHAPTER 9: Partition Capabilities and Policies

An application partition can be configured to provide a range of different functions. The Partition Security Officer can customize this functionality using partition policies. This configuration is governed by the following settings:

- **Partition Capabilities** are features of partition functionality that are inherited from the parent HSM policies (see HSM Capabilities and Policies). The HSM SO can configure HSM policies to allow or disallow partition capabilities. Some capabilities have corresponding modifiable partition policies.
- **Partition Policies** are configurable settings that allow the Partition Security Officer to modify the function of their corresponding capabilities.

The table below describes all partition capabilities, their corresponding policies, and the results of changing their settings. This section contains the following procedures:

- "Setting Partition Policies Manually" on page 353
- "Setting Partition Policies Using a Template" on page 355

> **NOTE** Regarding Capabilities and Policies - as a general rule, when firmware is updated, a given policy retains whatever value it had (default or your setting), before the update. Some firmware versions introduce new capabilities with their accompanying policies. The listed default setting of a policy is the expected setting
>
> - if the Capability and Policy was not in existence,
>   or
> - if the Policy was not changed,
>
> before a firmware update.
>
> Rolling back the HSM firmware or resetting the HSM to factory conditions restores that version's default settings for all policies.
>
> Applying a non-default policy setting should be [re-]done after updating firmware from factory settings.

> **TIP** The Partition Policy Template (PPT) feature is ***not intended*** as a way to reconfigure application partitions that are already initialized.
>
> It is for the consistent, orderly setup of a new partition or partitions, or of a previously existing partition that has been zeroized.

Destructive Policies

As a security measure, changing some partition policies forces deletion of all cryptographic objects on the partition. These policies are listed as **destructive** in the table below. Some policy changes are destructive in either direction (**OFF-to-ON** and **ON-to-OFF**), while others are destructive only in the direction resulting in lowered partition security.

Some destructiveness settings can be customized using a partition policy template to initialize the partition. Refer to "Editing a Partition Policy Template" on page 356 for details. All destructiveness information on this page reflects the default settings for each policy.

Use lunacm:> **partition showpolicies -verbose** to check whether the policy you want to enable/disable is destructive.

Policy descriptions and settings

#	Partition Capability	Partition Policy
0	**Enable private key cloning** Always **1**. This capability allows private keys to be cloned to another Luna HSM partition (required for backup and HA). **NOTE** The HSM SO can disable cloning for all partitions on the HSM by turning off **HSM policy 7:** Allow cloning. In this case, cloning is not possible on the partition, regardless of this capability/policy's setting. See also "Cloning vs Key Management" on page 353.	**Allow private key cloning** *Destructive OFF-to-ON* > **1** (default): The partition is capable of cloning private keys to another partition. This policy must be enabled to back up partitions or create HA groups. Public keys and objects can always be cloned, regardless of this policy's setting. > **0**: Private keys can never be cloned to another application partition. Partition policies **0** and **1** may not be set to **1** (ON) at the same time. **NOTE** Enabling or disabling this policy requires Luna HSM Firmware 7.1.0 or newer. Key attributes can be set modifiable, and a key can then be set with (for example) attribute -extractable (see cmu generatekeypair), but Partition Policies overrule object attributes; Cloning ON and Private Key Wrapping OFF would prevent export despite the attribute settings.

#	Partition Capability	Partition Policy
1	**Enable private key wrapping** Always **1**. This capability allows private keys to be encrypted (wrapped) and exported off the partition.	**Allow private key wrapping** *Destructive OFF-to-ON* > **1**: Private keys may be wrapped and saved to an encrypted file off the partition. Public keys and objects can always be wrapped and exported, regardless of this policy's setting. This applies to V0 partitions only; V1 partitions cannot enable this policy. > **0** (default): Private keys can never be wrapped and exported off the partition. Partition policies **0** and **1** may not be set to **1** (ON) at the same time. **NOTE** Enabling or disabling this policy requires Luna HSM Firmware 7.1.0 or newer. Key attributes can be set modifiable, and a key can then be set with (for example) attribute - extractable (see cmu generatekeypair), but Partition Policies overrule object attributes; Cloning ON and Private Key Wrapping OFF would prevent export despite the attribute settings.
2	**Enable private key unwrapping** Always **1**. This capability allows wrapped private keys to be imported to the partition.	**Allow private key unwrapping** *Not destructive* > **1** (default): Private keys can be unwrapped and stored on the partition. > **0**: Private keys cannot be unwrapped onto the partition.
3	**Enable private key masking** Private keys can be masked off the partition.	**Allow private key masking** *Destructive OFF-to-ON* > **1** (default for V1 partitions): Private keys can be masked off the partition. > **0** (default for V0 partitions): Private keys cannot be masked off the partition.

#	Partition Capability	Partition Policy
4	**Enable secret key cloning** Always **1**. This capability allows secret keys to be cloned to another Luna HSM partition (required for backup and HA). **NOTE** The HSM SO can disable cloning for all partitions on the HSM by turning off **HSM policy 7:** Allow cloning. In this case, cloning is not possible on the partition, regardless of this capability/policy's setting. See also "Cloning vs Key Management" on page 353.	**Allow secret key cloning** *Destructive OFF-to-ON* > **1** (default): Secret keys on the partition can be cloned to another partition. This is required for partition backup and HA groups. > **0**: Secret keys cannot be backed up, and will not be cloned to other HA group members.
5	**Enable secret key wrapping** Always **1**. This capability allows secret keys to be encrypted (wrapped) and exported off the partition.	**Allow secret key wrapping** *Destructive OFF-to-ON* > **1** (default): Secret keys can be wrapped and saved to an encrypted file off the partition. > **0**: Secret keys can never be wrapped and exported off the partition.
6	**Enable secret key unwrapping** Always **1**. This capability allows wrapped secret keys to be imported to the partition.	**Allow secret key unwrapping** *Not destructive* > **1** (default): Secret keys can be unwrapped and stored on the partition. > **0**: Secret keys cannot be unwrapped onto the partition.
7	**Enable secret key masking** Enable masking secret keys off the partition.	**Allow secret key masking** *Destructive OFF-to-ON* > **1** (default for V1 partitions): Secret keys can be masked and stored off the partition. > **0** (default for V0 partitions): Secret keys cannot be masked off the partition.

#	Partition Capability	Partition Policy
9	**Enable DigestKey** Always **1**. Enable the C_DigestKey function to hash a symmetric key and return the hash to the calling application. The hashing is a subset of steps performed in many HASH/HMAC-based KDFs. The HSM firmware checks the policy every time LUNA_DIGEST_KEY is called, and returns LUNA_RET_OPERATION_RESTRICTED if the policy is off. Only FIPS-compliant hashes are allowed, so the state of the policy does not affect overall FIPS compliance. **NOTE** DigestKey can allow replication of Key Derive Functions externally, permitting some keys to be derived outside the HSM. Requires Luna HSM Firmware 7.8.0 or newer.	**Allow DigestKey** *Destructive OFF-to-ON* > **1** : Key Derive Functions can be performed using DigestKey > **0** (default): Key Derive Functions cannot use DigestKey.
10	**Enable multipurpose keys** Always **1**. This capability allows keys that are created or unwrapped on the partition to have more than one of the following attributes enabled (set to **1**), and can therefore be used for multiple types of operation: > Encrypt/Decrypt > Sign/Verify > Wrap/Unwrap > Derive	**Allow multipurpose keys** *Destructive OFF-to-ON* > **1** (default): Keys that are created or unwrapped on the partition may be used for multiple operations. > **0**: Keys that are created or unwrapped on the partition may have only one of the affected attributes enabled. Thales recommends that you create keys with only the attributes required for their intended purpose. Disabling this policy enforces this rule on the partition. **NOTE** This policy does not affect Diffie-Hellman keys, which are always created with only Derive set to **1**.

#	Partition Capability	Partition Policy
11	**Enable changing key attributes** Always **1**. This capability allows the Crypto Officer to modify the following non-sensitive attributes of keys on the partition, changing key functions: > CKA_ENCRYPT > CKA_DECRYPT > CKA_WRAP > CKA_UNWRAP > CKA_SIGN > CKA_SIGN_RECOVER > CKA_VERIFY > CKA_VERIFY_RECOVER > CKA_DERIVE > CKA_EXTRACTABLE	**Allow changing key attributes** *Destructive OFF-to-ON* > **1** (default): The Crypto Officer can modify the non-sensitive attributes of keys on the partition. > **0**: Keys created on the partition cannot be modified.
15	**Allow failed challenge responses** Always **1**. This capability/policy applies to multifactor quorum-authenticated Luna Network HSM 7 only. It determines whether failed login attempts using a challenge secret count towards a partition lockout.	**Ignore failed challenge responses** *Destructive OFF-to-ON* > **1** (default): Failed challenge secret login attempts are not counted towards a partition lockout. Only failed PED key authentication attempts increment the counter. > **0**: Failed login attempts using either a PED key or a challenge secret will count towards a partition lockout. See "Activation on Multifactor Quorum-Authenticated Partitions" on page 373 and "Logging In to the Application Partition" on page 366 for more information.
16	**Enable operation without RSA blinding** Always **1**. RSA blinding is a technique that introduces random elements into the signature process to prevent timing attacks on the RSA private key. Some security policies may require this technique, but it does affect performance.	**Operate without RSA blinding** *Destructive OFF-to-ON* > **1** (default): The partition does not use RSA blinding. > **0**: The partition uses RSA blinding. Performance will be affected.

#	Partition Capability	Partition Policy
17	**Enable signing with non-local keys** Always **1**. Keys generated on the HSM have the attribute CKA_LOCAL=1. Keys that are imported (unwrapped) to the HSM have CKA_LOCAL=0. These attributes are maintained if keys are backed up or cloned to another HSM partition.	**Allow signing with non-local keys** *Not destructive* > **1** (default): Keys with attribute CKA_LOCAL=0 can be used for signing and their trust history is not assured. > **0**: Only keys with attribute CKA_LOCAL=1 can be used to sign data on the partition.
18	**Enable raw RSA operations** Always **1**. This capability enables the RSA mechanism CKM_RSA_X_509 on the partition, which allows weak encryption.	**Allow raw RSA operations** *Destructive OFF-to-ON* > **1** (default): The partition allows operations using the RSA mechanism CKM_RSA_X_509. > **0**: Operations using CKM_RSA_X_509 are blocked on the partition.
20	**Max failed user logins allowed** Displays the maximum number of failed partition login attempts (**10**) before the partition is locked out (see "Logging In to the Application Partition" on page 366).	**Max failed user logins allowed** *Not destructive* The Partition SO can lower the effective number of failed logins below the maximum if desired. Default: **10**
21	**Enable high availability recovery** Always **1**. This capability enables the RecoveryLogin feature on the partition. This feature allows other HA group members to restore the login state of the partition in the event of a power outage or other such deactivation.	**Allow high availability recovery** *Not destructive* > **1** (default): RecoveryLogin is enabled on the partition. This feature must be configured in advance (see role recoveryinit and role recoverylogin). > **0**: RecoveryLogin is disabled on the partition.
22	**Enable activation** This capability allows the partition to be activated. See "Activation on Multifactor Quorum-Authenticated Partitions" on page 373. > **1**: Always enabled on PED-authenticated HSMs. > **0**: Always disabled on password-authenticated HSMs.	**Allow activation** *Not destructive* > **1**: The black and/or gray PED keysecrets can be encrypted and cached, so that only a keyboard-entered challenge secret is required to log in. > **0** (default): PED keys must be presented at each login, whether via LunaCM or a client application. This policy is overridden and activation is disabled if a tamper event occurs, or if an uncleared tamper event is detected on reboot. See Tamper Events for more information.

#	Partition Capability	Partition Policy
23	**Enable auto-activation** This capability allows the partition to remain activated for up to two hours if the Luna Network HSM 7 loses power. See "Activation on Multifactor Quorum-Authenticated Partitions" on page 373. > **1**: Always enabled on multifactor quorum-authenticated HSMs. > **0**: Always disabled on password-authenticated HSMs.	**Allow auto-activation** *Not destructive* > **1**: Partition activation (see policy 22 above) is maintained after an HSM power loss of up to two hours. > **0** (default): The partition is deactivated in the event of a power loss. When power is restored, the black and/or gray PED keys must be presented to re-activate the partition.
25	**Minimum PIN length** > Luna HSM Firmware 7.7.2 and newer: Always **247** (**8** characters). > Luna HSM Firmware 7.7.1 and older: Always **248** (**7** characters). The absolute minimum length for a role password/challenge secret. This is displayed as a value subtracted from 255. The reason for this inversion is that a policy can only be set to a value equal to or lower than the value set by its capability. If the absolute minimum length was set to 8, the Partition SO would be able to set the preferred minimum to 2, a less-secure policy. The Partition SO may only change the minimum length to increase security by forcing stronger passwords.	**Minimum PIN length** *Not destructive* The Partition SO can choose to increase the effective minimum length of a role password/challenge secret by setting this policy. The policy value is determined as follows: Subtract the desired minimum length from 255 (the absolute maximum length), and set policy 25 to that value. **255 - (desired length) = (policy value)** For example, to set the minimum length to 10 characters, set the value of this policy to 245: **255 - 10 = 245** Default: > Luna HSM Firmware 7.7.2 and newer: **247** (**8** characters). > Luna HSM Firmware 7.7.1 and older: **248** (**7** characters).
26	**Maximum PIN length** Always **255**. The absolute maximum length for a role password/challenge secret is 255 characters.	**Maximum PIN length** *Not destructive* The effective maximum role password/challenge secret length may be changed by the Partition SO. It must always be greater than or equal to the effective minimum length, determined by the formula described in policy 25 (above). Default: **255**

#	Partition Capability	Partition Policy
28	**Enable Key Management Functions** Always **1**. This capability allows cryptographic objects to be created, deleted, generated, derived, modified on the partition. See also "Cloning vs Key Management" on page 353.	**Allow Key Management Functions** *Destructive OFF-to-ON* > **1** (default): The Crypto Officer can manage (create/delete, etc.) objects on the partition. The Crypto User is restricted to read-only operations. > **0**: Partition objects are read-only for both the CO and CU roles.
29	**Enable RSA signing without confirmation** Always **1**. This capability governs the HSM's internal signing verification.	**Perform RSA signing without confirmation** *Destructive OFF-to-ON* > **1** (default): No internal signing verification is performed. > **0**: The HSM performs an internal verification of signing operations to validate the signature. This has a performance impact on signature operations.
31	**Enable private key unmasking** Always **1**. Private keys can be unmasked onto the partition.	**Allow private key unmasking** *Not destructive* > **1** (default for V1 partitions): Private keys can be unmasked onto the partition (meaning they also can be migrated from legacy Luna HSMs that used SIM). > **0** (default for V0 partitions): Private keys cannot be unmasked onto the partition (meaning that migration of private keys from legacy HSMs using SIM is also not possible).
32	**Enable secret key unmasking** Enable unmasking of a secret key onto the partition.	**Allow secret key unmasking** *Not destructive* > **1** (default for V1 partitions): Secret keys can be masked and stored onto the partition. > **0** (default for V0 partitions): Secret keys cannot be masked onto the partition.

#	Partition Capability	Partition Policy
33	**Enable RSA PKCS mechanism** Always **1**. The mechanism CKM_RSA_PKCS has known weaknesses, which you can address in your applications. If you are not prepared to address these issues, you can choose to disable the mechanism entirely.	**Allow RSA PKCS mechanism** *Destructive OFF-to-ON* > **1** (default): CKM_RSA_PKCS is enabled on the partition. Using Luna HSM Firmware 7.8.4 or newer, when the partition is in FIPS approved configuration (**HSM policy 12:** Allow non-FIPS algorithms or **partition policy 43:** "Allow non-FIPS algorithms" on page 350 set to **OFF**/**0**), the mechanism is disabled even if this policy is set to **1**. > **0**: CKM_RSA_PKCS is disabled on the partition.
34	**Enable CBC-PAD (un)wrap keys of any size** Always **1**. There are known vulnerabilities using small keys wrapped/unwrapped with CBC_PAD mechanisms (and with small keys in general). You can choose to enforce a size restriction so that small weak keys cannot be unwrapped onto the partition. The following mechanisms are affected: > CKM_AES_CBC_PAD > CKM_AES_CBC_PAD_IPSEC > CKM_ARIA_CBC_PAD > CKM_ARIA_L_CBC_PAD > CKM_CAST3_CBC_PAD > CKM_CAST5_CBC_PAD > CKM_DES_CBC_PAD > CKM_DES3_CBC_PAD > CKM_DES3_CBC_PAD_IPSEC > CKM_RC2_CBC_PAD > CKM_RC5_CBC_PAD > CKM_SEED_CBC_PAD > CKM_SM4_CBC_PAD	**Allow CBC-PAD (un)wrap keys of any size** *Destructive OFF-to-ON* > **1** (default): All keys can be wrapped or unwrapped using CBC_PAD mechanisms. > **0**: Only keys that are a multiple of 64 bits (8 bytes) can be wrapped or unwrapped using CBC_PAD mechanisms.

#	Partition Capability	Partition Policy
37	**Enable Secure Trusted Channel** Always **1**. This capability allows the partition to use STC for client access. **NOTE** If you are using Luna HSM Firmware 7.4.2 or older, the HSM SO must first enable STC by turning on **HSM policy 39:** Allow Secure Trusted Channel. This is not required using Luna HSM Firmware 7.7.0 or newer; STC is always enabled.	**Force Secure Trusted Channel** *Destructive ON-to-OFF* > **1**: If this policy is set, STC is used for all client access to this partition. You must first set up and register the STC identities (see "Creating an STC Connection" on page 128). > **0** (default): NTLS is used by default for client access to this partition , but STC can be used if desired.
39	**Enable Start/End Date Attributes** Always **1**. This capability allows you to enforce the CKA_START_DATE and CKA_END_DATE attributes of public, private, and secret partition objects for encrypt, sign, and wrap operations.	**Allow Start/End Date Attributes** *Destructive ON-to-OFF* > **1**: CKA_START_DATE and CKA_END_DATE attributes are enforced for all public, private, and secret partition objects for encrypt, sign, and wrap operation. > **0** (default): These attributes can be set, but their values are ignored.
40	**Enable Per-Key Authorization Data** Both assigned and unassigned secret keys (symmetric or private) are given per-key authorization attributes in the form of CKA_AUTH_DATA, in any newly created or upgraded Luna HSM Firmware 7.7.0 or newer partition. For V0 partitions PKA is ignored and applications can use the pre-existing APIs as before. For V1 partitions it is actively used, for eIDAS compliance with newer API.	**Require Per-Key Authorization Data** *Destructive ON-to-OFF and OFF-to-ON* > **1** (default for V1 partitions): Per-Key-Authorization is on by default, but can be turned off for performance. > **0** (default for V0 partitions): Per-Key-Authorization is off by default, and cannot be turned on - V0 partitions do not allow policy changes that would require new clients.

#	Partition Capability	Partition Policy
41	**Enable Partition Version** Always **1**. This capability is visible for any partition at Luna HSM Firmware 7.7.0 or newer, and allows you to switch a partition between version V0 and V1.	**Partition Version** *Destructive ON-to-OFF* > **1**: Version 1 (V1) partition supports all features of Luna HSM Firmware 7.7.0 or newer. • cloning is used/permitted only for SMKs • key objects are transferred using SKS • only HA Login version 2 is supported. > **0** (default): Version 0 (V0) supports older API and your pre-existing applications (used with Luna HSM Firmware 7.7.0), enhanced by fixes and security updates of Luna HSM Firmware 7.7.0 (or newer), but Per Key Authorization, SKS, and other V1-dependent features are not available. Pre-7.7.0 version of HA Login can be used (full use of v1.1 or version 2.0, while v1.0 HA Login for use as primary only)

#	Partition Capability	Partition Policy
42	**Enable CPv1** This capability is visible for any partition at Luna HSM Firmware 7.7.1 or newer, and allows the partition to use the cloning protocol needed for HA with older partitions.	**Allow CPv1** *Destructive OFF-to-ON* For V0 partitions created while the HSM is at Luna HSM Firmware 7.7.1 or newer. This policy was added in order to reintroduce CPv1 ability for cloning keys from Luna on-premises HSM to Luna Cloud HSM, or to other on-premises HSMs with lower firmware versions. When the HSM is in non-FIPS 140 approved configuration (formerly FIPS mode) (where HSM policy 12 is set to ON) > **1**: Cloning (CPv1) can be used by the partition for key objects. > **0** (default): Cloning (CPv1) is not allowed by the partition for key objects. When the HSM is in FIPS 140 approved configuration (formerly FIPS mode where HSM policy 12 is set to OFF), this policy setting cannot be changed. Cloning (CPv1) is not allowed, by the partition, for key objects. Therefore, cloning can use only CPv3 if it is available on both source and target. For Luna HSM Firmware 7.7.0 V0 partitions, "Allow CPv1" is OFF after firmware update from version 7.7.0 to Luna HSM Firmware 7.7.1 or newer. For Luna HSM Firmware 7.7.0 V1 partitions, "Allow CPv1" is always OFF. For pre-7.7.0 firmware partitions, CPv1 is turned ON after update (to firmware version 7.7.1 or newer), if the HSM is not in FIPS 140 approved configuration (formerly FIPS mode), OFF if the HSM is in FIPS 140 approved configuration (formerly FIPS mode). To back up objects from a partition with firmware older than Luna HSM Firmware 7.7.0 and restore them to a V0 partition with CPv1 allowed, you require Luna Backup HSM 7 Firmware 7.7.2 or newer. Enabling CPv1 disables CPv3 and CPv4.

#	Partition Capability	Partition Policy
43	**Enable non-FIPS algorithms** This capability is visible for any partition at Luna HSM Firmware 7.7.1 or newer, and allows the use of algorithms that are FIPS non-compliant, within the current partition. **The ability to modify this partition-level capability and policy is driven by the HSM-level FIPS policy** (see Allow non-FIPS algorithms), and requires that HSM policy 12 be set to ON. That is, with policy 12 ON for the HSM, *the overall HSM allows non-FIPS algorithms*, so you can then choose to allow or disallow non-FIPS 140 approved configuration *on an individual partition-by-partition basis* within that HSM by using this partition policy 43). If HSM-level policy 12 is OFF, then *this* partition-level policy cannot be modified.	**Allow non-FIPS algorithms** *Destructive OFF-to-ON* For V0 partitions created while the HSM is at Luna HSM Firmware 7.7.1 or newer When the HSM is not in FIPS 140 approved configuration (formerly FIPS mode) where HSM policy 12 is set to ON) > **1** (default): Non-FIPS-compliant algorithms can be used by the partition. > **0**: Non-FIPS-compliant algorithms are not permitted. When the HSM is in FIPS 140 approved configuration (where HSM policy 12 is set to OFF), this policy setting cannot be changed; non-FIPS-compliant algorithms are not permitted on any partition on the HSM. For Luna HSM Firmware 7.7.0 V0 partitions, "Allow non-FIPS" is OFF after firmware update from version 7.7.0 to Luna HSM Firmware 7.7.1 or newer. For Luna HSM Firmware 7.7.0 V1 partitions, "Allow non-FIPS" follows the HSM policy (on if on, off if off). For pre-7.7.1 firmware partitions, this partition policy follows the HSM policy.

#	Partition Capability	Partition Policy
44	**Enable Extended Domain Management** This capability is visible for any partition at Luna HSM Firmware 7.8.0 or newer, and allows multifactor quorum-authenticated HSMs to inter-operate with password-authenticated HSMs -- which also allows multifactor quorum-authenticated HSMs to inter-operate with Luna Cloud HSM services.	**Allow Extended Domain Management** *Not destructive* > **1** : Partition cloning/security domains, in addition to the original, can be specified and accessed by the partition. > **0** (default): For newly created partitions, the policy is off, and only the primary/native domain is available. For partitions that existed prior to upgrading to Luna HSM Firmware 7.8.0 or newer, the policy is off. This ensures that continuity of existing applications and processes requires no change due to firmware update. Changes come into play only when you set this policy to ON. When enabled, this policy provides the ability to specify the source of a partition's domain and the ability to have *more than one domain* on a partition. See the lunacm commands partition domainlist, partition domainadd, partition domaindelete, and partition domainchangelabel. When the policy is turned OFF, all domains except the primary domain are deleted, which also happens if you roll back to a firmware version that doesn't support multiple domains. This policy is non-destructive of partition contents when switched from OFF-to-ON or from ON-to-OFF, with that one exception of the non-primary domains. The destructiveness for either transition *can* be adjusted by the use of Partition Policy Template (PPT) at partition initialization time, but going from ON to OFF still loses any non-primary domains, while other aspects/contents of a partition can remain intact.

#	Partition Capability	Partition Policy
45	**Enable ECDSA/RSA Prehash SigVer** This capability is visible for any partition at Luna HSM Firmware 7.9.0 or newer, and enables the use of prehash implementation allowing mechanisms without a hash function to run verification. This capability is ON by default, so that the associated policy can always be modified. The following mechanisms are affected: > CKM_DSA > CKM_ECDSA > CKM_RSA_PKCS > CKM_RSA_PKCS_PSS > CKM_RSA_X9_31	**Allow ECDSA/RSA Prehash SigVer** *Not destructive* This policy controls whether mechanisms that do not include their own hash function can perform verification by using a prehash. > **1** (default): Prehash is implemented. > **0** : Prehash is not available.

A number of partition capabilities are linked to the corresponding HSM capabilities and policies including:

> Partition Policy (0) Enable private key cloning is dependent on HSM Policy (7) Allow cloning;

> Partition Policy (3) Enable private key masking is dependent on HSM Policy (6) Allow Masking;

> Partition Policy (4) Enable secret key cloning is dependent on HSM Policy (7) Allow cloning;

> Partition Policy (7) Enable secret key masking is dependent on HSM Policy (6) Allow Masking;

> Partition Policy (22) Enable Activation and Partition Policy (23) Enable Auto-Activation are dependent on HSM Policy (1) Allow PED-based authentication;

> Partition Policy (31) Enable private key unmasking is dependent on HSM Policy (6) Allow Masking; and

> Partition Policy (32) Enable secret key unmasking is dependent on HSM Policy (6) Allow Masking.

In addition – the following dependencies within the partition level policies are observed:

> Partition Policy (7) Allow cloning cannot be enabled at the same time as Partition Policy (1) Allow private key wrapping;

> Partition Policy (1) Allow private key wrapping cannot be enabled at the same time as either one of the policies, Partition Policy (0) Enable private key cloning, Partition Policy (3) Allow private key masking, Partition Policy (31) Enable private key unmasking;

> Partition Policy (23) Allow Auto-Activation is dependent on Partition Policy (22) Allow Activation being enabled;

> Partition Policies related to ‘Masking’ (3, 7, 31 and 32) can only be enabled when Partition Policy (41) Partition Version is ‘0’; and

> Partition Policy (41) Partition Version cannot be set to ‘1’ at the same time as either Partition Policy (40) Enable Per-Key Authorisation Data or any of the Partition Policies covering key masking (3, 7, 31 and 32).

> Partition Policy (40) Enable Per-Key Authorisation Data is enabled by default but is disabled if Partition Policy (41) Partition Version is set to ‘0’.

> Partition Policy (42) Allow CPv1 requires that HSM Policy (12) Allow non-FIPS algorithms is ON, and Partition Policy (43) Allow non-FIPS algorithms is on, setting the partition to FIPS non-approved mode.

> **NOTE** With the ***HSM*** in FIPS 140 approved configuration (formerly FIPS mode) then, at partition level:
>
> > cloning (CPV1) is not allowed, and
>
> > FIPS 140 approved configuration (formerly FIPS mode) cannot be turned off per partition.
>
> This is to prevent keys/objects in a more secure container from being transferred to a less-secure container.

Cloning vs Key Management

> **TIP** **Security Note -***Cloning policies (**0** and **4**)* permit or deny the ability to securely copy keys and objects into and out of a partition.
> The *Key Management Functions policy (**28**)* controls the ability to create, delete, generate, derive, or modify cryptographic objects in the current partition.
>
> These controls are independent of each other. With Key Management functions denied, you can still clone objects in and out of partitions where Cloning policy is allowed. Thus HA (high availability) operation can clone keys into a partition that disallows Key Management functions (creation, deletion, etc.). **Cloning a key or object into a partition is not considered creation** - the key or object already existed within the security / cloning domain that encompasses the partition.
>
> Ultimately the security administrators define where keys can exist by controlling distribution of the security / cloning domain, and by defining policies around those keys.
>
> Additionally, key owners can choose to make their keys non-modifiable and non-extractable, if those options are indicated by your use-case.

Setting Partition Policies Manually

The Partition Security Officer can change available policies to customize partition functionality. Policy settings apply to all roles/objects on the partition. Refer to "Partition Capabilities and Policies" on page 337 for a complete list of partition policies and their effects. In most cases, partition policies are either enabled (**1**) or disabled (**0**), but some allow a range of values.

To change multiple policy settings during partition initialization, see "Setting Partition Policies Using a Template" on page 355.

See also "Configuring the Partition for Cloning or Export of Private/Secret Keys" on page 358.

> "Setting Partition Policies Using LunaCM on the Luna HSM Client" below

> "Setting Partition Policies Using LunaSH on the Luna Network HSM 7" on the next page

Setting Partition Policies Using LunaCM on the Luna HSM Client

You can use LunaCM on the Luna HSM Client to set policies on an initialized partition.

Prerequisites

> The partition must be initialized (see "Initializing an Application Partition" on page 332).

> If you are changing a destructive policy, back up any important cryptographic objects (see "Partition Backup and Restore" on page 467).

To manually set or change a partition policy using LunaCM on the Luna HSM Client

1. Launch LunaCM and set the active slot to the partition.

   lunacm:> **slot set -slot** <slotnum>

2. [Optional] Display the existing partition policy settings.

   lunacm:> **partition showpolicies**

3. Log in as Partition SO (see "Logging In to the Application Partition" on page 366).

   lunacm:> **role login -name po**

4. Change the policy setting by specifying the policy number and the desired value (**0**, **1**, or a number in the accepted range for that policy). You can specify multiple policy changes in the same command by using comma-separated lists (for example, **-policy 33,37,40 -value 0,1,1**).

   lunacm:> **partition changepolicy -policy** <policy_ID> **-value** <value>

   If you are changing a destructive policy, you are prompted to enter **proceed** to continue the operation.

   > **NOTE** If you are running more than one LunaCM session against the same partition, and change a partition policy in one LunaCM session, the new policy setting is visible in that session only (although it is in effect). You must exit and restart the other LunaCM sessions to display the new policy setting.

Setting Partition Policies Using LunaSH on the Luna Network HSM 7

You can use LunaSH on the Luna Network HSM 7 appliance to set policies on an initialized partition.

To manually set or change a partition policy using LunaSH on the Luna Appliance Software

1. Log in to LunaSH as **admin** or **operator**, or a custom user with access to the next command (see Logging In to LunaSH).

2. Change the policy setting by specifying the partition, policy number, the desired value (**0**, **1**, or a number in the accepted range for that policy), and the Partition SO password (for multifactor quorum-authenticated partitions, the Luna PED prompts for the Partition SO credential).

   lunash:> partition changePolicy **-partition** <name> **-policy** <policy#> **-value** <value> **-psopin** <PSO_password>

   > **NOTE** This command requires Luna Network HSM 7 Appliance Software 7.8.1 or newer. It cannot be used on STC partitions; the Partition SO must use LunaCM at the client for partition management.

Setting Partition Policies Using a Template

A partition policy template (PPT) is a file containing a set of preferred partition policy settings, used to initialize partitions with those settings. That is, PPT is a way to rapidly and consistently deploy, or redeploy multiple application partitions, where you want all of them to have the same configuration. or when you might replace an HSM or partition and enforce that it has all the same settings as previously.

> **TIP** The Partition Policy Template (PPT) feature is ***not intended*** as a way to reconfigure application partitions that are already initialized.
>
> It is for the consistent, orderly setup of a new partition or partitions, or of a previously existing partition that has been zeroized.

You can use the same file to initialize multiple partitions, rather than changing policies manually after initialization. This can save time and effort when initializing partitions that are to function as an HA group, or must comply with your company's overall security strategy. Templates enable scalable policy management and simplify future audit and compliance requirements.

> **NOTE** This feature is for partitions in on-premises, physical Luna HSMs, and is not supported for Luna Cloud HSM (DPoD) services.

See also Setting HSM Policies Using a Policy Template.

> **NOTE** This feature requires minimum Luna HSM Firmware 7.1.0 and Luna HSM Client 7.1.0.

You can create a partition policy template file from an initialized or uninitialized partition, and edit it using a standard text editor. Partition policy templates have additional customization options.

Policy templates cannot be used to alter settings for an initialized partition. Once a partition has been initialized, the Partition SO must change individual policies manually (see "Setting Partition Policies Manually" on page 353).

This section provides instructions for the following procedures, and some general guidelines and restrictions:

- "Creating a Partition Policy Template" below
- "Editing a Partition Policy Template" on the next page
- "Applying a Partition Policy Template" on page 358

Creating a Partition Policy Template

The following procedure describes how to create a policy template for a partition. This can be done optionally at two points in the partition setup process:

- before the partition is initialized: this produces a template file containing the default policy settings, which can then be edited
- after initializing and setting the partition policies manually: this produces a template file with the current policy settings, which can then be used to initialize other partitions to take on the same settings.

To create a partition policy template

1. Launch LunaCM and set the active slot to the partition.

   lunacm:> **slot set -slot** <slotnum>

   > **NOTE** The command to create and export a partition policy template (ppt) file is a sub-command of **partition showpolicies**, which does not require login by the partition SO.

2. Create the partition policy template file. Specify an existing save directory and original filename. No file extension is required. If a template file with the same name exists in the specified directory, it is overwritten.

   lunacm:> **partition showpolicies -exporttemplate** <filepath/filename>

```
lunacm:> partition showpolicies -exporttemplate /usr/safenet/lunaclient/templates/ParPT

Partition policies for Partition: myPartition1 written to
/usr/safenet/lunaclient/templates/ParPT

Command Result : No Error
```

Editing a Partition Policy Template

Use a standard text editor to manually edit policy templates for custom configurations. This section provides template examples and customization guidelines.

Partition Policy Template Example

This example shows the contents of a partition policy template created using the factory default policy settings. Use a standard text editor to change the policy and/or destructiveness values (0=OFF, 1=ON, or the desired value 0-255).

Partition policy template entries have two additional fields: **Off to on destructive** and **On to off destructive** (see example below). Change these values to **0** or **1** to determine whether cryptographic objects on the partition should be deleted when this policy is changed in the future. Policies that lower the security level of the objects stored on the partition are normally destructive, but it may be useful to customize this behavior for your own security strategy. See "Partition Capabilities and Policies" on page 337 for more information.

> **CAUTION!** Setting policy destructiveness to **0** (OFF) makes partitions less secure. Use this feature only if your security strategy demands it.

If you export a policy template from an uninitialized partition, the **Sourced from partition** header field remains blank. This field is informational and you can still apply the template.

The **Policy Description** field is included in the template for user readability only. Policies are verified by the number in the **Policy ID** field.

```
# Policy template FW Version 7.9.0
# Field format - Policy ID:Policy Description:Policy Value:Off to on destructive:On to off
destructive
# Sourced from partition: myPartition1, SN: 154438865290


0:"Allow private key cloning":1:1:0
1:"Allow private key wrapping":0:1:0
```

```
2:"Allow private key unwrapping":1:0:0
3:"Allow private key masking":0:1:0
4:"Allow secret key cloning":1:1:0
5:"Allow secret key wrapping":1:1:0
6:"Allow secret key unwrapping":1:0:0
7:"Allow secret key masking":0:1:0
9:"Allow DigestKey":0:1:0
10:"Allow multipurpose keys":1:1:0
11:"Allow changing key attributes":1:1:0
15:"Ignore failed challenge responses":1:1:0
16:"Operate without RSA blinding":1:1:0
17:"Allow signing with non-local keys":1:0:0
18:"Allow raw RSA operations":1:1:0
20:"Max failed user logins allowed":10:0:0
21:"Allow high availability recovery":1:0:0
22:"Allow activation":0:0:0
23:"Allow auto-activation":0:0:0
25:"Minimum pin length (inverted  255 - min)":248:0:0
26:"Maximum pin length":255:0:0
28:"Allow Key Management Functions":1:1:0
29:"Perform RSA signing without confirmation":1:1:0
31:"Allow private key unmasking":1:0:0
32:"Allow secret key unmasking":1:0:0
33:"Allow RSA PKCS mechanism":1:1:0
34:"Allow CBC-PAD (un)wrap keys of any size":1:1:0
39:"Allow Start/End Date Attributes":0:1:0
40:"Require Per-Key Authorization Data":0:1:0
41:"Partition Version":0:0:1
42:"Allow CPv1":0:1:0
43:"Allow non-FIPS Algorithms":1:1:0]
44:"Allow Extended Domain Management":0:0:0
41:"Allow ECDSA/RSA Prehash SigVer":0:0:0
```

Editing Guidelines and Restrictions

When creating or editing partition policy templates:

- You can remove a policy from the template by adding **#** at the beginning of the line or deleting the line entirely. When you apply the template, the partition will use the default values for that policy.
- Partition policy templates from older Luna versions (6.x or earlier) cannot be applied to Luna 7.x partitions.
- This version of the partition policy template feature is available on Luna 7.x application partitions only. When the active slot is set to a Luna 6.x partition, the **-exporttemplate** option is not available.
- You cannot set partition policy 37: "Force Secure Trusted Channel" on page 347 to 1 using a policy template.
- The following restrictions apply when configuring partitions for Cloning or Key Export (see "Configuring the Partition for Cloning or Export of Private/Secret Keys" on the next page for more information):
  - **Partition policy 0: Allow private key cloning** and **partition policy 1: Allow private key wrapping** can never be set to **1** (ON) at the same time. Initialization fails if the template contains a value of **1** for both policies.
  - **Partition policy 1: Allow private key wrapping** must always have **Off-to-on** destructiveness set to **1** (ON). Initialization fails if the template contains a value of **0** in this field.

> You may not use invalid policy values (outside the acceptable range), or values that conflict with your HSM or partition's capabilities. For example, **Partition capability 3: Enable private key masking** is always **0**, so you cannot set the corresponding partition policy to **1**. If you attempt to initialize a partition with a template containing invalid policy values, an error is returned and initialization fails.

If there is a mismatch between template policies and the default values of newer or dependent policies, then the attempt to apply the old policy would fail with CKR_FAILED_DEPENDENCIES.

You have the option to edit a policy file before applying it, to add newer policies.

Applying a Partition Policy Template

The following procedure describes how to initialize a partition using a policy template.

To apply a policy template to a new partition

1. Ensure that the template file is saved on the client workstation.
2. Launch LunaCM and set the active slot to the new partition.

   lunacm:> **slot set -slot** <slotnum>
3. Initialize the partition, specifying a label and the policy template file. If the template file is not in the same directory as LunaCM, include the correct filepath.

   lunacm:> **partition init -label** <label> **-applytemplate** <filepath/filename>
4. [Optional] Verify that the template has been applied correctly by checking the partition's policy settings. Include the **-verbose** option to view the destructiveness settings.

   lunacm:> **partition showpolicies** [**-verbose**]

Configuring the Partition for Cloning or Export of Private/Secret Keys

By default, the Luna Network HSM 7 stores all keys in hardware, allowing private asymmetric and secret keys to be copied only to another Luna HSM (cloning). Cloning allows you to move or copy key material from a partition to a backup HSM or to another partition in the same HA group. You might, however, want to export private or secret keys to an encrypted file for off-board storage or use. Individual partitions can be configured in one of three modes for handling private keys.

> **NOTE** This feature requires Luna HSM Firmware 7.1.0 or newer.

The Partition SO can set the mode by changing the following policies (see "Partition Capabilities and Policies" on page 337 for more information):

> **Partition policy 0: Allow private key cloning** (default: **1**)

> **Partition policy 1: Allow private key wrapping** (default: **0**)

> **NOTE** These partition policies can never be set both to **1** (ON) at the same time. An error will result (CKR_CONFIG_FAILS_DEPENDENCIES) if it is attempted.

The policies can be set at the time of initialization, using a policy template (see "Setting Partition Policies Using a Template" on page 355) or by following the procedures described below:

> "Cloning Mode" below
> "Key Export Mode" on the next page
> "No Backup Mode" on page 361

NOTE Partition configurations are listed in LunaCM as "Key Export With Cloning Mode". This indicates that the partition is *capable* of being configured for either Key Export or Cloning, with the mode of operation defined by the policies listed above. You can never configure a partition to allow both export and cloning of private keys at once.

TIP **Security Note -***Cloning policies (**0** and **4**)* permit or deny the ability to securely copy keys and objects into and out of a partition.
The *Key Management Functions policy (**28**)* controls the ability to create, delete, generate, derive, or modify cryptographic objects in the current partition.

These controls are independent of each other. With Key Management functions denied, you can still clone objects in and out of partitions where Cloning policy is allowed. Thus HA (high availability) operation can clone keys into a partition that disallows Key Management functions (creation, deletion, etc.). **Cloning a key or object into a partition is not considered creation** - the key or object already existed within the security / cloning domain that encompasses the partition.

Ultimately the security administrators define where keys can exist by controlling distribution of the security / cloning domain, and by defining policies around those keys.

Additionally, key owners can choose to make their keys non-modifiable and non-extractable, if those options are indicated by your use-case.

Cloning Mode

A partition in Cloning mode has the following capabilities and restrictions:

> All keys/objects can be cloned to another partition or Luna Backup HSM in the same cloning domain.
> All keys/objects are replicated within the partition's HA group.
> Private asymmetric keys cannot be wrapped off the HSM (cannot be exported to a file encrypted with a wrapping key).

In this mode, private keys are never allowed to exist outside of a trusted Luna HSM in the designated cloning domain. Cloning mode is the default setting for new partitions.

Setting Cloning Mode on a Partition

Cloning mode is the default setting on new partitions. If another mode was set previously, the Partition SO can use the following procedure to set Cloning mode. Use lunacm:> **partition showpolicies** to see the current policy settings.

> **CAUTION!** **Partition policy 0: Allow private key cloning** is Off-to-On destructive by default. Back up any important cryptographic material on the partition before continuing. This destructiveness setting can be customized by initializing the partition with a policy template (see "Editing a Partition Policy Template" on page 356).

To manually set Cloning mode on a partition

1. Log in to the partition as Partition SO.

   lunacm:> **slot set -slot** <slotnum>

   lunacm:> **role login -name po**

2. Set **partition policy 1: Allow private key wrapping** to **0** (OFF).

   lunacm:> **partition changepolicy -policy 1 -value 0**

3. Set **partition policy 0: Allow private key cloning** to **1** (ON).

   lunacm:> **partition changepolicy -policy 0 -value 1**

To initialize a partition in Cloning mode using a policy template

Use a standard text editor to include the following lines in the policy template file (see "Editing a Partition Policy Template" on page 356):

```
0:"Allow private key cloning":1:1:0
1:"Allow private key wrapping":0:1:0
```

Key Export Mode

A partition in Key Export mode has the following capabilities and restrictions:

- Private asymmetric keys cannot be cloned to other partitions nor to a Luna Backup HSM.
- The partition cannot be part of an HA group (private keys will not be replicated).
- All keys/objects, including private keys, can be wrapped off the HSM (can be exported to a file encrypted with a wrapping key).

This mode is useful when generating key pairs for identity issuance, where transient key-pairs are generated, wrapped off, and embedded on a device. They are not used on the HSM, but generated and issued securely, and then deleted from the HSM. This applies to V0 partitions only; V1 partitions cannot enable this mode.

Setting Key Export Mode on a Partition

The Partition SO can use the following procedure to set Key Export mode. Use lunacm:> **partition showpolicies** to see the current policy settings.

> **CAUTION!** **Partition policy 1: Allow private key wrapping** is always Off-to-On destructive. Back up any important cryptographic material on the partition before continuing. This destructiveness setting cannot be changed with a policy template (see "Editing Guidelines and Restrictions" on page 357).

To manually set Key Export mode on a partition

1. Launch LunaCM and log in to the partition as Partition SO.

   lunacm:> **slot set -slot** <slotnum>

   lunacm:> **role login-name po**

2. Set **partition policy 0: Allow private key cloning** to **0** (OFF).

   lunacm:> **partition changepolicy -policy 0 -value 0**

3. Set **partition policy 1: Allow private key wrapping** to **1** (ON).

   lunacm:> **partition changepolicy -policy 1 -value 1**

To initialize a partition in Key Export mode using a policy template

Use a standard text editor to include the following lines in the policy template file (see "Editing a Partition Policy Template" on page 356):

```
0:"Allow private key cloning":0:1:0
1:"Allow private key wrapping":1:1:0
```

No Backup Mode

A partition in No Backup mode has the following restrictions:

> Private asymmetric keys cannot be cloned to other partitions or to a Luna Backup HSM. All other objects can still be cloned.

> Private asymmetric keys cannot be wrapped off the HSM (exported to a file encrypted with a wrapping key). All other objects can still be wrapped off.

Without backup capability, private keys can never leave the HSM. This mode is useful when keys are intended to have short lifespans, and are easily replaced.

Setting No Backup Mode on a Partition

The Partition SO can use the following procedure to set No Backup mode. Use lunacm:> **partition showpolicies** to see the current policy settings.

To manually set No Backup mode on a partition

1. Launch LunaCM and log in to the partition as Partition SO.

   lunacm:> **slot set -slot** <slotnum>

   lunacm:> **role login -name po**

2. If **partition policy 0: Allow private key cloning** is set to **1** (ON), set it to **0** (OFF).

   lunacm:> **partition changepolicy -policy 0 -value 0**

3. If **partition policy 1: Allow private key wrapping** is set to **1** (ON), set it to **0** (OFF).

   lunacm:> **partition changepolicy -policy 1 -value 0**

To initialize a partition in No Backup mode using a policy template

Use a standard text editor to include the following lines in the policy template file (see "Editing a Partition Policy Template" on page 356):

```
0:"Allow private key cloning":0:1:0
1:"Allow private key wrapping":0:1:0
```


CHAPTER 10: Partition Roles

The security of an HSM and its cryptographic contents depends on well-controlled access to that HSM. A controlled access policy is defined by:

- the set of users with valid login credentials for the appliance, the HSM and the application partition
- the actions each user is allowed to perform when logged in (the user's role)

For example, an access policy that adheres to the PKCS#11 standard requires two roles: the security officer (SO), who administers the user account(s), and the standard user, who performs cryptographic operations. When a user logs in to the HSM, they can perform only those functions that are permitted for their role.

All cryptographic operations take place on an application partition. This partition is created on the HSM by the HSM SO and assigned to a registered client over a network (see Application Partitions). Partition roles allow the partition to function as an independent virtual HSM, with its own Security Officer and users. This design provides more flexibility in meeting the security needs of your organization. Personnel holding the roles described below must have administrative access to a client workstation with a partition assigned to it and Luna HSM Client installed. They do not require SSH access to LunaSH on the Luna Network HSM 7 appliance.

The partition-level roles are as follows:

Partition Security Officer (PO)

The Partition SO handles all administrative and configuration tasks on the application partition, including:

- Initializing the partition, setting the PO credential, and setting a cloning domain for the partition (see "Initializing an Application Partition" on page 332)
- Configuring partition policies (see "Partition Capabilities and Policies" on page 337)
- Initializing the Crypto Officer role (see "Initializing the Crypto Officer Role" on page 368)
- Activating the partition (see "Activation on Multifactor Quorum-Authenticated Partitions" on page 373)

Managing the Partition SO Role

Refer also to the following procedures to manage the PO role:

- "Logging In to the Application Partition" on page 366
- "Changing a Partition Role Credential" on page 370

Crypto Officer (CO)

The Crypto Officer is the primary user of the application partition and the cryptographic objects stored on it. The Crypto Officer has the following responsibilities:

- Creating, deleting, and modifying cryptographic objects via user applications
- Performing cryptographic operations via user applications
- Managing backup and restore operations for partition objects (see "Partition Backup and Restore" on page 467)

> Create and configure HA groups (see "Setting Up an HA Group" on page 432)
> Initializing the Crypto User role (see "Initializing the Crypto User Role" on page 369)
> The CO can modify keys - must provide per-key authorisation (PKA) data for unassigned keys
> The CO can unblock blocked (due to per-key auth failures) PKA keys
> The CO can increment usage counters and change/set the limit
> The CO can perform SMK rollover

Managing the Crypto Officer Role
Refer also to the following procedures to manage the CO role:

> "Logging In to the Application Partition" on page 366
> "Changing a Partition Role Credential" on page 370

Limited Crypto Officer (LCO)

The Limited Crypto Officer is a role needed for eIDAS compliance and the performance of Per Key Authorization functions, with a subset of the abilities and responsibilities of the Crypto Officer, but wider authority and ability than the Crypto User. The LCO is created by the partition CO. The LCO role is visible and accessible in V1 partitions if the Client software version is 10.3 or newer. The Limited Crypto Officer has the following abilities and responsibilities:

> Creating, deleting, and modifying cryptographic objects via user applications
> Performing cryptographic operations via user applications
  - The LCO can copy and modify keys and private objects- must provide per-key authorisation (PKA) data for unassigned keys
  - The LCO can increment usage counters, but cannot change/set the limit
  - The LCO cannot unblock blocked (due to per-key auth failures) PKA keys
  - The LCO can wrap/unwrap keys - must specify the per-key auth data for both the wrapping/unwrapping keys and the wrapped/unwrapped keys
  - The LCO can derive keys - must provide the per-key auth data for the key used for derivation and specify the per-key auth data for the key being derived in the template
  - The LCO can derive-and-wrap - must provide per-key auth data as above
  - The LCO can perform SKS operations (SIMExtract / SIMInsert)
  - The LCO cannot perform SMK rollover
> Creating and configuring HA groups (see "Setting Up an HA Group" on page 432)
> Initializing the Crypto User role (see "Initializing the Crypto User Role" on page 369)

Managing the Limited Crypto Officer Role
Refer also to the following procedures to manage the LCO role:

> "Logging In to the Application Partition" on page 366
> "Changing a Partition Role Credential" on page 370

- The LCO role does not support cloning
- The LCO role is not visible for V0 partitions.
- The LCO role is subject to role-affecting partition policies like
  - Minimum PIN length [25]
  - Maximum PIN length [26]
  - Maximum failed challenge responses [15]
  - Maximum failed user logins allowed [20]
    - Upon reaching the limit, the LCO is locked out; CO and CU remain operational
    - Partition CO can unlock a locked LCO by resetting its credentials
- The LCO can create and destroy private objects
- The LCO can generate keys assigned or unassigned, but cannot assign a key after it is generated.
- The LCO can delete keys
  - Unlike CO role, LCO must provide per-key authorization (PKA) data
  - LCO supports the “single-use signing keys” scenario where a user generates a key, signs with that key, and deletes the key
- The LCO can modify keys - must provide per-key authorisation (PKA) data for unassigned keys
- The LCO can increment usage counters but, unlike CO, cannot change/set the limit
- The LCO can wrap/unwrap
  - PKA behaviour for wrap: must provide the per-key auth data for both the wrapping and the wrapped keys
  - PKA behaviour for unwrap: must provide the per-key auth data for unwrapping key and specify the per-key auth data for the unwrapped key in the template
- For PKA operation
  - The LCO can derive keys
  - The LCO can derive-and-wrap
- The LCO can extract/insert in all scenarios
  - Including SKS key migration (old SKS: Insert; no Extract)
  - Including new SKS (Extract and Insert)
- The LCO cannot clone/replicate in any scenario - this means that LCO is not self-sufficient for HA; the CO is needed to clone SMK(s)
- Unlike the CO, the LCO cannot perform SMK rollover

Crypto User (CU)

The Crypto User is an optional role that can perform cryptographic operations using partition objects in a read-only capacity, but can create only public objects. This role is useful in that it provides limited access; the Crypto Officer is the only role that can make significant changes to the contents of the partition. The Crypto User has the following capabilities:

- Performing operations like encrypt/decrypt and sign/verify using objects on the partition

> Creating and backing up public objects (see "Partition Backup and Restore" on page 467)

> The CU can increment usage counters but, unlike CO, cannot change/set the limit

Managing the Crypto User Role
Refer also to the following procedures to manage the CU role:

> "Logging In to the Application Partition" below

> "Changing a Partition Role Credential" on page 370

Logging In to the Application Partition

Before you can perform administrative tasks on the partition or its stored cryptographic objects, you must log in with the appropriate role:

> Partition Security Officer (specify **po** for <role>)

> Crypto Officer (specify **co** for <role>)

> Crypto User (specify **cu** for <role>)

To log in to the application partition

1. Launch LunaCM on the Luna Network HSM 7 client workstation.
2. Set the active slot to the desired partition.

   lunacm:> **slot set -slot** <slotnum>
3. Log in by specifying your role on the partition.

   lunacm:> **role login -name** <role>

   You are prompted for the role's credential.

Failed Partition Login Attempts

The consequences of multiple failed login attempts vary by role, depending on the severity of the security risk posed by that role being compromised. This is a security feature meant to thwart repeated, unauthorized attempts to access your cryptographic material.

> **NOTE** The system must actually receive some erroneous/false information before it logs a failed attempt; if you merely forget to insert the PED key, or insert the wrong color key, that is not counted as a failed attempt. You must insert an incorrect PED key of the correct type, or enter an incorrect PIN or challenge secret, to fail a login attempt.

Partition Security Officer
If you fail ten consecutive Partition SO login attempts, the partition is zeroized and all cryptographic objects are destroyed. The Partition SO must re-initialize the partition and Crypto Officer role, who can restore key material from a backup device.

Crypto Officer

If you fail ten consecutive Crypto Officer login attempts, the CO and CU roles are locked out. But see below for the exception. The default lockout threshold of 10 is governed by partition policy 20: Max failed user logins allowed, and the Partition SO can set this threshold lower if desired (see "Partition Capabilities and Policies" on page 337).

Is recovery possible from lockout or loss of the partition role credential?

Yes, and no, depending on configuration options you might choose.

Separation of roles ensures that,

- while the Partition Crypto Officer (and subsidiary roles) can see and manage the *content* of an application partition,
- the partition SO cannot access or manage the content of a partition; SO manages at the provisioning and security level for the partition.

If you lose the use of your CO credential, the contents of the partition are no longer accessible. The Partition SO might not be able to help in that situation, for the following reason.

The partition SO cannot just reset the password of the partition CO if you have disallowed it

Recovery from partition role lockout depends on the setting of **HSM policy 15: Enable SO reset of partition PIN**:

- If HSM policy 15 is set to **1** (enabled), the CO and CU and LCO roles are temporarily locked out by too many bad authentication attempts. The Partition SO must unlock the CO role and reset the credential (see "Resetting the Crypto Officer, Limited Crypto Officer, or Crypto User Credential" on page 372).
- If HSM policy 15 is set to **0** (disabled), the CO and CU and LCO roles are permanently locked out and the partition contents are no longer accessible. The Partition SO must re-initialize the partition (destroying all contents) and the Crypto Officer role, who can restore key material from a backup. This is the default setting.

> **NOTE** If you have a backup and know its password, you can recover material. If you do not have a backup, or the backup that you have is not secured by a known password, then the material is lost.

> **CAUTION!** If loss of partition contents is not the desired outcome, ensure that the HSM SO *en*ables this destructive policy *before* creating partitions and assigning to clients.

Crypto User

If you fail ten consecutive Crypto User login attempts, the CU role is locked out. The default lockout threshold of 10 is governed by partition policy **20: Max failed user logins allowed**, and the Partition SO can set this threshold lower if desired (see "Partition Capabilities and Policies" on page 337). The CO must unlock the CU role and reset the credential (see "Resetting the Crypto Officer, Limited Crypto Officer, or Crypto User Credential" on page 372).

Initializing Crypto Officer and Crypto User Roles for an Application Partition

The following procedures will allow you to initialize the Crypto Officer (CO) and Crypto User (CU) roles and set an initial credential.

As of Luna Network HSM 7 Appliance Software 7.7.1 (and newer), in addition to creating an application partition, the administrator (HSM SO) can also initialize the partition, creating the PSO role. The administrator can then use the new PSO credential on that partition to initialize the Crypto Officer role. The Crypto User role is still created from the client side, via lunacm.

Initializing the Crypto Officer Role

The Crypto Officer (CO) is the primary user of the application partition and the cryptographic objects stored on it. The Partition Security Officer (PO) must initialize the CO role and assign an initial credential.

To initialize the Crypto Officer role using LunaCM on the Luna HSM Client

1. In LunaCM, log in to the partition as Partition SO (see "Logging In to the Application Partition" on page 366).

   lunacm:> **role login -name po**

2. Initialize the Crypto Officer role. If you are using a password-authenticated partition, specify a CO password. If you are using a multifactor quorum-authenticated partition, ensure that you have a blank or rewritable black PED key available. Refer to "Creating PED keys" on page 287 for details on creating PED keys.

   Passwords and activation challenge secrets must be 8-255 characters in length. Spaces are allowed; to specify a password with spaces using command-line options, enclose the password in double quotation marks. The space character may not be used as the first character in a password.The following characters are allowed:
   `!#$%'()*+,-./0123456789:=? @ABCDEFGHIJKLMNOPQRSTUVWXYZ[]^_abcdefghijklmnopqrstuvwxyz{}~`
   This character set is enforced when using Luna Appliance Software 7.9.0 or Luna HSM Client 10.8.0 or newer, and recommended for all previous versions. Previously-set passwords and challenge secrets are unaffected, but the new character set is enforced when these passwords are changed.

   lunacm:> **role init -name co**

3. Provide the CO credential to your designated Crypto Officer.

> **NOTE** If **HSM policy 21: Force user PIN change after set/reset** is enabled (this is the default setting), the CO must change the credential before any other actions are permitted. See "Changing a Partition Role Credential" on page 370.

To initialize the Crypto Officer role using LunaSH on the Luna Network HSM 7

> **NOTE** This command requires Luna Network HSM 7 Appliance Software 7.8.1 or newer. It cannot be used on STC partitions; the Partition SO must use LunaCM at the client for partition management.

The following steps assume that the Network HSM administrator has created the partition (**partition create**) and has initialized the partition (**partition init**), thus initializing the PSO role for that partition. You do not need to log in to initialize the CO, because the command requires you to provide the credential of the Partition Owner/Partition Security Officer that was created at partition initialization.

1. Initialize the Crypto Officer role, providing the partition name, the PSO credential and the credential for the CO that is being created. If you are using a password-authenticated partition, specify a CO password. If you are using a multifactor quorum-authenticated partition, ensure that you have a blank or rewritable black PED key available. Refer to "Creating PED keys" on page 287 for details on creating PED keys.

   Passwords and activation challenge secrets must be 8-255 characters in length. Spaces are allowed; to specify a password with spaces using command-line options, enclose the password in double quotation marks. The space character may not be used as the first character in a password.The following characters are allowed:
   `!#$%'()*+,-./0123456789:=? @ABCDEFGHIJKLMNOPQRSTUVWXYZ[]^_abcdefghijklmnopqrstuvwxyz{}~`
   This character set is enforced when using Luna Appliance Software 7.9.0 or Luna HSM Client 10.8.0 or newer, and recommended for all previous versions. Previously-set passwords and challenge secrets are unaffected, but the new character set is enforced when these passwords are changed.

   lunash:> **partition init co -partition** <partition name> **-psopin** <PSO's password> **-copin** <CO's password>

2. Provide the CO credential to your designated Crypto Officer, if you are not retaining/performing all roles. The CO should then change the credential, unless HSM policy 21 has been unset/disabled - see Note. If you are managing and performing all roles (no separation of responsibilities), then "provide the CO credential means to provide it to your application(s) that will be using that credential to access the partition for read-write operations.

> **NOTE** If **HSM policy 21: Force user PIN change after set/reset** is enabled (this is the default setting), the CO *must* change the credential before any other actions are permitted. See "Changing a Partition Role Credential" on the next page.

Any crypto operations, performed by the CO, are done from a registered client via a suitable API.

Initializing the Crypto User Role

The Crypto User (CU) is an optional role that can perform cryptographic operations using partition objects in a read-only capacity, but can only create public objects. The Crypto Officer must initialize the CU role and assign an initial credential.

To initialize the Crypto User role using LunaCM on the Luna HSM Client

1. In LunaCM, log in to the partition as Crypto Officer (see "Logging In to the Application Partition" on page 366).

   lunacm:> **role login -name co**

2. Initialize the Crypto User role. If you are using a password-authenticated partition, specify a CU password. If you are using a multifactor quorum-authenticated partition, ensure that you have a blank or rewritable gray PED keys available. Follow the instructions on the Luna PED screen. Refer to "Creating PED keys" on page 287 for details on creating PED keys.

   Passwords and activation challenge secrets must be 8-255 characters in length. Spaces are allowed; to specify a password with spaces using command-line options, enclose the password in double quotation marks. The space character may not be used as the first character in a password.The following characters

are allowed:
`!#$%'()*+,-./0123456789:=? @ABCDEFGHIJKLMNOPQRSTUVWXYZ[]^_abcdefghijklmnopqrstuvwxyz{}~`
This character set is enforced when using Luna Appliance Software 7.9.0 or Luna HSM Client 10.8.0 or newer, and recommended for all previous versions. Previously-set passwords and challenge secrets are unaffected, but the new character set is enforced when these passwords are changed.

lunacm:> **role init -name cu**

3. Provide the CU credential to your designated Crypto User.

> **NOTE** If **HSM policy 21: Force user PIN change after set/reset** is enabled (this is the default setting), the CU must change the credential before any other actions are permitted. See "Changing a Partition Role Credential" below.

To initialize the Crypto User role using LunaSH on the Luna Network HSM 7

> **NOTE** This command requires Luna Network HSM 7 Appliance Software 7.8.1 or newer. It cannot be used on STC partitions; the Partition SO must use LunaCM at the client for partition management.

You do not need to log in to initialize the crypto user because, as part of the command, you supply the credential of the Crypto Officer:

- who already exists (has already been initialized), and
- whose password has been changed from the one that the CO was given when first initialized (unless HSM policy 21 was changed from default).

1. Initialize the Crypto User role. If you are using a password-authenticated partition, specify a CU password. If you are using a multifactor quorum-authenticated partition, ensure that you have a blank or rewritable gray PED keys available. Follow the instructions on the Luna PED screen. Refer to "Creating PED keys" on page 287 for details on creating PED keys.

   lunash:> **partition init cu -partition**<partition name> [**-copin** <crypto officer credential>] [**-cupin** <crypto user initial credential>]

2. Provide the CU credential to your designated Crypto User. If you are managing and performing all roles (no separation of responsibilities), then "provide the CU credential means to provide it to your application(s) that will be accessing the partition for read-only operations.

> **NOTE** If **HSM policy 21: Force user PIN change after set/reset** is enabled (this is the default setting), the CU must change the credential before any other actions are permitted. See "Changing a Partition Role Credential" below.

Changing a Partition Role Credential

Use the instructions on this page to change your current credential for a role in an HSM application partition.

From time to time, it might be necessary to change the secret associated with a role on an HSM appliance, a role on a cryptographic module (HSM) or a partition of an HSM, or a cloning domain secret. Reasons for changing credentials include:

- Regular credential rotation as part of your organization's security policy
- Compromise of a role or secret due to loss or theft of a PED key
- Personnel changes in your organization or changes to individual security clearances
- Changes to your security scheme (implementing/revoking M of N, PINs, or shared secrets)

The following procedure allows you to change the credential for a partition role (Partition SO, Crypto Officer, Crypto User). You must first log in *using the role's current credential* (this is not a way to recover from lockout or from lost credentials).

> **NOTE** If **HSM policy 21: Force user PIN change after set/reset** is set to **1** (default), this procedure is required after initializing or resetting the CO or CU role and/or creating a challenge secret.

To change a partition role credential using LunaCM on the Luna HSM Client

1. In LunaCM, log in using the role's current credential (see "Logging In to the Application Partition" on page 366).

   lunacm:> **role login -name** <role>

2. Change the credential for the logged-in role. If you are using a password-authenticated partition, specify a new password. If you are using a multifactor quorum-authenticated partition, ensure that you have a blank or rewritable PED key available. Refer to "Creating PED keys" on page 287 for details on creating PED keys.

   Passwords and activation challenge secrets must be 8-255 characters in length. Spaces are allowed; to specify a password with spaces using command-line options, enclose the password in double quotation marks. The space character may not be used as the first character in a password.The following characters are allowed:
   `!#$%'()*+,-./0123456789:=? @ABCDEFGHIJKLMNOPQRSTUVWXYZ[]^_abcdefghijklmnopqrstuvwxyz{}~`
   This character set is enforced when using Luna Appliance Software 7.9.0 or Luna HSM Client 10.8.0 or newer, and recommended for all previous versions. Previously-set passwords and challenge secrets are unaffected, but the new character set is enforced when these passwords are changed.

   lunacm:> **role changepw -name** <role>

3. To change the CO or CU challenge secret for an activated PED-authenticated partition, specify the **-oldpw** and/or **-newpw** options.

   lunacm:> **role changepw -name** <role> **-oldpw** <oldpassword> **-newpw** <newpassword>

> **TIP** Where you have an HA Indirect Login setup (see High Availability Indirect Login), your HSM is made accessible by other HSMs. Adding a challenge secret to your role, that is unknown to other parties, does not prevent other parties from logging into your HSM. Rather it prevents other parties from using your particular role without that extra credential. To prevent other parties accessing your HSM, change the PIN.

To change a partition role credential using LunaSH on the Luna Network HSM 7

[Section added/modified for LUNA-31064]

> **NOTE** If you need to change the Partition SO credential, you must use LunaCM on the Luna HSM Client as described above.

1. In LunaSH, log in as SO (see hsm login).

   lunash:> **hsm login**

2. Change the credential for the role on the specified partition. By default, the Crypto Officer role is changed; to change the Crypto User credential, include the **-cu** option. If you are using a password-authenticated partition, specify a new password. If you are using a multifactor quorum-authenticated partition, ensure that you have a blank or rewritable PED key available. Refer to "Creating PED keys" on page 287 for details on creating PED keys.

   lunash:> **partition changePw -partition** <partitionname>

   lunash:> **partition changePw -partition** <partitionname> **-cu**

3. To change the CO or CU challenge secret for an activated multifactor quorum-authenticated partition, specify the **-oldpw** and/or **-newpw** options, like.

   lunash:> **partition changePw -partition** <partitionname> **-oldpw** <oldpassword> **-newpw** <newpassword> [**-cu**]

> **TIP** Where you have an HA Indirect Login setup (see High Availability Indirect Login), your HSM is made accessible by other HSMs. Adding a challenge secret to your role, that is unknown to other parties, does not prevent other parties from logging into your HSM. Rather it prevents other parties from using your particular role without that extra credential. To prevent other parties accessing your HSM, change the PIN.

Resetting the Crypto Officer, Limited Crypto Officer, or Crypto User Credential

If necessary, the Crypto Officer can reset the Crypto User credential at any time, without providing the current credential. This is useful in cases where the Crypto User credential has been lost or otherwise compromised.

Prerequisites for Crypto Officer Reset

The Partition SO can also reset the Crypto Officer's credential, if **HSM policy 15: Enable SO reset of partition PIN** is enabled. By default, this policy is not enabled, and changing it is destructive. If you want the Partition SO to be able to reset the CO's credential, the HSM SO must enable this policy before creating the application partition (see "Partition Capabilities and Policies" on page 337).

> **CAUTION!** HSM policy 15 is destructive when turned on. All partitions on the HSM and their contents will be erased.

To reset the Crypto Officer, Limited Crypto Officer, or Crypto User credential

1. Log in with the appropriate role (see "Logging In to the Application Partition" on page 366).
2. Reset the desired role's credential.

Passwords and activation challenge secrets must be 8-255 characters in length. Spaces are allowed; to specify a password with spaces using command-line options, enclose the password in double quotation marks. The space character may not be used as the first character in a password.The following characters are allowed:
`!#$%'()*+,-./0123456789:=? @ABCDEFGHIJKLMNOPQRSTUVWXYZ[]^_abcdefghijklmnopqrstuvwxyz{}~`
This character set is enforced when using Luna Appliance Software 7.9.0 or Luna HSM Client 10.8.0 or newer, and recommended for all previous versions. Previously-set passwords and challenge secrets are unaffected, but the new character set is enforced when these passwords are changed.

lunacm:> **role resetpw -name** <role>

You are prompted to set a new credential for the role.

3. Provide the new credential to the Crypto Officer, Limited Crypto Officer(*), or Crypto User.

> **NOTE** If **HSM policy 21: Force user PIN change after set/reset** is enabled, the user must change the credential before any other actions are permitted. See "Changing a Partition Role Credential" on page 370.
>
> The CO can reset the LCO's primary credentials (lunacm:> **role resetpw**) regardless of the status of "Enable SO reset of a partition PIN" policy 15.

(*LCO is applicable to Luna HSM Firmware 7.7.0 and newer.)

Activation on Multifactor Quorum-Authenticated Partitions

A multifactor quorum-authenticated partition requires a user's PED key each time a role (Partition SO, Crypto Officer, Limited Crypto Officer, Crypto User) logs in. For some use cases, such as key vaulting, this physical key requirement is desirable. For many applications, however, it is impractical to require the full Luna PED interaction every time.

For these use cases, the Partition SO can activate the partition and set a secondary password referred to as a challenge secret. When a partition is activated, the HSM caches the Crypto Officer and Limited Crypto Officer and Crypto User PED key secrets upon first login, and subsequent logins require the challenge secret only. The PED key secret remains cached until the role is explicitly deactivated or the HSM loses power due to a reboot or power outage.

Activation does not provide much advantage for clients that log in to the partition and remain logged in. It is an indispensable advantage in cases where the client application repeatedly logs in to perform a task, and then logs out or closes the cryptographic session after the task is completed.

Auto-activation

Auto-activation allows PED key credentials to remain cached even in the event of a reboot or a brief power outage (up to 2 hours).

Tamper events and activation/auto-activation

When a tamper event occurs, or if an uncleared tamper event is detected on reboot, the cached PED key data is zeroized, and activation/auto-activation is disabled. See Tamper Events and "Partition Capabilities and Policies" on page 337 for more information.

This section contains instructions for the following procedures:

Enabling Activation on a Partition

The Partition SO can enable activation on a partition by setting **partition policy 22: Allow activation** to **1** (on). This setting enables activation for the Crypto Officer and Limited Crypto Officer and Crypto User roles. When partition policy 22 is enabled, the Partition SO can set an initial challenge secret for the Crypto Officer.

Prerequisites

> The partition must be initialized (see "Initializing an Application Partition" on page 332).

To enable activation on a partition using LunaSM on Luna HSM Client

1. Log in to the partition as Partition SO (see "Logging In to the Application Partition" on page 366).

   lunacm:> **role login -name po**

2. Enable partition policy 22.

   lunacm:> **partition changepolicy -policy 22 -value 1**

Activating a Role

After enabling partition policy 22, activate the CO or LCO or CU roles on the partition. You must set a challenge secret password for each role you want to activate. The Partition SO must set the initial challenge secret for the Crypto Officer, who must set it for the Limited Crypto Officer or Crypto User. The role becomes activated the first time the user logs in to the partition.

Prerequisites

> **Partition policy 22: Allow activation** must be enabled on the partition (see "Enabling Activation on a Partition" above).
> The role you wish to activate must be initialized on the partition (see "Initializing Crypto Officer and Crypto User Roles for an Application Partition" on page 368).
> The partition must be assigned to the client and visible as a slot in LunaCM (see "Client-Partition Connections" on page 107).

To activate a role via lunacm on a client

1. Log in to the partition using the appropriate role (see "Logging In to the Application Partition" on page 366):
   - If you are activating the Crypto Officer role, log in as Partition SO.
   - If you are activating the Crypto User or Limited Crypto Officer role, log in as Crypto Officer.

   lunacm:> **role login -name** <role>

2. Set an initial challenge secret for the role you wish to activate. The length of the challenge secret is configurable by the Partition SO (see **partition policy 25:** "Minimum PIN length" on page 344).

   Passwords and activation challenge secrets must be 8-255 characters in length. Spaces are allowed; to specify a password with spaces using command-line options, enclose the password in double quotation marks. The space character may not be used as the first character in a password.The following characters are allowed:
   `!#$%'()*+,-./0123456789:=? @ABCDEFGHIJKLMNOPQRSTUVWXYZ[]^_abcdefghijklmnopqrstuvwxyz{}~`
   This character set is enforced when using Luna Appliance Software 7.9.0 or Luna HSM Client 10.8.0 or newer, and recommended for all previous versions. Previously-set passwords and challenge secrets are unaffected, but the new character set is enforced when these passwords are changed.

   lunacm:> **role createchallenge -name** <role>

   > **NOTE** Activation requires that a challenge secret is set for the specified role. If the role does not have a challenge secret, you are prompted for the PED key, regardless of the policy setting.

3. Log out of the partition.

   lunacm:> **role logout**

4. Provide the initial challenge secret to the designated CO or CU by secure means. The PED key secret is cached when they log in for the first time. The CO or CU can store the black or gray PED key in a safe place. The cached PED key secret allows their application(s) to open and close sessions and perform operations within those sessions.

   > **NOTE** If **HSM policy 21:** Force user PIN change after set/reset is enabled (this is the default setting), the CO or CU must change the challenge secret before any other actions are permitted. See "Changing a Partition Role Credential" on page 370.

   > **NOTE** The Luna PED screen prompts for a black PED key for any of
   > - "User",
   > - "Crypto Officer",
   > - "Limited Crypto Officer",
   > - "Crypto User".
   >
   > The Luna PED is not aware that the key you present has a black or a gray sticker on it. The colored stickers are visual identifiers for your convenience in keeping track of your PED keys. You differentiate by how you label, and how you use, a given physical key that the Luna PED sees as "black" (once it has been imprinted with a secret).

To activate a role using LunaSH on the Luna Network HSM 7

1. Log in to the HSM.

   lunash:> **hsm login**

2. Set an initial challenge secret for the role you wish to activate. The length of the challenge secret is configurable by the Partition SO (see **partition policy 25:** "Minimum PIN length" on page 344).

Passwords and activation challenge secrets must be 8-255 characters in length. Spaces are allowed; to specify a password with spaces using command-line options, enclose the password in double quotation marks. The space character may not be used as the first character in a password.The following characters are allowed:
`!#$%'()*+,-./0123456789:=? @ABCDEFGHIJKLMNOPQRSTUVWXYZ[]^_abcdefghijklmnopqrstuvwxyz{}~`
This character set is enforced when using Luna Appliance Software 7.9.0 or Luna HSM Client 10.8.0 or newer, and recommended for all previous versions. Previously-set passwords and challenge secrets are unaffected, but the new character set is enforced when these passwords are changed.

lunash:> **partition activate -partition** <partition name> **-password** <co challenge secret>

lunash:> **partition activate -partition** <partition name> **-cu -partition** <cu challenge secret>

> **NOTE** Activation requires that a challenge secret is set for the specified role. If the role does not have a challenge secret, you are prompted for the PED key, regardless of the policy setting.

3. Provide the initial challenge secret to the application(s) that will use the partition as CO or CU. The PED key secret is cached when they log in for the first time. The cached PED key secret allows those application(s) to open and close sessions and perform operations within those sessions.

> **NOTE** If **HSM policy 21:** Force user PIN change after set/reset is enabled (this is the default setting), the CO or CU must change the challenge secret before any other actions are permitted. See "Changing a Partition Role Credential" on page 370.

> **NOTE** The Luna PED screen prompts for a black PED key for any of
> - "User",
> - "Crypto Officer",
> - "Limited Crypto Officer",
> - "Crypto User".
>
> The Luna PED is not aware that the key you present has a black or a gray sticker on it. The colored stickers are visual identifiers for your convenience in keeping track of your PED keys. You differentiate by how you label, and how you use, a given physical key that the Luna PED sees as "black" (once it has been imprinted with a secret).

Enabling Auto-activation

Auto-activation allows PED key credentials to be cached even in the event of a reboot or a brief power outage (up to 2 hours). Clients can re-connect and continue using the application partition without needing to re-authenticate using a PED key.

The Partition SO can enable auto-activation on a partition by setting **partition policy 23:** "Allow auto-activation" on page 344.

Prerequisites

- **Partition policy 22: Allow activation** must be enabled on the partition (see "Enabling Activation on a Partition" on page 374).

To enable auto-activation on a partition

1. Log in to the partition as Partition SO (see "Logging In to the Application Partition" on page 366).

   lunacm:> **role login -name po**

2. Enable partition policy 23.

   lunacm:> **partition changepolicy -policy 23 -value 1**

   Auto-activation takes effect for each affected role (CO and/or CU) the next time the role is authenticated.

3. [Optional] For optimal reliability, the Luna Network HSM 7 **admin** or **operator** can set the appliance to reboot automatically if it fails to complete a normal shutdown. Log in to LunaSH to change this setting.

   lunash:> **sysconf appliance rebootonpanic enable**

Deactivating a Role

An activated role on a partition remains activated until it is explicitly deactivated, or the HSM loses power due to a reboot or power outage (with auto-activation disabled). This deletes the cached PED key secret for the role.

Prerequisites

> You must be authorized to deactivate the role. The CO and CU can manually deactivate their own or each other's roles. The Partition SO can deactivate both the CO and CU roles.

To deactivate a role on a partition

1. Log in to the partition with the appropriate role (see "Logging In to the Application Partition" on page 366).

   lunacm:> **role login -name** <role>

2. Specify the role you wish to deactivate.

   lunacm:> **role deactivate -name** <role>

   This deletes the cached authentication credential for the role. The next time the role logs in, the credential is re-cached.

3. If you wish to disable activation entirely, so that credentials are not re-cached at the next login, the Partition SO can disable **partition policy 22:** "Allow activation" on page 343.

   lunacm:> **partition changepolicy -policy 22 -value 0**

4. If partition policy 22 is disabled, auto-activation is also disabled (even though **partition policy 23:** "Allow auto-activation" on page 344 is set to **1**). When partition policy 22 is enabled again, auto-activation resumes. To turn off auto-activation, you must disable partition policy 23.

   lunacm:> **partition changepolicy -policy 23 -value 0**

Security of Your Partition Challenge

For Luna Network HSM 7s with Password Authentication, the partition password used for administrative access by the Crypto Officer is also the partition challenge secret or password used by client applications.

For Luna Network HSM 7s with multifactor quorum authentication, the partition authentication used for administrative access by the Crypto Officer is the secret on the black PED key(s) for that partition. The partition challenge secret or password used by client applications is a separate character string, set by the Partition SO and then changed by the Crypto Officer (mandatory) for the CO's use. This is one way in which we implement separation of roles in the Luna HSM security paradigm.

How Secure Is the Challenge Secret or Password?

The underlying concern is that a password-harvesting attack might eventually crack the secret that protects the partition. Layers of protection are in place, to minimize or eliminate such a risk.

First, such an attack must be run from a Luna HSM Client computer. For interaction with HSM partitions on a Luna Network HSM 7, a Luna HSM Client computer is one with Luna software installed, on which you have performed the exchange of certificates to create a Network Trust Link (NTL). That exchange requires the knowledge and participation of the appliance administrator and the Partition SO (who might, or might not, be the same person). It is not possible to secretly turn a computer into a Client of a Luna HSM partition - an authorized person within your organization must participate.

Second, for Luna HSMs with password authentication, you set the partition password directly when you create the partition, so you can make it as secure as you wish (for an example of guidance on password strength, see http://howsecureismypassword.net/ or http://xkcd.com/936/)

For Luna HSMs with multifactor quorum authentication, an optional partition password (also called a challenge secret) may be added for the initialized Crypto Officer (CO) and/or Limited Crypto Officer (LCO) and/or Crypto User (CU) roles. See role createchallenge for the proper command syntax.

Using LunaCM or LunaSH, you can change the partition password (or challenge secret) if you suspect it has been compromised, or if you are complying with a security policy that dictates regular password changes.

As long as you replace any password/challenge secret with one that is equally secure, the possible vulnerability is extremely small.

Conversely, you can choose to replace a secure, random password/challenge-secret with one that is shorter or more memorable, but less secure - you assume the risks inherent in such a tradeoff.

Third, Luna HSM **partition policy 15: Ignore failed challenge responses** can be set to **0** (off). When that policy is off, the HSM stops ignoring bad challenge responses (that is, attempts to submit the partition secret) and begins treating them as failed login attempts. Each bad login attempt is counted. **Partition policy 20: Max failed user logins allowed** determines how high that count can go before the partition is locked out.

Once a partition is locked by bad login attempts, it cannot be accessed until the HSM Security Officer (SO) unlocks it. This defeats an automated harvesting attack that relies on millions of attempts occurring at computer-generated speeds. As well, after one or two lockout cycles, the HSM SO would realize that an attack was under way and would rescind the NTL registration of the attacking computer. That computer would no longer exist as far as the HSM partition was concerned. The SO or your security organization would then investigate how the client computer had been compromised, and would correct the problem before allowing any new NTL registration from that source. See "Logging In to the Application Partition" on page 366 for more information.

As the owner/administrator of the HSM, you determine any tradeoffs with respect to security, convenience, and other operational parameters.

Name, Label, and Password Requirements

This page describes length and character requirements for setting names, labels, domains, passwords, and challenge secrets on the Luna Network HSM 7. This information can also be found in relevant sections throughout the documentation. Refer to the applicable section below:

Custom Appliance User Accounts

LunaSH user names can be 1-32 characters in length, chosen from letters a-z, or A-Z, numbers 0-9, the dash, the dot, or the underscore:

```
abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789-._
```

No spaces are allowed. User names cannot begin with a dot, dash, or number. As with any secure system, no two users (regardless of role) can have the same name.

Attempting to add system keywords as appliance usernames is not allowed and will result in an error message like:

```
Error: '<service name or keyword>' is reserved for system use and cannot be added.
```

Custom Appliance Roles

LunaSH role names can be 1-64 characters in length. The following characters are allowed:

```
abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789_
```

No spaces are allowed. Creating a role name that begins with a number is not recommended. As with any secure system, no two roles can have the same name.

Appliance User Passwords

Using Luna Appliance Software 7.9.0 or newer, LunaSH passwords must be at least eight characters in length, and include characters from each of the following four categories. Previous versions require characters from three categories:

> lowercase alphabetic: `abcdefghijklmnopqrstuvwxyz`
> uppercase alphabetic: `ABCDEFGHIJKLMNOPQRSTUVWXYZ`
> numeric: `0123456789`
> special (spaces allowed): `!@#$%^&*()-_=+[]{}\|/;:'",.<>?`~`

HSM Labels

The HSM label created during initialization must be 1-32 characters in length. If you specify a longer label, it will automatically be truncated to 32 characters. Only alphanumeric characters and the underscore are allowed:

```
abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789_
```

Cloning Domains

The domain string must be 1-128 characters in length. The following characters are allowed:

```
abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789 !@#$%^*-_=+[]{}()/:',.~
```

The following characters are problematic or invalid and must not be used in a domain string: `"&;<>?\`|`

Spaces are allowed, as long as the leading character is not a space; to specify a domain string with spaces using the **-domain** option, enclose the string in double quotation marks.

For password-authenticated HSMs, the domain string should match the complexity of the partition password.

Partition Names

Partition names created in LunaSH must be 1-32 characters in length. The following characters are allowed:

```
abcdefghijklmnopqurstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ 0123456789!@#$%^*()-_=+{}[]:',./~
```

Spaces are allowed; enclose the partition name in double quotes if it includes spaces.

The following characters are not allowed: `&\|;<>`"?`

No two partitions can have the same name.

Partition Labels

In LunaSH, the partition label created during initialization must be 1-32 characters in length. The following characters are allowed:

```
abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789 !@#$%^*()-_=+[]{}/:',.~
```

Spaces are allowed; enclose the label in double quotation marks if it includes spaces.

In LunaCM, the partition label created during initialization must be 1-32 characters in length. If you specify a longer label, it will automatically be truncated to 32 characters. The following characters are allowed:

```
abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789 !@#$%^&*()-_=+[]{}\|/;:',.<>`~
```

Spaces are allowed; enclose the label in double quotation marks if it includes spaces.

HSM/Partition Role Passwords or Challenge Secrets

Passwords and activation challenge secrets must be 8-255 characters in length. Spaces are allowed; to specify a password with spaces using command-line options, enclose the password in double quotation marks. The space character may not be used as the first character in a password.

The following characters are allowed:

```
!#$%'()*+,-./0123456789:=? @ABCDEFGHIJKLMNOPQRSTUVWXYZ[]^_abcdefghijklmnopqrstuvwxyz{}~
```

This character set is enforced when using Luna Appliance Software 7.9.0 or Luna HSM Client 10.8.0 or newer, and recommended for all previous versions. Previously-set passwords and challenge secrets are unaffected, but the new character set is enforced when these passwords are changed.

CHAPTER 11: Verifying HSM Authenticity or Key Attestation

Hardware Security Modules have traditionally been deployed in the corporate data center's most secure zone. Establishing trust with the HSM is, in part, achieved by physical access control. In cases of remote client usage (such as cloud cryptography), the client needs a way to verify the authenticity of the device protecting their most valued cryptographic keys.

Thales also provides a tool for validating the attestation of keys created by the HSM, to verify that they were created by a Luna HSM.

Public Key Confirmations

Thales's Luna HSMs include factory-issued device identities certified by a Thales authority. The root of this authority is maintained by Thales in HSMs locked in a vault with layered physical and logical access controls. These certificates are used as the root of trust for the issuance of "public key confirmations" (PKCs), certificates issued by the HSM attesting to the life cycle of a specific private key. A Luna HSM will issue confirmations only for private keys that were created by a Luna cryptographic module and that can never exist outside the security perimeter of a Luna HSM. A valid confirmation is cryptographic proof that a specific key is inside the identified HSM. The confirmation is also proof that the identified HSM is real.

The key pair within the HSM that signs the confirmation is called a Hardware Origin Key (HOK). It is protected inside the cryptographic module's FIPS 140-2 Level 3 security boundary. Each HOK is unique and there is no way to extract or replace it. The HOK is created in the HSM at the time of manufacture and certified by Thales's secure manufacturing authority, which is certified by Thales's root authority.

Public key confirmations are automatically generated for RSA key pairs in the HSM. A user can get a confirmation through the PKCS #11 API or the Luna **cmu** utility, and use it to verify that any RSA key is protected and has always been protected by a Luna HSM. A PKC bundle contains the following certificates:

> **MIC:** Manufacturing Integrity Certificate; corresponds to the Manufacturing Integrity Private Key (MIK), signed by the Thales root.
> **HOC:** Hardware Origin Certificate; corresponds to the Hardware Origin Private Key (HOK). Unique to each HSM. Signed by MIK.
> **DAC:** Device Authentication Certificate; corresponds to the Device Authentication Private Key (DAK). Unique to each HSM. Signed by HOK.
> **PKC:** Public Key Confirmation Certificate; certificate for a private key on the HSM. Signed by DAK.

Public key confirmations are delivered as PKCS #7 files containing a certificate chain. The PKCS #7 files can be viewed using tools like OpenSSL and Microsoft's Certificates snap-in for MMC.

> **NOTE** While third-party tools are capable of cryptographically validating the certificate signature chain, they may display some certificate errors, since they do not recognize some Thales-specific key usage attributes included in the certificates.

Chains of Trust

The chain of trust available via the **cmu** utility included with the Luna HSM Client, **Chrysalis-ITS**, is built in by default, and originates from Thales's SafeNet root certificate authority. It uses the MIC, HOC, DAC, and the PKC.

> **NOTE** Since the introduction of Functionality Modules, HSMs are shipped from the factory with FM-ready hardware. This means that they contain, and use, the HOK and the HOC, but they also have the FM-HOK and FM-HOC on standby. If FMs are enabled on the HSM, the original HOK and HOC are deleted, and the chain-of-trust, thereafter, proceeds through the FM-HOC.

Verifying the HSM's Authenticity

The **cmu** utility also includes a command, **cmu verifyhsm**, that tests an HSM's authenticity by creating and verifying a confirmation on a temporary key created in the HSM. The test includes a proof of possession that asks the HSM to sign a user-entered string as proof the associated private key is present within the target HSM.

> **NOTE** This confirmation procedure is currently not supported on FM-enabled HSMs. Refer to FM Deployment Constraints for details.

The test requires the SafeNet root certificate, provided below:

safenet-root.pem

> **NOTE** The current certificate is valid until 2031-12-31, but it might change before this date at Thales's discretion. Ensure that you have the most recent version of this documentation.

To verify the HSM's authenticity

1. Right-click the link above and save the root certificate to the Luna HSM Client directory.
2. Open a command line and navigate to the Luna HSM Client directory.
3. Use the **cmu** utility to authenticate the HSM. You must specify a challenge string for the HSM to sign, and the root certificate file:

   # **cmu verifyhsm -challenge** <string> **-rootcert safenet-root.pem**

   When prompted, specify the partition you wish to use and the Crypto Officer credential for that partition.

```
>cmu verifyhsm -challenge "1234567890" -rootcert safenet-root.pem
Select token
 [0] Token Label: mypartition-1
 [1] Token Label: mypartition-2
 Enter choice: 0
Please enter password for token in slot 0 : ********
Reading rootcert from file "safenet-root.pem"... ok.
Generating temporary RSA keypair in HSM... ok.
Extracting PKC bundle from HSM... ok.
Verifying PKC certificate... ok.
Verifying DAC certificate... ok.
Verifying HOC certificate... ok.
Verifying MIC certificate... ok.
Verifying MIC against rootcert... ok.
Signing and verifying challenge... ok.
Verifying HSM serial number... ok.
Overall status: Success.
```

If this test fails, contact the HSM SO.

Verifying Key Attestation

Thales provides a Java utility, **Luna PKC Validator**, that allows you to verify that a key was created by a Luna HSM. This tool can be compiled from the repository below:

https://github.com/ThalesGroup/luna-pkc-validator

> **NOTE** The private key used must have attribute **sign=true** because the cmu does a sign verify on test data internally.

The utility requires one of the following root certificates for validation.

- To check the attestation of RSA keys, use the SafeNet root certificate, provided above.
- To check the attestation of ECC keys, use the Thales ECC Manufacturing Integrity Certificate provided below.
- To check the attestation of PQC and ECC keys, use the Thales ECC Manufacturing Integrity Certificate. PQC attestation requires Luna HSM Firmware 7.9.1 or newer, and the version of cmu included with Luna HSM Client 10.9.1 or newer.

Thales-Luna-MIC-ECC.cer

CHAPTER 12: Migrating Keys to Your New HSM

This chapter describes how to migrate your keys and configuration from a Luna HSM 5.x or 6.x partition to a Luna HSM 7.x partition by using one of three methods; backup and restore, cloning, or cloning using a temporary HA group:

- "Luna Network HSM 5.x/6.x to Luna Network HSM 7" on page 388
- "Luna USB HSM 6.x to Luna Network HSM 7" on page 398
- "Luna PCIe HSM 5.x/6.x to Luna Network HSM 7" on page 403
- "Moving from Pre-7.7.0 to Firmware 7.7.0 or Newer" on page 411

Refer also to the chapter on "Key Cloning" on page 193, particularly "Cloning Keys Between Luna 6, Luna 7, and Luna Cloud HSM, Password or Multifactor Quorum" on page 210.

This document guides you through several migration scenarios consisting of older and newer Luna HSMs, using each applicable migration method. Before migrating, preconditions are provided for each scenario that must be met. There are specific user roles that are identified for performing the migration. In addition, both authentication methods (password and PED-authenticated) are supported.

Supported Luna HSMs

This document describes key migration for these Luna HSMs:

- Luna Network HSM, version 5.x or 6.x to 7.x
- Luna USB HSM, version 5.x or 6.x to 7.x
- Luna PCIe HSM, version 5.x or 6.x to 7.x

TIP When cloning objects:

- by direct clone command, or
- by backup/restore, or
- by synchronization in an HA group)

...between 5.x or 6.x source HSMs and 7.x target HSMs, the common domain between the HSMs *must be* the designated *primary domain* on any HSM that is at firmware version 7.8.0 or newer.

This is because the cloning protocol on HSMs prior to firmware 7.8.0 is unaware of the ability to have multiple domains and therefore the older HSM can interact with *only* the primary domain on the firmware 7.8.0+ HSM. So, if the domain of the old HSM exists on the firmware 7.8.0-and-newer HSM, set that domain to be Primary before cloning. If it doesn't already exist, set it to -primary while you are creating it.

Additionally, when migrating keys from a Multifactor Quorum (PED)-authenticated 5.x or 6.x HSM to a password-authenticated Luna 7 HSM,

- password-authenticated HSMs do not have the ability to use Remote PED
- so a PED must be connected directly/locally to the password-authenticated HSM.

Order of operations

As indicated below, a variety of migration methods and paths are possible, and in your own situation, you might be performing different parts of an overall migration strategy at different times over some period. We advise to think carefully about the order in which actions are performed, and this section is a quick summary of considerations that might apply in your situation.

> If you have the older Luna HSMs and are using the older Luna Backup HSM G5, then maintain that Backup HSM at firmware older than Luna Backup HSM G5 Firmware 6.26.0. Firmware 6.26.0 and above will work only with Luna HSMs 7.x. By keeping the pre-6.26.0 Backup firmware, initially, you maintain the option to make new backups from your older HSMs if that need arises before your 5.x or 6.x HSMs are decommissioned.

> Until you are ready to advance further, maintain your new 7.x HSMs at firmware older than Luna HSM Firmware 7.7.0 (Luna HSM Firmware 7.4.2is recommended) - with Luna Backup HSM G5 Firmware 6.26.0, you can backup Luna 7.x HSMs up to Luna HSM Firmware 7.4.2 which benefits from many fixes and improvements / features compared to prior 7.x versions.

> In the case of multifactor quorum-authenticated HSMs, the Luna PED must be at minimum Luna PED Firmware 2.7.1.

> Assuming that everything goes well in your migration from HSMs version 5.x or 6.x to new version 7.x HSMs, you can then upgrade the Luna Backup HSM G5 and Luna PED to their latest firmware versions, and also upgrade the new Luna 7.x HSMs to the target Luna HSM Firmware 7.7.0 or newer.

> These are the eventual target versions for all components:

- Luna PED Firmware 2.9.0 or newer (for the USB-powered PED) or Luna PED Firmware 2.7.4 or newer (for the adapter-powered PED) are the minimum required to support Luna HSMs at firmware 7.7.x
- Luna Backup HSM G5 Firmware 6.28.0 or newer, which is needed to support backup of Luna HSMs at firmware 7.7.x, and is compatible with Luna HSM Client 10.3.0.
- Luna HSM Client 10.3.0 or newer is needed to support Luna HSMs at Luna HSM Firmware 7.7.0 or newer

Refer to Luna HSM Firmware Releases to find the latest FIPS and CC evaluated versions and to the NIST and Common Criteria sites for completed and in-progress applications).

Migration methods

The three migration methods used in this guide are:

> Backup and restore

The backup and restore method uses the LunaCM **partition archive backup** command to backup key material on an HSM (5.x or 6.x) partition and the Restore command to then restore this material to an HSM 7.x partition.

> Cloning

The cloning method uses the LunaCM **partition clone** command to clone from an HSM (5.x or 6.x) partition to an HSM 7.x partition. It is also referred to as slot-to slot cloning.

> Cloning using an HA group

The HA group method uses the LunaCM **ha synchronize** command on members of a temporary HA group consisting of a 5.x or 6.x HSM and a 7.x HSM, set up solely for the purpose of migration. After migration, this group should be removed since the members are not using the same software version.

All three methods invoke the cloning protocols, behind the commands that you specifically issue. This table is a quick summary of the options and cloning protocols that are available for migration from various source levels, up to, and including Luna HSM Firmware 7.8.0 (or newer) and client Luna HSM Client 10.5.0 (or newer), with the cloned objects going to targets that are at Luna HSM Firmware 7.8.0 (depending on settings at the target).

Source slot	Source cloning protocol	Destination cloning protocol (firmware 7.8.0)	Cloning protocol used
Firmware version < 7.7.0	CPv1	CPv1, CPv3, CPv4	CPv1
Firmware version = 7.7.0	CPv3	CPv1 (disabled here by partition policy 42), CPv3 , CPv4 (not matched at source)	CPv3
	CPv3	CPv1 (enabled here by partition policy 42), CPv3 (disabled because CPv1 enabled), CPv4 (not matched at source)	N/A

Source slot	Source cloning protocol	Destination cloning protocol (firmware 7.8.0)	Cloning protocol used
Firmware version > 7.7.0 < 7.8.0	CPv1-out, CPv3	CPv1 (enabled by partition policy 42), CPv3 (disabled because CPv1 enabled), CPv4 (not matched at source)	CPv1
	CPv1-out, CPv3	CPv1 (disabled by partition policy 42), CPv3 (enabled because CPv1 disabled), CPv4 (not matched at source)	CPv3
Firmware version = 7.8.0	CPv1-out, CPv3, CPv4	CPv1 (enabled by partition policy 42), CPv3, CPv4 (these are disabled if CPv1 ensabled)	CPv1
	CPv1-out, CPv3, CPv4	CPv1 (not enabled by partition policy 42), CPv3, CPv4 (these are enabled if CPv1 disabled)	CPv4 (if both slots have at least one CPv4 cipher-suite in common), otherwise CPv3, unless CPv3 cipher-suite is also disabled, then N/A

Preconditions

Each migration procedure in this document is prefaced by a "Preconditions" section that specifies the hardware and software requirements along with any assumptions the procedure is using to perform the migration steps. Examples are a 5.x or 6.x HSM, a 7.x HSM, 5.x, 6.x or 7.x client software, user roles and the slot #s used in the procedure.

Roles required for migration

The following partition roles are needed to migrate key material:

- Partition Security Officer. The partition security officer role is needed to perform LunaCM HA operations and to create the Crypto Officer role.
- Partition Crypto Officer. The partition Crypto Officer role is needed to perform LunaCM backup/restore and cloning operations.

NOTE When logging in to a partition, be mindful of whether you're working with pre-PPSO or PPSO firmware. Use the **partition login** command if your HSM has pre-PPSO firmware (version 6.21.2 and earlier). Use the **role login** command if your HSM has PPSO firmware (version 6.22.0 and later). Also, with PPSO firmware 6.22.0 and later (up to but not including firmware 7.x), be careful with user names; that is, type **Crypto Officer** in full (is case sensitive) and not the abbreviation **co**.

In firmware version release 7.x, partition login name requirements allow for abbreviations. That is, you can log in using **po** for Partition Security Officer or **co** for Crypto Officer.

Luna Network HSM 5.x/6.x to Luna Network HSM 7

This chapter describes how to migrate your key material from a release 5.x or 6.x Luna Network HSM partition to a Luna Network HSM 7 partition. You can migrate your key material using one of the following three methods:

> "Backup and Restore" below

> "Cloning" on page 391

> "Cloning Using an HA Group" on page 396

TIP **When cloning objects:**

- by direct clone command, or
- by backup/restore, or
- by synchronization in an HA group)

...between 5.x or 6.x source HSMs and 7.x target HSMs, the common domain between the HSMs *must be* the designated *primary domain* on any HSM that is at firmware version 7.8.0 or newer.

This is because the cloning protocol on HSMs prior to firmware 7.8.0 is unaware of the ability to have multiple domains and therefore the older HSM can interact with *only* the primary domain on the firmware 7.8.0+ HSM. So, if the domain of the old HSM exists on the firmware 7.8.0-and-newer HSM, set that domain to be Primary before cloning. If it doesn't already exist, set it to -primary while you are creating it.

Additionally, when migrating keys from a Multifactor Quorum (PED)-authenticated 5.x or 6.x HSM to a password-authenticated Luna 7 HSM,

- password-authenticated HSMs do not have the ability to use Remote PED
- so a PED must be connected directly/locally to the password-authenticated HSM.

Backup and Restore

Cryptographic key material can be backed up and then restored to a Luna Network HSM 7 partition using a Luna Backup HSM.

The new configuration's operating system must be compatible with both the new 7.x and the old 5.x/6.x hardware. Consult the 5.x/6.x CRN for a list of compatible operating systems.

To backup and restore cryptographic keys from one HSM to another, the HSMs must share the same cloning domain. For password-authenticated HSMs, this domain should have been specified when the partition was initialized. For multifactor quorum-authenticated HSMs, the red key determines the cloning domain. You will need the same red PED key that was imprinted during 5.x/6.x partition creation to initialize the 7.x partition (see "Initializing an Application Partition" on page 332).

The Luna HSM Client 7/10 software should be installed, and the connection to both the source and destination partitions verified, before attempting this procedure (see "Luna HSM Client Software Installation" on page 20 for details). The source and destination partitions must both be assigned to the client machine issuing the backup and restore commands (see "Client-Partition Connections" on page 107 for details). Use **slot list** to ensure both partitions are visible to the client.

> **CAUTION!** If you are upgrading/migrating from a Luna HSM at 5.x or 6.x be aware that Partition Policy 28 - Allow Key Management Functions is non-destructive to change in your old HSM, but defaults to being destructive OFF-to-ON for 7.x HSMs. This could have impact on the function of your applications. The workaround is to
>
> > create a Partition Policy Template file, in which the destructiveness for policy 28 is set to off, and then
> >
> > create your new partitions using the template file.

Preconditions

The following instructions assume that:

> the 10.x client software has been installed
>
> an uninitialized partition has been created on the 7.x HSM
>
> the source and destination partitions are both registered with the client (visible)
>
> the source partition's security policy allows cloning of private and secret keys

In the following example:

> Slot 0: the source 5.x/6.x partition
>
> Slot 1: the destination 7.x partition
>
> Slot 2: the Backup HSM partition

> **NOTE** Partition login name requirements have changed with the hardware versions. With release 7.x , you can log in using the abbreviated **po** (Partition Security Officer) or **co** (Crypto Officer).

To migrate cryptographic keys from a 5.x/6.x partition to a 7.x partition using a Backup HSM

Follow these steps to back up all cryptographic material on a 5.x/6.x partition to a Backup HSM, and restore to a new 7.x partition.

1. Run LunaCM, set the current slot to the 7.x partition, and initialize the partition and the Partition SO role.

   slot set -slot 0

   partition init -label <7.x_partition_label>

a. If you are backing up a multifactor quorum-authenticated 5.x/6.x partition, use the 5.x/6.x partition's red key when prompted.

b. If you are backing up a password-authenticated 5.x/6.x partition, enter the same cloning domain when prompted.

2. Log in as the **po** (Partition Security Officer) and initialize the **co** (Crypto Officer) role.

   role login -name po

   role init -name co

   If you are backing up a multifactor quorum-authenticated 5.x/6.x partition, you can create an optional challenge secret for the Crypto Officer.

   role createchallenge -name co -challengeSecret <password>

3. Connect your backup HSM and make sure it is visible to the client, along with the 5.x/6.x and 7.x HSMs.

4. Set the current slot to the source 5.x/6.x slot.

   slot list

   slot set -slot 0

5. Log in as the Crypto Officer.

   > **NOTE** Be mindful of whether you're working with pre-PPSO or PPSO firmware and use the **partition login** or **role login** commands as specified below. Also, with PPSO firmware 6.22.0 and up, be careful with user names, i.e., type **Crypto Officer** in full (is case sensitive) and not **co**.

   a. If you are cloning a release 5.x or 6.x pre-PPSO partition (up to and including Firmware 6.21.2), use:

      partition login

   b. If you are cloning a release 6.x PPSO partition (Firmware 6.22.0 and up) , use:

      role login -name Crypto Officer

6. Optional: To verify the objects in the 5.x/6.x partition to be cloned, issue the "partition contents" command.

   partition contents

7. Back up the 5.x/6.x partition contents to the Backup HSM.

   partition archive backup -slot 2 -partition <backup_label>

   a. If you are backing up a multifactor quorum-authenticated 5.x/6.x partition, use the 5.x/6.x partition's red key when prompted.

   b. If you are backing up a password-authenticated 5.x/6.x partition, enter the same cloning domain when prompted.

   Optionally, verify that all objects were backed up successfully on the Backup HSM by checking the partition contents.

8. Set the current slot to the 7.x partition, log in as the Crypto Officer, and restore from backup.

   slot set -slot 1

   role login -name co

partition archive restore -slot 2 -partition <backup_label>

Afterwards, you can verify the partition contents on the 7.x partition:

partition contents

Cloning

The simplest method of migrating key material to a new 7.x partition is slot-to-slot cloning. This procedure copies all permitted cryptographic material from a 5.x/6.x Luna Network HSM partition to a Luna Network HSM 7 partition.

The new configuration's operating system must be compatible with both the new 7.x and the old 5.x/6.x hardware. Consult the 5.x/6.x CRN for a list of compatible operating systems.

To clone cryptographic keys from one HSM to another, the HSMs must share the same cloning domain. For password-authenticated HSMs, this domain should have been specified when the partition was initialized. For multifactor quorum-authenticated HSMs, the red key determines the cloning domain. You will need the same red key that was imprinted during 5.x/6.x partition creation to initialize the 7.x partition (see "Initializing an Application Partition" on page 332).

The Luna HSM Client 7/10 software should be installed, and the connection to both the source and destination partitions verified, before attempting this procedure (see "Luna HSM Client Software Installation" on page 20 for details). The source and destination partitions must both be assigned to the client machine issuing the cloning commands (see "Client-Partition Connections" on page 107 for details). Use **slot list** to ensure both partitions are visible to the client.

If the source partition contains asymmetric keys, its security policy must allow cloning of private and secret keys. Use lunacm:> **partition showpolicies** to ensure that your source partition's security template allows this. If the 5.x/6.x HSM's security template does not allow cloning of private/secret keys, the HSM Admin may be able to turn this feature on using lunacm:> **partition changepolicy**.

> **CAUTION!** Check your source partition policies and adjust them to be sure you can clone private and symmetric keys. Depending on the configuration of your partition and HSM, these policies may be destructive.

Preconditions

The following instructions assume that:

- the Luna HSM Client 7/10 software has been installed
- an uninitialized partition has been created on the Luna Network HSM 7
- the source and destination partitions must be registered with the client (visible)
- the source 5.x/6.x partition's security policy allows cloning of private and secret keys

In the following examples:

- Slot 0: the source 5.x/6.x partition
- Slot 1: the destination 7.x partition

> **NOTE** Partition login name requirements have changed with the hardware versions. With release 7.x, you can log in using the abbreviated **PO** (Partition Security Officer) or **CO** (Crypto Officer).

To clone cryptographic keys from a 5.x/6.x partition to a 7.x partition (pre-firmware 7.8.0)

Follow these steps to clone all cryptographic material on a 5.x/6.x partition to a 7.x partition.

1. Run LunaCM, set the current slot to the 7.x partition, and initialize the Partition SO role.

   slot list

   slot set -slot 1

   partition init -label <7.x_partition_label>

   a. If you are cloning a multifactor quorum-authenticated 5.x/6.x partition, use the 5.x/6.x partition's red key when prompted.

   b. If you are cloning a password-authenticated 5.x/6.x partition, enter the same cloning domain when prompted.

2. Log in as the **po** (Partition Security Officer) and initialize the **co** (Crypto Officer) role.

   role login -name po

   role init -name co

   If you are cloning a multifactor quorum-authenticated 5.x/6.x partition, you can create an optional challenge secret for the Crypto Officer.

   role createchallenge -name co -challengesecret <password>

3. Set the current slot to the source 5.x/6.x slot, log in as the Crypto Officer.

   slot set -slot 0

   > **NOTE** Be mindful of whether you're working with pre-PPSO or PPSO firmware and use the "partition login" or "role login" commands as specified below. Also, with PPSO firmware 6.22.0 and up, be careful with user names, i.e., type "Crypto Officer" in full (is case sensitive) and not "co".

   a. If you are cloning a release 5.x or 6.x pre-PPSO partition (up to and including Firmware 6.21.2), use:

      partition login

   b. If you are cloning a release 6.x PPSO partition (Firmware 6.22.0 and up) , use:

      role login -name Crypto Officer

4. Optional: To verify the objects in the 5.x/6.x partition to be cloned, issue the "partition contents" command.

   partition contents

5. Clone the objects to the 7.x partition slot (see partition clone for correct syntax).

   partition clone -objects 0 -slot 1

   Afterward, you can set the current slot to the 7.x partition and verify that all objects have cloned successfully.

   slot set -slot 1

role login -name co -password <password>

partition contents

You should see the same number of objects that existed on the 5.x/6.x HSM. You can now decommission the old 5.x/6.x HSM.

To clone partition objects from on-premises multifactor quorum-authenticated partition to on-premises password-authenticated partition using Luna HSM Firmware 7.8.0 or newer

Requires Luna HSM Client 10.5.0 or newer.

This procedure is for :

> an on-premises multifactor quorum-authenticated Luna Network HSM 7 partition as the source, which could be for:

- a routine cloning between two HSM partitions that are at Luna HSM Firmware 7.8.0 or newer,
- *migration cloning of keys and objects* from a legacy HSM partition (firmware 5.x, 6.x), or from firmware older than Luna HSM Firmware 7.8.0.

> an on-premises password-authenticated Luna Network HSM 7 partition as the target (at Luna HSM Firmware 7.8.0 or newer).

1. Ensure that the two partitions can both use a common cloning protocol
   a. if the source has partition policy 42 - Enable CPv1 on , then that protocol is chosen and others are disabled (or if the source has firmware earlier than Luna HSM Firmware 7.7.0, meaning that CPv1 is the only protocol); this imposes restrictions on operations, see "Cloning Protocols and Cipher Suite Selection" on page 217

      lunacm:> **slot set -slot** <slotnum>

      lunacm:> **partition showpolicies**

   b. if partition policy 42 - Enable CPv1 is OFF, then negotiation of common cipher suites is attempted between partitions; this is preferred when available.

   c. if CPv1 has not been forced, and all cipher suites for CPv4 have been disabled on one of the participating partitions, then only CPv3 remains and a common CPv4 cipher suite cannot be negotiated.

2. Ensure that the source and target partitions have a cloning domain in common.
   a. If the source is a Luna HSM Firmware 7.8.0 or newer partition, then it can accept the target's domain string (password-authenticated) into the multifactor quorum-authenticated source partition, avoiding the need to connect a Luna PED to the target, in which case, skip to step **d.**; otherwise, go to step **b.**

   b. If the source multifactor quorum-authenticated partition is at any firmware version older than Luna HSM Firmware 7.8.0, it cannot have more than one domain, so its PED key secret must be brought to the target; connect a Luna PED locally to the password-authenticated target.

   c. In LunaCM, set the active slot to the target partition and log in as Partition SO.

      lunacm:> **slot set -slot** <slotnum>

      lunacm:> **role login -name po**

   d. View the partition domains and note their labels.

      lunacm:> **partition domainlist**

e. If the two partitions share a common domain, proceed to cloning.

f. If the two partitions do not share a common domain, then make room, if necessary, by deleting one domain you can do without.

lunacm:>partition domaindelete

g. Add a domain that matches one from the other partition.

lunacm:> **partition domainadd -domain** <text domain secret> **-domainlabel** <label of the text domain being duplicated>

3. In LunaCM, set the active slot to the source partition and log in as Crypto Officer.

lunacm:> **slot set -slot** <slotnum>

lunacm:> **role login -name co**

4. [Optional] View the partition objects and their object handles.

lunacm:> **partition contents**

5. Clone objects on the partition to the target partition by specifying the target slot. You can choose which objects to clone by specifying a comma-separated list of object handles, or specify **all** to clone all objects on the partition. Present the target partition's Crypto Officer credential when prompted.

lunacm:> **partition clone -slot** <slotnum> **-objects** <comma-separated_list/**all**>

The specified objects are cloned to the target partition. Any objects that already exist on the target are not cloned.

6. [OPTIONAL] You can retain an added domain on a partition as long as it remains useful

- as long as the partition contains objects encrypted under that particular domain, or
- while you think the current partition might clone (as source or as target) objects with a partition or service using that domain.

Or you can delete a domain using partition domaindelete if it is no longer needed.

To clone partition objects from on-premises password-authenticated partition to on-premises multifactor quorum-authenticated partition using Luna HSM Firmware 7.8.0 or newer

Requires Luna HSM Client 10.5.0 or newer.

This procedure is for:

> an on-premises password-authenticated Luna Network HSM 7 partition as the source, which could be for:

- a routine cloning between two HSM partitions that are at Luna HSM Firmware 7.8.0 or newer,
- *migration cloning of keys and objects* from a legacy HSM partition (firmware 5.x, 6.x), or from firmware older than Luna HSM Firmware 7.8.0.

> an on-premises multifactor quorum-authenticated Luna Network HSM 7 partition as the target (at Luna HSM Firmware 7.8.0 or newer).

1. Ensure that the two partitions can both use a common cloning protocol.

a. for HSMs (both legacy and 7.x) before Luna HSM Firmware 7.7.0, only protocol CPv1 is available

b. for Luna HSM Firmware 7.7.1 and newer, if partition policy 42 - Enable CPv1 is ON, then that protocol is chosen and others are disabled

lunacm:> **slot set -slot** <slotnum>

lunacm:> **partition showpolicies**

c. if partition policy 42 - Enable CPv1 is OFF, then negotiation of common cipher suites is attempted between partitions; this is preferred when available.

d. if CPv1 has not been forced, and all cipher suites for CPv4 have been disabled on one of the participating partitions, then only CPv3 remains and a common CPv4 cipher suite cannot be negotiated.

2. Ensure that the source and target partitions have a cloning domain in common.

a. In LunaCM, set the active slot to the target multifactor quorum-authenticated partition and log in as Partition SO (po).

lunacm:> **slot set -slot** <slotnum>

lunacm:> **role login -name po**

b. View the partition domains and note their labels.

lunacm:> **partition domainlist**

c. If the two partitions share a common domain, proceed to cloning.

d. If the two partitions do not share a common domain, then make room, if necessary, by deleting one domain you can do without on the target partition.

lunacm:> partition domaindelete

e. Add a domain that matches one from the source partition

lunacm:> **partition domainadd -domain** <text domain secret> **-domainlabel** <label of the text domain being duplicated>

3. In LunaCM, set the active slot to the source partition and log in as Crypto Officer.

lunacm:> **slot set -slot** <slotnum>

lunacm:> **role login -name co**

4. [Optional] View the partition objects and their object handles.

lunacm:> **partition contents**

5. Clone objects on the current partition to the target partition by specifying the target slot. You can choose which objects to clone by specifying a comma-separated list of object handles, or specify **all** to clone all objects on the partition. Present the target partition's Crypto Officer credential when prompted.

lunacm:> **partition clone -slot** <slotnum> **-objects** <comma-separated_list/**all**>

The specified objects are cloned to the target partition. Any objects that already exist on the target are not cloned.

6. [OPTIONAL] You can retain an added domain on a partition as long as it remains useful

- as long as the partition contains objects encrypted under that particular domain, or
- while you think the current partition might clone (as source or as target) objects with a partition or service using that domain.

Or you can delete a domain using partition domaindelete if it is no longer needed.

Cloning Using an HA Group

High Availability (HA) groups duplicate key material between the HSMs in the group. This function can be used to copy all cryptographic key material from a 5.x/6.x Luna Network HSM partition to a new Luna Network HSM 7 partition.

The new configuration's operating system must be compatible with both the new 7.x and the old 5.x/6.x hardware. Consult the 5.x/6.x CRN for a list of compatible operating systems.

To clone cryptographic keys from one HSM to another, the HSMs must share the same cloning domain. For password-authenticated HSMs, this domain should have been specified when the partition was initialized. For multifactor quorum-authenticated HSMs, the red key determines the cloning domain. You will need the same red key that was imprinted during 5.x/6.x partition creation to initialize the 7.x partition (see "Initializing an Application Partition" on page 332).

The Luna HSM Client 7/10 software should be installed, and the connection to both the source and destination HSM partitions verified, before attempting this procedure (see "Luna HSM Client Software Installation" on page 20 for details). The source and destination partitions must both be assigned to the client machine issuing the cloning commands (see "Client-Partition Connections" on page 107 for details). Use **slot list** to ensure both partitions are visible to the client.

> **NOTE** It is not recommended to maintain an HA group with different versions of the Luna Network HSM hardware.

Preconditions

The following instructions assume that:

- the 7.x client software has been installed
- an uninitialized partition has been created on the 7.x Luna Network HSM 7
- the source and destination partitions are both registered with the client (visible)

In the following examples:

- Slot 0 = the source 5.x/6.x partition
- Slot 1 = the destination 7.x partition

> **NOTE** Partition login name requirements have changed between hardware versions. With release 7.x, you can log in using the abbreviated **po** (Partition Security Officer) or **co** (Crypto Officer).

To clone cryptographic keys from a 5.x/6.x partition to a 7.x partition using an HA group

Follow these steps to copy cryptographic material from an 5.x/6.x partition to a new 7.x partition by creating an HA group that includes both partitions.

1. Run LunaCM, set the current slot to the SA7 partition, and initialize the Partition SO role.

   slot set -slot 1

   partition init -label <7.x_partition_label>

a. If you are cloning a multifactor quorum-authenticated 5.x/6.x partition, use the 5.x/6.x partition's red key when prompted.

b. If you are cloning a password-authenticated 5.x/6.x partition, enter the same cloning domain when prompted.

2. Log in as the **po** (Partition Security Officer) and initialize the **co** (Crypto Officer) role.

   role login -name po

   role init -name co

   If you are cloning a multifactor quorum-authenticated 5.x/6.x partition, create a challenge secret for the Crypto Officer. This is required to set an HA activation policy.

   role createchallenge -name co -challengesecret <password>

3. Set the current slot to the source 5.x/6.x slot, log in as the Crypto Officer.

   slot set -slot 0

   > **NOTE** Be mindful of whether you're working with pre-PPSO or PPSO firmware and use the **partition login** or **role login** commands as specified below. Also, with PPSO firmware 6.22.0 and up, be careful with user names, i.e., type **Crypto Officer** in full (is case sensitive) and not **co**.

   a. If you are cloning a release 5.x or 6.x pre-PPSO partition (up to and including Firmware 6.21.2), use:

      partition login

   b. If you are cloning a release 6.x PPSO partition (Firmware 6.22.0 and up) , use:

      role login -name Crypto Officer

4. Optional: To verify the objects in the 5.x/6.x partition to be cloned, use:

   partition contents

5. Using LunaCM, create an HA group of the 5.x/6.x slot and the 7.x slot.

   > **NOTE** HA requires that all members have an activation policy set. See "Activation on Multifactor Quorum-Authenticated Partitions" on page 373 for details.

   a. Via LunaSH, log in as Security Officer and set policy 22 on the 5.x/6.x partition:

      partition changepolicy -partition <5.x_partition_label> **-policy 22 -value 1**

   b. In LunaCM, log in to the 7.x partition as Partition Security Officer, and set the activation policy from the client machine:

      slot set -slot 1

      role login -name po

      partition changepolicy -policy 22 -value 1

   c. Create the HA group with the 5.x/6.x partition as the primary partition. Select the "copy" option to preserve objects.

      hagroup creategroup -label <group_label> **-slot 0 -password** <password>

d. Add the 7.x partition slot to the HA group. Repeat this step to add multiple 7.x partitions to the group.

hagroup addmember -group <group_label> **-slot 1 -password** <password>

6. Synchronize the group to clone the objects to the 7.x member(s).

hagroup synchronize -group <group_label> **-password** <password>

7. Check synchronization status of the group.

hagroup listgroups

Notice the entry "`Needs sync: no`". This means that the objects have been successfully cloned among all members of the HA group. You can also log in to the 7.x slot as the Crypto Officer and check the partition contents.

Luna USB HSM 6.x to Luna Network HSM 7

This chapter describes how to migrate your key material from a release 6.x Luna USB HSM partition to a Luna Network HSM 7 partition. You can migrate your key material using one of the following methods:

> "Backup and Restore" below

> "Cloning" on page 401

TIP When cloning objects:

- by direct clone command, or
- by backup/restore, or
- by synchronization in an HA group)

...between 5.x or 6.x source HSMs and 7.x target HSMs, the common domain between the HSMs *must be* the designated *primary domain* on any HSM that is at firmware version 7.8.0 or newer.

This is because the cloning protocol on HSMs prior to firmware 7.8.0 is unaware of the ability to have multiple domains and therefore the older HSM can interact with *only* the primary domain on the firmware 7.8.0+ HSM. So, if the domain of the old HSM exists on the firmware 7.8.0-and-newer HSM, set that domain to be Primary before cloning. If it doesn't already exist, set it to -primary while you are creating it.

Additionally, when migrating keys from a Multifactor Quorum (PED)-authenticated 5.x or 6.x HSM to a password-authenticated Luna 7 HSM,

- password-authenticated HSMs do not have the ability to use Remote PED
- so a PED must be connected directly/locally to the password-authenticated HSM.

Backup and Restore

Cryptographic key material can be backed up from a release 6.x Luna USB HSM partition and then restored to a Luna Network HSM 7 partition using a Luna Backup HSM. The following procedure performs a backup of a 6.x partition on an older operating system to a Luna Backup HSM. The Backup HSM is then moved to a newer operating system where the 6.x key material is restored to a 7.x partition.

Consult the 6.x/7.x CRN for a list of compatible operating systems.

To clone cryptographic keys from one HSM to another, the HSMs must share the same cloning domain. For password-authenticated HSMs, this domain should have been specified when the 6.x partition was initialized. For multifactor quorum-authenticated HSMs, the red key determines the cloning domain. You will need the same red key that was imprinted during 6.x partition creation to initialize the 7.x partition (see "Initializing an Application Partition" on page 332).

HSM Client software must be installed before attempting this procedure (see "Luna HSM Client Software Installation" on page 20 for details). The source and destination partitions must be assigned to the client machine issuing the backup or restore command (see "Client-Partition Connections" on page 107 for details). Use **slot list** to ensure both partitions are visible to the client.

Preconditions

On the older operating system, the following instructions assume that:

> 6.x Luna HSM Client Software is installed

> the source 6.x partition is visible

> the source partition's security policy allows cloning of private and secret keys

> the destination Backup HSM partition is visible

On the new operating system, the following instructions assume that:

> Luna HSM Client 7/10 software is installed

> you have created an uninitialized partition on the Luna Network HSM 7

> the destination 7.x partition is registered with the client software (visible)

> the source Backup HSM partition's security policy allows cloning of private and secret keys

Slots used in the following instructions:

> On the older operating system running 6.x client software:
  - Slot 0: the source 6.x partition
  - Slot 2: the destination Luna Backup HSM partition

> On the new operating system running 7.x client software:
  - Slot 1: the destination 7.x partition
  - Slot 2: the source Luna Backup HSM partition (with the backup of the 6.x partition)

> **NOTE** Partition login name requirements have changed with the hardware versions. With release 7.x , you can log in using the abbreviated **PO** (Partition Security Officer) or **CO** (Crypto Officer).

To backup/restore cryptographic keys from a 5.x/6.x partition to a 7.x partition using a Backup HSM

Follow these steps to back up all cryptographic material on a 5.x/6.x partition to a Luna Backup HSM, and restore to a new 7.x partition.

1. On the old operating system running 5.x/6.x client software, run LunaCM and set the current slot to the 5.x/6.x partition.

   slot list

slot set -slot 0

2. Log in as the Crypto Officer.

> **NOTE** Be mindful of whether you're working with pre-PPSO or PPSO firmware and use the "partition login" or "role login" commands as specified below. Also, with PPSO firmware 6.22.0 and up, be careful with user names, i.e., type "Crypto Officer" in full (is case sensitive) and not "co".

   a. If you are backing up a release 5.x or 6.x pre-PPSO partition (up to and including Firmware 6.21.2), use:

      partition login

   b. If you are backing up a release 6.x PPSO partition (Firmware 6.22.0 and up), use:

      role login -name Crypto Officer

3. Optional: To verify the objects in the 5.x/6.x partition to be backed up, use:

   partition contents

4. Back up the 5.x/6.x partition contents to the Luna Backup HSM.

   partition archive backup -slot 2 -partition <backup_label>

   a. If you are backing up a multifactor quorum-authenticated 5.x/6.x partition, use the 5.x/6.x partition's red PED key when prompted.

   b. If you are backing up a password-authenticated 5.x/6.x partition, enter the same cloning domain when prompted.

   Optionally, verify that all objects were backed up successfully on the Luna Backup HSM by issuing the **partition contents** command.

5. Move the Luna Backup HSM (with the backup of the 5.x/6.x partition) to the new operating system running the 7.x client software, and make sure it is visible to the client along with the 7.x HSM.

6. On the new operating system running the 7.x client software, run LunaCM, set the current slot to the 7.x partition, and initialize the partition and the PPSO role.

   slot set -slot 1

   partition init -label <7.x_partition_label>

   a. If you are backing up a multifactor quorum-authenticated 5.x/6.x partition, use the 5.x/6.x partition's red PED key when prompted.

   b. If you are backing up a password-authenticated 5.x/6.x partition, enter the same cloning domain when prompted.

7. Log in as the **po** (Partition Security Officer) and initialize the **co** (Crypto Officer) role.

   role login -name po

   role init -name co

   If you are backing up a PED-authenticated 5.x/6.x partition, you can create an optional challenge secret for the Crypto Officer.

   role createchallenge -name co -challengesecret <password>

8. Set the current slot to the 7.x partition, log in as the Crypto Officer, and restore from backup.

slot set -slot 1

role login -name co

partition archive restore -slot 2 -partition <backup_label>

Afterwards, you can verify the partition contents on the 7.x partition:

partition contents

Cloning

The simplest method of migrating key material to a new 7.x partition is slot-to-slot cloning. This procedure copies all permitted cryptographic material from a 6.x Luna USB HSM partition to a Luna Network HSM 7 partition.

> **NOTE** The Luna cloning protocol has been updated several times since the previous generation of Luna HSMs, for standards compliance and added security. If you are planning to migrate your key material from a Luna 5/6 in FIPS mode to a Luna 7 HSM in FIPS mode, you must use an intermediate firmware version before updating to Luna HSM Firmware 7.7.1 or newer. Thales recommends migrating to Luna Network HSM 7 7 with one of the following FIPS-validated firmware versions:
>
> > Luna HSM Firmware 7.3.3
> > Luna HSM Firmware 7.0.3
>
> After the migration process is complete, you can update the Luna Network HSM 7 to Luna HSM Firmware 7.7.1 or newer (see Updating the Luna HSM Firmware).

The new configuration's operating system must be compatible with both the new 7.x and the old 6.x hardware. Consult the 6.x CRN for a list of compatible operating systems.

To clone cryptographic keys from one HSM to another, the HSMs must share the same cloning domain. For password-authenticated HSMs, this domain should have been specified when the partition was initialized. For multifactor quorum-authenticated HSMs, the red key determines the cloning domain. You will need the same red key that was imprinted during 6.x partition creation to initialize the 7.x partition (see "Initializing an Application Partition" on page 332).

The Luna HSM Client 7/10 software should be installed, and the connection to both the source and destination partitions verified, before attempting this procedure (see "Luna HSM Client Software Installation" on page 20 for details). The source and destination partitions must both be assigned to the client machine issuing the cloning commands (see "Client-Partition Connections" on page 107 for details). Use **slot list** to ensure both partitions are visible to the client.

If the source partition contains asymmetric keys, its security policy must allow cloning of private and secret keys. Use lunacm:> **partition showpolicies** to ensure that your source partition's security template allows this. If the 6.x HSM's security template does not allow cloning of private/secret keys, the HSM Admin may be able to turn this feature on using lunacm:> **partition changepolicy**.

> **CAUTION!** Check your source partition policies and adjust them to be sure you can clone private and symmetric keys. Depending on the configuration of your partition and HSM, these policies may be destructive.

Preconditions

The following instructions assume that:

- the 7.x client software has been installed
- an uninitialized partition has been created on the Luna Network HSM 7
- the destination 7.x partition must be registered with the client (visible)
- the source 6.x partition's security policy allows cloning of private and secret keys

In the following examples:

- Slot 0: the source 6.x partition
- Slot 1: the destination 7.x partition

> **NOTE** Partition login name requirements have changed with the hardware versions. With release 7.x, you can log in using the abbreviated **PO** (Partition Security Officer) or **CO** (Crypto Officer).

To clone cryptographic keys from a 5.x/6.x partition to a 7.x partition (pre-firmware 7.8.0)

Follow these steps to clone all cryptographic material on a 5.x/6.x partition to a 7.x partition.

1. Run LunaCM, set the current slot to the 7.x partition, and initialize the Partition SO role.

   slot list

   slot set -slot 1

   partition init -label <7.x_partition_label>

   a. If you are cloning a multifactor quorum-authenticated 5.x/6.x partition, use the 5.x/6.x partition's red key when prompted.

   b. If you are cloning a password-authenticated 5.x/6.x partition, enter the same cloning domain when prompted.

2. Log in as the **po** (Partition Security Officer) and initialize the **co** (Crypto Officer) role.

   role login -name po

   role init -name co

   If you are cloning a multifactor quorum-authenticated 5.x/6.x partition, you can create an optional challenge secret for the Crypto Officer.

   role createchallenge -name co -challengesecret <password>

3. Set the current slot to the source 5.x/6.x slot, log in as the Crypto Officer.

   slot set -slot 0

   > **NOTE** Be mindful of whether you're working with pre-PPSO or PPSO firmware and use the "partition login" or "role login" commands as specified below. Also, with PPSO firmware 6.22.0 and up, be careful with user names, i.e., type "Crypto Officer" in full (is case sensitive) and not "co".

   a. If you are cloning a release 5.x or 6.x pre-PPSO partition (up to and including Firmware 6.21.2), use:

partition login

b. If you are cloning a release 6.x PPSO partition (Firmware 6.22.0 and up) , use:

role login -name Crypto Officer

4. Optional: To verify the objects in the 5.x/6.x partition to be cloned, issue the “partition contents” command.

partition contents

5. Clone the objects to the 7.x partition slot (see partition clone for correct syntax).

partition clone -objects 0 -slot 1

Afterward, you can set the current slot to the 7.x partition and verify that all objects have cloned successfully.

slot set -slot 1

role login -name co -password <password>

partition contents

You should see the same number of objects that existed on the 5.x/6.x HSM. You can now decommission the old 5.x/6.x HSM.

Luna PCIe HSM 5.x/6.x to Luna Network HSM 7

This chapter describes how to migrate your key material from a release 5.x or 6.x Luna PCIe HSM 7 partition to a Luna Network HSM 7 partition. You can migrate your key material using one of the following methods:

> "Backup and Restore" on the next page

> "Cloning when both source and target already have the same domain" on page 406

TIP When cloning objects:

- by direct clone command, or
- by backup/restore, or
- by synchronization in an HA group)

...between 5.x or 6.x source HSMs and 7.x target HSMs, the common domain between the HSMs *must be* the designated *primary domain* on any HSM that is at firmware version 7.8.0 or newer.

This is because the cloning protocol on HSMs prior to firmware 7.8.0 is unaware of the ability to have multiple domains and therefore the older HSM can interact with *only* the primary domain on the firmware 7.8.0+ HSM. So, if the domain of the old HSM exists on the firmware 7.8.0-and-newer HSM, set that domain to be Primary before cloning. If it doesn't already exist, set it to -primary while you are creating it.

Additionally, when migrating keys from a Multifactor Quorum (PED)-authenticated 5.x or 6.x HSM to a password-authenticated Luna 7 HSM,

- password-authenticated HSMs do not have the ability to use Remote PED
- so a PED must be connected directly/locally to the password-authenticated HSM.

Backup and Restore

Cryptographic key material can be backed up from a release 5.x or 6.x Luna PCIe HSM 7 partition and then restored to a Luna Network HSM 7 partition using a Luna Backup HSM. The following procedure performs a backup of a 5.x/6.x partition on an older operating system to a Luna Backup HSM. The Backup HSM is then moved to a newer operating system where the 5.x/6.x key material is restored to a 7.x partition.

Consult the 5.x/6.x/7.x CRN for a list of compatible operating systems.

To clone cryptographic keys from one HSM to another, the HSMs must share the same cloning domain. For password-authenticated HSMs, this domain should have been specified when the 5.x/6.x partition was initialized. For multifactor quorum-authenticated HSMs, the red key determines the cloning domain. You will need the same red key that was imprinted during 5.x/6.x partition creation to initialize the 7.x partition (see "Initializing an Application Partition" on page 332).

Luna HSM Client 7/10 software must be installed on both operating systems (older and new) before attempting this procedure (see "Luna HSM Client Software Installation" on page 20 for details). The destination partition must be assigned to the client machine (see "Client-Partition Connections" on page 107 for details). Use **slot list** to ensure partitions are visible to the client.

Preconditions

On the older operating system, the following instructions assume that:

> 5.x/6.x Luna HSM Client software is installed with "Luna Backup HSM" option selected.

> the source 5.x/6.x partition is visible

> the source partition's security policy allows cloning of private and secret keys

> the destination Backup HSM partition is visible

On the new operating system, the following instructions assume that:

> Luna HSM Client 7/10 software is installed with "Luna Backup HSM" option selected.

> you have created an uninitialized partition on the Luna Network HSM 7

> the destination 7.x partition is registered with the client software (visible)

> the source Backup HSM partition's security policy allows cloning of private and secret keys

Slots used in the following instructions:

On the older operating system running 5.x/6.x client software:

- Slot 0: the source 5.x/6.x partition
- Slot 2: the destination Luna Backup HSM partition

On the new operating system running 7.x client software:

- Slot 1: the destination 7.x partition
- Slot 2: the source Backup HSM partition (with the backup of the 5.x/6.x partition)

> **NOTE** Partition login name requirements have changed with the hardware versions. With release 7.x, you can log in using the abbreviated **po** (Partition Security Officer) or **co** (Crypto Officer).

To backup/restore cryptographic keys from a 5.x/6.x partition to a 7.x partition using a Backup HSM

Follow these steps to back up all cryptographic material on a 5.x/6.x partition to a Luna Backup HSM, and restore to a new 7.x partition.

1. On the old operating system running 5.x/6.x client software, run LunaCM and set the current slot to the 5.x/6.x partition.

   slot list

   slot set -slot 0

2. Log in as the Crypto Officer.

   > **NOTE** Be mindful of whether you're working with pre-PPSO or PPSO firmware and use the "partition login" or "role login" commands as specified below. Also, with PPSO firmware 6.22.0 and up, be careful with user names, i.e., type "Crypto Officer" in full (is case sensitive) and not "co".

   a. If you are backing up a release 5.x or 6.x pre-PPSO partition (up to and including Firmware 6.21.2), use:

      partition login

   b. If you are backing up a release 6.x PPSO partition (Firmware 6.22.0 and up), use:

      role login -name Crypto Officer

3. Optional: To verify the objects in the 5.x/6.x partition to be backed up, use:

   partition contents

4. Back up the 5.x/6.x partition contents to the Luna Backup HSM.

   partition archive backup -slot 2 -partition <backup_label>

   a. If you are backing up a multifactor quorum-authenticated 5.x/6.x partition, use the 5.x/6.x partition's red PED key when prompted.

   b. If you are backing up a password-authenticated 5.x/6.x partition, enter the same cloning domain when prompted.

   Optionally, verify that all objects were backed up successfully on the Luna Backup HSM by issuing the **partition contents** command.

5. Move the Luna Backup HSM (with the backup of the 5.x/6.x partition) to the new operating system running the 7.x client software, and make sure it is visible to the client along with the 7.x HSM.

6. On the new operating system running the 7.x client software, run LunaCM, set the current slot to the 7.x partition, and initialize the partition and the PPSO role.

   slot set -slot 1

   partition init -label <7.x_partition_label>

   a. If you are backing up a multifactor quorum-authenticated 5.x/6.x partition, use the 5.x/6.x partition's red PED key when prompted.

   b. If you are backing up a password-authenticated 5.x/6.x partition, enter the same cloning domain when prompted.

7. Log in as the **po** (Partition Security Officer) and initialize the **co** (Crypto Officer) role.

   role login -name po

   role init -name co

   If you are backing up a PED-authenticated 5.x/6.x partition, you can create an optional challenge secret for the Crypto Officer.

   role createchallenge -name co -challengesecret <password>

8. Set the current slot to the 7.x partition, log in as the Crypto Officer, and restore from backup.

   slot set -slot 1

   role login -name co

   partition archive restore -slot 2 -partition <backup_label>

   Afterwards, you can verify the partition contents on the 7.x partition:

   partition contents

Cloning when both source and target already have the same domain

The simplest method of migrating key material to a new 7.x partition is slot-to-slot cloning. This procedure copies all permitted cryptographic material from a 5.x/6.x Luna PCIe HSM 7 partition to a Luna Network HSM 7 partition.

> **NOTE** The Luna cloning protocol has been updated several times since the previous generation of Luna HSMs, for standards compliance and added security. If you are planning to migrate your key material from a Luna 5/6 in FIPS mode to a Luna 7 HSM in FIPS mode, you must use an intermediate firmware version before updating to Luna HSM Firmware 7.7.1 or newer. Thales recommends migrating to Luna Network HSM 7 7 with one of the following FIPS-validated firmware versions:
>
> > Luna HSM Firmware 7.3.3
>
> > Luna HSM Firmware 7.0.3
>
> After the migration process is complete, you can update the Luna Network HSM 7 to Luna HSM Firmware 7.7.1 or newer (see Updating the Luna HSM Firmware).

The new configuration's operating system must be compatible with both the new 7.x and the old 5.x/6.x hardware. Consult the 5.x/6.x CRN for a list of compatible operating systems.

To clone cryptographic keys from one HSM to another, the HSMs must share the same cloning domain. For password-authenticated HSMs, this domain should have been specified when the partition was initialized. For multifactor quorum-authenticated HSMs, the red key determines the cloning domain. You will need the same red key that was imprinted during 5.x/6.x partition creation to initialize the 7.x partition (see "Initializing an Application Partition" on page 332).

The Luna HSM Client 7/10 software should be installed, and the connection to both the source and destination partitions verified, before attempting this procedure (see "Luna HSM Client Software Installation" on page 20 for details). The destination partition must be assigned to the client machine issuing the cloning commands (see "Client-Partition Connections" on page 107 for details). Use the **slot list** command to ensure both partitions are visible to the client.

If the source partition contains asymmetric keys, its security policy must allow cloning of private and secret keys. Use lunacm:> **partition showpolicies** to ensure that your source partition's security template allows this. If the 5.x/6.x HSM's security template does not allow cloning of private/secret keys, the HSM Admin may be able to turn this feature on using lunacm:> **partition changepolicy**.

> **CAUTION!** Check your source partition policies and adjust them to be sure you can clone private and symmetric keys. Depending on the configuration of your partition and HSM, these policies might be destructive.

Preconditions

On the operating system running 5.x/6/x client software, verify:

- that the 5.x/6.x PCIe HSM partition's security policy allows cloning of private and secret keys
- all key material on the 5.x/6.x Luna PCIe HSM 7 partition to be cloned

Regarding the operating system running 7.x client software, the following instructions assume that:

- the Luna HSM Client 7/10 software has been installed with "Luna PCIe HSM 7" option selected.
- an uninitialized partition has been created on the 7.x HSM
- the destination 7.x HSM partition must be registered with the client (visible)
- the Luna PCIe HSM card (with 5.x/6.x key material) has been installed

Slots used in the following instructions:

- Slot 0: the source 5.x/6.x Luna PCIe HSM 7 partition
- Slot 1: the destination 7.x partition

> **NOTE** Partition login name requirements have changed with the hardware versions. With release 7.x, you can log in using the abbreviated **po** (Partition Security Officer) or **co** (Crypto Officer).

To clone cryptographic keys from a 5.x/6.x partition to a 7.x partition (pre-firmware 7.8.0)

Follow these steps to clone all cryptographic material on a 5.x/6.x partition to a 7.x partition.

1. Run LunaCM, set the current slot to the 7.x partition, and initialize the Partition SO role.

   slot list

   slot set -slot 1

   partition init -label <7.x_partition_label>

   a. If you are cloning a multifactor quorum-authenticated 5.x/6.x partition, use the 5.x/6.x partition's red key when prompted.

   b. If you are cloning a password-authenticated 5.x/6.x partition, enter the same cloning domain when prompted.

2. Log in as the **po** (Partition Security Officer) and initialize the **co** (Crypto Officer) role.

   role login -name po

   role init -name co

If you are cloning a multifactor quorum-authenticated 5.x/6.x partition, you can create an optional challenge secret for the Crypto Officer.

role createchallenge -name co -challengesecret <password>

3. Set the current slot to the source 5.x/6.x slot, log in as the Crypto Officer.

   slot set -slot 0

   > **NOTE** Be mindful of whether you're working with pre-PPSO or PPSO firmware and use the "partition login" or "role login" commands as specified below. Also, with PPSO firmware 6.22.0 and up, be careful with user names, i.e., type "Crypto Officer" in full (is case sensitive) and not "co".

   a. If you are cloning a release 5.x or 6.x pre-PPSO partition (up to and including Firmware 6.21.2), use:

      partition login

   b. If you are cloning a release 6.x PPSO partition (Firmware 6.22.0 and up) , use:

      role login -name Crypto Officer

4. Optional: To verify the objects in the 5.x/6.x partition to be cloned, issue the "partition contents" command.

   partition contents

5. Clone the objects to the 7.x partition slot (see partition clone for correct syntax).

   partition clone -objects 0 -slot 1

   Afterward, you can set the current slot to the 7.x partition and verify that all objects have cloned successfully.

   slot set -slot 1

   role login -name co -password <password>

   partition contents

   You should see the same number of objects that existed on the 5.x/6.x HSM. You can now decommission the old 5.x/6.x HSM.

Cloning keys and objects when source and target partitions have different cloning domains

To perform a migration

To clone partition objects from on-premises multifactor quorum-authenticated partition to on-premises password-authenticated partition using Luna HSM Firmware 7.8.0 or newer

Requires Luna HSM Client 10.5.0 or newer.

This procedure is for :

> an on-premises multifactor quorum-authenticated Luna Network HSM 7 partition as the source, which could be for:
>
> - a routine cloning between two HSM partitions that are at Luna HSM Firmware 7.8.0 or newer,
> - *migration cloning of keys and objects* from a legacy HSM partition (firmware 5.x, 6.x), or from firmware older than Luna HSM Firmware 7.8.0.

> an on-premises password-authenticated Luna Network HSM 7 partition as the target (at Luna HSM Firmware 7.8.0 or newer).

1. Ensure that the two partitions can both use a common cloning protocol
   a. if the source has partition policy 42 - Enable CPv1 on , then that protocol is chosen and others are disabled (or if the source has firmware earlier than Luna HSM Firmware 7.7.0, meaning that CPv1 is the only protocol); this imposes restrictions on operations, see "Cloning Protocols and Cipher Suite Selection" on page 217

      lunacm:> **slot set -slot** <slotnum>

      lunacm:> **partition showpolicies**

   b. if partition policy 42 - Enable CPv1 is OFF, then negotiation of common cipher suites is attempted between partitions; this is preferred when available.

   c. if CPv1 has not been forced, and all cipher suites for CPv4 have been disabled on one of the participating partitions, then only CPv3 remains and a common CPv4 cipher suite cannot be negotiated.
2. Ensure that the source and target partitions have a cloning domain in common.
   a. If the source is a Luna HSM Firmware 7.8.0 or newer partition, then it can accept the target's domain string (password-authenticated) into the multifactor quorum-authenticated source partition, avoiding the need to connect a Luna PED to the target, in which case, skip to step **d.**; otherwise, go to step **b.**

   b. If the source multifactor quorum-authenticated partition is at any firmware version older than Luna HSM Firmware 7.8.0, it cannot have more than one domain, so its PED key secret must be brought to the target; connect a Luna PED locally to the password-authenticated target.

   c. In LunaCM, set the active slot to the target partition and log in as Partition SO.

      lunacm:> **slot set -slot** <slotnum>

      lunacm:> **role login -name po**

   d. View the partition domains and note their labels.

      lunacm:> **partition domainlist**

   e. If the two partitions share a common domain, proceed to cloning.

   f. If the two partitions do not share a common domain, then make room, if necessary, by deleting one domain you can do without.

      lunacm:>partition domaindelete

   g. Add a domain that matches one from the other partition.

      lunacm:> **partition domainadd -domain** <text domain secret> **-domainlabel** <label of the text domain being duplicated>
3. In LunaCM, set the active slot to the source partition and log in as Crypto Officer.

   lunacm:> **slot set -slot** <slotnum>

   lunacm:> **role login -name co**
4. [Optional] View the partition objects and their object handles.

   lunacm:> **partition contents**

5. Clone objects on the partition to the target partition by specifying the target slot. You can choose which objects to clone by specifying a comma-separated list of object handles, or specify **all** to clone all objects on the partition. Present the target partition's Crypto Officer credential when prompted.

   lunacm:> **partition clone -slot** <slotnum> **-objects** <comma-separated_list/**all**>

   The specified objects are cloned to the target partition. Any objects that already exist on the target are not cloned.

6. [OPTIONAL] You can retain an added domain on a partition as long as it remains useful
   - as long as the partition contains objects encrypted under that particular domain, or
   - while you think the current partition might clone (as source or as target) objects with a partition or service using that domain.

   Or you can delete a domain using partition domaindelete if it is no longer needed.

To clone partition objects from on-premises password-authenticated partition to on-premises multifactor quorum-authenticated partition using Luna HSM Firmware 7.8.0 or newer

Requires Luna HSM Client 10.5.0 or newer.

This procedure is for:

> an on-premises password-authenticated Luna Network HSM 7 partition as the source, which could be for:
   - a routine cloning between two HSM partitions that are at Luna HSM Firmware 7.8.0 or newer,
   - *migration cloning of keys and objects* from a legacy HSM partition (firmware 5.x, 6.x), or from firmware older than Luna HSM Firmware 7.8.0.

> an on-premises multifactor quorum-authenticated Luna Network HSM 7 partition as the target (at Luna HSM Firmware 7.8.0 or newer).

1. Ensure that the two partitions can both use a common cloning protocol.
   a. for HSMs (both legacy and 7.x) before Luna HSM Firmware 7.7.0, only protocol CPv1 is available
   b. for Luna HSM Firmware 7.7.1 and newer, if partition policy 42 - Enable CPv1 is ON, then that protocol is chosen and others are disabled

      lunacm:> **slot set -slot** <slotnum>

      lunacm:> **partition showpolicies**

   c. if partition policy 42 - Enable CPv1 is OFF, then negotiation of common cipher suites is attempted between partitions; this is preferred when available.
   d. if CPv1 has not been forced, and all cipher suites for CPv4 have been disabled on one of the participating partitions, then only CPv3 remains and a common CPv4 cipher suite cannot be negotiated.
2. Ensure that the source and target partitions have a cloning domain in common.
   a. In LunaCM, set the active slot to the target multifactor quorum-authenticated partition and log in as Partition SO (po).

      lunacm:> **slot set -slot** <slotnum>

      lunacm:> **role login -name po**

   b. View the partition domains and note their labels.

lunacm:> **partition domainlist**

c. If the two partitions share a common domain, proceed to cloning.

d. If the two partitions do not share a common domain, then make room, if necessary, by deleting one domain you can do without on the target partition.

lunacm:> partition domaindelete

e. Add a domain that matches one from the source partition

lunacm:> **partition domainadd -domain** <text domain secret> **-domainlabel** <label of the text domain being duplicated>

3. In LunaCM, set the active slot to the source partition and log in as Crypto Officer.

lunacm:> **slot set -slot** <slotnum>

lunacm:> **role login -name co**

4. [Optional] View the partition objects and their object handles.

lunacm:> **partition contents**

5. Clone objects on the current partition to the target partition by specifying the target slot. You can choose which objects to clone by specifying a comma-separated list of object handles, or specify **all** to clone all objects on the partition. Present the target partition's Crypto Officer credential when prompted.

lunacm:> **partition clone -slot** <slotnum> **-objects** <comma-separated_list/**all**>

The specified objects are cloned to the target partition. Any objects that already exist on the target are not cloned.

6. [OPTIONAL] You can retain an added domain on a partition as long as it remains useful

- as long as the partition contains objects encrypted under that particular domain, or
- while you think the current partition might clone (as source or as target) objects with a partition or service using that domain.

Or you can delete a domain using partition domaindelete if it is no longer needed.

Moving from Pre-7.7.0 to Firmware 7.7.0 or Newer

Luna HSM Firmware 7.7.0 is a major release for the Luna Network HSM 7 that introduces significant changes to various product components. Some of these changes have been introduced to help customers attain eIDAS compliance, PP 419-221.5 compliance, or conformity with FIPS SP 800-131A (revised). If you are migrating keys from a Luna Network HSM 7 with Luna HSM Firmware 7.4.2 or older to Luna HSM Firmware 7.7.0 or newer, all information that is relevant to the migration process can be found in the following sections:

> Special Considerations for Luna HSM Firmware 7.7.0 and Newer for information you must be aware of and procedures you must complete before and after updating to Luna HSM Firmware 7.7.0 or newer.

> "V0 and V1 Partitions" on page 148 for information about the distinction between V0 and V1 partitions; one of the major changes introduced to Luna HSM partitions with Luna HSM Firmware 7.7.0.

> "Converting Partitions from V0 to V1 or V1 to V0" on page 161 for information about converting partitions from V0 to V1 after updating to Luna HSM Firmware 7.7.0 or newer.

NOTE If you are operating an HA group and would like to migrate the pool of HA member partitions to Luna HSM Firmware 7.7.0 or newer, relevant information is contained or linked to within the topics above.

CHAPTER 13: High-Availability Groups

Luna HSMs can provide scalability and redundancy for cryptographic applications that are critical to your organization. For applications that require continuous, uninterruptible uptime, the Luna HSM Client allows you to combine application partitions on multiple HSMs into a single logical group, known as a High-Availability (HA) group.

High Availability Options for Luna HSMs

Luna HSMs support three, non-overlapping approaches to High Availability:

- client-mediated HA creation and management of HA groups, by means of the LunaCM hagroup commands, with preparation, management, and usage described at "High-Availability Groups" above ; the "participating" member HSMs are unaware that they are part of a group
- extensions to the PKCS#11 API to program your own complete HA environment, in full, or to tie into (integrate with) suitable COTS High Availability solutions that provide a suitable Application Interface, using HA Indirect Login calls to handle common authentication among HSM partitions in groups; refer to High Availability Indirect Login..
- peer clustering, where the member HSM appliances coordinate among themselves - makes use of HSM-appliance clusters and keyrings rather than classic "virtual-HSM" partitions

Clustering

Clustering is a peer-mediated load-sharing and redundancy approach among Luna Network HSM 7 appliances. See Clusters for more information.

Network Latency and Luna Cloud HSM as Active HA Member

Requests performed by cloud services like Luna Cloud HSM may experience greater network latency than those sent to on-premise HSMs. Thales recommends using a Luna Cloud HSM service as a standby HA member to achieve the best performance. By default, you can add a Luna Cloud HSM service as a standby HA member only. If all other HA members fail and the Luna Cloud HSM service becomes active, it will revert to standby when another member recovers.

If you prefer to use the Luna Cloud HSM service as an active HA member, you must first edit the following toggle in the **Chrystoki.conf/crystoki.ini** configuration file (see "Configuration File Summary" on page 76):

```
[Toggles]
lunacm_cv_ha_ui = 0
```

> **CAUTION!** HA failover from multifactor quorum-authenticated Luna partitions to Luna Cloud HSM requires minimum Luna HSM Client 10.5.0. Refer to known issue LUNA-23945.

Key/object replication options for HA and other uses

HA ensures that your applications can continue to access a functioning partition as long as at least one partition is running, and reachable, in the HA group. Proper HA operation requires that all member partitions be populated with the same keys and objects, relying on replication and synchronization of keys among group-member partitions.

Luna HSMs have three ways to replicate partition objects (keys) among partitions, depending on partition type and settings. Each method can have limitations related to key type.

The replication of *symmetric* keys is enabled by default for each partition that belongs to the same HA group.

Different private-key replication options are offered for *asymmetric* keys, to adapt the level of security needed for customer use cases.

Each partition of an HSM can be configured in any of the following modes:

> Key Export

> Cloning

> the newer Scalable Key Storage (SKS)

Key Export and Cloning are set by partition policies, while availability of SKS is determined by selecting V1 as the partition type at partition creation time (requires Luna HSM Firmware 7.7.0 or newer). Cloning is the default mode, so Key Export needs to be explicitly selected if desired.

Cloning

To maintain the highest security level, enable the cloning mode (and thus turn off the export mode).

In cloning mode, the asymmetric private keys are cloned automatically between all the partitions of the HA group. In that mode, neither private keys nor secret keys are ever allowed to exist outside of a trusted Luna HSM that is a member of the designated cloning domain (sometimes also called "security domain"). As indicated above, symmetric keys are replicated. Cloning mode is the default setting for new partitions.

PQC

> **NOTE** By design (approved by NIST) HSS keys cannot be copied/cloned and therefore are not for use in an HA group, and cannot be backed-up or restored.
>
> - Do not generate an HSS key pair on an HA virtual slot.
> - Do not add a partition to an HA group if the partition has an HSS private key on it.
>
> LMS-HSS key creation and use is supported only in partition mode on the Luna HSM, is not supported in key rings, and does not support PKA (per-key authentication).

Key Export

For use cases that require asymmetric private keys, generated on the HSM, to be embedded or used on a device (for example identity issuance), cloning mode must be turned off and key export mode enabled. In that mode, your application is responsible:

> to ensure the availability of the generated asymmetric private keys (by storing the keys in a file or database), and

> to ensure the availability and security of the storage solution.

Cloning and Key Export are mutually exclusive
For a use case where applications require both modes, then it is possible to create separate partitions on an HSM, configured for cloning mode or export mode - an individual partition is set to be one or the other, but cannot be both at the same time. Changing the replication mode of a partition requires recreating the partition, which loses all content (and removes it from any HA group). Backup any important keys and objects before taking such action.

Where a partition belongs to an HA group, all member partitions must be configured for the same replication mode. Thus, if your use case needed to employ some partitions in cloning mode and some partitions in key export mode, you would need to have separate HA groups to accommodate the separate sets of keys.

Keys cannot pass between replication modes.

An HA group in Key Export mode replicates only the symmetric keys, and not the asymmetric keys.

SKS
For a use case that generates greater numbers of keys than can be stored within a cryptographic module (HSM), but also demands greater security of externally stored keys than is provided by Key Export mode, the Scalable Key Storage (SKS) mode is provided in Luna HSM Firmware 7.7.0 and newer.

- SKS ensures that the HSM remains the assurance boundary for your asymmetric private keys.
- In that mode SKS blobs, containing the key material and the associated objects, are encrypted with a symmetric key (SMK) that is cloned when managed within an HA group. Your application is responsible to ensure the availability of the SKS blobs (by storing the blobs into a file, or database and to ensure the availability of that storage solution). Because replication of the SMK is done by cloning, the SMK never exists outside a Luna HSM, thus the security of all keys and objects encrypted by that SMK is as high as security for the cloned key option, but without the storage limitation of the HSM.
- SKS requires that your partition be created as type V1. See "Scalable Key Storage" on page 165 and "V0 and V1 Partitions" on page 148.

Client-driven High Availability

(To develop your own HA solution or integrate with a commercial one, see High Availability Indirect Login For HSM Firmware 7.7.0 and Newer instead.)

This feature is best suited to provide redundancy to the Luna Network HSM 7 and Luna PCIe HSM 7 products. Luna Network HSM 7 partitions can be grouped with other Luna Network HSM 7 partitions or with a Luna Cloud HSM Service. Luna PCIe HSM 7 partitions can be grouped with other Luna PCIe HSM 7 partitions or with a Luna Network HSM 7 Service.

An HA group allows your client application to access cryptographic services as long as one member HSM is functional and network-connected. This allows you to perform maintenance on any individual member without ever pausing your application, and provides redundancy in the case of individual failures. Cryptographic requests are distributed across all active group members, enabling a performance gain for each member added. Cryptographic objects are replicated across the entire group, so HA can also be used to keep a current, automatic, remote backup of the group contents.

HA functionality is handled by the Luna HSM Client software. The individual partitions have no way to know they are configured in an HA group, so you can configure HA on a per-application basis. The way you group your HSMs depends on your circumstances and desired performance.

This chapter contains the following sections:

If you plan to create an HA group consisting of different kinds of Luna HSMs, refer also to:

Performance

For repetitive operations (for example, many signings using the same key), an HA group provides linear performance gains as group members are added. The best approach is to maintain an HA group at a size that best balances application server capability and the expected loads, with an additional unit providing capacity for bursts of traffic.

For best overall performance, keep all group members running near their individual performance ideal, about 30 simultaneous threads per HSM. If you assemble an HA group that is significantly larger than your server(s) can manage, you might not achieve full performance from all members. Gigabit Ethernet connections are recommended to maximize performance.

Performance is also affected by the kind of cryptographic operations being requested. For some operations, an HA group can actually hinder performance by requiring extra operations to replicate new key objects. For example, if the operation involves importing and unwrapping keys:

Using an HA group	Using an individual partition

1. Encryption (to wrap the key)
2. Decryption on the primary member partition (to unwrap the key)
3. Object creation on the primary member partition (the unwrapped key is created and stored as a key object)
4. Key replication across the HA group:
  a. RSA 4096-bit operation is used to derive a shared secret between HSMs
  b. Encryption of the key on the primary HA member using the shared secret
  c. Decryption of the key on each HA member using the shared secret
  d. Object creation on each HA member
5. Encryption (using the unwrapped key object to encrypt the data)
	1. Encryption (to wrap the key)
	2. Decryption (to unwrap the key)
	3. Object creation (the unwrapped key is created and stored as a key object)
	4. Encryption (using the unwrapped key object to encrypt the data)

In this case, the HA group must perform many more operations than an individual partition, most significantly the RSA-4096-bit operation and creating the additional objects. Those two operations are by far the most time-consuming on the list, and so this task would have much better performance on an individual partition.

The crucial HA performance consideration is whether the objects on the partitions are constant, or always being created and replaced. If tasks make use of already-existing objects, those objects exist on all HA group members; operations can be performed by different group members, boosting performance. If new objects are created, they must be replicated across the entire group, causing a performance loss while the replication occurs.

> **NOTE** The way your application uses the **C_FindObjects** function to search for objects in a virtual HA slot can have a significant impact on your application performance (see "Application Object Handles" on page 423).

Load Balancing

Cryptographic requests sent to the HA group's virtual slot are load-balanced across all active members of the HA group. The load-balancing algorithm sends requests for cryptographic operations to the least busy partition in the HA group. This scheme accounts for operations of variable length, ensuring that queues are balanced even when some partitions are assigned very long operations. When an application requests a repeated set of operations, this method works. When the pattern is interrupted, however, the request type becomes relevant, as follows:

- Single-part (stateless) cryptographic operations are load-balanced.
- Multi-part (stateful) cryptographic operations are load-balanced.

- Multi-part (stateful) information retrieval requests are not load-balanced. In this case, the cost of distributing the requests to different HA group members is generally greater than the benefit. For this reason, multi-part information retrieval requests are all targeted at one member.
- Key management requests are not load-balanced. Operations affecting the state of stored keys (creation, deletion) are performed on a single HA member, and the result is then replicated to the rest of the HA group.

For example, when a member partition is signing and an asymmetric key generation request is issued, additional operations on that member are queued while the partition generates the key. In this case, the algorithm schedules more operations on other partitions in the HA group.

The load-balancing algorithm operates independently in each application process. Multiple processes on the same client or on different clients do not share information when scheduling operations. Some mixed-use cases might cause applications to use some partitions more than others. If you increase key sizes, interleave other cryptographic operations, or if network latency increases, performance may drop for individual active members as they become busier.

> **NOTE** Partitions designated as standby members are not used to perform cryptographic operations, and are therefore not part of the load-balancing scheme (see "Standby Members" on page 422).

The Primary Partition

The primary partition is the first partition you specify as a member of the HA group. While cryptographic operations are load-balanced across all the partitions in the group, new keys are always created on the primary partition, and then replicated on the other partitions (see "Key Replication" on the next page). Depending on how many new keys you are creating on your HA group, this can mean that the primary partition has a heavier workload than the other partitions in the group. If your HSMs are in different remote locations, you could select one with the least latency as the primary partition.

Despite its name, the primary partition is not more critical than any other partition in the HA group. If the primary partition fails, its operations fail over to other partitions in the group, and the next member added to the group becomes the new primary partition.

> **NOTE** The mechanism list available to the current primary partition is the only mechanism list that is available to HA virtual slot.

Network Topography

The network topography of the HA group is generally not important to the functioning of the group. As long as the client has a network path to each member, the HA logic will function. Different latencies between the client and each HA member cause a command scheduling bias towards the low-latency members. Commands scheduled on the long-latency devices have a longer overall latency associated with each command.

In this case, the command latency is a characteristic of the network. To achieve uniform load distribution, ensure that partitions in the group have similar network latency.

Key Replication

Objects (session or token) are replicated immediately to all members in an HA group when they are generated in the virtual HA slot. Similarly, deletion of objects (session or token) from the virtual HA slot is immediately replicated across all group members. Therefore, when an application creates a key on the virtual HA slot, the HA library automatically replicates the key across all group members before reporting back to the application. Keys are created on one member partition and replicated to the other members. If a member fails during this process, the HA group reattempts key replication to that member until it recovers, or failover attempts time out. Once the key exists on all active members of the HA group, a success code is returned to the application.

> **NOTE** If you are using Luna HSM Client 10.4.0 or newer and are setting up an HA group with a mix of FIPS and non-FIPS partitions as members, objects will not replicate across all HSMs in the group in the following cases:
>
> > If you have set a non-FIPS primary, a FIPS secondary, and created a non-FIPS approved key on the group, the key will not replicate to the FIPS secondary. No error is returned when this occurs.
> >
> > If you synchronize group members with the **hagroup synchronize** LunaCM command, any non-FIPS keys will fail to replicate to the FIPS member(s). An error is returned when this occurs, but lunaCM synchronizes everything else.

> **NOTE** If your application bypasses the virtual slot and creates or deletes directly in a physical member slot, the action occurs only in that single physical slot, and can be overturned by the next synchronization operation. For this reason we generally advise to enable HA-only, unless you have specific reason to access individual physical slots, and are prepared (in your application) to perform the necessary housekeeping.

Key replication, for pre-firmware-7.7.0 HSM partitions and for V0 partitions, uses the Luna cloning protocol, which provides mutual authentication, confidentiality, and integrity for each object that is copied from one partition to another. Therefore, *prior to Luna HSM Firmware 7.8.0*, all HA group member partitions must be initialized with the same cloning domain.

Key replication, for Luna HSM Firmware 7.8.0 (and newer) HSM partitions and for V0 partitions, and Luna HSM Client 10.5.0 (and newer), becomes more versatile with Extended Domain Management, as each member partition can have as many as three cloning/security domains. It becomes possible to easily mix password-authenticated and multi-factor (PED) authenticated partitions in HA groups. Any member must have at least one of its domains in common with the current primary member. [For reasons of redundancy and overlap, we recommend that you *not* create (say) a 4-member group where the primary has domains A, B, C, and the three secondary members include one member with domain A, one member with domain B, and one member with domain C, where no other domains belong to the group -- such a group could function only until the primary failed/went-offline, at which point the next primary would have no domain peers with which to synchronize. Therefore, consider redundancy overlap when using Extended Domain Management with HA group members.

Key replication for V1 partitions uses the Luna cloning protocol to ensure that all HA group members have the same SMK, and uses SKS to export a key originating at one member and to import and decrypt that key (using the common SMK) on each other member in the group. Again, all HA group member partitions must be initialized with the same cloning domain in order that the common SMK can be available on every member.

The cloning protocol (for pre-firmware-7.7.0 or V0 or mixed-version HA), or the SKS protocol (for blob transfers in a V1 partition HA) is invoked separately for each object to be replicated and the sequence of required calls must be issued by an authorized client library residing on a client platform that has been authenticated to each of the partitions in the HA group). That is, the client must have an authenticated NTLS or STC channel to the HSM appliance; only an authorized client can perform object synchronization across all HA members.

Failover

When any active HA group member fails, a failover event occurs – the affected partition is dropped from the list of available HA group members, and all operations that were pending on the failed partition are transparently rescheduled on the remaining member partitions. The Luna HSM Client continuously monitors the health of member partitions at two levels:

- network connectivity – disruption of the network connection causes a failover event after a 20-second timeout.
- command completion – any command that is not executed within 20 seconds causes a failover event.

> **NOTE** Most commands are completed within milliseconds. Some can take longer, either because the command itself is time-consuming (for example, key generation), or because the HSM is under extreme load. The HSM automatically sends a "heartbeat" signal every two seconds for commands that are pending or in progress. The client extends the 20-second timeout whenever it receives a heartbeat, preventing false failover events. When an HA group is active, the client sends another heartbeat signal every 10 seconds to check the status of HA member partitions.

When an HA group member fails, the HA group status (see hagroup listgroups) reports a device error for the failed member. The client tries to reconnect the failed member at a minimum retry rate of once every 60 seconds, for the specified number of times (see "Recovery" on the next page).

When a failover occurs, the application experiences a latency stall on the commands in process on the failing unit, but otherwise there is no impact on the transaction flow. The scheduling algorithm described in "Load Balancing" on page 417 automatically minimizes the number of commands that stall on a failing unit during the 20-second timeout.

As long as one HA group member remains functional, cryptographic service is maintained no matter how many other group members fail. As described in "Recovery" on the next page, members can be returned to service without restarting the application.

Mid-operation failures

Any operation that fails mid-point needs to be re-sent from the calling application. The entire operation returns a failure (CKR_DEVICE_ERROR). This is more likely to happen in a multi-part operation, but a failure could conceivably happen during a single atomic operation as well.

For example, multi-part operations could be block encryption/decryption or any other command where the previous state of the HSM is critical to the processing of the next command. These operations must be re-sent, since the HA group does not synchronize partitions' internal memory state, only the stored key material.

> **NOTE** You must ensure that your applications can deal with the rare possibility of a mid-operation failure, by re-issuing the affected commands.

Possible Causes of Failure

In most cases, a failure is a brief service interruption, like a system reboot. These temporary interruptions are easily dealt with by the failover and auto-recovery functions. In some cases, additional actions may be required before auto-recovery can take place. For example, if a partition becomes deactivated, it must be reactivated by the Crypto Officer (see "Activation on Multifactor Quorum-Authenticated Partitions" on page 373). Some permanent failures may require manual recovery (see "Recovery" below). Possible failure events include:

> **HSM-side failures**
  - HSM card failure
  - HSM re-initialization
  - HSM reboot
  - HSM power failure
  - Deactivated partition
  - NTLS service failure
  - STC service failure

> **Client-side failures**
  - Client workstation power failure
  - Client workstation reboot
  - Network keepalive failure

> **Network failures**
  - Network failure near the HSM (one member partition disappears from client's view)
  - Network failure near the client (client loses contact with all member partitions)

Recovery

Recovery of a failed HA group member is designed to be automatic in as many cases as possible. You can configure your auto-recovery settings to require as much manual intervention as is convenient for you and your organization. In either an automated or manual recovery process, there is no need to restart your application. As part of the recovery process:

> Any cryptographic objects created while the member was offline are automatically replicated to the recovered partition.

> The recovered partition becomes available for its share of load-balanced cryptographic operations.

Auto-recovery

When auto-recovery is enabled, Luna HSM Client performs periodic recovery attempts when it detects a member failure. You can adjust the frequency (maximum once per minute) and the total number of retries (no limit). If the failed partition is not recovered within the scheduled number of retries, it remains a member of the HA group, but the client will no longer attempt to recover it. You must then address whatever equipment or network issue caused the failure, and execute a manual recovery of the member partition.

With each recovery attempt, a single application thread experiences a slight latency delay of a few hundred milliseconds while the client uses the thread to recover the failed member partition.

There are two HA auto-recovery modes:

- **activeBasic** – uses a separate, non-session-based Active Recovery Thread to perform background checks of HA member availability, recover failed members, and synchronize the contents of recovered members with the rest of the group. It does not restore existing sessions if all members fail simultaneously and are recovered.
- **activeEnhanced** – works the same as activeBasic, but restores existing sessions and login states if all members fail and are recovered.

HA auto-recovery is disabled by default. It is automatically enabled when you set the recovery retry count (see "Configuring HA Auto-Recovery" on page 439). Thales recommends enabling auto-recovery in all configurations.

> **NOTE** If a member partition loses Activation when it fails (it remains offline for more than two hours) you must present the black Crypto Officer PED key to re-cache the authentication secret before the member can be recovered.

Manual Recovery

When auto-recovery is disabled, or fails to recover the partition within the scheduled number of retries, you must execute a manual recovery in LunaCM. Even if you use manual recovery, you do not need to restart your application. When you execute the recovery command, the client makes a recovery attempt the next time the application uses the group member (see "Manually Recovering a Failed HA Group Member" on page 452).

Even with auto-recovery enabled and configured for a large number of retries, there are some rare occasions where a manual recovery may be necessary (for example, when a member partition and the client application fail at the same time).

> **CAUTION!** Never attempt a manual recovery while the application is running and auto-recovery is enabled. This can cause multiple concurrent recovery processes, resulting in errors and possible key corruption.

Failure of All Group Members

If all members of an HA group fail (and no standby members are configured), all logged-in sessions are lost, and operations that were active when the last member failed are terminated. If you have set the HA auto-recovery mode to activeEnhanced, all sessions will be restarted when one or more members are recovered, and normal operations will resume. Otherwise, you must restart the client application once the group members have been recovered.

Permanent Failures

Sometimes an HSM failure is permanent (from the perspective of the HA group). For example, if the HSM is re-initialized, the member partition is erased and must be recreated. In this case, you can decide to recreate the original member or deploy a new member to the group. The client automatically replicates cryptographic objects to the new member and begins assigning operations to it (see "Replacing an HA Group Member" on page 453).

Standby Members

After you add member partitions to an HA group, you can designate some as standby members. Cryptographic objects are replicated on all members of the HA group, including standby members, but standby members do not perform any cryptographic operations unless all the active members go offline. In this event, all standby

members are immediately promoted to active service, and operations are load-balanced across them. This provides an extra layer of assurance against a service blackout for your application.

Since standby members replicate keys but do not perform operations, they can also serve as an automatic backup partition for the cryptographic objects on the HA group. The contents of standby partitions are always kept up-to-date, so it is not possible to keep multiple backups (different generations of preserved material) using an HA group (see "Planning Your HA Group Deployment" on page 426). You can consider HA standby members to be your backup only in the case where the most recent sync always replicates all objects you are interested in preserving and recovering.

If you have audit-compliance rules or other mandate to preserve earlier partition contents (keys and objects), then you should perform intentional backups with dedicated backup devices (see "Partition Backup and Restore" on page 467).

Mixed-Version HA Groups

Generally, Thales recommends using HSMs with the same software/firmware in HA groups; different versions have different capabilities, and a mixed HA group is limited to those functions that are common to the versions involved. A mixed-version HA group may have access to fewer cryptographic mechanisms, or have different restrictions in FIPS mode. However, HA groups containing both Luna 6 and 7 partitions and Luna Cloud HSM services are supported. This mixed-version configuration is useful for migrating keys to a new Luna 7 HSM or the cloud, or to gradually upgrade your production environment from Luna 6 to Luna 7.

Process Interaction

At the lowest communication level, the transport protocol (TCP) maintains communication between the client and the appliance (whether HA is involved or not). For HA groups involving member partitions on Luna Network HSM 7, the protocol timeout is 10 seconds. This means:

- In a period of no activity by client or appliance, the appliance's TCP sends a packet after 10 seconds of silence.
- If that packet is acknowledged, the 10-second TCP timer restarts, and the cycle repeats indefinitely.
- If the packet is not acknowledged, TCP sends another every 10 seconds. If there is no response after 2 minutes, the connection is considered dead, and higher levels are alerted to perform their cleanup.

Above that level, the NTLS/STC layer provides the connection security and some other services. Any time a client sends a request for a cryptographic operation, the HSM on the appliance begins working on that operation.

While the HSM processes the request, appliance-side NTLS/STC sends a "keep-alive" ping every 2 seconds, until the HSM completes the request. NTLS/STC does not perform any interpretation of the ping, but simply keeps the TCP layer active. If your client application requests a lengthy operation (for example, an 8192-bit keygen), the random-number-generation portion of that operation could take multiple minutes, during which the HSM would legitimately be sending nothing back to the client. The NTLS ping ensures that the connection remains alive during long pauses.

Application Object Handles

Application developers should be aware that the PKCS #11 object handle model is fully virtualized when using an HA slot. The application must not assume fixed handle numbers across instances of an application. A handle’s value remains consistent for the life of a process; but it might be a different value the next time the application is executed.

When you use an HA slot with your applications, the client behaves as follows when interacting with the application:

1. Intercept the call from the application.
2. Translate virtual object handles to physical object handles using the mappings specified by the virtual object table. The virtual object table is created and updated for the current session only, and only contains of list of the objects accessed in the current session.
3. Launch any required actions on the appropriate HSM or partition.
4. Receive the result from the HSM or partition and forward the result to your application,
5. Propagate any changes in objects on the physical HSM that performed the action to all of the other members of the HA group.

Virtual slots and virtual objects

When an application uses a non-HA physical slot, it addresses all objects in the slot by their physical object handles. When an application uses an HA slot, however, a virtual layer of abstraction overlays the underlying physical slots that make up the HA group, and the HA group is presented to the application as a virtual slot. This virtual slot contains virtual objects that have virtual object handles. The object handles in an HA slot are virtualized since the object handles on each of the underlying physical slots might be different from slot to slot. Furthermore, the physical object handles could change if a member of the HA group drops out (fails or loses communication) and is replaced.

The virtual object table

HA slots use a virtual object table to map the virtual objects in the virtual HA slot to the real objects in the physical slots that make up the HA group. The HA client builds a virtual object table for each application that loads the library. The table is ephemeral, and only exists for the current session. It is created and updated, if necessary, each time an application makes a request to access an object. To maximize performance and efficiency, the table only contains a list of the objects accessed in the current session. For example, the first time an application accesses an object after application start up, the table is created, a look up is performed to map the virtual object to its underlying physical objects, and an entry for the object is added to the table. For each subsequent request for that object, the data in the table is used and no look up is required. If the application then accesses a different object that is not listed in the table, a new look up is performed and the table is updated to add an entry for the new object.

C_FindObjects behavior and application performance

Since the client must perform a lookup to create the virtual object table, the way you use the C_FindObjects function can have a significant impact on the performance of your applications. For example, if you use the C_FindObjects function to ask for specific attributes, the client only needs to update the table to include the requested objects. If, however, you use the C_FindObjects function to find all objects, the client queries each HSM/partition in the group, for each object, to create the table. This can take a significant amount of time if the slot contains a large number of objects, or if the HA group includes many members.

To mitigate performance degradation when using the C_FindObjects function to list the objects on an HA slot, we recommend that you structure your applications to search by description or other attributes, rather than searching for all objects. Doing so minimizes the number of objects returned and the time required to create or update the table. If your application must find all objects, we recommend that you add the C_FindObjects all function call to the beginning of your application so that the table is built on application start up, so that the table is available to the application for all subsequent C_FindObjects function calls.

Example: Database Encryption

This section walks through a sample use case of some of the HA logic with a specific application – a transparent database encryption.

Typical Database Encryption Key Architecture

Database engines typically use a two-layered key architecture. At the top layer is a master encryption key that is the root of data protection. Losing this key is equivalent to losing the database, so it obviously needs to be highly durable. At the second layer are table keys used to protect table-spaces and/or columns. These table keys are stored with the database as blobs encrypted by the master encryption key (MEK). This architecture maps to the following operations on the HSM:

1. Initial generation of master key for each database.
2. Generation and encryption of table keys with the master key.
3. Decryption of table keys when the database needs to access encrypted elements.
4. Generation of new master keys during a re-key and then re-encrypting all table keys with it.
5. Generation and encryption of new table keys for storage in the database (often done in a software module).

The HSM is not involved in the use of table keys. Instead it provides the strong protection of the MEK which is used to protect the table keys. Users must follow backup procedures to ensure their MEK is as durable as the database itself ("Partition Backup and Restore" on page 467).

HSM High Availability with Database Encryption

When the HSMs are configured as an HA group, the database's master key is automatically and transparently replicated to all the members when the key is created or re-keyed. If an HSM group member was offline or fails during the replication, it does not immediately receive a copy of the key. Instead the HA group proceeds after replicating to all of the active members. Once a member is re-joined to the group the HSM client automatically replicates the new master keys to the recovered member.

Before every re-key event, the user must ensure the HA group has sufficient redundancy. A re-key will succeed as long as one HA group member exists, but proceeding with too few HSMs will result in an availability risk. For example, proceeding with only one HSM means the new master key will be at risk since it exists only on a single HSM. Even with sufficient redundancy, Thales Group recommends maintaining an offline backup of a database's master key.

HSM Load Balancing with Database Encryption

While a database is up and running, the master key exists on all members in the HA group. Requests to encrypt or decrypt table keys are distributed across the entire group. The load-balancing feature is able to deliver improved performance and scalability when the database requires a large number of accesses to the table keys. Most deployments will not need much load balancing as the typical database deployment results in a small number of table keys.

While the table keys are re-keyed, new keys are generated in the HSM and encrypted for storage in the database. Within an HA group, these keys are generated on the primary member and then replicated to the entire HA group, even though they exist on the HSM for only a moment. These events are infrequent enough that this extra replication has minimal impact.

Planning Your HA Group Deployment

This section describes important considerations and constraints to keep in mind as you plan your Client-mediated High-Availability (HA) group deployment (**NOTE**: Not to be confused with Clusters and Keyrings). The benefits of HA are described in detail in "High-Availability Groups" on page 413. There are several sample configurations described in this section that take advantage of different HA features. Depending on your organization's security needs, you might choose one of these configurations, or your own variation.

> "HSM and Partition Prerequisites" below

> "Sample Configurations" on the next page

- "Performance and Load Balancing" on the next page
- "Redundancy and Failover" on page 429
- "Automatic Remote Backup" on page 431
- "HA Group Sharing" on page 431

If you plan to create an HA group consisting of different kinds of Luna HSMs, refer also to:

> "Cloning Keys Between Luna 6, Luna 7, and Luna Cloud HSM, Password or Multifactor Quorum" on page 210

HSM and Partition Prerequisites

The HSM partitions you plan to use in an HA group must meet the following prerequisites before you can use them in an HA group.

Compatible HSM Software/Firmware Versions

Generally, Thales recommends using HSMs with the same software/firmware in HA groups; different versions have different capabilities, and a mixed-version HA group is limited to those functions that are common to the versions involved. This means they have access to fewer cryptographic mechanisms, or have different restrictions in FIPS 140 approved configuration (formerly FIPS mode). However, mixed-version HA groups containing Luna 6 and 7 member partitions and Luna Cloud HSM services are supported. See "Cloning Keys Between Luna 6, Luna 7, and Luna Cloud HSM, Password or Multifactor Quorum" on page 210 for more information.

Common Cloning Domain

All key replication in an HA group uses the Luna cloning protocol, which provides mutual authentication, confidentiality, and integrity for each object that is copied from one partition to another. Therefore, all HA group member partitions must be initialized with the same cloning domain. If you are planning to combine already-existing partitions into an HA group, you must first re-initialize them using the same domain string or red PED key.

Common Crypto Officer Credentials

An HA group essentially allows you to log in to all its member partitions simultaneously, using a single credential. Password-authenticated partitions must all be initialized with the same Crypto Officer password. Multifactor Quorum-authenticated partitions must all be initialized with the same black Crypto Officer PED key and activated with the same CO challenge password.

Formerly, it was not possible to create an HA group made up of both password- and multifactor quorum-authenticated partitions. From the advent of Extended Domain Management (see "Allow Extended Domain Management" on page 351) it is possible to mix authentication types as long as:

- the affected partitions share the same cloning domain
- the multifactor quorum partition is activated (and preferably also auto-activated)
- the challenge secret for the multifactor quorum partition is the same as the password for the password authenticated partition.

Common HSM/Partition Policies (FIPS 140 approved configuration [formerly FIPS mode])
Generally, all HSMs/partitions used in an HA group must have the same policy configuration.

If you are planning to set up an HA group with a mix of FIPS and non-FIPS partitions as members, note the following:

- If you are using Luna HSM Client 10.4.0 or newer, you *can* set up an HA group with a mix of FIPS and non-FIPS partitions as members. However, some limitations must be considered. For more information, refer to "Key Replication" on page 419.
- If you are using Luna HSM Client 10.3.0 or older, you *cannot* set up an HA group with a mix of FIPS and non-FIPS partitions as members.

Functionality Modules
If you intend to use Functionality Modules (FMs) with your HA group, all HSMs containing HA group members must have FMs enabled and they must all have the same FM(s) loaded. See FM Deployment Constraints for details. FMs are not supported for Luna Cloud HSM services.

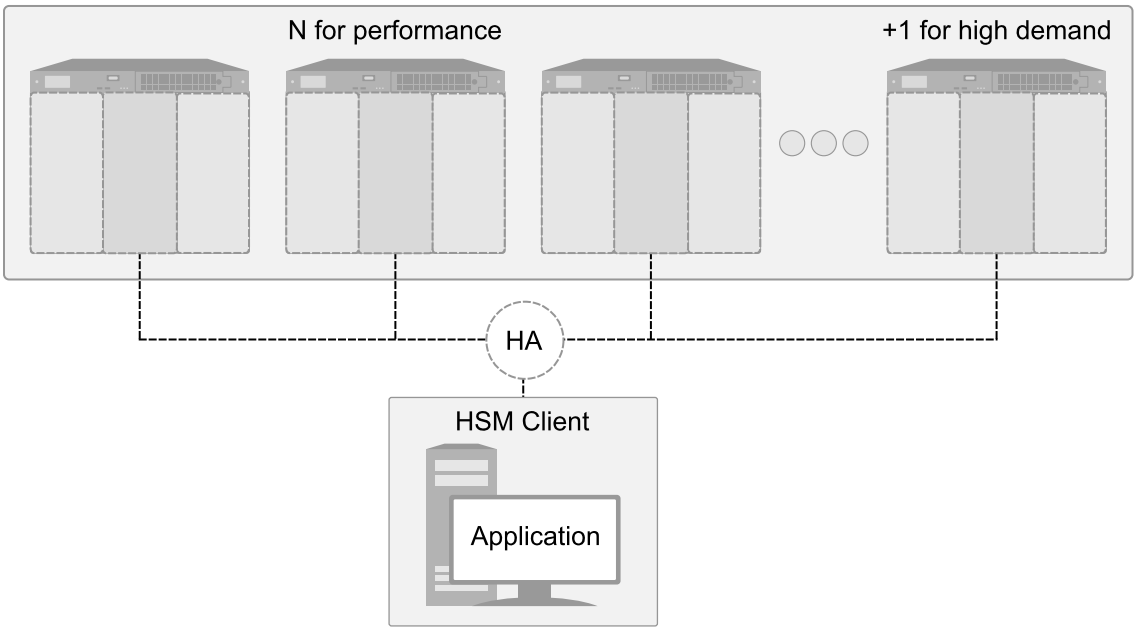
Sample Configurations

Your ideal HA group configuration depends on the number of HSMs you have available and the purpose of your application(s). Sample configurations for different types of deployment are described below.

Performance and Load Balancing
If your application is designed to perform many cryptographic operations as quickly as possible, using keys or other objects that do not change often, you can create a large HA group using partitions on many HSMs. This deployment uses load balancing to provide linear performance gains for each HSM added to the group.

For example: your application uses keys stored on the HSM to perform many encrypt/decrypt or sign/verify operations. You want to minimize transaction latency by providing enough HSMs to handle capacity.

The Luna HSM Client allows HA groups with up to 32 member partitions. The best approach in this example is to add enough group members to handle the usual number of operations, plus enough extra members to handle periods of high demand.


N for performance
+1 for high demand
HA
HSM Client
Application

TIP HA Benefits

While client-mediated HA can have benefits with respect to

> up time
> redundancy and failover
> performance increases

...be aware that HA also comes with costs, and that there are exceptions.

Replication of objects across the group takes time; this happens when objects are created or updated via the HA virtual slot.

Finding objects takes longer when multiple partitions are being searched.

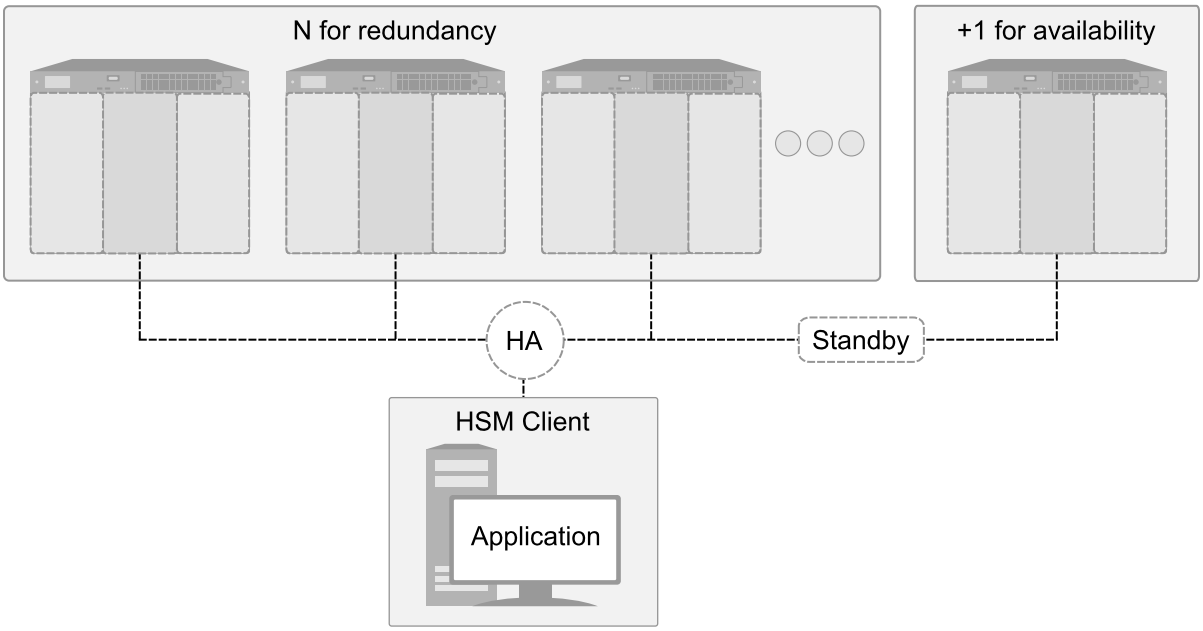
Maximum benefit from HA is achieved when objects are created once and used many times, where

> all operations by your application should be directed toward the virtual slot, which takes care of replicating objects, and updates that are performed on objects, to all member partitions/slots and,
> all operations by your application should generally not be directed toward individual slots (in fact, the 'physical' slots that make up an HA group should be hidden by the HAOnly configuration option and not addressed directly) or you risk
  - breaking synchronization among members,
  - skew of contents with differing versions of objects or keys in different member slots
  - error conditions if a call is made against the virtual HA slot and all members do not have the key or object that is expected to undergo parallel operations in the member partitions/slots

If your application creates objects frequently, against the HA virtual slot, the automatic replication across all slots will slow the overall performance. In the case where objects or keys are created to be used once it can be useful to address individual 'physical' slots independently. In such case, you retain standby/rollover advantage of HA, just not a performance enhancement. As soon as keys and objects are no longer useful, they must be deleted, or HA performance is affected by the lookup burden.

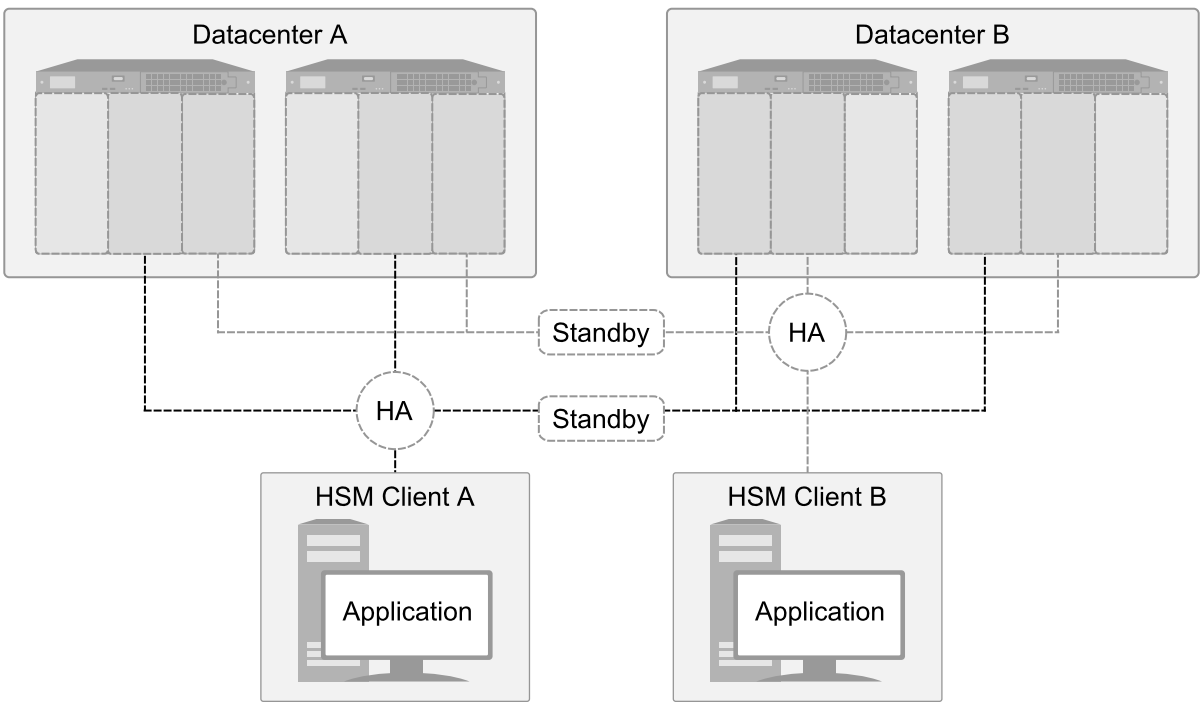
Redundancy and Failover

If your application requires continuous, uninterruptible uptime, operations assigned to an HA group are reassigned to other group members in the event of a member failure (see "Failover" on page 420 for details). Additional group members can be added and set to standby mode for an extra layer of redundancy (see "Standby Members" on page 422 for details).



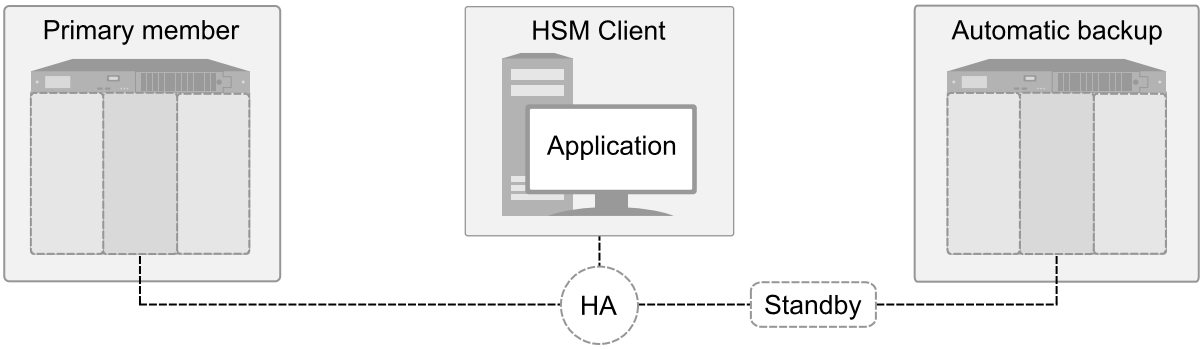
To maximize the use of your HSMs, plan which member partitions you will set to standby mode. Although the configuration above is a straightforward example of an HA group with a single standby member, it is not an ideal production configuration, because the standby member is idle unless all the other members fail. The configuration below is a more useful implementation of two HA groups, each with standby members on the other's HSMs.

As depicted below, applications can be deployed in geographically dispersed locations. In this scenario, Luna's standby capability allows you to use the HSMs in Datacenter B to cost-effectively improve availability for the local HA group at Datacenter A, and vice-versa. This approach allows the HA groups to avoid using remote HSMs with high latency, unless they are urgently required. If all local members fail, the standby partitions are automatically promoted to active status.



Automatic Remote Backup

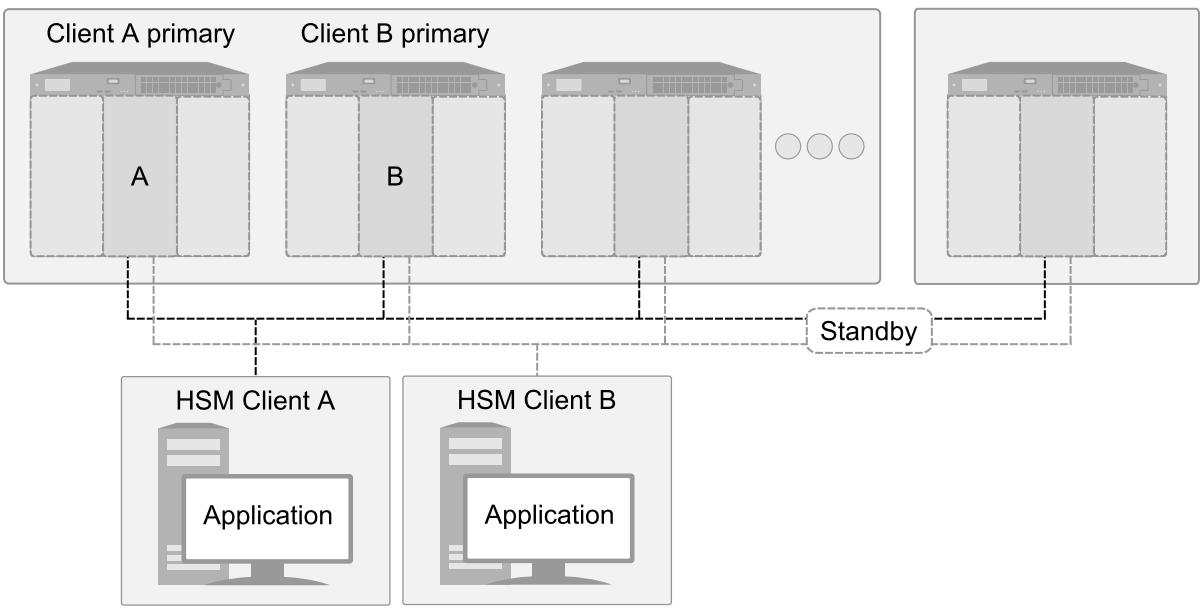
Since the contents of member partitions are always kept up-to-date, you can use an HA group to keep an automatic backup of your cryptographic objects. Set the backup member to standby mode so that it does not perform operations. If the regular member(s) fail, the standby member takes over operations.



HA Group Sharing

Generally, an HA group is defined on a single client, which runs an application against the virtual HA group. You can share the HA group across multiple clients by assigning all member partitions to both clients and creating the HA group independently on each one.

> **TIP** When an HA group is shared across multiple clients, the group can be defined with a different primary member (the first partition assigned to the group) on each client. This approach optimizes an HA group to distribute the key management and/or multi-part cryptographic operation load more equally.



Setting Up an HA Group

Use LunaCM to create an HA group from partitions assigned to your client. This procedure is completed by the Crypto Officer. Ensure that you have met all necessary prerequisites before proceeding with group creation. For a detailed description of HA functionality, see "High-Availability Groups" on page 413.

> **NOTE** Your LunaCM instance needs to update the **Chrystoki.conf** (Linux/UNIX) or **crystoki.ini** file (Windows) when setting up or reconfiguring HA. Ensure that you have Administrator privileges on the client workstation.
>
> **V1 partitions:** If you add an application partition with an existing SMK to an HA group, the primary member's SMK overwrites the existing SMK of the joining partition.If a partition's SMK has ever been used to encrypt important SKS objects, save a backup of the SMK before adding that partition to any HA group.

Prerequisites

HA groups are set up in LunaCM by the Crypto Officer. Before the CO can perform this setup, however, all HSMs and member partitions must meet the following prerequisites, completed by the HSM and Partition Security Officers.

HSMs

The HSM SO must ensure that all HSMs containing HA group member partitions meet the following prerequisites:

- All HSMs must use the same authentication method (password/multifactor quorum). Luna Cloud HSM Services support HA groups using password authentication only.
- HA groups cannot contain both Luna PCIe HSM 7s and Luna Network HSM 7s.
- All must be running one of the supported software/firmware versions. Generally, Thales recommends using HSMs with the same software/firmware for HA. However, mixed-version HA groups containing Luna 6 and 7 member partitions and Luna Cloud HSM services are supported. See "Cloning Keys Between Luna 6, Luna 7, and Luna Cloud HSM, Password or Multifactor Quorum" on page 210 for more information.
- For Luna Network HSM 7s, network setup must be complete, including network time synchronization, and the appliances must be accessible via SSH.
- HSM policies **7: Allow Cloning** and **16: Allow Network Replication** must be set to **1**(see Setting HSM Policies Manually).
- HSM policies must be consistent across all HSMs.

> **NOTE** For HSM policy **12: Allow non-FIPS algorithms**, note the following:
>
> - If you are using Luna HSM Client 10.4.0 or newer, you *can* set up an HA group with a mix of FIPS and non-FIPS partitions as members. However, some limitations must be considered. For more information, refer to "Key Replication" on page 419.
> - If you are using Luna HSM Client 10.3.0 or older, you *cannot* set up an HA group with a mix of FIPS and non-FIPS partitions as members.

> The client must be able to access all the application partitions using NTLS or STC links for Network HSMs, or XTC/REST for Luna Cloud HSM Services (see "Client-Partition Connections" on page 107).

A client can have a session with an STC slot, making use of only one appID. When running an HA command if you are already logged in or have a session open with an appID to a member of that HA slot, you will not be able to log into that slot at this time. When you run a command like "ha list", lunacm logs into each member using a randomly created appID. If any one of these slots already has a login session, such an attempt is rejected with CKR_ACCESS_ID_ALREADY_EXISTS. The workaround is to close the problem session first.

Partitions

The Partition SO must ensure that all partitions in an HA group meet the following prerequisites:

> The partitions must be created on different HSMs; partitions on a single HSM cannot provide failover for each other, as they have a single point of failure.

> All partitions must be visible in LunaCM on the client workstation.

> All partitions must be initialized with the same cloning domain:

- Password-authenticated partitions must share the same domain string.
- multifactor quorum-authenticated partitions must share the same red domain PED key.

> Partition policies **0: Allow private key cloning** and **4: Allow secret key cloning** must be set to **1** on all partitions.

> Partition policies must be consistent across all member partitions.

> The Crypto Officer role on each partition must be initialized with the same CO credential (password or black PED key).

> PED-authenticated partitions must have partition policies **22: Allow activation** and **23: Allow auto-activation** set to **1**. All partitions must be activated and have auto-activation enabled, so that they can retain their login state after failure/recovery. Each partition must have the same activation challenge secret set (see "Activation on Multifactor Quorum-Authenticated Partitions" on page 373)

> **NOTE** If HSM policy **21: Force user PIN change after set/reset** is set to **1** (the default setting), the Crypto Officer must change the initial CO credential before using the partition for cryptographic operations. This applies to the activation challenge secret as well (see role changepw).

To set up an HA group

1. Decide which partition will serve as the primary member (see "The Primary Partition" on page 418). Create a new HA group, specifying the following information:
   - the group label (do not call the group "HA")
   - the Serial number OR the slot number of the primary member partition
   - the Crypto Officer password or challenge secret for the partition

   lunacm:>**hagroup creategroup -label** <label> {**-slot** <slotnum> | **-serialnumber** <serialnum>}

```
lunacm:> hagroup creategroup -label myHAgroup -slot 0

        Enter the password: ********
```

```
        New group with label "myHAgroup" created with group number 1154438865287.
        Group configuration is:


         HA Group Label:  myHAgroup
        HA Group Number:  1154438865287
       HA Group Slot ID:  Not Available
        Synchronization:  enabled
          Group Members:  154438865287
             Needs sync:  no
        Standby Members:  <none>


Slot #    Member S/N                    Member Label    Status
======    ==========                    ============    ======
     0  154438865287                             par0     alive


Command Result : No Error
```

LunaCM generates a serial number for the HA group (by adding a "1" before the primary partition serial number), assigns it a virtual slot number, and automatically restarts.

```
lunacm (64-bit) v7.3.0. Copyright (c) 2018 SafeNet. All rights reserved.


        Available HSMs:

        Slot Id ->              0
        Label ->                par0
        Serial Number ->        154438865287
        Model ->                LunaSA 7.3.0
        Firmware Version ->     7.3.0
        Configuration ->        Luna User Partition With SO (PW) Key
                                Export With Cloning Mode
        Slot Description ->     Net Token Slot

        Slot Id ->              1
        Label ->                par1
        Serial Number ->        1238700701509
        Model ->                LunaSA 7.3.0
        Firmware Version ->     7.3.0
        Configuration ->        Luna User Partition With SO (PW) Key
                                Export With Cloning Mode
        Slot Description ->     Net Token Slot

        Slot Id ->              5
        HSM Label ->            myHAgroup
        HSM Serial Number ->    1154438865287
        HSM Model ->            LunaVirtual
        HSM Firmware Version -> 7.3.0
        HSM Configuration ->    Luna Virtual HSM (PW) Key Export With
                                Cloning Mode
        HSM Status ->           N/A - HA Group


Current Slot Id: 0
```

2. Add another partition to the HA group, specifying either the slot or the serial number. If the new member contains cryptographic objects, you are prompted to decide whether to replicate the objects within the HA group, or delete them.

   lunacm:>**hagroup addmember -group** <grouplabel> {**-slot** <slotnum> | **-serialnumber** <serialnum>}

```
lunacm:> hagroup addmember -group myHAgroup -slot 1


        Enter the password: ********


Warning:  There are objects currently on the new member.
          Do you wish to propagate these objects within the HA
          group, or remove them?

          Type 'copy' to keep and propagate the existing
          objects, 'remove' to remove them before continuing,
          or 'quit' to stop adding this new group member.
          >  copy

        Member 1238700701509 successfully added to group myHAgroup. New group
        configuration is:

         HA Group Label:  myHAgroup
        HA Group Number:  1154438865287
       HA Group Slot ID:  5
        Synchronization:  enabled
          Group Members:  154438865287, 1238700701509
             Needs sync:  no
        Standby Members:  <none>


Slot #    Member S/N                    Member Label    Status
======    ==========                    ============    ======
     0  154438865287                           par0     alive
     1  1238700701509                          par1     alive


        Please use the command "ha synchronize" when you are ready
        to replicate data between all members of the HA group.
        (If you have additional members to add, you may wish to wait
        until you have added them before synchronizing to save time by
        avoiding multiple synchronizations.)

Command Result : No Error
```

   Repeat this step for each additional HA group member.

3. If you are adding member partitions that already have cryptographic objects stored on them, initiate a manual synchronization. You can tell whether this step is required by checking the line **Needs Sync : yes/no** in the HA group output. This will also confirm that the HA group is functioning correctly.

   lunacm:> **hagroup synchronize -group** <grouplabel>

4. [Optional] If you created an HA group out of empty partitions, and you want to verify that the group is functioning correctly, see "Verifying an HA Group" on the next page.

5. Specify which member partitions, if any, will serve as standby members.

See "Setting an HA Group Member to Standby" on page 438.

6. Set up and configure auto-recovery (recommended). If you choose to use manual recovery, you will have to execute a recovery command whenever a group member fails.

   See "Configuring HA Auto-Recovery" on page 439.

7. [Optional] Enable HA Only mode (recommended).

   See "Enabling/Disabling HA Only Mode" on page 440.

8. [Optional] Configure HA logging.

   See "HA Logging" on page 445 for procedures and information on reading HA logs.

The HA group is now ready for your application.

Verifying an HA Group

After creating an HA group in LunaCM, you can see the group represented as a virtual slot alongside the physical slots:

```
lunacm (64-bit) v7.3.0. Copyright (c) 2018 SafeNet. All rights reserved.


        Available HSMs:

        Slot Id ->              0
        Label ->                par0
        Serial Number ->        154438865287
        Model ->                LunaSA 7.3.0
        Firmware Version ->     7.3.0
        Configuration ->        Luna User Partition With SO (PW) Key Export With Cloning Mode
        Slot Description ->     Net Token Slot

        Slot Id ->              1
        Label ->                par1
        Serial Number ->        1238700701509
        Model ->                LunaSA 7.3.0
        Firmware Version ->     7.3.0
        Configuration ->        Luna User Partition With SO (PW) Key Export With Cloning Mode
        Slot Description ->     Net Token Slot

        Slot Id ->              5
        HSM Label ->            myHAgroup
        HSM Serial Number ->    1154438865287
        HSM Model ->            LunaVirtual
        HSM Firmware Version -> 7.3.0
        HSM Configuration ->    Luna Virtual HSM (PW) Key Export With Cloning Mode
        HSM Status ->           N/A - HA Group


Current Slot Id: 0
```

The following procedure is one way to verify that your HA group is working as intended:

To verify an HA group

1. Exit LunaCM and run **multitoken** against the HA group slot number (slot 5 in the example) to create some objects on the HA group partitions.

   ./multitoken -mode <keygen_mode> **-key** <key_size> **-nodestroy -slots** <HA_virtual_slot>

   You can hit **Enter** at any time to stop the process before the partitions fill up completely. Any number of created objects will be sufficient to show that the HA group is functioning.

2. Run LunaCM and check the partition information on the two physical slots. Check the object count under "Partition Storage":

   lunacm:> **partition showinfo**

```
        Current Slot Id: 0

lunacm:> partition showinfo

...(clip)...

        Partition Storage:
                Total Storage Space:  325896
                Used Storage Space:   22120
                Free Storage Space:   303776
                Object Count:         14
                Overhead:             9648

Command Result : No Error


lunacm:> slot set slot 1

        Current Slot Id:    1     (Luna User Slot 7.0.1 (PW) Signing With Cloning Mode)

Command Result : No Error


lunacm:> partition showinfo

...(clip)...

        Partition Storage:
                Total Storage Space:  325896
                Used Storage Space:   22120
                Free Storage Space:   303776
                Object Count:         14
                Overhead:             9648

Command Result : No Error
```

3. To remove the test objects, login to the HA virtual slot and clear the virtual partition.

   lunacm:> **slot set -slot** <HA_virtual_slot>

   lunacm:> **partition login**

   lunacm:> **partition clear**

If you are satisfied that your HA group is working, you can begin using your application against the HA virtual slot. The virtual slot assignment will change depending on how many more application partitions are added to your client configuration. If your application invokes the HA group label, this will not matter. If you have applications that invoke the slot number, see "Enabling/Disabling HA Only Mode" on page 440.

Setting an HA Group Member to Standby

Some HA group members can be designated as standby members. Standby members do not perform any cryptographic operations unless all active members have failed (see "Standby Members" on page 422 for details). They are useful as a last resort against loss of application service.

Prerequisites

> The partition you want to designate as a standby member must already be a member of the HA group (see "Adding/Removing an HA Group Member" on page 449).

> The Crypto Officer must perform this procedure.

To set an HA group member to standby

1. [Optional] Check the serial number of the member you wish to set to standby mode.

   lunacm:> **hagroup listgroups**

2. Set the desired member to standby mode by specifying the serial number.

   lunacm:> **hagroup addstandby -group** <label> **-serialnumber** <member_serialnum>

```
lunacm:> hagroup addstandby -group myHAgroup -serialnumber 2855496365544

        The member 2855496365544 was successfully added to the standby list for the HA Group
myHAgroup.


Command Result : No Error
```

To make a standby HA member active

> **NOTE** By default, a Luna Cloud HSM service from Thales DPoD is always added to an HA group as a standby member. If you prefer to use the Luna Cloud HSM service as an active HA member, you must first edit the following toggle in the **Chrystoki.conf/crystoki.ini** configuration file (see "Configuration File Summary" on page 76):
>
> ```
> [Toggles]
> lunacm_cv_ha_ui = 0
> ```

1. [Optional] Check the serial number of the standby member.

   lunacm:> **hagroup listgroups**

```
        If you would like to see synchronization data for group myHAgroup,
        please enter the password for the group members. Sync info
        not available in HA Only mode.

        Enter the password: ********
```

```
            HA auto recovery:  disabled
            HA recovery mode:  activeBasic
   Maximum auto recovery retry:  0
   Auto recovery poll interval:  60 seconds
                   HA logging:  disabled
            Only Show HA Slots:  no

              HA Group Label:  myHAgroup
             HA Group Number:  11238700701509
            HA Group Slot ID:  5
             Synchronization:  enabled
               Group Members:  154438865287, 1238700701509
                  Needs sync:  no
            Standby Members:  2855496365544


Slot #    Member S/N                    Member Label    Status
======    ==========                    ============    ======
     0  154438865287                            par0     alive
     1  1238700701509                           par1     alive
     2  2855496365544                           par2     alive
```

2. Remove the member from standby and return it to active HA use.

   lunacm:> **hagroup removestandby -group** <label> **-serialnumber** <member_serialnum>

```
lunacm:> hagroup removestandby -group myHAgroup -serialnumber 2855496365544

        The member 2855496365544 was successfully removed from the standby list for the HA
Group myHAgroup.


Command Result : No Error
```

Configuring HA Auto-Recovery

When auto-recovery is enabled, Luna HSM Client performs periodic recovery attempts when it detects a member failure. HA auto-recovery is disabled by default for new HA groups. To enable it, you must set a maximum number of recovery attempts. You can also set the frequency of recovery attempts, and the auto-recovery mode (**activeBasic** or **activeEnhanced**). These settings will apply to all HA groups configured on the client.

To configure HA auto-recovery

1. Set the desired number of recovery attempts by specifying the retry count as follows:
   - Set a value of **0** to disable HA auto-recovery
   - Set a value of **-1** for unlimited retries
   - Set any specific number of retries from **1** to **500**

   lunacm:> **hagroup retry -count** <retries>
2. [Optional] Set the desired frequency of recovery attempts by specifying the time in seconds. The acceptable range is **60-1200** seconds (default: **60**).

lunacm:> **hagroup interval -interval** <seconds>

3. [Optional] Set the auto-recovery mode. The default is **activeBasic**.

   lunacm:> **hagroup recoverymode -mode** {**activeBasic** | **activeEnhanced**}

4. [Optional] Check that auto-recovery has been enabled. You are prompted for the Crypto Officer password/challenge secret.

   lunacm:> **hagroup listgroups**

Enabling/Disabling HA Only Mode

By default, client applications can see both physical slots and virtual HA slots. Directing applications at the physical slots bypasses the high availability and load balancing functionality. An application must be directed at the virtual HA slot to use HA load balancing and redundancy. HA Only mode hides the physical slots and leaves only the HA group slots visible to applications, simplifying the PKCS#11 slot numbering (see "Slot Numbering and Behavior" on page 564), especially "Settings Affecting Slot Order" on page 565.

If an HA group member partition fails and/or is recovered, *while HAonly is enabled*, then slot numbers do not change.

If an HA group member partition fails and/or is recovered, *while HAonly is **not** enabled*, all visible slot numbers can change, including the HA group virtual slots. This can cause applications to direct operations to the wrong slot.

If a physical slot in the HA group receives a direct request, the results will not be replicated on the other partitions in the group (see "HA Troubleshooting" on page 463) When HA Only mode is enabled, the HA virtual slots are not affected by partition slot changes.

The virtual slot defaults to slot 0, unless this behavior is modified by
*Thales recommends **enabling HAonly mode on all clients running HA groups**.*

> **NOTE** Individual partition slots are still visible in LunaCM when HA Only mode is enabled. They are hidden only from client applications. Use **CKdemo** (Option **11**) to see the slot numbers to use with client applications.

To enable HA Only mode

1. Enable HA Only mode in LunaCM.

   lunacm:> **hagroup haonly -enable**

2. [Optional] Since LunaCM still displays the partitions, you can check the status of HA Only mode at any time.

   lunacm:> **hagroup haonly -show**

To disable HA Only mode

1. Disable HA Only mode in LunaCM.

   lunacm:> **hagroup haonly -disable**

Example config file for a large HA group

This chrystoki.conf file and the 16 network HSMs (with slightly altered IP addresses) were used in our testing.

```
Chrystoki2 = {
  LibUNIX = /usr/safenet/lunaclient/lib/libCryptoki2.so;
  LibUNIX64 = /usr/safenet/lunaclient/lib/libCryptoki2_64.so;
}
Luna = {
  DefaultTimeOut = 500000;
  PEDTimeout1 = 100000;
  PEDTimeout2 = 200000;
  PEDTimeout3 = 20000;
  KeypairGenTimeOut = 2700000;
  CloningCommandTimeOut = 300000;
  CommandTimeOutPedSet = 720000;
}
  CardReader = {
  RemoteCommand = 1;
}
Misc = {
  PE1746Enabled = 0;
  ValidateHost = 0;
  ToolsDir = /usr/safenet/lunaclient/bin;
  PartitionPolicyTemplatePath = /usr/safenet/lunaclient/data/partition_policy_templates;
  ProtectedAuthenticationPathFlagStatus = 0;
  MutexFolder = /usr/safenet/lunaclient/lock;
  PluginModuleDir = /usr/safenet/lunaclient/plugins;
}
LunaSA Client = {
  ReceiveTimeout = 20000;
  SSLConfigFile = /usr/safenet/lunaclient/bin/openssl.cnf;
  ClientPrivKeyFile = /usr/safenet/lunaclient/cert/client/192.168.143.48Key.pem;
  ClientCertFile = /usr/safenet/lunaclient/cert/client/192.168.143.48.pem;
  ServerCAFile = /usr/safenet/lunaclient/cert/server/CAFile.pem;
  NetClient = 1;
```

```
TCPKeepAlive = 1;
ServerName00 = 192.121.10.63;
ServerPort00 = 1792;
ServerHtl00 = 0;
ServerName01 = 192.121.10.53;
ServerPort01 = 1792;
ServerHtl01 = 0;
ServerName02 = 192.121.10.62;
ServerPort02 = 1792;
ServerHtl02 = 0;
ServerName03 = 192.121.10.59;
ServerPort03 = 1792;
ServerHtl03 = 0;
ServerName04 = 192.121.10.52;
ServerPort04 = 1792;
ServerHtl04 = 0;
ServerName05 = 192.121.10.64;
ServerPort05 = 1792;
ServerHtl05 = 0;
ServerName06 = 192.121.10.50;
ServerPort06 = 1792;
ServerHtl06 = 0;
ServerName07 = 192.121.10.51;
ServerPort07 = 1792;
ServerHtl07 = 0;
ServerName08 = 192.121.10.65;
ServerPort08 = 1792;
ServerHtl08 = 0;
ServerName09 = 192.121.10.58;
ServerPort09 = 1792;
ServerHtl09 = 0;
ServerName10 = 192.121.10.60;
ServerPort10 = 1792;
ServerHtl10 = 0;
ServerName11 = 192.121.10.56;
```

```
  ServerPort11 = 1792;
  ServerHtl11 = 0;
  ServerName12 = 192.121.10.57;
  ServerPort12 = 1792;
  ServerHtl12 = 0;
  ServerName13 = 192.121.10.55;
  ServerPort13 = 1792;
  ServerHtl13 = 0;
  ServerName14 = 192.121.10.54;
  ServerPort14 = 1792;
  ServerHtl14 = 0;
  ServerName15 = 192.121.10.61;
  ServerPort15 = 1792;
  ServerHtl15 = 0;
  ServerName16 = 192.168.141.93;
  ServerPort16 = 1792;
  ServerHtl16 = 0;
  ServerName17 = 192.168.141.198;
  ServerPort17 = 1792;
  ServerHtl17 = 0;
}
Secure Trusted Channel = {
  SoftTokenDir = /usr/safenet/lunaclient/configData/token;
  ClientIdentitiesDir = /usr/safenet/lunaclient/data/client_identities;
  PartitionIdentitiesDir = /usr/safenet/lunaclient/data/partition_identities;
  ClientTokenLib = /usr/safenet/lunaclient/lib/libSoftToken.so;
}
PedServer = {
  ServerCAFile = /usr/safenet/lunaclient/PEDserver/CAFile.pem;
  PedConfigFile = /etc/pedServer.conf;
}
VirtualToken = {
  VirtualToken00Label=My_HA;
  VirtualToken00SN = 11287408863039;
```

```
 VirtualToken00Members =
1287408863039,1327020333026,1335064630247,1335062301941,1377509648637,1327024989629,137877
8575411,1378780903715,1305890956067,1305921224049,1372948497173,1459759386384,123865646369
6,1485871338177,1358801709921,1259264300111,1382217483700,1335066958603;
}
HASynchronize = {
}
 HAConfiguration = {
 haLogStatus = enabled;
 reconnAtt = -1;
 haLogPath = /usr/safenet/lunaclient/;
}
 CkLog2 = {
 Enabled = ;
 NewFormat = ;
 File = ;
 FileSize = ;
 Error = ;
 LibUNIX = ;
 LibUNIX64 = ;
}
 Ped Server = {
 PedConfigFile = /etc/pedServer.conf;
}
RBS = {
 CmdProcessor = /usr/safenet/lunaclient/rbs/lib/librbs_processor2.so;
 DaemonName = RBSD;
 HostPort = 1792;
 ClientAuthFile = /usr/safenet/lunaclient/rbs/clientauth.dat;
 ServerSSLConfigFile = /usr/safenet/lunaclient/rbs/server/server.cnf;
 ServerPrivKeyFile = /usr/safenet/lunaclient/rbs/server/serverkey.pem;
 ServerCertFile = /usr/safenet/lunaclient/rbs/server/server.pem;
 NetServer = 1;
 HostName = 0.0.0.0;
}
```


HA Logging

Logging of HA-related events takes place on the Luna HSM Client workstation. The log file **haErrorLog.txt** shows HA errors, as well as add-member and delete-member events. It does not record status changes of the group as a whole (like adding or removing the group). Some information is recorded in the log file to help Thales customer support staff troubleshoot operational issues.

The HA log rotates after the configured maximum length is reached. When it finishes writing the current record (even if that record slightly exceeds the configured maximum), the file is renamed to include the timestamp and the next log entry begins a new haErrorLog.txt.

> "Configuring HA Logging" below

> "HA Log Messages" on the next page

Configuring HA Logging

Using Luna HSM Client 7.2.0 or newer, logging is automatically enabled when you configure an HA group (see "Setting Up an HA Group" on page 432), but you must configure a valid destination path before logging can begin. HA groups are configured on the client using LunaCM. The HA configuration settings are saved to the **Chrystoki.conf** (Linux/Unix) or **crystoki.ini** (Windows) file, as illustrated in the following example:

```
VirtualToken = {
VirtualToken00Label = haGroup1; // The label of the HA group.
VirtualToken00SN = 11234840370164; // The pseudo serial number of the HA group.
VirtualToken00Members = 1234840370164, 1234924189183; // The serial number of the members.
VirtualTokenActiveRecovery = activeEnhanced; // The recovery mode.
}
HASynchronize = {
haGroup1 = 1; // Enable automatic synchronization of objects.
}
HAConfiguration = {
HAOnly = 1; // Enable listing HA groups only via PKCS#11 library.
haLogPath = /tmp/halog; // Base path of the HA log file; i.e., "/tmp/halog/haErrorLog.txt".
haLogStatus = enabled; // Enable HA log.
logLen = 100000000; // Maximum size of HA log file in bytes.
failover_on_deactivation = 1; // if a partition becomes deactivated then the client will
immediately
                              // failover and resume its operation on the other HA partitions.
This
                              // is currently an alpha feature
reconnAtt = 120; // Number of recovery attempts.
}
HARecovery = {
haGroup1 = 1; // Deprecated in this release as auto recovery will cover the use case. When
cryptoki
              // loads into memory it reads the number and if the number changes (gets
incremented)
              // then cryptoki interprets this as a manual recovery attempt.
}
```

To configure HA logging

Use the LunaCM command **hagroup halog**.

1. Set a valid path for the log directory. You must specify an existing directory.

lunacm:> **hagroup halog -path** <filepath>

```
lunacm:> hagroup halog -path "C:\Program Files\Safenet\Lunaclient\halog"

        HA Log path successfully set to C:\Program Files\Safenet\Lunaclient\halog.

Command Result : No Error
```

2. [Optional] Set the maximum length for individual log files.

   lunacm:> **hagroup halog -maxlength** <max_file_length>

```
lunacm:> hagroup halog -maxlength 500000

        HA Log maximum file size was successfully set to 500000.

Command Result : No Error
```

3. [Optional] Enable or disable HA logging at any time.

   lunacm:> **hagroup halog -disable**

   lunacm:> **hagroup halog -enable**

```
lunacm:> hagroup halog -disable

        HA Log was successfully disabled.

Command Result : No Error
```

4. [Optional] View the current status of the HA logging configuration.

   lunacm:> **hagroup halog -show**

```
lunacm:> hagroup halog -show

         HA Log: enabled
       Log File: C:\Program Files\Safenet\Lunaclient\halog\haErrorLog.txt
Max File Length: 500000 bytes

Command Result : No Error
```

HA Log Messages

The following table provides descriptions of the messages generated by the HA sub-system and saved to the HA log. The HA log is saved to the location specified by **haLogPath** in the **Chrystoki.conf** (Linux/Unix) or **crystoki.ini** (Windows) file.

Message Format

Every HA log message has a consistent prefix consisting of the date, time, process id, and serial number (of the affected HA group). For example:

```
Wed Oct  4 16:29:21 2017 : [17469] HA group: 11234840370164 …
```

Message Descriptions

In the message descriptions, the term **connection** refers to the connection between the Luna HSM Client and the Luna Network HSM 7 appliance. A connection is considered **valid** if the appliance responds correctly on the

IP address and port. The connection can transition to **invalid** for a number of reasons. Some examples include if the appliance Ethernet cable is detached, if the appliance is shutdown/rebooted, or if the NTLS service is stopped/restarted.

Message ID	Message/Description
HALOG_CONFIGURED_AS_PASSWORD	`<MessagePrefix> configured as a "PASSWORD Based" virtual device` **Description:** Message advising that the virtual partition is password-authenticated. This means that you cannot add a PED-authenticated member to the group.
HALOG_CONFIGURED_AS_PED	`<MessagePrefix> configured as a "PED Based" virtual device` **Description:** Message advising that the virtual partition is PED-authenticated. This means that you cannot add a password-authenticated member to the group.
HALOG_DROPMEMBER	`<MessagePrefix> has dropped member: <SerialNumber>` **Description:** The connection changed from valid to invalid, determined after an HSM command (such as C_Sign) fails.
HALOG_DROPUNRECOVERABLE	`<MessagePrefix> unable to reach member: <SerialNumber>. Manual Recover or Auto Recovery will be able to recover this member` **Description:** The connection is invalid, as determined during a call to C_Initialize.
HALOG_LOGINFAILED	`<MessagePrefix> can not login to member: <SerialNumber>, autorecovery will be disabled. Code: <ErrorCodeHex> : <ErrorCodeString>` **Description:** The connection changed from valid to invalid, as determined during a call to C_Login.
HALOG_MEMBER_DEACTIVATED	`<MessagePrefix> member: <SerialNumber> deactivated` **Description:** The user manually deactivated the partition, as determined after an HSM command (such as C_Sign) fails.
HALOG_MEMBER_NOW_ACTIVATED	`<MessagePrefix> recovery attempt <AttemptNumber> member <SerialNumber> is now activated and will be reintroduce back into the HA group.` **Description:** Additional info about the recovered partition, which was deactivated and is now becoming activated.
HALOG_MEMBER_REVOKED	`<MessagePrefix> member: <SerialNumber> revoked` **Description:** The user manually revoked the partition, as determined during a periodic recovery attempt.
HALOG_MEMBERS_OFFLINE	`<MessagePrefix> all members gone offline.` **Description:** A situation where all members go offline. Recovery is not possible at this point.

Message ID	Message/Description
HALOG_MGMT_THREAD_START	`<MessagePrefix> management thread started` **Description:** This thread is responsible for managing all members and HA in general while the HA group is active. The thread starts up when the application first launches.
HALOG_MGMT_THREAD_TERMINATE	`<MessagePrefix> management thread terminated` **Description:** This thread is responsible for managing all members and HA in general while the HA group is active. If the client application shuts down, this thread will simply terminate. The thread will start up again once the application re-launches.
HALOG_NEWMEMBER	`<MessagePrefix> detected new member member: <SerialNumber>` **Description:** The user manually added a member to the HA group without restarting the application, as determined during a periodic recovery attempt.
HALOG_RECOVERED	`<MessagePrefix> recovery attempt <Integer> succeeded for member: <SerialNumber>` **Description:** The connection changed from invalid to valid, as determined during a periodic recovery attempt.
HALOG_RECOVERY_ATTEMPT_#_REINTRODUCING	`<MessagePrefix> recovery attempt <AttemptNumber> reintroducing <Number> token objects to recovered token <TokenNumber>` **Description:** Additional info about the recovered partition at which some objects were cloned.

HALOG_RECOVERYFAILED	`<MessagePrefix> recovery attempt <Integer> failed for member: <SerialNumber>. Code: <ErrorCodeHex> : <ErrorCodeString>.`
	If autorecovery fails, then a second message is logged, as follows:
	`<MessagePrefix> exceeded maximum number of autorecovery attempts for member: <SerialNumber>. Autorecovery will be disabled`
	Description: The connection remains invalid, as determined during a periodic recovery attempt.

HALOG_RENABLEMEMBER (deprecated)	`<MessagePrefix> Re-enable auto recovery process for member: <SerialNumber>` **Description:** The user manually requested partition recovery, as determined during a periodic recovery attempt before an HSM command.
HALOG_UNRECOVERABLE (deprecated)	`<MessagePrefix> recovery attempt <Integer> failed for member: <SerialNumber>. Manual Recover or Auto Recovery will not be able to recover this member. Code: <ErrorCodeHex> : <ErrorCodeString>` **Description:** The connection is invalid and is not eligible for recovery.

Message ID	Message/Description
No ID*	`<MessagePrefix> member <SerialNumber> is not activated and is excluded from the HA group` **Description:** The HA member was not activated at the time when a C_Initialize call was made, and is therefore excluded from the HA group. Once the partition is activated, the HA group will attempt an automatic recovery, resulting in one of the two messages below
No ID*	`<MessagePrefix> recovery attempt <SerialNumber> is not activated and cannot be reintroduced back into the HA group\n` **Description:** Recovery failed
No ID*	`<MessagePrefix> recovery attempt <SerialNumber> is now activated and will be reintroduce back into the HA group.\n` **Description:** Recovery succeeded

* You might encounter these extra messages in the HA logs. They were added for HA development testing and therefore have no Message IDs assigned to them. They could duplicate information covered by other log messages as defined above.

Adding/Removing an HA Group Member

You can add a new member to an HA group at any time using LunaCM, even if your application is running. Cryptographic objects will be replicated on the new partition and operations will be scheduled according to the load-balancing algorithm (see "Load Balancing" on page 417).

Likewise, you can remove a member at any time, and currently-scheduled operations will fail over to the rest of the group members (see "Failover" on page 420).

> **NOTE** If you remove the partition that was used to create the group, the HA group serial number changes to reflect this. This is to prevent another HA group from being assigned the same serial number as the original. If your application queries the HA group serial number, it must redirect operations to the new serial.

Prerequisites

The new member partition must:

- be assigned to the client and visible in LunaCM
- be initialized with the same domain string/red domain PED key as the other partitions in the group
- have the Crypto Officer role initialized with the same credentials as the other partitions in the group
- be activated and have auto-activation enabled (multifactor quorum-authenticated)

> **NOTE V1 partitions:** If you add an application partition with an existing SMK to an HA group, the primary member's SMK overwrites the existing SMK of the joining partition.
>
> If a partition's SMK has ever been used to encrypt important SKS objects, save a backup of the SMK before adding that partition to any HA group.

To add an HA group member

1. Open LunaCM on the client workstation and ensure that the new partition is visible.

```
lunacm (64-bit) v7.3.0. Copyright (c) 2018 SafeNet. All rights reserved.


        Available HSMs:

        Slot Id ->              0
        Label ->                par0
        Serial Number ->        154438865287
        Model ->                LunaSA 7.3.0
        Firmware Version ->     7.3.0
        Configuration ->        Luna User Partition With SO (PW) Key Export With Cloning
Mode
        Slot Description ->     Net Token Slot

        Slot Id ->              1
        Label ->                par1
        Serial Number ->        1238700701509
        Model ->                LunaSA 7.3.0
        Firmware Version ->     7.3.0
        Configuration ->        Luna User Partition With SO (PW) Key Export With Cloning
Mode
        Slot Description ->     Net Token Slot

        Slot Id ->              2
        Label ->                par2
        Serial Number ->        2855496365544
        Model ->                LunaSA 7.3.0
        Firmware Version ->     7.3.0
        Configuration ->        Luna User Partition With SO (PW) Key Export With Cloning
Mode
        Slot Description ->     Net Token Slot

        Slot Id ->              5
        HSM Label ->            myHAgroup
        HSM Serial Number ->    1154438865287
        HSM Model ->            LunaVirtual
        HSM Firmware Version -> 7.3.0
        HSM Configuration ->    Luna Virtual HSM (PW) Key Export With Cloning Mode
        HSM Status ->           N/A - HA Group


Current Slot Id: 0
```

2. Add the new partition to the HA group by specifying either the slot or the serial number. You are prompted for the Crypto Officer password/challenge secret.

   lunacm:> **hagroup addmember -group** <label> {**-slot** <slotnum> | **-serial** <serialnum>}

```
lunacm:> hagroup addmember -group myHAgroup -slot 2

         Enter the password: ********
         Member 2855496365544 successfully added to group myHAgroup. New group
         configuration is:

          HA Group Label:  myHAgroup
         HA Group Number:  1154438865287
        HA Group Slot ID:  5
        Synchronization: enabled
           Group Members:  154438865287, 1238700701509, 2855496365544
              Needs sync:  no
         Standby Members:  <none>


Slot #    Member S/N                    Member Label    Status
======    ==========                    ============    ======
     0  154438865287                           par0     alive
     1  1238700701509                           par1     alive
     2  2855496365544                           par2     alive


         Please use the command "ha synchronize" when you are ready
         to replicate data between all members of the HA group.
         (If you have additional members to add, you may wish to wait
         until you have added them before synchronizing to save time by
         avoiding multiple synchronizations.)

Command Result : No Error
```

To remove an HA group member

1. Remove the partition from the group by specifying either the slot or the serial number.

   lunacm:> **hagroup removemember -group** <label> {**-slot** <slotnum> | **-serial** <serialnum>}

```
lunacm:> hagroup removemember -group myHAgroup -slot 0

         Member 154438865287 successfully removed from group myHAgroup.


         Note: Serial number for the group changed to 11238700701509.
Command Result : No Error
```

> **NOTE** If you remove the partition that was used to create the group, the HA group serial number changes to reflect this. This is to prevent another HA group from being assigned the same serial number as the original. If your application queries the HA group serial number, it must redirect operations to the new serial.

   LunaCM restarts.

```
lunacm (64-bit) v7.3.0. Copyright (c) 2018 SafeNet. All rights reserved.


         Available HSMs:

         Slot Id ->              0
```

```
        Label ->                par0
        Serial Number ->        154438865287
        Model ->                LunaSA 7.3.0
        Firmware Version ->     7.3.0
        Configuration ->        Luna User Partition With SO (PW) Key Export With Cloning
Mode
        Slot Description ->     Net Token Slot

        Slot Id ->              1
        Label ->                par1
        Serial Number ->        1238700701509
        Model ->                LunaSA 7.3.0
        Firmware Version ->     7.3.0
        Configuration ->        Luna User Partition With SO (PW) Key Export With Cloning
Mode
        Slot Description ->     Net Token Slot

        Slot Id ->              2
        Label ->                par2
        Serial Number ->        2855496365544
        Model ->                LunaSA 7.3.0
        Firmware Version ->     7.3.0
        Configuration ->        Luna User Partition With SO (PW) Key Export With Cloning
Mode
        Slot Description ->     Net Token Slot

        Slot Id ->              5
        HSM Label ->            myHAgroup
        HSM Serial Number ->    11238700701509
        HSM Model ->            LunaVirtual
        HSM Firmware Version -> 7.3.0
        HSM Configuration ->    Luna Virtual HSM (PW) Key Export With Cloning Mode
        HSM Status ->           N/A - HA Group


Current Slot Id: 0
```

2. [Optional] Check that the partition was removed from the group.

   lunacm:> **hagroup listgroups**

Manually Recovering a Failed HA Group Member

Thales recommends using auto-recovery for all HA group configurations (see "Configuring HA Auto-Recovery" on page 439). If you do not enable auto-recovery and a member partition fails, or if the recovery retry count expires before the partition comes back online, you must recover the partition manually using LunaCM. You do not need to pause your application(s) to perform a manual recovery; the HA group handles load-balancing and automatically replicates any new or changed keys to the recovered member.

To perform a manual recovery of a failed HA group member

1. [Optional] Ensure that the failed member is available and visible in LunaCM by addressing the problem that caused the failure. Display the HA group to see the failed members. You are prompted for the Crypto Officer password/challenge secret.

   lunacm:> **hagroup listgroups**

```
          HA Group Label:  myHAgroup
         HA Group Number:  1154438865287
       HA Group Slot ID:  5
       Synchronization: enabled
          Group Members:  154438865287, 1238700701509
             Needs sync:  no
         Standby Members:  <none>


Slot #    Member S/N                    Member Label    Status
======    ==========                    ============    ======
------  154438865287                           par0     alive
------  1238700701509                     ------------     down
```

2. If you are using a multifactor quorum-authenticated partition with auto-activation disabled, or if the partition was down for longer than two hours, log in to the partition as Crypto Officer and present the black CO PED key.

   lunacm:> **slot set -slot** <slotnum>

   lunacm:> **role login -name co**

3. Execute the manual recovery command, specifying the HA group label.

   lunacm:>**hagroup recover**

   If you have an application running on the HA group, the failed members will be recovered the next time an operation is scheduled. Load-balancing and key replication is automatic.

4. If you do not currently have an application running, you can manually synchronize the contents of the HA group.

   > **CAUTION!** Never use manual synchronization if you have an application running. The HA group performs this automatically. Using this command on an HA group that is running an application could create conflicting key versions.

   lunacm:> **hagroup synchronize -group** <label>

Replacing an HA Group Member

Sometimes an HSM failure is permanent (from the perspective of the HA group). For example, if the HSM is re-initialized, the member partition is erased and must be recreated. In this case, you can recreate a partition on the same HSM or another HSM, and deploy the new member to the group. You do not need to pause your application to replace an HA group member.

Prerequisites

The Crypto Officer must complete this procedure, but any new member partition must first be created and assigned to the client by the HSM SO, and initialized by the Partition SO. All the prerequisites listed in "Setting Up an HA Group" on page 432 must be met.

> **NOTE V1 partitions:** If you add an application partition with an existing SMK to an HA group, the primary member's SMK overwrites the existing SMK of the joining partition.
>
> If a partition's SMK has ever been used to encrypt important SKS objects, save a backup of the SMK before adding that partition to any HA group.

To replace an HA group member

1. [Optional] Display the HA group to see the failed member. You are prompted for the Crypto Officer password/challenge secret.

   lunacm:> **hagroup listgroups**

```
         HA Group Label:  myHAgroup
        HA Group Number:  1154438865287
      HA Group Slot ID:  5
      Synchronization: enabled
         Group Members:  154438865287, 1238700701509
            Needs sync:  no
        Standby Members:  <none>


Slot #    Member S/N                    Member Label    Status
======    ==========                    ============    ======
------  154438865287                           par0     alive
------  1238700701509                     ------------     down
```

2. Prepare the new HA group member, whether that means creating a new partition on the original HSM or configuring a new Luna Network HSM 7, and assign the new partition to the HA client. Ensure that the new member partition and the HSM on which it resides meet the prerequisites outlined in "Setting Up an HA Group" on page 432 and is visible in LunaCM.

```
lunacm (64-bit) v7.3.0. Copyright (c) 2018 SafeNet. All rights reserved.


        Available HSMs:

        Slot Id ->              0
        Label ->                par0
        Serial Number ->        154438865287
        Model ->                LunaSA 7.3.0
        Firmware Version ->     7.3.0
        Configuration ->        Luna User Partition With SO (PW) Key Export With Cloning
Mode
        Slot Description ->     Net Token Slot

        Slot Id ->              1
        Label ->                par1
        Serial Number ->        1238700701510
        Model ->                LunaSA 7.3.0
        Firmware Version ->     7.3.0
        Configuration ->        Luna User Partition With SO (PW) Key Export With Cloning
Mode
        Slot Description ->     Net Token Slot

        Slot Id ->              5
        HSM Label ->            myHAgroup
        HSM Serial Number ->    1154438865287
```

```
        HSM Model ->            LunaVirtual
        HSM Firmware Version -> 7.3.0
        HSM Configuration ->    Luna Virtual HSM (PW) Key Export With Cloning Mode
        HSM Status ->           N/A - HA Group


Current Slot Id: 0
```

3. Add the new partition to the HA group by specifying either the slot or the serial number. You are prompted for the Crypto Officer password/challenge secret.

   lunacm:> **hagroup addmember -group** <label> {**-slot** <slotnum> | **-serial** <serialnum>}

```
lunacm:> hagroup addmember -group myHAgroup -slot 1

        Enter the password: ********
        Member 1238700701510 successfully added to group myHAgroup. New group
        configuration is:

         HA Group Label:  myHAgroup
        HA Group Number:  1154438865287
       HA Group Slot ID:  5
       Synchronization: enabled
          Group Members:  154438865287, 1238700701509, 1238700701510
             Needs sync:  no
        Standby Members:  <none>


Slot #    Member S/N                    Member Label    Status
======    ==========                    ============    ======
     0  154438865287                           par0     alive
------  1238700701509                     ------------     down
     1  1238700701510                           par1     alive


        Please use the command "ha synchronize" when you are ready
        to replicate data between all members of the HA group.
        (If you have additional members to add, you may wish to wait
        until you have added them before synchronizing to save time by
        avoiding multiple synchronizations.)

Command Result : No Error
```

   The new partition is now an active member of the HA group. If you have an application currently running, cryptographic objects are automatically replicated to the new member and it is assigned operations according to the load-balancing algorithm.

4. Remove the old partition from the group by specifying the serial number.

   lunacm:> **hagroup removemember -group** <label> **-serial** <serialnum>

   LunaCM restarts.

5. [Optional] If you do not currently have an application running, you can manually synchronize the contents of the HA group.

> **CAUTION!** Never use manual synchronization if you have an application running. The HA group performs this automatically. Using this command on an HA group that is running an application could create conflicting key versions.

lunacm:> **hagroup synchronize -group** <label>

6. [Optional] If you intend to have the new partition serve as a standby member, see "Setting an HA Group Member to Standby" on page 438.

Deleting an HA Group

Use LunaCM to delete an HA group from your configuration.

> **NOTE** This procedure only removes the HA group virtual slot; the member partitions and all their contents remain intact. Only the HSM SO can delete individual partitions.

To delete an HA group

1. Stop any applications currently using the HA group.
2. Delete the group by specifying its label (see **hagroup listgroups**).

   lunacm:> **hagroup deletegroup -label** <label>

```
lunacm:> hagroup deletegroup -label myHAgroup

        The HA group myHAgroup was successfully deleted.
Command Result : No Error
```

Changing passwords for an HA group

Changing CO passwords has previously been an operation performed on a single HSM application partition by means of the role changepw command, which is fine for situations where a partition is used in stand-alone fashion. HA groups require all member partitions to have the same password. Performing the multiple operations in quick sequence invites human error. Changing the password of one member, while an HA group is in use in a production scenario, breaks the group functionality.

The usual approach has been to individually remove all but one member, change their passwords while they are not part of the group, change the password of the single member in the group, and then re-add the other members, one-by-one, which is labor-intensive and leaves the operation of your application at risk with diminished or no redundancy until the group membership is restored.

Using Luna HSM Client 10.7.0 or newer, the partition changepw [for HA] command is added to LunaCM, to change the passwords for an entire HA group, quickly and efficiently, with minimum disruption and downtime. This is especially significant where organizational security regimes mandate regular rotation of credentials.

The **partition changepw** command operates on the current virtual slot for the HA group, to perform password change for the entire group.

For description of command options see partition changepw [for HA].

Caveats

If your application generates new connections during the execution of the group CO password change, some authentication failures could occur, in which case see below for appropriate responses. If you are sure that no such new-connection attempts will be made (since you control your application), you could use the **-noRollback** option. Otherwise, we suggest that allowing the automatic rollback of an interrupted/failed PW change for a group is the best option, so that you can simply try again. With -noRollback, the members are in whatever old/new password state they attained before the update failure, and any new connections are refused until the mismatch of passwords within the group is corrected.

Change the password for an HA group

Prerequisites

The HA group is functional.

All members are connected.

To change the CO password on all members of an HA group

1. On your client, open lunacm.

```
lunacm (64-bit) v10.7.0. Copyright (c) 2023 Thales Group. All rights reserved.


        Available HSMs:

        Slot Id ->              0
        Label ->                par
        Serial Number ->        1385675017771
        Model ->                LunaSA 7.8.4
        Firmware Version ->     7.8.4
        Bootloader Version ->   1.1.5
        Configuration ->        Luna User Partition With SO (PW) Key Export With Cloning
Mode
        Slot Description ->     Net Token Slot
        FM HW Status ->         FM Ready

        Slot Id ->              1
        Label ->                par1
        Serial Number ->        1238696044902
        Model ->                LunaSA 7.8.4
        Firmware Version ->     7.8.4
        Bootloader Version ->   1.1.5
        Configuration ->        Luna User Partition With SO (PW) Key Export With Cloning
Mode
        Slot Description ->     Net Token Slot
        FM HW Status ->         FM Ready

        Slot Id ->              2
        HSM Label ->            HA
        HSM Serial Number ->    11385675017771
        HSM Model ->            LunaVirtual
        HSM Firmware Version -> 7.8.4
        HSM Configuration ->    Luna Virtual HSM (PW) Key Export With Cloning Mode
```

```
          HSM Status ->           N/A - HA Group
          HSM Certificates ->     *** Test Certs ***
```

2. Select the virtual slot of the HA group.

```
lunacm> slot set slot 2
```

3. Use the partition changePw command to launch password change for all of the HSM partitions managed by the HA virtual slot.

```
lunacm> partition changePw -n co -oldPw userpin123 -newPw userpin1234 -logoutOther

  Confirming all members of HA are online... [OK]

  Confirming all members of HA can be logged into... [OK]

  Changing password of all members of HA group... [OK]

  Final summary of members:


Member S/N           Member Label                  Password Status
==========           ============                  ===============
1213473506146        LNH_143.184_NTLS_v0_par1      Changed
91351086532          LNH_10.202_NTLS_v0_par1       Changed

Command Result : No Error
```

What to do in the event of a failure

Failure of the initial check

```
lunacm> partition changePw -n co -oldPw userpin123 -newPw userpin1234 -logoutOther

        Confirming all members of HA are online... [OK]

        Confirming all members of HA can be logged into... [FAILED]

        Failed at member: 1213473506146

        The role is logged out due to incorrect old password.

        Final summary of members:


Member S/N           Member Label                  Password Status
=========            ============                  ===============
1213473506146        LNH_143.184_NTLS_v0_par1      Unchanged
91351086532          LNH_10.202_NTLS_v0_par1       Unchanged

Command Result : 0xa0 (CKR_PIN_INCORRECT)
```

Response: ensure that all members are logged in and redo. If any member cannot be corrected and logged in, then remove it from the group, proceed with the remaining members, and afterward add the failed member or a new/replacement member that has the new group CO password.

Unsuccessful password change on all members with the -noRollback flag specified

```
lunacm> partition changePw -n co -oldPw userpin123 -newPw userpin1234 -noRollback

        Confirming all members of HA are online... [OK]

        Confirming all members of HA can be logged into... [OK]

        Changing password of all members of HA group... [FAILED]

        The following member(s) failed: 91351086532

        The role is logged out due to incorrect old password.

        Final summary of members:


Member S/N            Member Label                  Password Status
==========            ============                  ===============
1213473506146         LNH_143.184_NTLS_v0_par1      Changed
91351086532           LNH_10.202_NTLS_v0_par1       Unchanged

Command Result : 0xa0 (CKR_PIN_INCORRECT)
```

Response: Change the password for the partition that failed. Use **partition changePw** with the **-memberlist** option and the serial number of the failed partition. Until you do that, existing crypto operations against the group continue uninterrupted, because the prior session (using the oldPw) has not closed, but any attempt at a new operation against the virtual slot will fail while any members do not have the updated newPw.

Unsuccessful password change *without* the -noRollback flag specified, and a successful rollback

```
lunacm> partition changePw -n co -oldPw userpin123 -newPw userpin1234

        Confirming all members of HA are online... [OK]

        Confirming all members of HA can be logged into... [OK]

        Changing password of all members of HA group... [FAILED]

        Failed at member: 91351086532

        Rolling back members to old password... [OK]

        Members have successfully been rolled back to their original password.

        The role is logged out due to incorrect old password.

        Final summary of members:


Member S/N            Member Label                  Password Status
==========            ============                  ===============
1213473506146         LNH_143.184_NTLS_v0_par1      Unchanged
91351086532           LNH_10.202_NTLS_v0_par1       Unchanged

Command Result : 0xa0 (CKR_PIN_INCORRECT)
```

Response: remove the failed member from the HA group and try again. The remaining members can function with your client application both before and after a successful group password change. The failed one[former] member can be diagnosed and corrected, while separated from the group, then have its password updated so it can rejoin the group at your convenience.

Monitoring HA Status

Rapid HA group status checking

To check the status of HA group members, the old PKCS#11 Extensions function `CA_GetHAState` has been superseded by `CA_GetCurrentHaState()`, which is invoked programmatically and is demonstrated by CKDemo option 49, as in the following example.

Environment Requirements

When an HA group member becomes unavailable, the `CA_GetCurrentHaState()` function (once called) detects its unavailability within 3 seconds (in optimal conditions, depending on various factors including network hazards). For performance near the speed that was verified in testing, your operational environment must embody good IT conditions (network, CPU, RAM, storage) and reasonable perturbations (<=20% of optimal conditions). When those conditions are met, `CA_GetCurrentHaState()` can be repeated quickly and continually for ongoing monitoring. Requires Luna HSM Client 10.7.0 or newer.

When a previously unavailable HA group member becomes available again, the recovery process must detect the member and then confirm that it can be used for PKCS#11 operations. The recovery process is affected by several configuration file parameters, including "AutoReconnectInterval" on page 88 and "statusTimeout" on page 90; if the status takes longer than the time specified by **statusTimeout**, the API terminates and reports the status of all members fetched so far.

Example of HA Current Status check using CKDemo

Assume that the group (12 members in this example) is already created and the Crypto Officer is logged in.

1. Initially, all members are connected and working. Check the status.

```
(TITLE) menu titles, (99 or FULL) Full Help, (NONE) No help, (0 or EXIT) Quit


Status: Doing great, no errors (CKR_OK)
Enter your choice : 49

Slots available:
        slot#0 - Net Token Slot
        slot#1 - Net Token Slot
        slot#2 - Net Token Slot
        slot#3 - Net Token Slot
        slot#4 - Net Token Slot
        slot#5 - Net Token Slot
        slot#6 - Net Token Slot
        slot#7 - Net Token Slot
        slot#8 - Net Token Slot
        slot#9 - Net Token Slot
        slot#10 - Net Token Slot
        slot#11 - Net Token Slot
        slot#12 - Net Token Slot
```

```
        slot#19 - HA Virtual Card Slot
Select a slot (last selected slot = 19): 19

HA group 11327020333032 status:

     HSM 1327020333032   - CKR_OK
     HSM 1327024989635   - CKR_OK
     HSM 1378778575417   - CKR_OK
     HSM 1378780903721   - CKR_OK
     HSM 1305890956073   - CKR_OK
     HSM 1305921224055   - CKR_OK
     HSM 1372948497179   - CKR_OK
     HSM 1459759386390   - CKR_OK
     HSM 1238656463702   - CKR_OK
     HSM 1485871338183   - CKR_OK
     HSM 1358801709927   - CKR_OK
     HSM 1259264300119   - CKR_OK
     HSM 1382217483713   - CKR_OK
Status: Doing great, no errors (CKR_OK)
```

2. Some members are disconnected or disabled, and the scan is repeated.

```
Enter your choice : 49

Slots available:
        slot#0 - Net Token Slot
        slot#1 - Net Token Slot
        slot#2 - Net Token Slot
        slot#3 - Net Token Slot
        slot#4 - Net Token Slot
        slot#5 - Net Token Slot
        slot#6 - Net Token Slot
        slot#7 - Net Token Slot
        slot#8 - Net Token Slot
        slot#9 - Net Token Slot
        slot#10 - Net Token Slot
        slot#11 - Net Token Slot
        slot#12 - Net Token Slot
        slot#19 - HA Virtual Card Slot
Select a slot (last selected slot = 19): 19

HA group 11327020333032 status:

     HSM 1327020333032   - CKR_OK
     HSM 1327024989635   - CKR_OK
     HSM 1378778575417   - CKR_TOKEN_NOT_PRESENT
     HSM 1378780903721   - CKR_TOKEN_NOT_PRESENT
     HSM 1305890956073   - CKR_OK
     HSM 1305921224055   - CKR_OK
     HSM 1372948497179   - CKR_OK
     HSM 1459759386390   - CKR_TOKEN_NOT_PRESENT
     HSM 1238656463702   - CKR_TOKEN_NOT_PRESENT
     HSM 1485871338183   - CKR_TOKEN_NOT_PRESENT
     HSM 1358801709927   - CKR_TOKEN_NOT_PRESENT
     HSM 1259264300119   - CKR_OK
     HSM 1382217483713   - CKR_TOKEN_NOT_PRESENT
Status: Doing great, no errors (CKR_OK)
```

The HA group remains in operation, with reduced functional members, with non-functional members identified.

Assume that the connection, routing, or other problems are corrected, as indicated by log entries...

```
Mon Nov 27 16:56:28 2023 : [14686] HA group: 11327020333032 recovery attempt #3 succeeded
for member: 1378778575417
Mon Nov 27 16:56:31 2023 : [14686] HA group: 11327020333032 recovery attempt #3 succeeded
for member: 1378780903721
Mon Nov 27 16:56:33 2023 : [14686] HA group: 11327020333032 recovery attempt #3 succeeded
for member: 1459759386390
Mon Nov 27 16:56:35 2023 : [14686] HA group: 11327020333032 recovery attempt #3 succeeded
for member: 1238656463702
Mon Nov 27 16:56:38 2023 : [14686] HA group: 11327020333032 recovery attempt #3 succeeded
for member: 1485871338183
Mon Nov 27 16:56:41 2023 : [14686] HA group: 11327020333032 recovery attempt #3 succeeded
for member: 1358801709927
Mon Nov 27 16:56:48 2023 : [14686] HA group: 11327020333032 recovery attempt #5 succeeded
for member: 1382217483713
```

3. Check status again.

> **NOTE** Checking could have been performed every few seconds while corrective actions were in progress.

```
Enter your choice : 49

Slots available:
        slot#0 - Net Token Slot
        slot#1 - Net Token Slot
        slot#2 - Net Token Slot
        slot#3 - Net Token Slot
        slot#4 - Net Token Slot
        slot#5 - Net Token Slot
        slot#6 - Net Token Slot
        slot#7 - Net Token Slot
        slot#8 - Net Token Slot
        slot#9 - Net Token Slot
        slot#10 - Net Token Slot
        slot#11 - Net Token Slot
        slot#12 - Net Token Slot
        slot#19 - HA Virtual Card Slot
Select a slot (last selected slot = 19): 19

HA group 11327020333032 status:

     HSM 1327020333032   - CKR_OK
     HSM 1327024989635   - CKR_OK
     HSM 1378778575417   - CKR_OK
     HSM 1378780903721   - CKR_OK
     HSM 1305890956073   - CKR_OK
     HSM 1305921224055   - CKR_OK
     HSM 1372948497179   - CKR_OK
     HSM 1459759386390   - CKR_OK
     HSM 1238656463702   - CKR_OK
     HSM 1485871338183   - CKR_OK
     HSM 1358801709927   - CKR_OK
     HSM 1259264300119   - CKR_OK
```

```
     HSM 1382217483713   - CKR_OK
Status: Doing great, no errors (CKR_OK)

(TITLE) menu titles, (99 or FULL) Full Help, (NONE) No help, (0 or EXIT) Quit


Status: Doing great, no errors (CKR_OK)
Enter your choice :
```

And the log entries for that action say:

```
Mon Nov 27 16:56:59 2023 : [14686] HA group: 11327020333032 Initializing HA State API
Mon Nov 27 16:56:59 2023 : [14686] HA group: 11327020333032 Retrieved Current HA Status
```

HA Troubleshooting

If you encounter problems with an HA group, refer to this section.

Cryptographic Operations Blocked During Remote PED Operations When Audit Logging Is Enabled

With audit logging enabled on the HSM, crypto operations are blocked on all application partitions during Remote PED operations. During this time, requests sent to HA member partitions on this HSM will not fail over to other members. When the Remote PED operation is complete, all crypto operations resume normally. If your application has its own timeout programmed, it may incorrectly conclude that the entire HA group has failed.

Using Luna HSM Client 10.7.2 or newer, you can configure the "ProbeTimeout" on page 89 setting in the **Chrystoki.conf/crystoki.ini** file to trigger an HA failover after a specified time. This allows operations to continue normally during Remote PED operations.

Administration Tasks on HA Groups

Do not attempt to run administrative tasks on an HA group virtual slot (such as altering partition policies). These virtual slots are intended for cryptographic operations only. It is not possible to use an HA group to make administrative changes to all partitions in the group simultaneously; the exception is lunacm:> partition changepw using Luna HSM Client 10.7.0 or neewer.

Unique Object IDs (OUID)

If two applications using the same HA group modify the same object using different members, the object fingerprint might conflict.

Client-Side Limitations

New features or abilities, or new cryptographic mechanisms added by firmware update, or previously usable mechanisms that become restricted for security reasons, can have an impact on the working of an HA group, when the Luna HSM Clientversion is older. Luna Clients are "universal" in the sense that they are able to work fully with current Luna HSMs/partitions, and with earlier versions, as well as with cloud crypto solutions (DPoD Luna Cloud HSM service), but a client version cannot be aware of HSM versions that were not yet developed when the Client was released.

Client-Side Failures

Any failure of the client (such as operating system problems) that does not involve corruption or removal of files, should resolve itself when the client is rebooted.

If the client workstation seems to be working fine otherwise, but you have lost visibility of the HSMs in LunaCM or your client, try the following remedies:

- verify that the Thales drivers are running, and retry
- reboot the client workstation
- restore your client configuration from backup
- re-install Luna HSM Client and re-configure the HA group

Failures Between the HSM Appliance and Client

The only failure that could likely occur between a Luna Network HSM 7 (or multiple HSMs) and a client computer coordinating an HA group is a network failure. In that case, the salient factor is whether the failure occurred near the client or near one (or more) of the Luna Network HSM 7 appliances.

If the failure occurs near the client, and you have not set up port bonding on the client, then the client would lose sight of all HA group members, and the application fails. The application resumes according to its timeouts and error-handling capabilities, and HA resumes automatically if the members reappear within the recovery window that you had set.

If the failure occurs near a Luna Network HSM 7 member of the HA group, then that member disappears from the group until the network failure is cleared, but the client can still see other members, and normal failover occurs.

Avoid direct access to individual HA group members when securing with STC

This is best ensured by having `HAonly` setting turned ON, in the configuration file, so that only the HA virtual slot is visible and all requests and responses are handled transparently by the HA system (see "Configuration File Summary" on page 76). If you cannot avoid directly accessing an individual HA member slot, then be sure to *log out of it before your application attempts to use the HA virtual slot*. This is especially important when STC is invoked (see "Client-Partition Connections" on page 107).

Each HSM keeps track of any appid registered against a remote connection, and rejects any attempt to create a new session with different appID from the same client. That is, only one access ID is permitted per STC channel. If a client opens a session directly to an individual HA member partition, then an ID is assigned. If the client next attempts operation via the HA virtual slot, then as part of that process, random appids are assigned to each member partition for the open channel, but one of those member partitions already has the earlier ID, so the HSM responds with CKR_ACCESS_ID_ALREADY_EXISTS and the operation fails.

Log out of any individual member slot, before invoking the HA slot, to avoid this problem.

Some security settings and implications

> **TIP** **Security Note -***Cloning policies (**0** and **4**)* permit or deny the ability to securely copy keys and objects into and out of a partition.
> The *Key Management Functions policy (**28**)* controls the ability to create, delete, generate, derive, or modify cryptographic objects in the current partition.
>
> These controls are independent of each other. With Key Management functions denied, you can still clone objects in and out of partitions where Cloning policy is allowed. Thus HA (high availability) operation can clone keys into a partition that disallows Key Management functions (creation, deletion, etc.). **Cloning a key or object into a partition is not considered creation** - the key or object already existed within the security / cloning domain that encompasses the partition.
>
> Ultimately the security administrators define where keys can exist by controlling distribution of the security / cloning domain, and by defining policies around those keys.
>
> Additionally, key owners can choose to make their keys non-modifiable and non-extractable, if those options are indicated by your use-case.

Guidelines and Recommendations For Updating or Converting HA Member Partitions

This section lists some general guidelines for users that are updating High Availability (HA) member partitions to a new firmware version or converting them to V0 or V1. Refer to the following relevant topics before reading the information in this section:

- Updating the Luna HSM Firmware
- Special Considerations for Luna HSM Firmware 7.7.0 and Newer
- Behavior of "V0 and V1 Partitions" on page 148
- "Converting Partitions from V0 to V1 or V1 to V0" on page 161

Adhere to the following guidelines and recommendations when you are updating or converting HA member partitions:

- Before beginning the update or conversion process, remove the partition from the HA group and stop all ongoing operations with HA.
- Update or convert members one at a time, leaving the primary member for last.

> **NOTE** You must update/convert secondary partitions first and the primary partition last. If you do not adhere to this guideline, you may experience issues while updating/converting.

- Resume HA group operation after *all* members have been updated or converted.
- If HA operation *cannot* be stopped for an update or conversion, you must be aware of the following constraints and guidelines:
  - Expect HA functionality (with some caveats) when members have been updated, but not *during* firmware update to Luna HSM Firmware 7.7.0 (or newer) or *during* conversion of member partitions from V0 to V1.

- Thales does not recommend operating HA groups with mixed firmware and partition versions because the HA group will not work as expected. For proper HA functionality, all members of a working HA group should have the following properties:
  - Identical firmware versions. As a result, firmware updates should take place while the partition is *not* a member of an HA group.
  - Identical partition types; that is, either all partitions are V0 or all partitions are V1. As a result, partition conversions to V0 (which happens as a result of updating from pre-7.7.0 firmware to Luna HSM Firmware 7.7.0 or newer) or V1 should take place while the partition is *not* a member of an HA group.
  - Identical history of functionality module (FM) deployment; that is, either all members are FM-enabled or all members are FM-never-enabled, but not some of each.
- Converting a partition from V1 to V0 is destructive, so the partition *cannot* remain an HA member.
- Cloning of keys and objects can proceed only from a lower-security environment to a higher-security environment but not in the other direction; that is, cloning of keys and objects can proceed in the following directions:
  - Older firmware to newer firmware.
  - Pre-7.7.0 partitions to V0 partitions.
  - V0 partitions to V1 partitions.
  - Partitions of an HSM that has had FMs enabled to partitions that have never had functionality modules enabled.

  The above constraint has implications for the update and conversion process; that is, the primary partition must be at the same firmware and partition version as the remaining HA member partitions, or lower. If this requirement is not met, attempts to synchronize the HA group will fail. For example, V1 partition can be added to an existing HA group that already has HA members made up of partitions from a pre-7.7.0-firmware HSM. However, when V1 partition becomes the primary member of the HA group, synchronization with remaining member of the HA group will no longer function.
- Avoid direct access to individual HA group members when securing with STC. For more information, refer to "Avoid direct access to individual HA group members when securing with STC" on page 464.

> For best results, the client library that enables HA among several HSMs should be up-to-date.

CHAPTER 14: Partition Backup and Restore

Luna Network HSM 7 allows secure creation, storage, and use of cryptographic data (keys and other objects). It is critically important to safeguard your important cryptographic objects against unforeseen damage or data loss. No device can offer total assurance against equipment failure, physical damage, or human error. Therefore, a comprehensive strategy for making regular backups is essential. There are multiple ways to perform these operations, depending on your implementation.

This section contains the following information:

Luna Network HSM 7 can perform backup and restore operations using the legacy "Luna Backup HSM G5" on page 538, the updated "Luna Backup HSM 7" on page 474, or a "Backup to Luna Cloud HSM" on page 473 service. Refer to the section describing the variant you wish to use:

Key Concepts for Backup and Restore Operations

A Crypto Officer (CO) can use the backup HSM to back up and restore the objects in any partition they can log in to, provided that:

> The application partition and the backup HSM partition share the same domain.
> The application partition and the backup HSM use the same authentication method (multifactor quorum or password).
> The CO has the required credentials on the backup HSM.

You can perform backup/restore operations on your application partitions by connecting the backup HSM to the Luna HSM Client workstation. When you connect the backup HSM to a Luna HSM Client workstation, the backup HSM Admin partition is added to the slots listed in LunaCM, allowing you to clone objects between the source application partition and the target backup partition.

> **NOTE** To perform backup operations on Luna HSM Firmware 7.7.0 or newer (V0 or V1 partitions) you require at minimum:
>
> > Luna Backup HSM 7 Firmware 7.7.1
>
> > Luna Backup HSM G5 Firmware 6.28.0
>
> You can use a Luna Backup HSM with older firmware to restore objects to a V0 or V1 partition, but this is supported for purposes of getting your objects from the older partitions onto the newer V0 or V1 partitions only. V0 and V1 partitions are considered more secure than partitions at earlier firmware versions - any attempt to restore from a higher-security status to lower-security status fails gracefully.
>
> When the Luna Backup HSM is connected directly to the Luna Network HSM 7 appliance, only the SMK can be backed up from or restored to a V1 partition.

Backups are created and stored as partitions within the Admin partition on the backup HSM.

Credentials Required to Perform Backup and Restore Operations

You require the following credentials to perform backup/restore operations:

Luna Network HSM 7	**Remote PED (orange) PED key**. Required for multifactor quorum-authenticated backups only, using a local or remote Luna Backup HSM 7 v1, or a remote Luna Backup HSM G5 or Luna Backup HSM 7 v2.
Source Luna Network HSM 7 partition	**Crypto Officer (CO)**. Required to access the objects in the source application partition that will be backed up. **Domain**. Required to allow objects to be cloned between the sourceapplication partition and target backup partition. The domains for the source application partition and target backup partition must match, otherwise the backup will fail.
Target Backup HSM	**HSM Security Officer (SO)**. Required to create or access the target backup partition in the Admin slot, where all backups are archived. **Remote PED (orange) PED key**. Required for multifactor quorum-authenticated backups only, using a local or remote Luna Backup HSM 7 v1, or a remote Luna Backup HSM 7 v2 or Luna Backup HSM G5, to establish a remote PED connection to the HSM that hosts the target backup partition. **Note:** You create new credentials for both roles on HSM initialization, and use them for subsequent backups to the target backup HSM.
Target Backup Partition	**Partition Security Officer (PO)**. Required to access the target backup partition on a Luna Backup HSM 7. **Crypto Officer (CO)**. Required to access the objects in the target backup partition. **Note:** You create new credentials on the initial backup, and use them for subsequent backups to the target backup partition.

Client Software Required to Perform Backup and Restore Operations

You must install the Luna HSM Client software and USB driver for the backup HSM on the workstation you intend to use to perform backup and restore operations. The Luna Backup HSM 7 v1 requires minimum Luna HSM Client 10.1.0. The Luna Backup HSM 7 v2 requires minimum Luna HSM Client 10.4.0. Refer to "Luna HSM Client Software Installation" on page 20.

> **NOTE** Ensure that the backup HSM is not connected to the Luna HSM Client workstation when you install or uninstall the client software. Failure to do so may result in the backup HSM becoming unresponsive.

When you install the client software, you must select the following options:

- The **Backup** option. This installs the driver for the backup HSM and components required for the Remote Backup Service (RBS).
- The **USB** option. This installs the driver for the backup HSM.
- The **Network** and/or **PCIe** options, depending on which type of HSM you intend to back up.
- The **Remote PED** option, if you want to back up multifactor quorum-authenticated partitions. Note that you can install and use a remote PED on the same workstation used to host the backup HSM, or on a different workstation. This option is mandatory for the Luna Backup HSM 7 v1, but a local PED connection can be used for the Luna Backup HSM 7 v2 or Luna Backup HSM G5.

Multifactor Quorum Authentication with Luna Backup HSM 7 v1

The Luna Backup HSM 7 v1 is equipped with a single USB port that is used to connect the backup HSM to a Luna HSM Client workstation or Luna Network HSM 7 appliance. As such, any PED connections to the backup HSM must use a remote PED and the **pedserver** service.

Planning Your Backup HSM Deployment

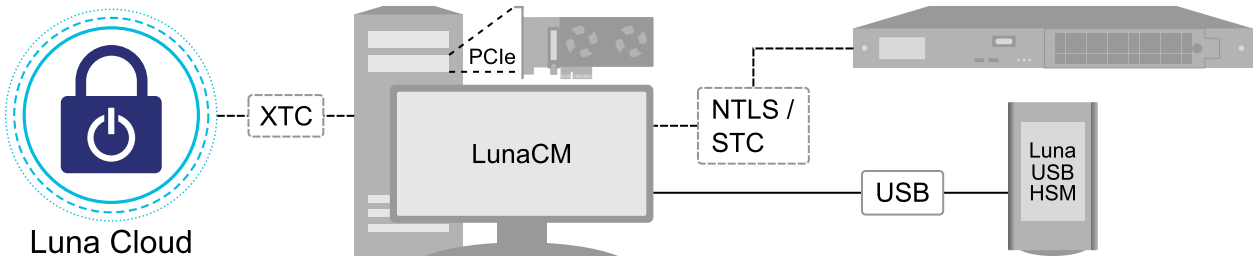
When setting up your backup deployment, you have multiple configuration options. This section will help you choose the right configuration, depending on where you prefer to keep your backups. You can use a Luna Backup HSM, Luna Cloud HSM service, or an application partition on another Luna HSM for backup/restore operations.

Backup and restore operations require that cloning be enabled.

- "Partition to Partition" below
- "Backup to Luna Cloud HSM" on the next page
- "Backup HSM Connected to the Client Workstation" on the next page
- "Backup HSM Installed Using Remote Backup Service" on page 472

Partition to Partition

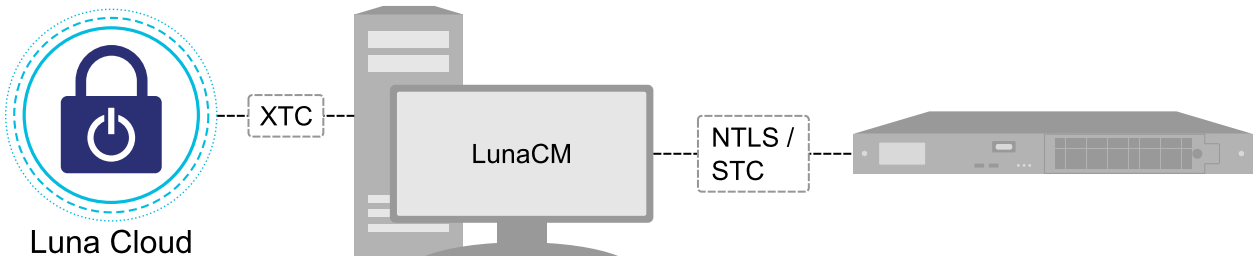
You can clone objects from any Luna 7 application partition to any other Luna 7 partition that shares its cloning domain. You must have the Crypto Officer credential for both partitions. Both partitions must use the same authentication method (either password or PED key).



See "Cloning Objects to Another Application Partition" on page 201.

Backup to Luna Cloud HSM

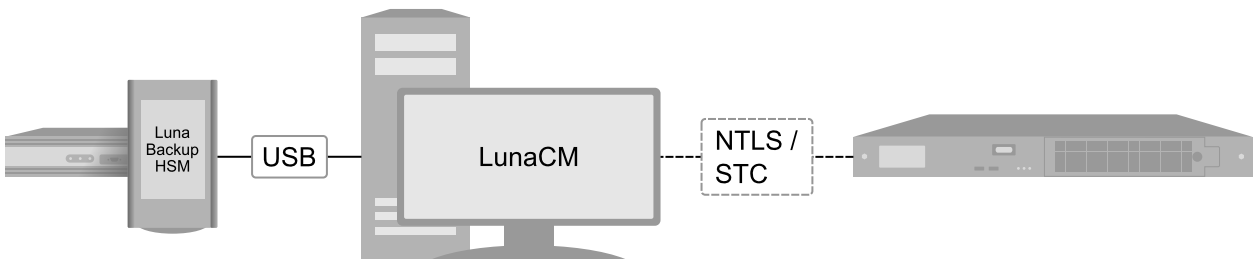
You can securely back up the contents of any password- or multifactor quorum-authenticated Luna 7 partition to a Luna Cloud HSM service.



See "Cloning Objects to Another Application Partition" on page 201.

Backup HSM Connected to the Client Workstation

In this configuration, the Luna Backup HSM is connected to a USB port on the client workstation. It is useful in deployments where the partition Crypto Officer keeps backups at the client. This allows you to perform backup/restore operations for all application partitions that appear as visible slots in LunaCM. You can restore a partition backup to the original source partition or to another existing Luna application partition that shares the same cloning domain.

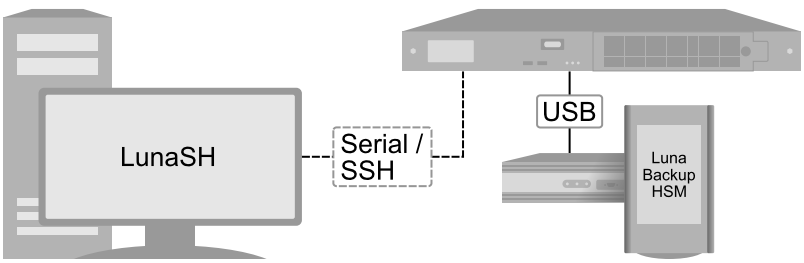


Depending on your Luna Backup HSM and Luna HSM Client version, refer to:

Hardware/Software Requirements	Available Procedures
> "Luna Backup HSM 7" on page 474 **v2** > Luna HSM Client 10.4.0 or newer	> "Luna Backup HSM 7 Connected to Luna HSM Client Using Direct Multifactor Quorum Authentication" on page 509 > "Luna Backup HSM 7 Connected to Luna HSM Client Using Remote Multifactor Quorum Authentication" on page 520 > "Luna Backup HSM 7 Connected to Luna HSM Client Using Password Authentication" on page 531
> "Luna Backup HSM 7" on page 474 **v1 or v2** > Luna HSM Client 10.1.0 or newer	> "Luna Backup HSM 7 Connected to Luna HSM Client Using Remote Multifactor Quorum Authentication" on page 520 > "Luna Backup HSM 7 Connected to Luna HSM Client Using Password Authentication" on page 531
> "Luna Backup HSM G5" on page 538	> "Backup/Restore Using Luna Backup HSM G5 Connected to Luna HSM Client" on page 557

Backup HSM Connected to the Luna Network HSM 7 Appliance

In this configuration, the Luna Backup HSM is connected to a USB port on the Luna Network HSM 7 appliance. It is useful in deployments where the partition Crypto Officer has **admin**-level access to LunaSH on the appliance. This allows you to perform backup/restore operations for all application partitions that appear in LunaSH using partition list. You can restore a partition backup to the original source partition or to another existing Luna application partition that shares the same cloning domain.



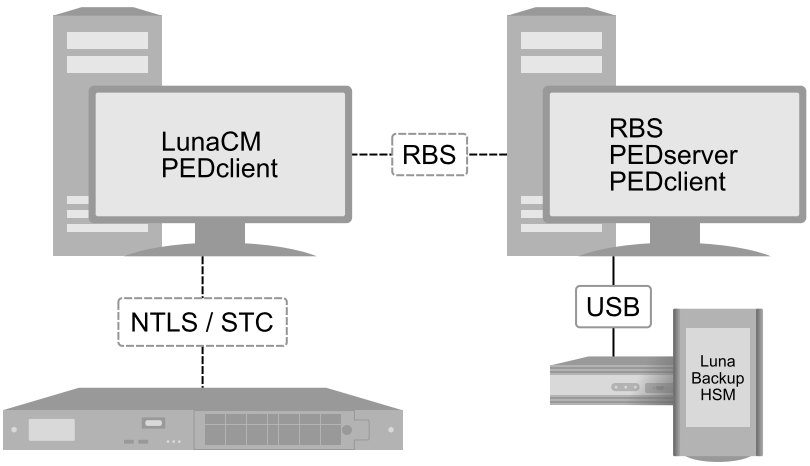
Depending on your Luna Backup HSM and Luna Appliance Software version, refer to:

Hardware/Software Requirements	Available Procedures
> "Luna Backup HSM 7" on page 474 **v2** > Luna Appliance Software 7.8.4 or newer	> "Luna Backup HSM 7 Connected to Luna Network HSM 7 Using Direct Multifactor Quorum Authentication" on page 483 > "Luna Backup HSM 7 Connected to Luna Network HSM 7 Using Luna PED for Multifactor Quorum Authentication" on page 492 > "Luna Backup HSM 7 Connected to Luna Network HSM 7 Using Password Authentication" on page 503

Hardware/Software Requirements	Available Procedures
> "Luna Backup HSM 7" on page 474 **v1 or v2** > Luna Appliance Software 7.7.0 or newer	> "Luna Backup HSM 7 Connected to Luna Network HSM 7 Using Luna PED for Multifactor Quorum Authentication" on page 492 > "Luna Backup HSM 7 Connected to Luna Network HSM 7 Using Password Authentication" on page 503
> "Luna Backup HSM G5" on page 538	> "Backup/Restore Using Luna Backup HSM G5 Connected to Luna Network HSM 7" on page 552

Backup HSM Installed Using Remote Backup Service

In this configuration, the Luna Backup HSM is connected to a remote client workstation that communicates with the Luna Network HSM 7 client via the Remote Backup Service (RBS). It is useful in deployments where backups are stored in a separate location from the Luna Network HSM 7, to mitigate the consequences of catastrophic loss (fire, flood, etc).



Refer to "Configuring a Remote Backup Server" on page 561.

Backup and Restore Best Practices

To ensure that your data is protected in the event of a failure or other catastrophic event, Thales recommends that you use the following best practices as part of a comprehensive backup strategy:

> **CAUTION!** Failure to develop and exercise a comprehensive backup and recovery plan may prevent you from being able to recover from a catastrophic event. Although Thales provides a robust set of backup hardware and utilities, we cannot guarantee the integrity of your backed-up key material, especially if stored for long periods. Thales strongly recommends that you exercise your recovery plan at least semi-annually (every six months) to ensure that you can fully recover your key material.

Develop and document a backup and recovery plan
This plan should include the following:

- What is being backed up
- The backup frequency
- Where the backups are stored
- Who is able to perform backup and restore operations
- Frequency of exercising the recovery test plan

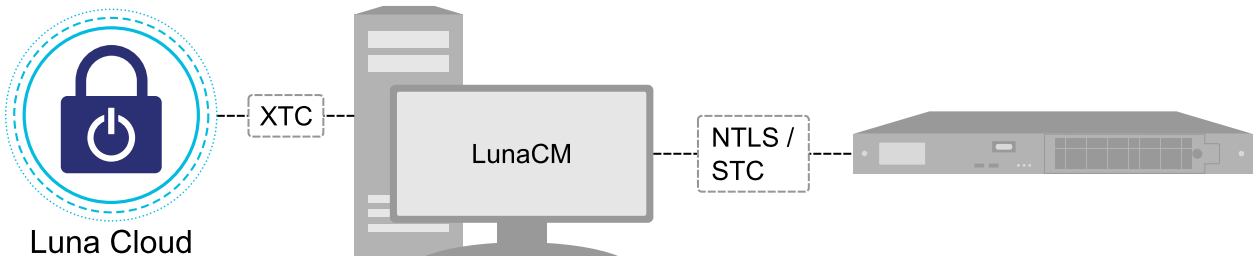
Make multiple backups
To ensure that your backups are always available, build redundancy into your backup procedures.

Use off-site storage
In the event of a local catastrophe, such as a flood or fire, you might lose both your working HSMs and locally-stored backup HSMs. To fully protect against such events, always store a copy of your backups at a remote location.

Regularly exercise your disaster recovery plan
Execute your recovery plan at least semi-annually (every six months) to ensure that you can fully recover your key material. This involves retrieving your stored Backup HSMs and restoring their contents to a test partition, to ensure that the data is intact and that your recovery plan works as documented.

Backup to Luna Cloud HSM

Luna Cloud HSM services allow you to back up your partition objects securely in the cloud, with no additional HSM hardware. With "Universal Cloning" on page 198 (includes Extended Domain Management and CPv4, and requires Luna HSM Firmware 7.8.0 or newer for the on-premises HSM and Luna HSM Client 10.5.0 or newer), Luna Cloud HSM can now clone keys from both password-and multifactor quorum-authenticated Luna HSMs. You can create Luna Cloud HSM backups using slot-to-slot cloning, or set up an HA group to synchronize your partition contents with Luna Cloud HSM.



Refer to the following sections to set up and use this backup method:

- "Adding a Luna Cloud HSM Service" on page 72 for instructions on adding and initializing Luna Cloud HSM for use with your deployment.
- "Cloning Objects to Another Application Partition" on page 201 for instructions on creating Luna Cloud HSM backups.

> "Setting Up an HA Group" on page 432 for instructions on setting up a synchronized Luna Cloud HSM service.

> "Cloning Keys Between Luna 6, Luna 7, and Luna Cloud HSM, Password or Multifactor Quorum" on page 210 for additional information about mixed-environment cloning.

Luna Backup HSM 7

The Luna Backup HSM 7 is a full-featured, hand-held, USB-attached backup HSM that includes an informational full-color display. The Luna Backup HSM 7 connects easily to a client workstation using the included USB 3.0 Type C cable, and includes a universal 5V external power supply, which may be required to power the device in some instances.

The refreshed v2 model includes a USB-C port, which, combined with a USB-A to USB-C adapter, allows you to insert PED keys directly into the HSM, greatly simplifying the multifactor quorum authentication procedure and, depending on your configuration, eliminating the need for a Luna PED in backup/restore operations.

The Luna Backup HSM 7 is available in the following models. All models can be initialized in multifactor quorum or password-authenticated mode for backing up either multifactor quorum or password authenticated partitions. In-field storage upgrades are not available.

B700	32 MB storage, up to 100 partitions of the same authentication type
B750	128 MB storage, up to 100 partitions of the same authentication type
B790	256 MB storage, up to 100 partitions of the same authentication type

For setup, management, and backup/restore procedures, refer to the following sections:

> "Luna Backup HSM 7 Hardware Installation" on the next page

> "Managing the Luna Backup HSM 7" on page 478

> "Configuring a Remote Backup Server" on page 561

Refer to the following procedures depending on your authentication method, Luna Backup HSM 7 hardware, and Luna Appliance Software or Luna HSM Client versions:

Multifactor Quorum Authentication

- "Luna Backup HSM 7 Connected to Luna Network HSM 7 Using Direct Multifactor Quorum Authentication" on page 483 (requires "Luna Backup HSM 7" on the previous page **v2** and Luna Appliance Software 7.8.4 or newer)
- "Luna Backup HSM 7 Connected to Luna Network HSM 7 Using Luna PED for Multifactor Quorum Authentication" on page 492 (requires Luna Appliance Software 7.7.0 or newer)
- "Luna Backup HSM 7 Connected to Luna HSM Client Using Direct Multifactor Quorum Authentication" on page 509 (requires "Luna Backup HSM 7" on the previous page **v2** and Luna HSM Client 10.4.0 or newer)
- "Luna Backup HSM 7 Connected to Luna HSM Client Using Remote Multifactor Quorum Authentication" on page 520 (requires Luna HSM Client 10.1.0 or newer)

Password Authentication

- "Luna Backup HSM 7 Connected to Luna Network HSM 7 Using Password Authentication" on page 503 (requires Luna Appliance Software 7.7.0 or newer)
- "Luna Backup HSM 7 Connected to Luna HSM Client Using Password Authentication" on page 531 (requires Luna HSM Client 10.1.0 or newer)

Luna Backup HSM 7 Hardware Installation

The following topics describe how to install and connect a Luna Backup HSM 7:

- "Luna Backup HSM 7 Required Items" below
- "Luna Backup HSM 7 Hardware Functions" on page 477
- "Installing the Luna Backup HSM 7 Hardware" on page 478

The Luna Backup HSM 7 complies with the following:

Luna Backup HSM 7 Required Items

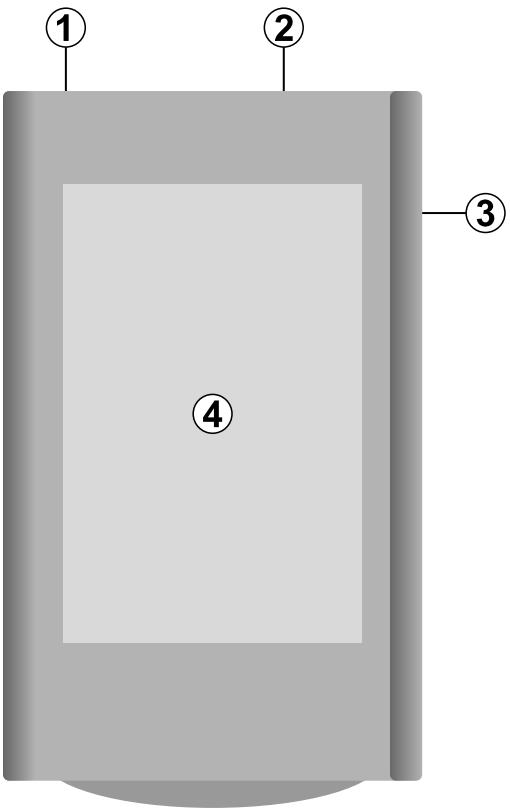
This section provides a list of the components you should have received with your Luna Backup HSM 7 order.

Qty	Item
1	**Luna Backup HSM 7 v1 or v2**  
1	**USB 3.0 Cable: Type A to Type C** 
1	**5V Power Supply** with replaceable plug modules for international use. 

Qty	Item
1	**USB-A to USB-C adapter (included with v2 only)**  Used to connect PED keys to the Luna Backup HSM 7 v2.

Luna Backup HSM 7 Hardware Functions

The Luna Backup HSM 7 hardware is illustrated below, with important features labeled.



1	5V power supply connector. Required only if the USB port connected to (**2**) does not supply adequate power to the Luna Backup HSM 7.
2	USB-C connector. Used for USB power and connecting to the client computer or Luna Network HSM 7.
3	USB-C connector (v2 only). Used for connecting PED keys to authenticate roles on the HSM. Requires the included USB-A to USB-C adapter.
4	LED touchscreen. Displays information about the Luna Backup HSM 7 and is used to input role-specific information like PINs.

The Luna Backup HSM 7 does not contain an internal battery, and maintains the integrity of its stored key material without being connected to power.

Installing the Luna Backup HSM 7 Hardware

The backup HSM is a USB device. To install the backup HSM, connect it to a USB port on a Luna HSM Client workstation or Luna Network HSM 7 appliance using the included USB cable. The Luna Network HSM 7 USB connection provides adequate power, and connecting the provided power supply is not recommended. The workstation must be running Luna HSM Client software that supports the backup HSM and provides the required drivers.

> **NOTE** On most Luna HSM Client computers, the USB port provides adequate power, and connecting the provided power supply is not recommended. Wait and confirm that the HSM boots properly. If the HSM fails to boot up:
>
> **1.** Disconnect the HSM from the USB port.
>
> **2.** Connect the HSM to power using the provided power supply. Wait for it to boot completely.
>
> **3.** Reconnect the HSM to the USB port on the client.
>
> If the HSM is connected to the USB port before the power supply, you may encounter an issue where the HSM occasionally loses contact with the client, and must be power cycled.

> **NOTE** If you are installing Luna HSM Client on Windows, the driver may not be installed unless the Luna device is connected to the computer first; refer to Backup/USB/PCIe Drivers Not Installed on Windows 10 or Windows Server 2022 Unless Device is Connected.

Managing the Luna Backup HSM 7

This section contains the following procedures for maintaining and using the Luna Backup HSM 7:

> "Recovering the Luna Backup HSM 7 from Secure Transport Mode" below
>
> "Configuring the Luna Backup HSM 7 for FIPS Compliance" on the next page
>
> "Updating the Luna Backup HSM 7 Firmware" on page 480
>
> "Rolling Back the Luna Backup HSM 7 Firmware" on page 482

Recovering the Luna Backup HSM 7 from Secure Transport Mode

The Luna Backup HSM 7 is shipped in Secure Transport Mode (STM). STM provides a logical check on the firmware and critical security parameters (such as configuration, keys, policies, roles, etc.) so that the authorized recipient can determine if these have been altered while the HSM was in transit.

> **NOTE** This procedure requires connection to a client machine with Luna HSM Client 10.1.0 or newer installed. This operation is not possible while the Backup HSM is connected to the Luna Network HSM 7 appliance.

To recover the Luna Backup HSM 7 from STM

1. Connect the Luna Backup HSM 7 to a USB port on a Luna HSM Client workstation with the **Backup** option installed (refer to "Luna HSM Client Software Installation" on page 20 for your client operating system).
2. Launch LunaCM on the client workstation.
3. Select the slot assigned to the Luna Backup HSM 7 Admin partition.

   lunacm:> **slot set -slot** <slot_id>
4. Recover the HSM from Secure Transport Mode. See Secure Transport Mode for more information about the Random User String:

   lunacm:> **stm recover -randomuserstring** <string>

> **NOTE** Recovering a Luna Backup HSM 7 from STM may take up to three minutes.

Configuring the Luna Backup HSM 7 for FIPS Compliance

Luna Backup HSM Firmware 7.7.1 and newer uses the same updated cloning protocol as Luna HSM Firmware 7.7.0 and newer. For the Luna Backup HSM 7 to be FIPS-compliant, it must restrict restore operations to application partitions that use the new protocol. This restriction is applied by setting **HSM policy 55:** Enable Restricted Restore to **1** on the backup HSM. The Luna Backup HSM 7 must be initialized and connected to a Luna HSM Client computer to set this policy.

When this policy is enabled on the Luna Backup HSM 7, objects that have been backed up from partitions using firmware older than Luna HSM Firmware 7.7.0 can be restored to Luna HSM Firmware 7.7.0 or newer (V0 or V1) partitions only.

> **CAUTION!** FIPS compliance requires that objects are never cloned or restored to an HSM using less secure firmware, and this includes restoring from Luna Backup HSM 7 firmware.
>
> If you have backups already stored on the Luna Backup HSM 7 that were taken from pre-7.7.0 partitions, turning this policy ON will prevent you from restoring them to the same source partition. You must update the HSM containing the source partition to Luna HSM Firmware 7.7.0 or newer before restoring from backup.

> **NOTE** **HSM policy 12:** Allow non-FIPS algorithms, which is used to set FIPS-compliant mode on other Luna HSMs, does not apply to the Luna Backup HSM 7. Attempts to change this policy will fail with the error `CKR_CANCEL`.

To configure the Luna Backup HSM 7 for FIPS compliance

1. On the Luna HSM Client computer, run LunaCM.
2. Set the active slot to the Luna Backup HSM 7.

   lunacm:> **slot set -slot** <slot_id>
3. Log in as Backup HSM SO.

   lunacm:> **role login -name so**
4. Set **HSM policy 55:** Enable Restricted Restore to **1**.

lunacm:> **hsm changehsmpolicy -policy 55 -value 1**

5. [Optional] Check that the Luna Backup HSM 7 is now in FIPS approved operation mode.

   lunacm:> **hsm showinfo**

   ```
   *** The HSM is in FIPS 140-2 approved operation mode. ***
   ```

Updating the Luna Backup HSM 7 Firmware

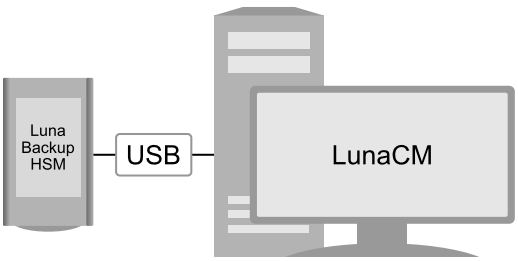
To update the Luna Backup HSM 7, download the desired firmware version from the Thales Support Portal. If you are updating a Backup HSM connected to a Luna Network HSM 7 appliance, the firmware update file is included in the appliance software update package. See Updating the Luna Network HSM 7 Appliance Software for the procedure. Depending on whether the Backup HSM is connected to a Luna Network HSM 7 appliance or a Luna HSM Client workstation, you can use LunaSH or LunaCM to perform the firmware update.

> "Updating the Client-Connected Luna Backup HSM 7 Firmware" below

> "Updating the Appliance-Connected Luna Backup HSM 7 Firmware" on the next page

Updating the Client-Connected Luna Backup HSM 7 Firmware

Use the following procedure to update the Luna Backup HSM 7 firmware using LunaCM. The Backup HSM SO must complete this procedure.



> **NOTE** This functionality requires minimum Luna HSM Client 10.3.0.

Prerequisites

> Luna Backup HSM 7 firmware update file (<filename>**.fuf**)

> firmware update authentication code file (<filename>**.txt**)

> If you have backups currently stored on the Backup HSM, they must take up less than 60% of storage capacity, or the firmware upgrade will not proceed.

> **NOTE** If you are updating from Luna Backup HSM 7 Firmware 7.3.2, objects and partitions must be re-sized to include additional object overhead associated with the new V1 partitions - this action is automatic (see "V0 and V1 Partitions" on page 148). This conversion can take a long times, depending on the number of objects stored on the Backup HSM (a few minutes to several hours). Ensure that you can leave the update operation uninterrupted for this amount of time. Do not interrupt the procedure even if the operation appears to have stalled.

To update the Luna Backup HSM 7 firmware using LunaCM

1. Copy the firmware file (<filename>**.fuf**) and the authentication code file (<filename>**.txt**) to the Luna HSM Client root directory.
   - Windows: C:\Program Files\SafeNet\LunaClient
   - Linux: /usr/safenet/lunaclient/bin
   - Solaris: /opt/safenet/lunaclient/bin

   > **NOTE** On some Windows configurations, you might not have authority to copy or unzip files directly into **C:\Program Files\...**. If this is the case, put the files in a known location that you can reference in a LunaCM command.

2. Launch LunaCM.
3. If more than one HSM is installed, set the active slot to the Admin partition of the HSM you wish to update.

   lunacm:> **slot set -slot** <slot_number>
4. [Multifactor Quorum-Authenticated]
   - If you are updating a Luna Backup HSM 7 v2, you will insert PED keys directly into the Backup HSM; skip to step 5.
   - If you are updating a Luna Backup HSM 7 v1, connect to the Remote PED server.

     lunacm:> **ped connect** [**-ip** <IP_address>] [**-port** <port#>]
5. Log in as HSM SO.

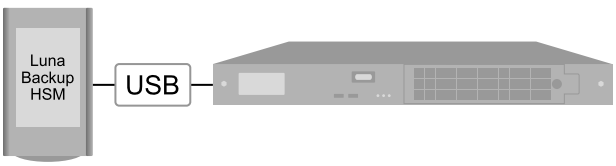
   lunacm:> **role login -name so**
6. Apply the new firmware update by specifying the update file and the authentication code file. If the files are not located in the Luna HSM Client root directory, specify the full filepaths.

   lunacm:> **hsm updatefw -fuf** <filename>**.fuf -authcode** <filename>**.txt**

The previous version of the firmware is stored in reserve on the HSM. To restore the previous firmware version, see "Rolling Back the Luna Backup HSM 7 Firmware" on the next page.

Updating the Appliance-Connected Luna Backup HSM 7 Firmware

Use the following procedure to update the Luna Backup HSM 7 firmware using LunaSH to the latest version that comes packaged with the appliance software. To install a different version, you must download the firmware update file (.fuf) from the Thales Support Portal and install it using LunaCM at the client (see "Updating the Client-Connected Luna Backup HSM 7 Firmware" on the previous page). The Backup HSM SO must complete this procedure.



> **NOTE** The Luna Network HSM 7 appliance software update includes the latest version of the Luna Backup HSM 7 firmware. Refer to the Customer Release Notes page for your appliance software version to see which Luna Backup HSM 7 is available for upgrade. This procedure requires Luna Appliance Software 7.7.0 or newer on the Luna Network HSM 7.

Prerequisites

> If you have backups currently stored on the Backup HSM, they must take up less than 60% of storage capacity, or the firmware upgrade will not proceed.

> **NOTE** If you are updating from Luna Backup HSM 7 Firmware 7.3.2, objects and partitions must be re-sized to include additional object overhead associated with the new V1 partitions - this action is automatic (see "V0 and V1 Partitions" on page 148). This conversion can take a long times, depending on the number of objects stored on the Backup HSM (a few minutes to several hours). Ensure that you can leave the update operation uninterrupted for this amount of time. Do not interrupt the procedure even if the operation appears to have stalled.

To update the Luna Backup HSM 7 firmware using LunaSH

1. Using a serial or SSH connection, log in to the appliance as **admin** (see Logging In to LunaSH).
2. [Optional] List the available Backup HSMs connected to the appliance and note the serial number of the one you wish to update.

   lunash:> **token backup list**
3. [Multifactor Quorum-Authenticated]
   - If you are updating a Luna Backup HSM 7 v2 and have Luna Appliance Software 7.8.4 or newer installed on the Luna Network HSM 7, you can insert PED keys directly into the Backup HSM; skip to step 4.
   - If you are updating a Luna Backup HSM 7 v1, or have Luna Appliance Software 7.8.3 or older installed on the Luna Network HSM 7, connect to the Remote PED server.

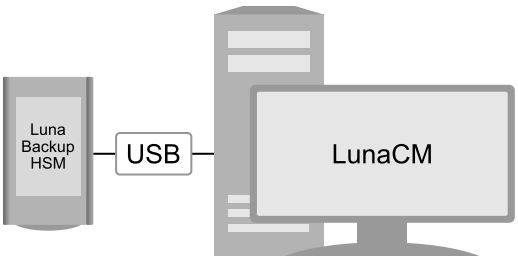
     lunacm:> **hsm ped connect** [**-ip** <IP_address>] [**-port** <port#>]
4. Log in to the Backup HSM as HSM SO.

   lunash:> **token backup login -serial** <serialnum>
5. Apply the Backup HSM firmware update.

   lunash:> **token backup update firmware -serial** <serialnum>

Rolling Back the Luna Backup HSM 7 Firmware

When you update the Luna Backup HSM 7 firmware, the previous version of the firmware is stored in reserve on the HSM. If required, you can use the following procedure to roll back the HSM firmware to the previous version. Firmware rollback must be initiated using LunaCM; the Backup HSM must be connected to a Luna HSM Client workstation.



CAUTION! Firmware rollback is destructive; earlier firmware versions might have older mechanisms and security vulnerabilities that a new version does not. Ensure that you do not have any important backups stored on the HSM before you proceed. This procedure zeroizes the HSM and all backups are erased.

Prerequisites

> Connect theLuna Backup HSM 7 to a Luna HSM Client workstation.

To roll back the Luna Backup HSM 7 firmware to the previous version

1. At the LunaCM prompt, set the active slot to the Backup HSM.

   lunacm:> **slot set -slot** <slot_number>

2. Check the previous firmware version that is available on the HSM.

   lunacm:> **hsm showinfo**

3. [Multifactor Quorum-Authenticated]
   - If you are rolling back a Luna Backup HSM 7 v2, you can insert PED keys directly into the Backup HSM; skip to step 5.
   - If you are rolling back a Luna Backup HSM 7 v1, connect to the Remote PED server.

     lunacm:> **ped connect** [**-ip** <IP_address>] [**-port** <port#>]

4. Log in as HSM SO.

   lunacm:> **role login -name so**

5. Roll back the Backup HSM firmware.

   lunacm:> **hsm rollbackfw**

Luna Backup HSM 7 Connected to Luna Network HSM 7 Using Direct Multifactor Quorum Authentication

In this configuration, you connect the Luna Backup HSM 7 to a USB port on the Luna Network HSM 7 appliance, and insert PED keys directly into the Luna Backup HSM 7. This allows you to perform backup/restore operations for all application partitions on that HSM. You can restore a partition backup to the original source partition or to another existing Luna application partition that shares the same cloning domain. To use this method, you require:

> "Luna Backup HSM 7" on page 474 **v2**

> Luna Appliance Software 7.8.4 or newer installed on the Luna Network HSM 7

NOTE

> **The Luna Backup HSM 7 is shipped in** Secure Transport Mode, and must be recovered from STM before first use. STM recovery requires LunaCM on a Luna HSM Client.
>
> See "Recovering the Luna Backup HSM 7 from Secure Transport Mode" on page 478.

> **If you require the Luna Backup HSM 7 to be FIPS-compliant**, you must complete an additional configuration step after initialization that requires LunaCM on a Luna HSM Client computer (see "Configuring the Luna Backup HSM 7 for FIPS Compliance" on page 479). Therefore, it may be simpler to initialize the Luna Backup HSM 7 at the client instead of using the initialization procedure below.

> **If you are backing up or restoring encrypted blobs stored on a V1 partition**, the Backup HSM must be connected to the client. Only the SMK can be backed up/restored using an appliance-connected Backup HSM.

> **If** "Secure Trusted Channel" on page 110 **is enabled on the partition**, the Backup HSM must be connected to the client.
>
> See "Luna Backup HSM 7 Connected to Luna HSM Client Using Direct Multifactor Quorum Authentication" on page 509.

This section provides instructions for the following procedures:

> "Initializing the Luna Backup HSM 7 for Multifactor Quorum Authentication" below

> "Backing Up a Multifactor Quorum-Authenticated Partition" on page 486

> "Restoring a Multifactor Quorum-Authenticated Partition From Backup" on page 489

Initializing the Luna Backup HSM 7 for Multifactor Quorum Authentication

You must initialize the Luna Backup HSM 7 prior to first use. You can initialize the backup HSM by connecting it to a Luna Network HSM 7 and using LunaSH commands to perform the initialization.

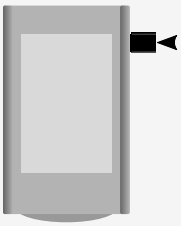
Prerequisites

> If necessary, recover the Luna Backup HSM 7 from Secure Transport Mode (see "Recovering the Luna Backup HSM 7 from Secure Transport Mode" on page 478).

> Ensure that you are familiar with the concepts in "Multifactor Quorum Authentication" on page 236. You will need the following PED keys:

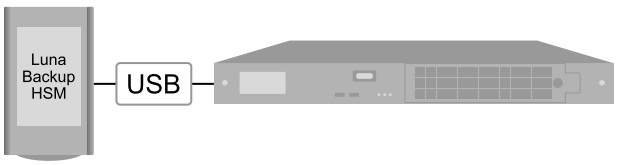
- N number of HSM SO (blue) PED keys, as defined by the M of N scheme you choose for the HSM SO role, plus the number required to create duplicate PED keys as necessary.
- Blank or reused Domain (red) PED key(s).

NOTE Whenever the Luna Backup HSM 7 prompts you to insert an PED key, use the USB-C adapter in the USB port on the right side of the Luna Backup HSM 7:



To initialize the Luna Backup HSM 7 for multifactor quorum authentication

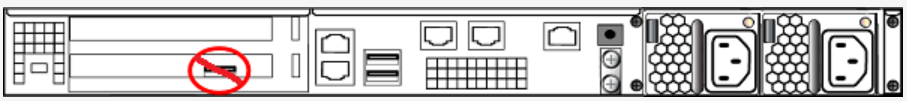
1. Connect your Luna Backup HSM 7 to a USB port on the Luna Network HSM 7:



   a. Open a network (SSH) or serial connection to the appliance and log in as **admin**, or other **admin**-level user, to start a LunaSH session.

   b. Connect the backup HSM directly to one of the USB ports on the Luna Network HSM 7 appliance using the included USB cable.

   NOTE The Luna Backup HSM 7 must be connected to one of the appliance USB ports, and not the one on the HSM card:



   The Luna Network HSM 7 USB connection provides adequate power, and connecting the provided power supply is not recommended.

2. Get the serial number of the backup HSM, or read the serial number from the Backup HSM display screen.

   lunash:> **token backup list**

3. Initialize the backup HSM:

   lunash:> **token backup init -label** <backup_hsm_label> **-serial** <backup_hsm_serial_number>

   You are prompted on the Luna Backup HSM 7 touchscreen to insert the blue HSM SO key(s) and red Domain key(s). Respond to the prompts and set the PINs on the required keys when requested. Ensure that you label any new PED keys that you create during this process.

   NOTE If your organization requires FIPS compliance, there is an additional procedure you must complete before using the Luna Backup HSM 7 to back up partitions. Refer to "Configuring the Luna Backup HSM 7 for FIPS Compliance" on page 479.

Backing Up a Multifactor Quorum-Authenticated Partition

Backups are created and stored as partitions within the Admin partition on the Luna Backup HSM 7. A new backup partition is created on initial backup. For subsequent backups, you can choose to replace the contents of the existing backup partition with the current source partition objects, or add new objects in the source partition to the existing backup partition. Like all cloning operations, the source and target backup partitions must be initialized with the same domain.

In addition to the credentials listed in "Credentials Required to Perform Backup and Restore Operations" on page 468, the Crypto Officer requires **admin**-level access to the appliance to access the LunaSH **partition backup** and **partition restore** commands (see Appliance Users and Roles).

Prerequisites

Before you begin, ensure that you have satisfied the following prerequisites:

> You are able to log in to the Luna Network HSM 7 using an **admin**-level account to access LunaSH.

> You have the required credentials:

If you are creating a new backup partition:

- New or reused Partition SO (blue) PED key(s) to initialize the backup partition
- New or reused Crypto Officer (black) PED key(s) to initialize the CO role on the backup partition
- The Domain (red) PED key(s) for the source partition, to initialize the domain on the backup

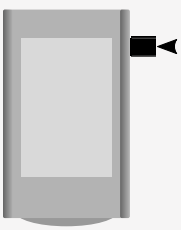
If you are backing up to an existing backup partition whose domain matches the source partition:

- The existing Partition SO (blue) PED key(s) for the backup partition, to log in
- The existing Crypto Officer (black) PED key(s) for the backup partition

> **TIP** If the source partition is activated, only the source partition Crypto Officer's challenge secret is required. To simplify the backup process and minimize interactions with the PED, it is recommended that you activate the CO role on the user partitions you want to back up. See "Activation on Multifactor Quorum-Authenticated Partitions" on page 373 for more information.

If the source partition is not activated, you also need:

- [Remote PED authentication] The Remote PED Vector (orange) PED key(s) for the source HSM
- The Crypto Officer (black) PED key(s) for the source partition

> **NOTE** Whenever the Luna Backup HSM 7 prompts you to insert an PED key, use the USB-C adapter in the USB port on the right side of the Luna Backup HSM 7:
>
> 

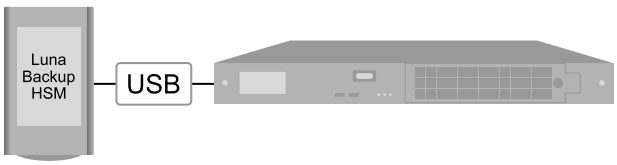
> The following policies are set:

- **HSM policy 16:** Allow network replication must be set to **1** (ON) on the HSM that hosts the user partition.
- [V0 partitions or firmware older than Luna HSM Firmware 7.7.0] **Partition policy 0:** "Allow private key cloning" on page 338 is set to **1** (ON) on the user partition.
- [V0 partitions or firmware older than Luna HSM Firmware 7.7.0] **Partition policy 4:** "Allow secret key cloning" on page 340 is set to **1** (ON) on the user partition.

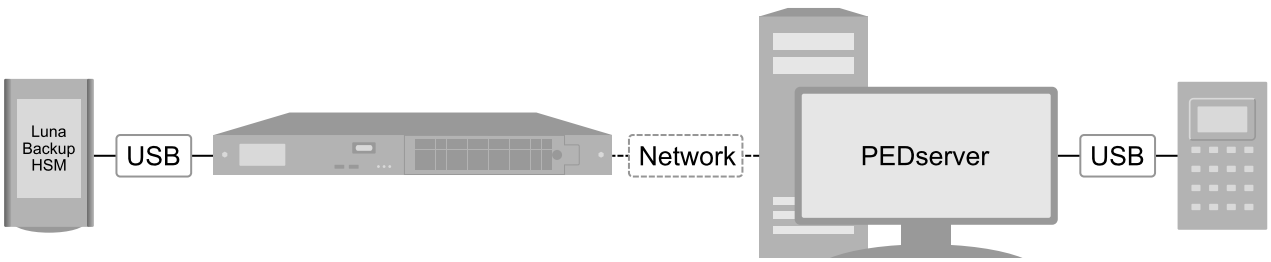
> **NOTE** HSS (PQC) private keys cannot be cloned, due to inherent restrictions of that key type. This means that backup and HA synchronization fail if HSS private keys are encountered in your application partition.

To back up a multifactor quorum-authenticated partition

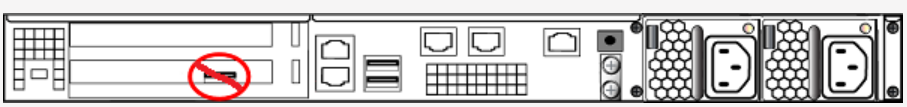
1. Configure your Luna Network HSM 7 appliance using one of the following configurations:
   - Activated source partition:



   - Non-activated source partition:

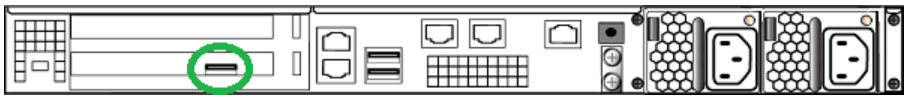


   a. Open a network (SSH) or serial connection to the appliance and log in as **admin**, or other admin-level user, to start a LunaSH session.
   b. Connect the backup HSM directly to one of the USB ports on the Luna Network HSM 7 appliance using the included USB cable.

> **NOTE** The Luna Backup HSM 7 must be connected to one of the appliance USB ports, and not the one on the HSM card:
>
> 
>
> The Luna Network HSM 7 USB connection provides adequate power, and connecting the provided power supply is not recommended.

c. [Non-activated source partition] Connect the Luna Network HSM 7 appliance to a Luna PED, using a local or remote connection:

   – [Local PED] Connect the Luna PED to the USB port on the HSM card:

   – [Remote PED] Connect the Luna Network HSM 7 appliance to the Remote PED server (see "Opening a Remote PED Connection" on page 265):

     lunash:> **hsm ped connect -ip** <remote_ped_host_ip_address>

2. Get the serial number of the backup HSM, or read the serial number from the backup HSM display screen.

   lunash:> **token backup list**

3. Display a list of application partitions; you require the label for the partition you are backing up.

   lunash:> **partition list**

4. If you plan to back up to an existing partition on the Backup HSM, display a list of the existing backups.

   lunash:> **token backup partition list -serial** <backup_hsm_serial_number>

5. Initiate the backup operation:

   lunash:> **partition backup -partition** <source_partition_label> **-serial** <backup_hsm_serial_number> [**-tokenpar** <target_backup_partition_label>] [**-add** | **-replace**]

   > **NOTE** You must specify **-add** or **-replace** when backing up to an existing backup partition. Use **-add** to add only new objects. Use **-replace** to erase the contents of the existing backup and replace them with the contents of the source partition. You do not need to specify these options when backing up a V1 partition, as only the SMK is backed up.
   >
   > If you omit the **-tokenpar** option when creating a new backup, the partition is assigned a default name (<source_partition_name>_<YYYYMMDD>) based on the source HSM's internally-set time and date.
   >
   > If the backup operation is interrupted (if the Backup HSM is unplugged, or if you fail to respond to prompts for PED keys, for example), the Backup HSM's full available space can become occupied with a single backup partition. If this occurs, delete the backup partition with lunash:> **token backup partition delete** before reattempting the backup operation.

6. Respond to the prompts on the Luna PED and/or Luna Backup HSM 7 touchscreen to insert the following keys in the following order:

   If the source partition is not activated:

   i. [Remote PED authentication] The Remote PED Vector (orange) PED key(s) for the source HSM

   ii. The Crypto Officer (black) PED key(s) for the source partition

   If you are creating a new backup partition:

   i. The HSM SO (blue) PED key(s) for the backup HSM, to log in

   ii. New or reused Partition SO (blue) PED key(s) to initialize the backup partition

   iii. The Partition SO (blue) PED key(s) you just created for the backup partition, to log in

iv. New or reused Crypto Officer (black) PED key(s) to initialize the CO role on the backup partition.

v. The Domain (red) PED key(s) for the source partition, to initialize the domain on the backup.

vi. The Crypto Officer (black) PED key(s) you just created for the backup partition, to log in

If you are backing up to an existing backup partition:

i. The HSM SO (blue) PED key(s) for the backup HSM, to log in

ii. The Crypto Officer (black) PED key(s) for the backup partition

The backup begins once you have completed the authentication process. Objects are backed up one at a time.

Restoring a Multifactor Quorum-Authenticated Partition From Backup

You can restore the objects from a multifactor quorum-authenticated backup partition to the same partition that was originally backed up, or to another partition that has been initialized with the same domain (red PED key).

Prerequisites

Before you begin, ensure that you have satisfied the following prerequisites:

> You are able to log in to the Luna Network HSM 7 using an **admin**-level account to access LunaSH.

> The target partition must be initialized using the same domain (red PED key) as the backup partition, the Crypto Officer role must be initialized and the CO role credential changed from its initial value.

> You have the required credentials:

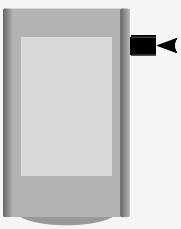
- The Crypto Officer challenge secret for the target partition
- The Crypto Officer (black) PED key(s) for the backup partition

> **TIP** If the target partition is activated, only the Crypto Officer's challenge secret is required. To simplify the backup process and minimize interactions with the PED, it is recommended that you activate the CO role on the user partitions you want to restore from backup. See "Activation on Multifactor Quorum-Authenticated Partitions" on page 373 for more information.

If the target partition is not activated, you also need:

- The Remote PED Vector (orange) PED key(s) for the target HSM
- The Crypto Officer (black) PED key(s) for the target partition

> **NOTE** Whenever the Luna Backup HSM 7 prompts you to insert an PED key, use the USB-C adapter in the USB port on the right side of the Luna Backup HSM 7:

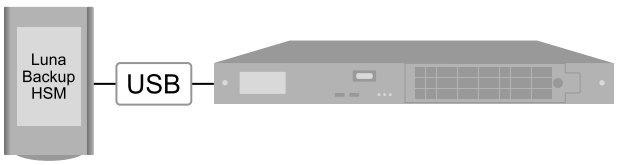


> The following policies are set:

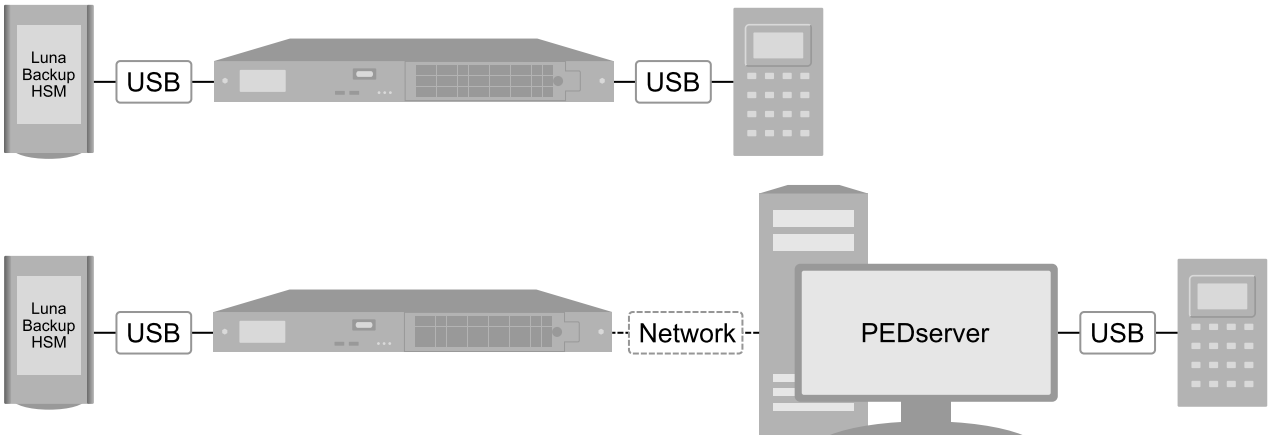
- **HSM policy 16:** Allow network replication must be set to **1** (ON) on the HSM that hosts the user partition you want to restore to.
- [V0 partitions or firmware older than Luna HSM Firmware 7.7.0] **Partition policy 0:** "Allow private key cloning" on page 338 is set to **1** (ON) on the user partition you want to restore to.
- [V0 partitions or firmware older than Luna HSM Firmware 7.7.0] **Partition policy 4:** "Allow secret key cloning" on page 340 is set to **1** (ON) on the user partition you want to restore to.

To restore a multifactor quorum-authenticated partition from backup

1. Configure your Luna HSM Client workstation using one of the following configurations:
   - Activated source partition:

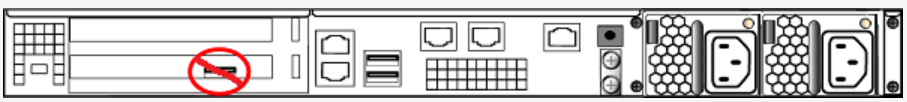


   - Non-activated source partition:



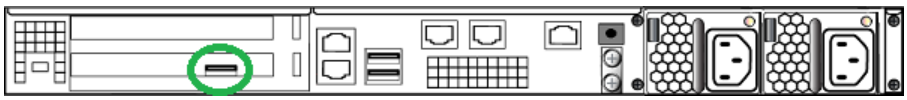
   a. Open a network (SSH) or serial connection to the appliance and log in as **admin**, or other admin-level user, to start a LunaSH session.

   b. Connect the backup HSM directly to one of the USB ports on the Luna Network HSM 7 appliance using the included USB cable.

> **NOTE** The Luna Backup HSM 7 must be connected to one of the appliance USB ports, and not the one on the HSM card:
>
> 
>
> The Luna Network HSM 7 USB connection provides adequate power, and connecting the provided power supply is not recommended.

   c. [Non-activated target partition] Connect the Luna Network HSM 7 appliance to a Luna PED, using a local or remote connection:

      – [Local PED] Connect the Luna PED to the USB port on the HSM card:



– [Remote PED] Connect the Luna Network HSM 7 appliance to the Remote PED server (see _____ _):

lunash:> **hsm ped connect -ip** <remote_ped_host_ip_address>

2. Get the serial number of the backup HSM, or read the serial number from the Backup HSM display screen:

lunash:> **token backup list**

3. Display a list of application partitions; you require the label for the partition you are restoring to.

lunash:> **partition list**

4. Display a list of the existing backups.

lunash:> **token backup partition list -serial** <backup_hsm_serial_number>

5. Initiate the restore operation:

lunash:> **partition restore -partition** <target_user_partition_label> **-tokenpar** <source_backup_partition_label> **-serial** <backup_hsm_serial_number> {**-add** | **-replace**}

Use the **-add** option to add only new objects, or the **-replace** option to erase the contents of the partition and replace them with the contents of the backup.

> **CAUTION!** If you are restoring a V1 backup to a V1 partition, use **-add** to restore the SMK. Use **-replace** only if you wish to erase any existing cryptographic objects on the target partition. By default, V1 backups only include the SMK.

6. You are prompted for the following credentials in the following order:

If the target partition is activated:

i. [In LunaSH] The Crypto Officer challenge secret for the target partition

ii. [On the Luna Backup HSM 7 touchscreen] The Crypto Officer (black) PED key(s) for the backup partition

If the target partition is not activated:

i. [On the Luna PED] The Remote PED Vector (orange) PED key(s) for the target HSM

ii. [On the Luna PED] The Crypto Officer (black) PED key(s) for the target partition

iii. [On the Luna Backup HSM 7 touchscreen] The Crypto Officer (black) PED key(s) for the backup partition

The restore operation begins once you have completed the authentication process. Objects are restored one at a time.

Luna Backup HSM 7 Connected to Luna Network HSM 7 Using Luna PED for Multifactor Quorum Authentication

In this configuration, you connect the Luna Backup HSM 7 to a USB port on the Luna Network HSM 7 appliance, and insert PED keys into a Remote Luna PED. This configuration allows you to perform backup/restore operations for all application partitions on that HSM. You can restore a partition backup to the original source partition or to another existing Luna application partition that shares the same cloning domain. To use this method, you require:

- "Luna Backup HSM 7" on page 474 **v1 or v2** with Luna Backup HSM 7 Firmware 7.7.1 or newer installed
- Luna Network HSM 7 Appliance Software 7.7.0 or newer installed on the Luna Network HSM 7

> **NOTE**
>
> - **The Luna Backup HSM 7 is shipped in** Secure Transport Mode, and must be recovered from STM before first use. STM recovery requires LunaCM on a Luna HSM Client.
>
>   See "Recovering the Luna Backup HSM 7 from Secure Transport Mode" on page 478.
> - **If you require the Luna Backup HSM 7 to be FIPS-compliant**, you must complete an additional configuration step after initialization that requires LunaCM on a Luna HSM Client computer (see "Configuring the Luna Backup HSM 7 for FIPS Compliance" on page 479). Therefore, it may be simpler to initialize the Luna Backup HSM 7 at the client instead of using the initialization procedure below.
> - **If you are backing up or restoring encrypted blobs stored on a V1 partition**, the Backup HSM must be connected to the client. Only the SMK can be backed up/restored using an appliance-connected Backup HSM.
> - **If** "Secure Trusted Channel" on page 110 **is enabled on the partition**, the Backup HSM must be connected to the client.
>
>   See "Luna Backup HSM 7 Connected to Luna HSM Client Using Remote Multifactor Quorum Authentication" on page 520.

This section provides instructions for the following procedures:

- "Initializing the Luna Backup HSM 7 for Multifactor Quorum Authentication" below
- "Backing Up a Multifactor Quorum-Authenticated Partition" on page 495
- "Restoring a Multifactor Quorum-Authenticated Partition From Backup" on page 500

Initializing the Luna Backup HSM 7 for Multifactor Quorum Authentication

You must initialize theLuna Backup HSM 7 prior to first use. You can initialize the backup HSM by connecting it to a Luna Network HSM 7 and using LunaSH commands to perform the initialization.

Prerequisites

- If necessary, recover the Luna Backup HSM 7 from Secure Transport Mode (see "Recovering the Luna Backup HSM 7 from Secure Transport Mode" on page 478).

> Ensure that you are familiar with the concepts in "Multifactor Quorum Authentication" on page 236. You will need the following PED keys:

- A blank Remote PED Vector (orange) PED key, plus the number required to create duplicate PED keys as necessary.

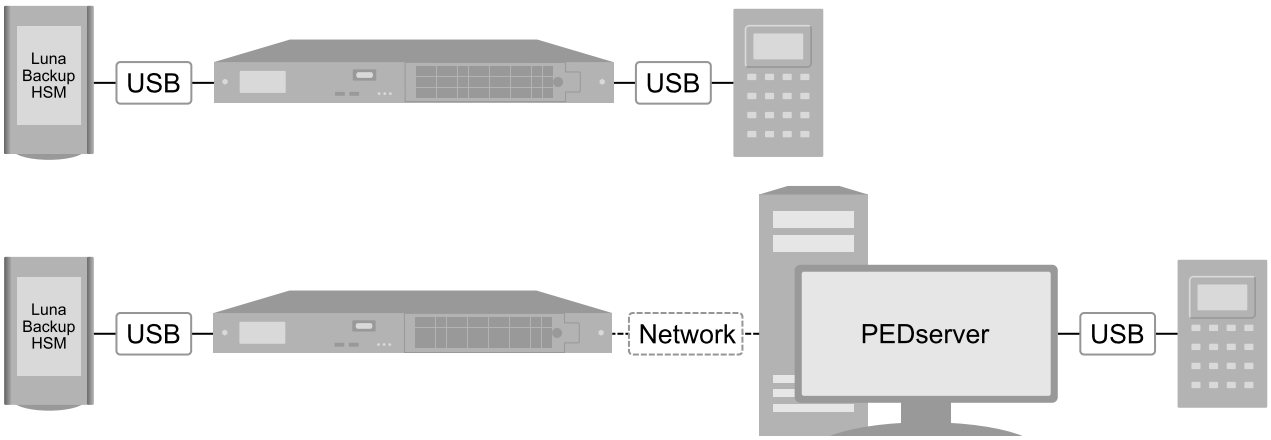
> **CAUTION!** Always make copies of your orange PED keys, or declare MofN as one-of-several, and store at least one safely. For the Luna Backup HSM 7 v1, *the orange PED key is as important as the HSM SO blue key or the Domain red key*.
>
> The orange PED key is required for all Luna Backup HSM 7 v1 operations. If this key is lost, your backups will become irretrievable. Thales recommends keeping multiple backups of all PED keys stored in a secure location.

- N number of HSM SO (blue) PED keys, as defined by the M of N scheme you choose for the HSM SO role, plus the number required to create duplicate PED keys as necessary.
- Blank or reused Domain (red) PED key(s).

> [Luna Backup HSM 7 Firmware 7.7.1 and newer only] Set the value of **-pedwritedelay** to **2000** to avoid experiencing frequent CKR_CALLBACK_ERRORS, which will prevent you from completing the procedure below. For more information about this error, refer to "Intermittent CKR_CALLBACK_ERROR: PED Cannot Service its USB Data Channel Fast Enough to Communicate with PEDserver" on page 276.

To initialize the Luna Backup HSM 7 for multifactor quorum authentication

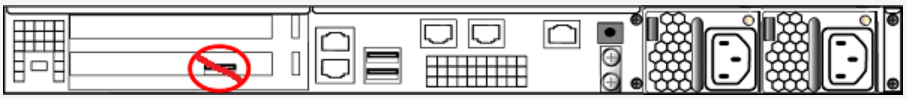
1. Configure your multifactor quorum-authenticated Luna Network HSM 7 using one of the following configurations:



   a. Open a network (SSH) or serial connection to the appliance and log in as **admin**, or other admin-level user, to start a LunaSH session.

   b. Connect the backup HSM directly to one of the USB ports on the Luna Network HSM 7 appliance using the included USB cable.

NOTE The Luna Backup HSM 7 must be connected to one of the appliance USB ports, and not the one on the HSM card:



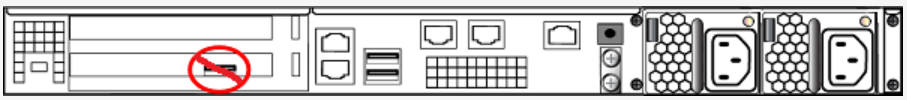
The Luna Network HSM 7 USB connection provides adequate power, and connecting the provided power supply is not recommended.

c. Get the serial number of the backup HSM, or read the serial number from the Backup HSM display screen.

lunash:> **token backup list**

d. Connect the Remote PED to the Luna Network HSM 7 appliance. You can connect a Remote PED directly to the Luna Network HSM 7 appliance using the included USB cable, or you can connect to a network-attached Luna HSM Client workstation that hosts a remote PED.

NOTE The Luna PED must be connected to one of the appliance USB ports, and not the one on the HSM card:



The Luna PED must be set to Remote PED mode (see "Modes of Operation" on page 248).

– If you connect the Remote PED directly to a USB port on the appliance, use the appliance loopback IP address (127.0.0.1) to connect to the local **pedserver** service running on the appliance, and specify the serial number of the connected backup HSM you want to use. You can read the serial number from the Backup HSM display screen. The **pedserver** service must be running on the appliance. You can use the lunash:> **service** commands to administer the service:

lunash:> **hsm ped connect -ip 127.0.0.1 -serial** <backup_hsm_serial_number>

NOTE A remote PED connected to the USB port on the appliance uses the appliance **pedserver** service. If the PED is not responding, use the lunash:> **service** commands to verify the service status and restart if necessary. The Luna PED must be in Remote mode.

– If you are using a network-attached Remote PED, connect to the IP address of the workstation used to host the Remote PED. This can be the same workstation you are using to host the LunaSH session, or a different workstation. Be sure to include the **-serial** option.

lunash:> **hsm ped connect -ip** <pedserver_host>**-serial**<backup_hsm_serial_number>

LunaSH generates and displays a one-time password that is used to set up a secure channel between the backup HSM and the PED, allowing you to securely initialize the Remote PED Vector (orange) PED key. Enter the displayed password on the PED when prompted to complete setup of the secure channel and respond to the prompts to create the Remote PED Vector (orange) PED key.

```
Please attend to the PED and enter following password: 94485995
```

2. Create Remote PED Vector (orange) PED key(s) for the backup HSM:

lunash:> **hsm ped vector init -serial** <backup_hsm_serial_number>

You are then prompted to insert the orange key again to authenticate the Remote PED connection.

> **CAUTION!** The orange PED key is required for all Luna Backup HSM 7 v1 operations. If this key is lost, your backups will become irretrievable. Thales recommends keeping multiple backups of all PED keys stored in a secure location.

3. Initialize the backup HSM:

   lunash:> **token backup init -label** <backup_hsm_label> **-serial** <backup_hsm_serial_number>

   You are prompted by the Luna PED for the blue HSM SO key(s) and red Domain key(s). Respond to the PED prompts and insert and set the PINs on the required keys when requested. Ensure that you label any new PED keys that you create during this process.

4. Use the **Duplicate** function on the PED to create and label duplicates of the new PED keys, as required. See "Duplicating Existing PED keys" on page 298 for details.

5. Disconnect the PED when done:

   - If you connected the Remote PED directly to a USB port on the appliance:

     lunash:> **hsm ped disconnect -serial** <backup_hsm_serial_number>

   - If you connected to a network-attached Remote PED:

     lunash:> **hsm ped disconnect**

> **NOTE** If your organization requires FIPS compliance, there is an additional procedure you must complete before using the Luna Backup HSM 7 to back up partitions. Refer to "Configuring the Luna Backup HSM 7 for FIPS Compliance" on page 479.

Backing Up a Multifactor Quorum-Authenticated Partition

Backups are created and stored as partitions within the Admin partition on the Luna Backup HSM 7. A new backup partition is created on initial backup. For subsequent backups, you can choose to replace the contents of the existing backup partition with the current source partition objects, or add new objects in the source partition to the existing backup partition. Like all cloning operations, the source and target backup partitions must be initialized with the same domain.

In addition to the credentials listed in "Credentials Required to Perform Backup and Restore Operations" on page 468, the Crypto Officer requires **admin**-level access to the appliance to access the LunaSH **partition backup** and **partition restore** commands (see Appliance Users and Roles).

Prerequisites

Before you begin, ensure that you have satisfied the following prerequisites:

> You are able to log in to the Luna Network HSM 7 using an **admin**-level account to access LunaSH.

> You have the required credentials:

If you are creating a new backup partition:

- The Remote PED Vector (orange) PED key(s) for the Backup HSM
- The HSM SO (blue) PED key(s) for the backup HSM
- New or reused Partition SO (blue) PED key(s) to initialize the backup partition

- New or reused Crypto Officer (black) PED key(s) to initialize the CO role on the backup partition
- The Domain (red) PED key(s) for the source partition, to initialize the domain on the backup

If you are backing up to an existing backup partition whose domain matches the source partition:

- The Remote PED Vector (orange) PED key(s) for the Backup HSM
- The HSM SO (blue) PED key(s) for the backup HSM
- The existing Crypto Officer (black) PED key(s) for the backup partition

> **TIP** If the source partition is activated, only the source partition Crypto Officer's challenge secret is required. To simplify the backup process and minimize interactions with the PED, it is recommended that you activate the CO role on the user partitions you want to back up. See "Activation on Multifactor Quorum-Authenticated Partitions" on page 373 for more information.

If the source partition is not activated, you also need:

- The Remote PED Vector (orange) PED key(s) for the source HSM
- The Crypto Officer (black) PED key(s) for the source partition

> The following policies are set:

- **HSM policy 16:** Allow network replication must be set to **1** (ON) on the HSM that hosts the user partition.
- [V0 partitions or firmware older than Luna HSM Firmware 7.7.0] **Partition policy 0:** "Allow private key cloning" on page 338 is set to **1** (ON) on the user partition.
- [V0 partitions or firmware older than Luna HSM Firmware 7.7.0] **Partition policy 4:** "Allow secret key cloning" on page 340 is set to **1** (ON) on the user partition.

> [Luna Backup HSM 7 Firmware 7.7.1 and newer only] Set the value of **-pedwritedelay** to **2000** to avoid experiencing frequent CKR_CALLBACK_ERRORS, which will prevent you from completing the procedure below. For more information about this error, refer to "Intermittent CKR_CALLBACK_ERROR: PED Cannot Service its USB Data Channel Fast Enough to Communicate with PEDserver" on page 276.

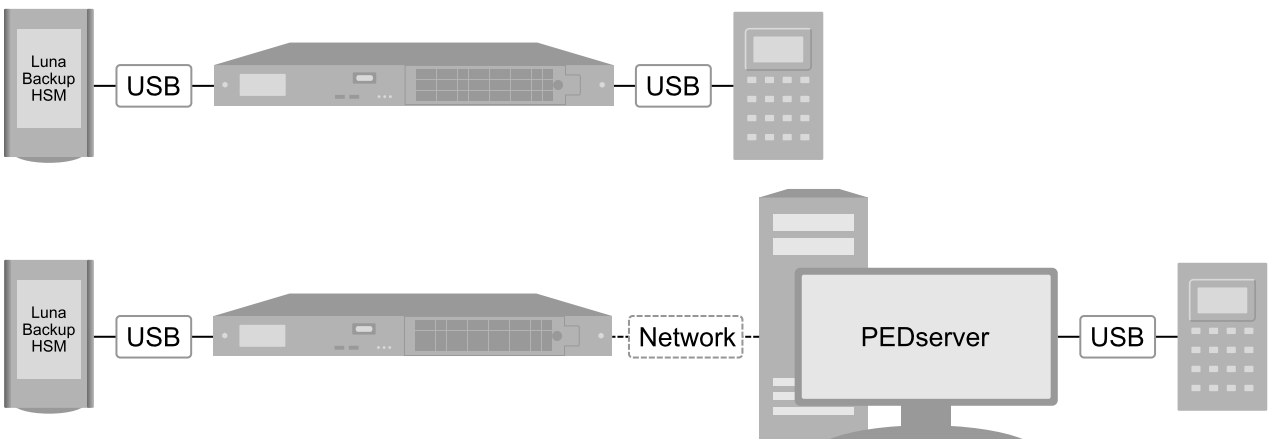
> **NOTE** HSS (PQC) private keys cannot be cloned, due to inherent restrictions of that key type. This means that backup and HA synchronization fail if HSS private keys are encountered in your application partition.

TIP HSS / LMS keys cannot be moved or copied off the HSM on which they are created. If NIST standard definition of HSS or other inputs change in a future release, the CRN and other Luna HSM documentation will be updated. In the interim, cloning attempts, like backup/restore or HA-group synchronization are refused by the HSM. For example, an attempt to backup a mixed-content partition might produce progress messages like this, where two among the source objects are HSS keys:

```
Logging in as the SO on slot 127.
Creating partition HSSBackuptest on slot 127.
Verifying that all objects can be backed up...
8 objects found; attempting to back up 8 objects
Backing up objects...
Cloned object 121 to partition HSSBackuptest (new handle 429).
Cloned object 120 to partition HSSBackuptest (new handle 505).
Cloned object 116 to partition HSSBackuptest (new handle 506).
Cloned object 115 to partition HSSBackuptest (new handle 510).
Cloned object 103 to partition HSSBackuptest (new handle 511).
Cloned object 102 to partition HSSBackuptest (new handle 514).
Failed to clone object 100 from partition HSSBackuptest (CKR_KEY_
TYPE_INCONSISTENT).
Failed to clone object 97 from partition HSSBackuptest (CKR_
ATTRIBUTE_TYPE_INVALID).
Not all objects can be cloned. Please verify HSM configuration.
Resizing partition HSSBackuptest on slot 127 to minimum necessary
space.
Backup Successfully Completed.
6 objects have been backed up to partition HSSBackuptest
on slot 127.
Command Result : No Error
```

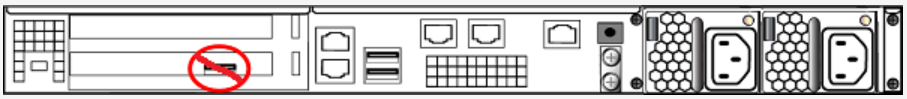
To back up a multifactor quorum-authenticated partition

1. Configure your Luna Network HSM 7 appliance using one of the following configurations:



   a. Open a network (SSH) or serial connection to the appliance and log in as **admin**, or other admin-level user, to start a LunaSH session.

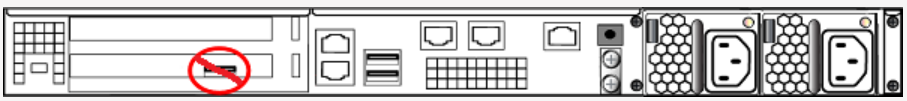
b. Connect the backup HSM directly to one of the USB ports on the Luna Network HSM 7 appliance using the included USB cable.

> **NOTE** The Luna Backup HSM 7 must be connected to one of the appliance USB ports, and not the one on the HSM card:
>
> 
>
> The Luna Network HSM 7 USB connection provides adequate power, and connecting the provided power supply is not recommended.

c. Get the serial number of the backup HSM, or read the serial number from the backup HSM display screen.

lunash:> **token backup list**

d. Connect the Remote PED to the Luna Network HSM 7 appliance. You can connect a Remote PED directly to the Luna Network HSM 7 appliance using the included USB cable, or you can connect to a network-attached Luna HSM Client workstation that hosts a remote PED:

> **NOTE** The Luna PED must be connected to one of the appliance USB ports, and not the one on the HSM card:
>
> 
>
> The Luna PED must be set to Remote PED mode (see "Modes of Operation" on page 248).

– If you connect the Remote PED directly to a USB port on the appliance, use the appliance loopback IP address (127.0.0.1) to connect to the local **pedserver** service running on the appliance, and specify the serial number of the connected backup HSM you want to use:

lunash:> **hsm ped connect -ip 127.0.0.1 -serial** <backup_hsm_serial_number>

> **NOTE** A remote PED connected to the USB port on the appliance uses the appliance **pedserver** service. If the PED is not responding, use the lunash:> **service** commands to verify the service status and restart if necessary. The Luna PED must be in Remote mode.

– If you are using a network-attached Remote PED, connect to the IP address of the workstation used to host the Remote PED. This can be the same workstation you are using to host the LunaSH session, or a different workstation. Be sure to include the **-serial** option.

lunash:> **hsm ped connect -ip** <remote_ped_host_ip_address> **-serial**<backup_hsm_serial_number>

Respond to the prompts on the PED to insert the Backup HSM's orange PED key.

2. Display a list of application partitions; you require the label for the partition you are backing up.

lunash:> **partition list**

3. If you plan to back up to an existing partition on the Backup HSM, display a list of the existing backups.

lunash:> **token backup partition list -serial** <backup_hsm_serial_number>

4. Initiate the backup operation:

   lunash:> **partition backup -partition** <source_partition_label> **-serial** <backup_hsm_serial_number> [**-tokenpar** <target_backup_partition_label>] [**-add** | **-replace**]

   > **NOTE** You must specify **-add** or **-replace** when backing up to an existing backup partition. Use **-add** to add only new objects. Use **-replace** to erase the contents of the existing backup and replace them with the contents of the source partition. You do not need to specify these options when backing up a V1 partition, as only the SMK is backed up.
   >
   > If you omit the **-tokenpar** option when creating a new backup, the partition is assigned a default name (<source_partition_name>_<YYYYMMDD>) based on the source HSM's internally-set time and date.
   >
   > If the backup operation is interrupted (if the Backup HSM is unplugged, or if you fail to respond to PED prompts, for example), the Backup HSM's full available space can become occupied with a single backup partition. If this occurs, delete the backup partition with lunash:> **token backup partition delete** before reattempting the backup operation.

5. You are prompted for the following credentials in the following order:

   If the source partition is not activated:

   i. The Remote PED Vector (orange) PED key(s) for the source HSM
   ii. The Crypto Officer (black) PED key(s) for the source partition
   iii. The Remote PED Vector (orange) PED key(s) for the Backup HSM

   If the source partition is activated:

   i. [In LunaSH] The Crypto Officer challenge secret for the source partition

   If you are creating a new backup partition:

   i. The HSM SO (blue) PED key(s) for the backup HSM, to log in
   ii. New or reused Partition SO (blue) PED key(s) to initialize the backup partition
   iii. The Partition SO (blue) PED key(s) you just created for the backup partition, to log in
   iv. New or reused Crypto Officer (black) PED key(s) to initialize the CO role on the backup partition
   v. The Domain (red) PED key(s) for the source partition, to initialize the domain on the backup
   vi. The Crypto Officer (black) PED key(s) you just created for the backup partition, to log in

   If you are backing up to an existing backup partition:

   i. The HSM SO (blue) PED key(s) for the backup HSM, to log in.
   ii. The existing Crypto Officer (black) PED key(s) for the backup partition

   The backup begins once you have completed the authentication process. Objects are backed up one at a time.

6. Disconnect the PED when done:

   - If you connected the Remote PED directly to a USB port on the appliance:

     lunash:> **hsm ped disconnect -serial** <backup_hsm_serial_number>

   - If you connected to a network-attached Remote PED:

lunash:> **hsm ped disconnect**

7. If this is the first backup to the backup partition, use the **Duplicate** function on the PED to create and label a set of backup keys for the new PO (blue) and CO (black) keys. See "Duplicating Existing PED keys" on page 298 for details.

Restoring a Multifactor Quorum-Authenticated Partition From Backup

You can restore the objects from a multifactor quorum-authenticated backup partition to the same partition that was originally backed up, or to another partition that has been initialized with the same domain (red PED key).

Prerequisites

Before beginning, ensure that you have satisfied the following prerequisites:

> You are able to log in to the Luna Network HSM 7 using an **admin**-level account to access LunaSH.

> The target partition must be initialized using the same domain (red PED key) as the backup partition, the Crypto Officer role must be initialized and the CO role credential changed from its initial value.

> You have the required credentials:

- The Remote PED Vector (orange) PED key(s) for the backup HSM
- The Crypto Officer challenge secret for the target partition
- The Crypto Officer (black) PED key(s) for the backup partition

> **TIP** If the target partition is activated, only the Crypto Officer's challenge secret is required. To simplify the backup process and minimize interactions with the PED, it is recommended that you activate the CO role on the user partitions you want to restore from backup. See "Activation on Multifactor Quorum-Authenticated Partitions" on page 373 for more information.

If the target partition is not activated, you also need:

- The Remote PED Vector (orange) PED key(s) for the target HSM
- The Crypto Officer (black) PED key(s) for the target partition

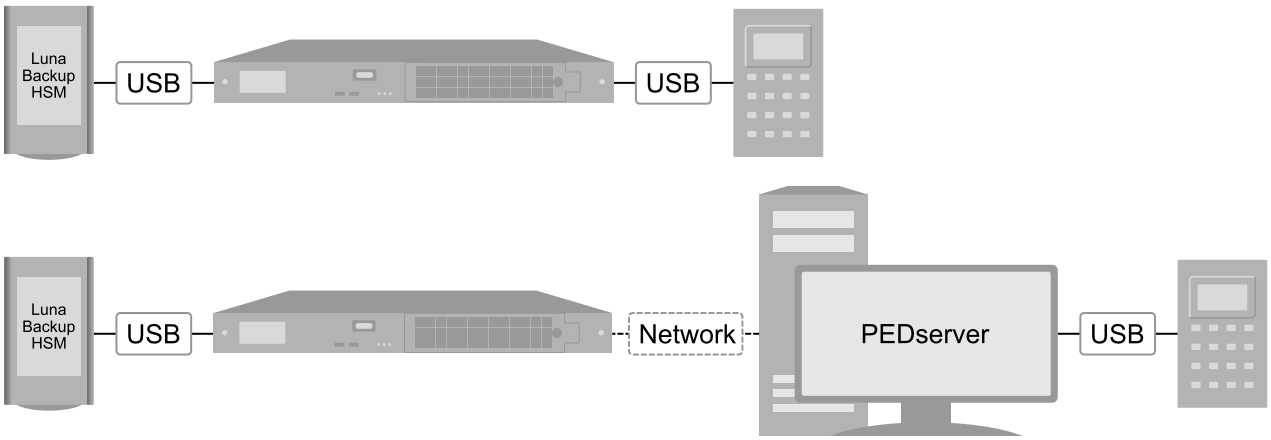
> The following policies are set:

- **HSM policy 16:** Allow network replication must be set to **1** (ON) on the HSM that hosts the user partition you want to restore to.
- [V0 partitions only] **Partition policy 0:** "Allow private key cloning" on page 338 is set to **1** (ON) on the user partition you want to restore to.
- [V0 partitions only] **Partition policy 4:** "Allow secret key cloning" on page 340 is set to **1** (ON) on the user partition you want to restore to.

> [Luna Backup HSM 7 Firmware 7.7.1 and newer only] Set the value of **-pedwritedelay** to **2000** to avoid experiencing frequent CKR_CALLBACK_ERRORS, which will prevent you from completing the procedure below. For more information about this error, refer to "Intermittent CKR_CALLBACK_ERROR: PED Cannot Service its USB Data Channel Fast Enough to Communicate with PEDserver" on page 276.

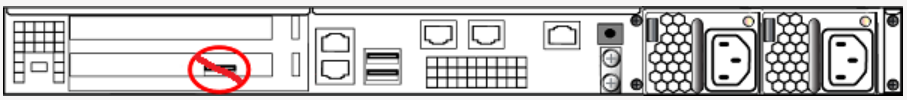
To restore a multifactor quorum-authenticated partition from backup

1. Configure your Luna HSM Client workstation using one of the following configurations:



a. Open a network (SSH) or serial connection to the appliance and log in as **admin**, or other admin-level user, to start a LunaSH session.

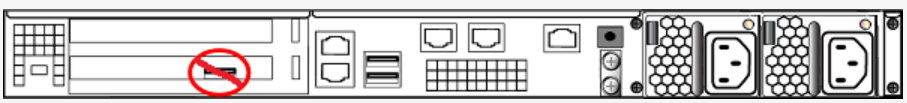
b. Connect the backup HSM directly to one of the USB ports on the Luna Network HSM 7 appliance using the included USB cable.

> **NOTE** The Luna Backup HSM 7 must be connected to one of the appliance USB ports, and not the one on the HSM card:
>
> 
>
> The Luna Network HSM 7 USB connection provides adequate power, and connecting the provided power supply is not recommended.

c. Get the serial number of the backup HSM, or read the serial number from the backup HSM display screen.

lunash:> **token backup list**

d. Connect the Remote PED to the Luna Network HSM 7 appliance. You can connect a Remote PED directly to the Luna Network HSM 7 appliance using the included USB cable, or you can connect to a network-attached Luna HSM Client workstation that hosts a remote PED:

> **NOTE** The Luna PED must be connected to one of the appliance USB ports, and not the one on the HSM card:
>
> 
>
> The Luna PED must be set to Remote PED mode (see "Modes of Operation" on page 248).

– If you connect the Remote PED directly to a USB port on the appliance, use the appliance loopback IP address (127.0.0.1) to connect to the local **pedserver** service running on the appliance, and specify the serial number of the connected backup HSM you want to use:

lunash:> **hsm ped connect -ip 127.0.0.1 -serial** <backup_hsm_serial_number>

> **NOTE** A remote PED connected to the USB port on the appliance uses the appliance **pedserver** service. If the PED is not responding, use the lunash:> **service** commands to verify the service status and restart if necessary. The Luna PED must be in Remote mode.

- If you are using a network-attached Remote PED, connect to the IP address of the workstation used to host the Remote PED. This can be the same workstation you are using to host the LunaSH session, or a different workstation. Be sure to include the **-serial** option.

  lunash:> **hsm ped connect -ip** <remote_ped_host_ip_address> **-serial**<backup_hsm_serial_number>

Respond to the prompts on the PED to insert the Backup HSM's orange PED key(s).

2. Display a list of application partitions; you require the label for the partition you are restoring to.

   lunash:> **partition list**

3. Display a list of the existing backups.

   lunash:> **token backup partition list -serial** <backup_hsm_serial_number>

4. Initiate the restore operation:

   lunash:> **partition restore -partition** <target_user_partition_label> **-tokenpar** <source_backup_partition_label> **-serial** <backup_hsm_serial_number> {**-add** | **-replace**}

   Use the **-add** option to add only new objects, or the **-replace** option to erase the contents of the partition and replace them with the contents of the backup.

   > **CAUTION!** If you are restoring a V1 backup to a V1 partition, use **-add** to restore the SMK. Use **-replace** only if you wish to erase any existing cryptographic material on the target partition. By default, V1 backups only include the SMK.

5. You are prompted for the following credentials in the following order:

   If the target partition is activated:

   i. [In LunaSH] The Crypto Officer challenge secret for the target partition
   ii. The Crypto Officer (black) PED key(s) for the backup partition

   If the target partition is not activated:

   i. The Remote PED Vector (orange) PED key(s) for the target HSM
   ii. The Crypto Officer (black) PED key(s) for the target partition
   iii. The Remote PED Vector (orange) PED key(s) for the backup HSM
   iv. The Crypto Officer (black) PED key(s) for the backup partition

   The restore operation begins once you have completed the authentication process. Objects are restored one at a time.

6. Disconnect the PED when done:

   - If you connected the Remote PED directly to a USB port on the appliance:

     lunash:> **hsm ped disconnect -serial** <backup_hsm_serial_number>

   - If you connected to a network-attached Remote PED:

lunash:> **hsm ped disconnect**

Luna Backup HSM 7 Connected to Luna Network HSM 7 Using Password Authentication

In this configuration, you connect the Luna Backup HSM 7 to a USB port on the Luna Network HSM 7 appliance, and enter passwords in LunaSH. This configuration allows you to perform backup/restore operations for all application partitions on that HSM. You can restore a partition backup to the original source partition or to another existing Luna application partition that shares the same cloning domain. To use this method, you require:

- "Luna Backup HSM 7" on page 474 **v1 or v2** with Luna Backup HSM 7 Firmware 7.7.1 or newer installed
- Luna Network HSM 7 Appliance Software 7.7.0 or newer installed on the Luna Network HSM 7

> **NOTE**
>
> - **The Luna Backup HSM 7 is shipped in** Secure Transport Mode, and must be recovered from STM before first use. STM recovery requires LunaCM on a Luna HSM Client. See "Recovering the Luna Backup HSM 7 from Secure Transport Mode" on page 478.
> - **If you require the Luna Backup HSM 7 to be FIPS-compliant**, you must complete an additional configuration step after initialization that requires LunaCM on a Luna HSM Client computer (see "Configuring the Luna Backup HSM 7 for FIPS Compliance" on page 479). Therefore, it may be simpler to initialize the Luna Backup HSM 7 at the client instead of using the procedure below (see "Luna Backup HSM 7 Connected to Luna HSM Client Using Password Authentication" on page 531).
> - **If you are backing up or restoring encrypted blobs stored on a V1 partition**, the Backup HSM must be connected to the client (see "Luna Backup HSM 7 Connected to Luna HSM Client Using Password Authentication" on page 531). Only the SMK can be backed up/restored using an appliance-connected Backup HSM.
> - **If** "Secure Trusted Channel" on page 110 **is enabled on the partition**, the Backup HSM must be connected to the client (see "Luna Backup HSM 7 Connected to Luna HSM Client Using Password Authentication" on page 531).

This section provides instructions for the following procedures:

- "Initializing the Luna Backup HSM 7 for Password Authentication" below
- "Backing Up a Password-Authenticated Partition" on the next page
- "Restoring a Password-Authenticated Partition From Backup" on page 507

Initializing the Luna Backup HSM 7 for Password Authentication

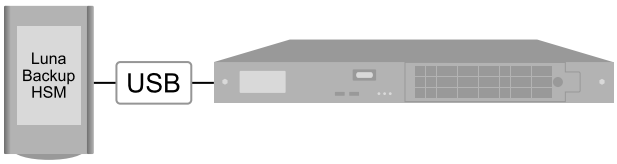
You must initialize the Luna Backup HSM 7 prior to first use. You can initialize the backup HSM by connecting it to a Luna Network HSM 7 and using LunaSH commands to perform the initialization.

Prerequisites

> If necessary, recover the Luna Backup HSM 7 from Secure Transport Mode (see "Recovering the Luna Backup HSM 7 from Secure Transport Mode" on page 478).

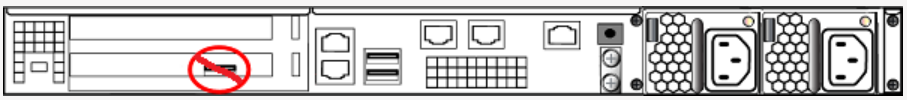
To initialize the Luna Backup HSM 7 for password authentication

1. Configure your password-authenticated Luna Network HSM 7 as illustrated below:



   a. Open a network (SSH) or serial connection to the appliance and log in as **admin**, or other admin-level user, to start a LunaSH session.

   b. Connect the backup HSM directly to the Luna Network HSM 7 using the included USB cable.

   > **NOTE** The Luna Backup HSM 7 must be connected to one of the appliance USB ports, and not the one on the HSM card:
   >
   > 
   >
   > The Luna Network HSM 7 USB connection provides adequate power, and connecting the provided power supply is not recommended.

2. Get the serial number of the backup HSM, or read the serial number from the Backup HSM display screen.

   lunash:> **token backup list**

3. Initialize the backup HSM:

   lunash:> **token backup init -label** <backup_hsm_label> **-serial** <backup_hsm_serial_number>

   You are prompted to set a new HSM SO password and the HSM domain string.

   > **NOTE** If your organization requires FIPS compliance, there is an additional procedure you must complete before using the Luna Backup HSM 7 to back up partitions. Refer to "Configuring the Luna Backup HSM 7 for FIPS Compliance" on page 479.

Backing Up a Password-Authenticated Partition

Backups are created and stored as partitions within the Admin partition on the backup HSM. A new backup partition is created on initial backup. For subsequent backups, you can choose to replace the contents of the existing backup partition with the current source partition objects, or add new objects in the source partition to the existing backup partition. Like all cloning operations, the source and target backup partitions must be initialized with the same domain.

In addition to the credentials listed in "Credentials Required to Perform Backup and Restore Operations" on page 468, the Crypto Officer requires **admin**-level access to the appliance to access the LunaSH **partition backup** and **partition restore** commands (see Appliance Users and Roles).

Prerequisites

Before you begin, ensure that you have satisfied the following prerequisites:

> You are able to log in to the Luna Network HSM 7 using an **admin**-level account to access LunaSH.

> You have the required credentials:

If you are creating a new backup:

- The Crypto Officer password and domain string for the source partition
- The HSM SO password for the backup HSM

If you are adding to an existing backup initialized with the same domain string as the source partition:

- The Crypto Officer password and domain string for the source partition
- The HSM SO password for the backup HSM
- The Partition SO and Crypto Officer passwords for the existing backup

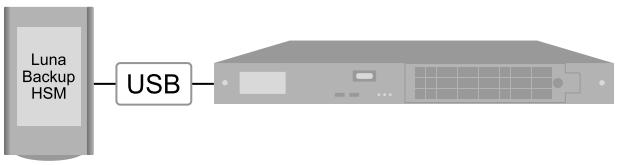
> The following policies are set (see HSM Capabilities and Policies and "Partition Capabilities and Policies" on page 337 for more information):

- **HSM policy 16:** Allow network replication must be set to **1** (ON) on the HSM that hosts the user partition.
- [V0 partitions or firmware older than Luna HSM Firmware 7.7.0] **Partition policy 0:** "Allow private key cloning" on page 338 is set to **1** (ON) on the user partition.
- [V0 partitions or firmware older than Luna HSM Firmware 7.7.0] **Partition policy 4:** "Allow secret key cloning" on page 340 is set to **1** (ON) on the user partition.

> **NOTE** HSS (PQC) private keys cannot be cloned, due to inherent restrictions of that key type. This means that backup and HA synchronization fail if HSS private keys are encountered in your application partition.

To back up a password-authenticated partition

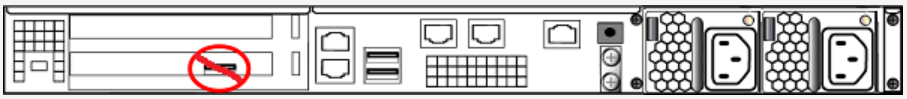
1. Configure your Luna Network HSM 7 as illustrated below:



   a. Open a network (SSH) or serial connection to the appliance and log in as **admin**, or other admin-level user, to start a LunaSH session.

   b. Connect the backup HSM directly to the Luna Network HSM 7 using the included USB cable.

NOTE The Luna Backup HSM 7 must be connected to one of the appliance USB ports, and not the one on the HSM card:



The Luna Network HSM 7 USB connection provides adequate power, and connecting the provided power supply is not recommended.

2. Get the serial number of the backup HSM, or read the serial number from the Backup HSM display screen.

   lunash:> **token backup list**

3. Display a list of application partitions; you require the label for the partition you are backing up.

   lunash:> **partition list**

4. If you plan to back up to an existing partition on the Backup HSM, display a list of the existing backups.

   lunash:> **token backup partition list -serial** <backup_hsm_serial_number>

5. Initiate the backup operation:

   lunash:> **partition backup -partition** <source_partition_label> **-serial** <backup_hsm_serial_number> [**-tokenpar** <target_backup_partition_label>] [**-tokensopwd** <backup_hsm_SO_password>] [**-add** | **-replace**]

   NOTE You must specify **-add** or **-replace** when backing up to an existing backup partition. Use **-add** to add only new objects. Use **-replace** to add new objects and overwrite existing objects. You do not need to specify these options when backing up a V1 partition, as only the SMK is backed up.

   If you omit the **-tokenpar** option when creating a new backup, the partition is assigned a default name (<source_partition_name>_<YYYYMMDD>) based on the source HSM's internally-set time and date.

   If the backup operation is interrupted (if the Backup HSM is unplugged, for example), the Backup HSM's full available space can become occupied with a single backup partition. If this occurs, delete the backup partition with lunash:> **token backup partition delete** before reattempting the backup operation.

6. Respond to the prompts for the following passwords, unless you specified them in the **partition backup** options:

   a. The Crypto Officer password for the source partition

   b. The HSM SO password for the backup HSM

   c. The Crypto Officer password for the target partition on the backup HSM (if you specified an existing backup). If you are creating a new backup, you must set its CO password now.

   d. If you are creating a new backup, you must provide the domain string for the source partition -- it is used to initialize the new backup partition so that objects can be cloned. If your target is an existing backup partition, the operation will proceed only if the domains already match.

   The backup begins once you have completed the authentication process. Objects are backed up one at a time.

Restoring a Password-Authenticated Partition From Backup

You can restore the objects from a multifactor quorum-authenticated backup partition to the same partition that was originally backed up, or to another partition that has been initialized with the same domain string.

Prerequisites

Before beginning, ensure that you have satisfied the following prerequisites:

> You are able to log in to the Luna Network HSM 7 appliance using an **admin**-level account to access LunaSH.

> The target partition must be initialized with the same domain string as the backup partition.

> You have the required credentials:

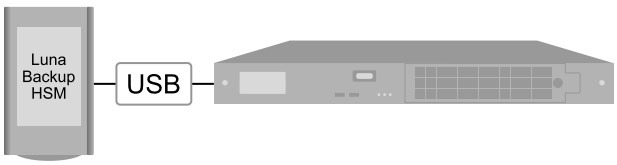
- The Crypto Officer password for the target partition
- The Crypto Officer password for the backup partition

> The following policies are set:

- **HSM policy 16:** Allow network replication must be set to **1** (ON) on the HSM that hosts the user partition you want to restore to.
- [V0 partitions only] **Partition policy 0:** "Allow private key cloning" on page 338 is set to **1** (ON) on the user partition you want to restore to.
- [V0 partitions only] **Partition policy 4:** "Allow secret key cloning" on page 340 is set to **1** (ON) on the user partition you want to restore to.

To restore a password-authenticated partition

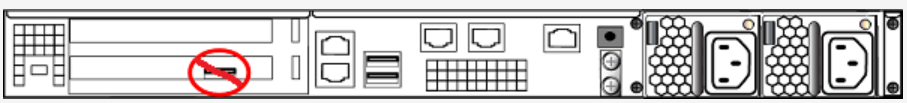
1. Configure your Luna Network HSM 7 as illustrated below:



   a. Open a network (SSH) or serial connection to the appliance and log in as **admin**, or other admin-level user, to start a LunaSH session.

   b. Connect the backup HSM directly to the Luna Network HSM 7 using the included USB cable.

   NOTE The Luna Backup HSM 7 must be connected to one of the appliance USB ports, and not the one on the HSM card:



   The Luna Network HSM 7 USB connection provides adequate power, and connecting the provided power supply is not recommended.

2. Display a list of application partitions; you require the label for the partition you are restoring to.

   lunash:> **partition list**

3. Display a list of the existing backups.

   lunash:> **token backup partition list -serial** <backup_hsm_serial_number>

4. Initiate the restore operation:

   lunash:> **partition restore -partition** <target_user_partition_label> **-tokenpar** <backup_partition_label> **-serial** <backup_hsm_serial_number> {**-add** | **-replace**}

   Use the **-add** option to add only new objects, or the **-replace** option to add new objects and overwrite existing objects.

   > **CAUTION!** If you are restoring a V1 backup to a V1 partition, use **-add** to restore the SMK. Use **-replace** only if you wish to erase any existing cryptographic material on the target partition. By default, V1 backups include only the SMK.

5. Respond to the prompts for the following passwords:

   a. The Crypto Officer password for the target partition

   b. The Crypto Officer password for the backup partition

   The restore operation begins once you have completed the authentication process. Objects are restored one at a time.

   > **TIP** HSS / LMS keys cannot be moved or copied off the HSM on which they are created. If NIST standard definition of HSS or other inputs change in a future release, the CRN and other Luna HSM documentation will be updated. In the interim, cloning attempts, like backup/restore or HA-group synchronization are refused by the HSM. For example, an attempt to backup a mixed-content partition might produce progress messages like this, where two among the source objects are HSS keys:

```
Logging in as the SO on slot 127.
Creating partition HSSBackuptest on slot 127.
Verifying that all objects can be backed up...
8 objects found; attempting to back up 8 objects
Backing up objects...
Cloned object 121 to partition HSSBackuptest (new handle 429).
Cloned object 120 to partition HSSBackuptest (new handle 505).
Cloned object 116 to partition HSSBackuptest (new handle 506).
Cloned object 115 to partition HSSBackuptest (new handle 510).
Cloned object 103 to partition HSSBackuptest (new handle 511).
Cloned object 102 to partition HSSBackuptest (new handle 514).
Failed to clone object 100 from partition HSSBackuptest (CKR_KEY_
TYPE_INCONSISTENT).
Failed to clone object 97 from partition HSSBackuptest (CKR_
ATTRIBUTE_TYPE_INVALID).
Not all objects can be cloned. Please verify HSM configuration.
Resizing partition HSSBackuptest on slot 127 to minimum necessary
space.
Backup Successfully Completed.
6 objects have been backed up to partition HSSBackuptest
on slot 127.
Command Result : No Error
```


Luna Backup HSM 7 Connected to Luna HSM Client Using Direct Multifactor Quorum Authentication

In this configuration, you connect the Luna Backup HSM 7 to a USB port on the Luna HSM Client, and insert PED keys directly into the Luna Backup HSM 7. This allows you to perform backup/restore operations for all application partitions that can be accessed by the client. You can restore a partition backup to the original source partition or to another existing Luna application partition that shares the same cloning domain. To use this method, you require:

- "Luna Backup HSM 7" on page 474 **v2**
- Luna HSM Client 10.4.0 or newer

This section provides instructions for the following procedures:

- "Initializing the Luna Backup HSM 7" below
- "Configuring the Luna Backup HSM 7 for FIPS Compliance" on page 511
- "Backing Up a Multifactor Quorum-Authenticated Partition" on page 511
- "Restoring To a Multifactor Quorum-Authenticated Partition" on page 516

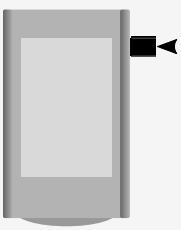
Initializing the Luna Backup HSM 7

You must initialize the Luna Backup HSM 7 prior to first use. You can initialize the backup HSM by connecting it to a Luna HSM Client and using LunaCM commands to perform the initialization.

Prerequisites

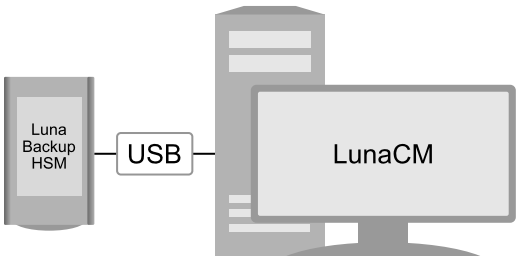
You need the following PED keys:

- N number of blue (HSM SO) PED keys, as defined by the M of N scheme you choose for the HSM SO role, plus the number required to create duplicate PED keys as necessary.
- Blank or reused red (Domain) PED key(s)

> **NOTE** Use the USB-C adapter in the USB port on the right side of the Luna Backup HSM 7 to insert PED keys:
>
> 

To initialize the Luna Backup HSM 7

1. Connect your Luna Backup HSM 7 to a workstation:



a. Install the required Luna HSM Client software on the workstation, including the **Backup** option. See "Client Software Required to Perform Backup and Restore Operations" on page 469 for details.

> **NOTE** If you are installing Luna HSM Client on Windows, the driver may not be installed unless the Luna device is connected to the computer first; refer to Backup/USB/PCIe Drivers Not Installed on Windows 10 or Windows Server 2022 Unless Device is Connected.

b. Connect the backup HSM directly to the Luna HSM Client workstation using the included USB cable.

> **NOTE** On most Luna HSM Client computers, the USB port provides adequate power, and connecting the provided power supply is not recommended. Wait and confirm that the HSM boots properly. If the HSM fails to boot up:
>
> **1.** Disconnect the HSM from the USB port.
>
> **2.** Connect the HSM to power using the provided power supply. Wait for it to boot completely.
>
> **3.** Reconnect the HSM to the USB port on the client.
>
> If the HSM is connected to the USB port before the power supply, you may encounter an issue where the HSM occasionally loses contact with the client, and must be power cycled.

2. Launch LunaCM on the client workstation.
3. Select the slot assigned to the backup HSM Admin partition.

   lunacm:> **slot set -slot** <slot_id>
4. If necessary, recover the HSM from Secure Transport Mode. See Secure Transport Mode for more information:

   lunacm:> **stm recover -randomuserstring** <string>

> **NOTE** Recovering a Luna Backup HSM 7 from secure transport mode may take up to three minutes.

5. Initialize the selected backup HSM, specifying a label and the **-iped** authentication mode.

   lunacm:> **hsm init -iped -label** <label>

> You are prompted by the touchscreen for the blue HSM SO PED key(s) and red Domain PED key(s). Respond to the prompts and insert and set the PINs on the required keys when requested. Ensure that you label any new PED keys that you create during this process.

Configuring the Luna Backup HSM 7 for FIPS Compliance

Luna Backup HSM Firmware 7.7.1 and newer uses the same updated cloning protocol as Luna HSM Firmware 7.7.0 and newer. For the Luna Backup HSM 7 to be FIPS-compliant, it must restrict restore operations to application partitions that use the new protocol. This restriction is applied by setting **HSM policy 55:** Enable Restricted Restore to **1** on the backup HSM. The Luna Backup HSM 7 must be initialized and connected to a Luna HSM Client computer to set this policy.

When this policy is enabled on the Luna Backup HSM 7, objects that have been backed up from partitions using firmware older than Luna HSM Firmware 7.7.0 can be restored to Luna HSM Firmware 7.7.0 or newer (V0 or V1) partitions only.

> **CAUTION!** FIPS compliance requires that objects are never cloned or restored to an HSM using less secure firmware, and this includes restoring from Luna Backup HSM 7 firmware.
>
> If you have backups already stored on the Luna Backup HSM 7 that were taken from pre-7.7.0 partitions, turning this policy ON will prevent you from restoring them to the same source partition. You must update the HSM containing the source partition to Luna HSM Firmware 7.7.0 or newer before restoring from backup.

> **NOTE** **HSM policy 12:** Allow non-FIPS algorithms, which is used to set FIPS-compliant mode on other Luna HSMs, does not apply to the Luna Backup HSM 7. Attempts to change this policy will fail with the error `CKR_CANCEL`.

To configure the Luna Backup HSM 7 for FIPS compliance

1. On the Luna HSM Client computer, run LunaCM.
2. Set the active slot to the Luna Backup HSM 7.

   lunacm:> **slot set -slot** <slot_id>
3. Log in as Backup HSM SO.

   lunacm:> **role login -name so**
4. Set **HSM policy 55:** Enable Restricted Restore to **1**.

   lunacm:> **hsm changehsmpolicy -policy 55 -value 1**
5. [Optional] Check that the Luna Backup HSM 7 is now in FIPS approved operation mode.

   lunacm:> **hsm showinfo**

   ```
   *** The HSM is in FIPS 140-2 approved operation mode. ***
   ```

Backing Up a Multifactor Quorum-Authenticated Partition

Backups are created and stored as partitions within the Admin partition on the Luna Backup HSM 7. A new backup partition is created on initial backup. For subsequent backups, you can choose to replace the contents of the existing backup partition with the current source partition objects, or add new objects in the source partition to the existing backup partition. Like all cloning operations, the source and target backup partitions must be initialized with the same domain.

Prerequisites

> You have the required credentials:

If the source partition is not activated:

- [Remote PED authentication] The Remote PED Vector (orange) PED key(s) for the source HSM
- The Crypto Officer (black) PED key(s) for the source partition

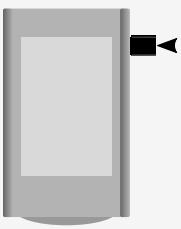
> **TIP** If the source partition is activated, only the source partition Crypto Officer's challenge secret is required. To simplify the backup process and minimize interactions with the PED, it is recommended that you activate the CO role on the user partitions you want to backup. See "Activation on Multifactor Quorum-Authenticated Partitions" on page 373 for more information.

If you are creating a new backup partition:

- New or reused Partition SO (blue) PED key(s) to initialize the backup partition
- The Domain (red) PED key(s) for the source partition, to initialize the domain on the backup
- New or reused Crypto Officer (black) PED key(s) to initialize the CO role on the backup partition

If you are backing up to an existing backup partition whose domain matches the source partition:

- The existing Partition SO (blue) PED key(s) for the backup partition, to log in
- The existing Crypto Officer (black) PED key(s) for the backup partition

> **NOTE** Use the USB-C adapter in the USB port on the right side of the Luna Backup HSM 7 to insert PED keys:
>
> 

> The following policies are set:

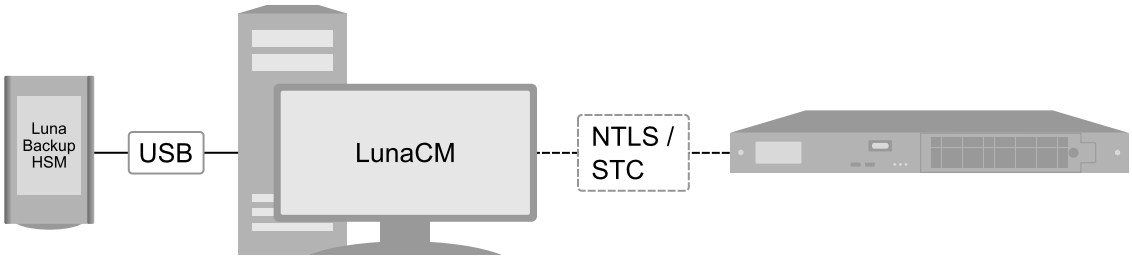
- **HSM policy 16:** Allow network replication must be set to **1** (ON) on the HSM that hosts the user partition.
- [V0 partitions or firmware older than Luna HSM Firmware 7.7.0] **Partition policy 0:** "Allow private key cloning" on page 338 is set to **1** (ON) on the user partition.
- [V0 partitions or firmware older than Luna HSM Firmware 7.7.0] **Partition policy 4:** "Allow secret key cloning" on page 340 is set to **1** (ON) on the user partition.

> **NOTE** HSS (PQC) private keys cannot be cloned, due to inherent restrictions of that key type. This means that backup and HA synchronization fail if HSS private keys are encountered in your application partition.

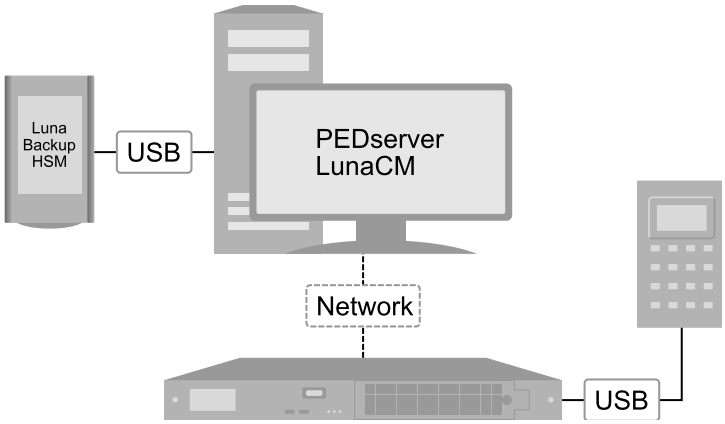
To back up a multifactor quorum-authenticated partition

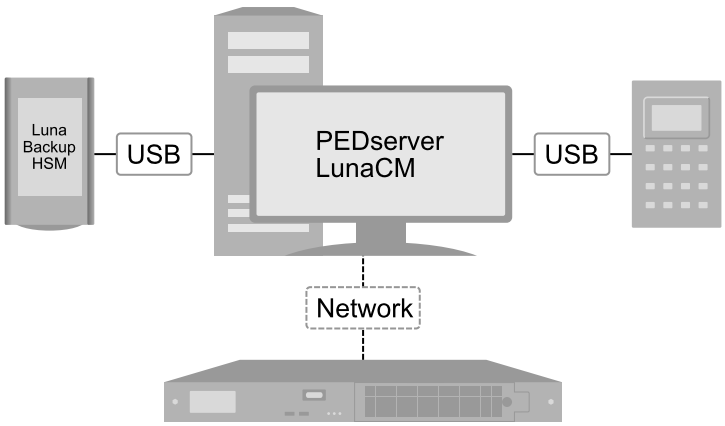
1. Configure your Luna HSM Client workstation using one of the following configurations:

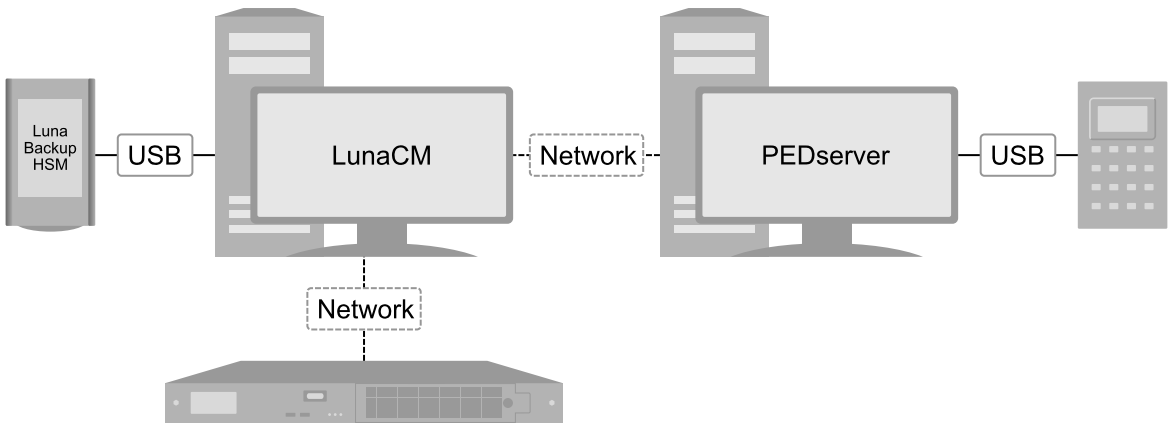
- Activated source partition:



- Non-activated source partition:







a. If you have not already done so, install the required client software on the Luna HSM Client workstation. See "Client Software Required to Perform Backup and Restore Operations" on page 469 for details.

> **NOTE** If you are installing Luna HSM Client on Windows, the driver may not be installed unless the Luna device is connected to the computer first; refer to Backup/USB/PCIe Drivers Not Installed on Windows 10 or Windows Server 2022 Unless Device is Connected.

b. Connect the backup HSM directly to the Luna HSM Client workstation using the included USB cable.

> **NOTE** On most Luna HSM Client computers, the USB port provides adequate power, and connecting the provided power supply is not recommended. Wait and confirm that the HSM boots properly. If the HSM fails to boot up:
>
> **1.** Disconnect the HSM from the USB port.
>
> **2.** Connect the HSM to power using the provided power supply. Wait for it to boot completely.
>
> **3.** Reconnect the HSM to the USB port on the client.
>
> If the HSM is connected to the USB port before the power supply, you may encounter an issue where the HSM occasionally loses contact with the client, and must be power cycled.

c. [Non-activated source partition, Local PED] Connect the PED to the USB port on the Luna Network HSM 7 card, using the PED USB cable.

d. [Non-activated source partition] Connect the PED to the Luna HSM Client workstation used to host the remote PED, using the PED USB cable.

> **NOTE** You connect to the remote PED using the IP address of the workstation used to host the PED. This can be the same workstation that hosts the user and backup partition slots, or a different workstation. The workstation used to host the PED must be running PEDserver.

2. [Non-activated source partition] Start the **pedserver** service on the workstation used to host the remote PED:

Windows	C:\Program Files\Safenet\LunaClient> **"pedserver -mode start" on page 316**
Linux	/usr/safenet/lunaclient> **"pedserver -mode start" on page 316**

3. Launch LunaCM on the workstation that hosts the Luna Network HSM 7 partition slots.

4. Identify the slot assignments for:

   - The Luna Network HSM 7 partition you want to back up.
   - The Luna Backup HSM 7 admin partition (where all backups are stored).

   lunacm:> **slot list**

   If you cannot see both slots, check your connections or configure your client as required.

5. Select the Luna Network HSM 7 partition:

   lunacm:> **slot set -slot** <slot_id>

6. Log in to the partition as Crypto Officer (CO):

- If the partition is activated, use the following command and provide the Crypto Officer (CO) challenge secret as prompted:

  lunacm:> **role login -name co**

- If the partition is not activated:

  i. Connect to the Luna HSM Client workstation that hosts the PED. If defaults are not set using lunacm:> **ped set**, specify an IP address (and port if required; 1503 is default).

  lunacm:> **ped connect** [**-ip** <pedserver_host_ip>]

  ii. Log in to the selected Luna Network HSM 7 partition as the Crypto Officer (CO):

  lunacm:> **role login -name co**

  iii. Respond to the prompts on the PED to provide the orange (PED vector) PED key(s) and PIN for the Luna Network HSM 7 and the black (CO) key(s) and PIN for the CO role on the application partition.

  iv. Disconnect the remote PED session. Note that you will remain logged in to the Luna Network HSM 7 partition:

  lunacm:> **ped disconnect**

7. Initiate the backup:

lunacm:> **partition archive backup -slot** <backup_HSM_admin_slot> [**-partition** <target_backup_label>] [**-append**] [**-replace**] [**-smkonly**]

If you omit the **-partition** option when creating a new backup, the partition is assigned a default name (<source_partition_name>_<YYYYMMDD>) based on the source HSM's internally-set time and date.

If you are backing up a V1 partition, include **-smkonly** to back up the SMK only. By default, the SMK and any encrypted cryptographic material on the partition are backed up.

The backup begins once you have completed the authentication process. Objects are backed up one at a time. If you are backing up to an existing backup partition, you can use the following options to define how individual objects are backed up:

-append	Add only new objects to an existing backup.
-replace	Delete the existing objects in a target backup partition and replace them with the contents of the source user partition. This is the default.
-append -replace	Add new objects and replace existing objects that have the same OUID but a different fingerprint (such as would occur if any of the object attributes were changed since the previous backup).

> **NOTE** If the backup operation is interrupted (if the Backup HSM is unplugged, or if you fail to respond to PED prompts, for example), the Backup HSM's full available space can become occupied with a single backup partition. If this occurs, delete the backup partition with lunacm:> **partition archive delete** before reattempting the backup operation.

8. Respond to the prompts on the Luna Backup HSM 7 touchscreen to insert the following PED keys:

If you are creating a new backup partition:

i. The blue HSM SO PED key(s) for the backup HSM.

ii. You are prompted to initialize the backup Partition SO role by creating a new blue PED key or reusing an existing key. After you initialize the role, you are prompted to insert the key again to log in as Partition SO.

iii. The red Domain PED key(s). This must be the same PED key(s) used for the Luna Network HSM 7 partition, otherwise the backup will fail.

iv. The blue Partition SO PED key(s) for the backup partition, to log in again.

v. You are prompted to initialize the Crypto Officer role for the backup by creating a new black PED key or reusing an existing key. After you initialize the role, you are prompted to insert the key again to log in as Crypto Officer.

If you are backing up to an existing backup partition whose domain matches the source partition:

i. The blue HSM SO PED key(s) for the backup HSM.

ii. The blue Partition SO PED key(s) for the backup.

iii. The black Crypto Officer PED key(s) for the backup.

Restoring To a Multifactor Quorum-Authenticated Partition

You can restore the objects from a multifactor quorum-authenticated backup partition to the same partition that was originally backed up, or to another partition that has been initialized with the same domain (red PED key).

Prerequisites

> The target partition must be initialized using the same domain (red PED key) as the backup partition, the Crypto Officer role must be initialized and the CO role credential changed from its initial value.

> You require the Crypto Officer challenge secret for the target partition.

If the target partition is not activated, you also require:

- [Remote PED authentication] The Remote PED Vector (orange) PED key(s) for the target HSM
- The Crypto Officer (black) PED key(s) for the target partition

> **TIP** If the target partition is activated, only the Crypto Officer's challenge secret is required. To simplify the backup process and minimize interactions with the PED, it is recommended that you activate the CO role on the user partitions you want to restore from backup. See "Activation on Multifactor Quorum-Authenticated Partitions" on page 373 for more information.

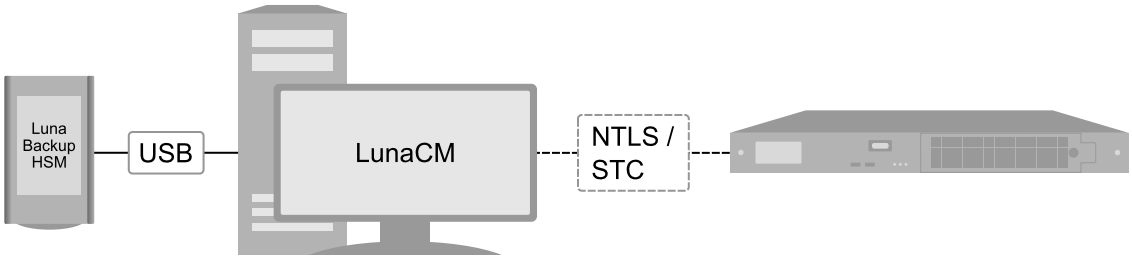
> The following policies are set:

- **HSM policy 16:** Allow network replication must be set to **1** (ON) on the HSM that hosts the user partition you want to restore to.
- [V0 partitions or firmware older than Luna HSM Firmware 7.7.0] **Partition policy 0:** "Allow private key cloning" on page 338 is set to **1** (ON) on the user partition you want to restore to.
- [V0 partitions or firmware older than Luna HSM Firmware 7.7.0] **Partition policy 4:** "Allow secret key cloning" on page 340 is set to **1** (ON) on the user partition you want to restore to.

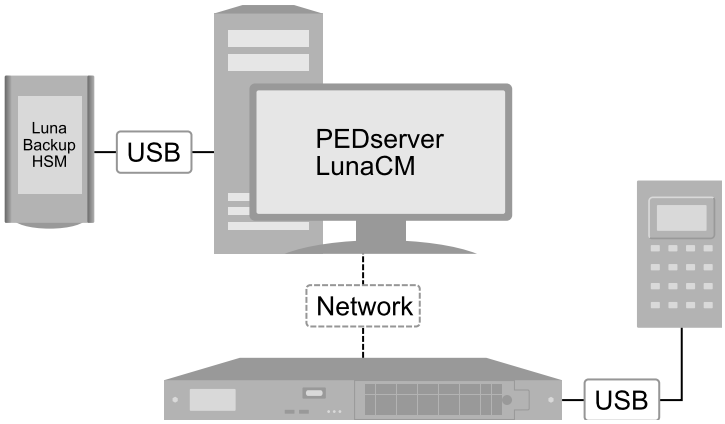
To restore a multifactor quorum-authenticated partition

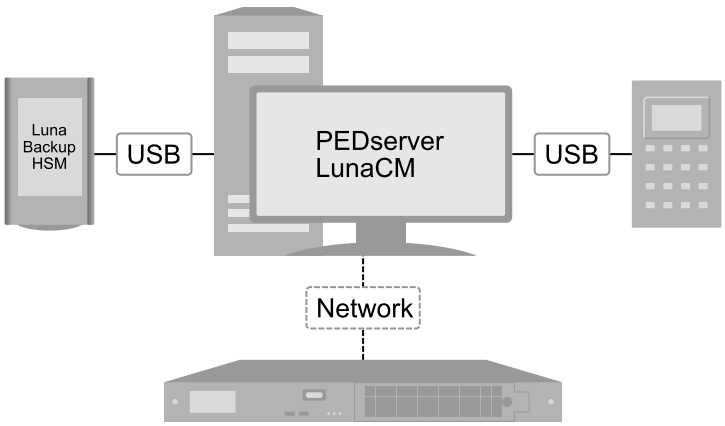
1. Configure your Luna HSM Client workstation using one of the following configurations:

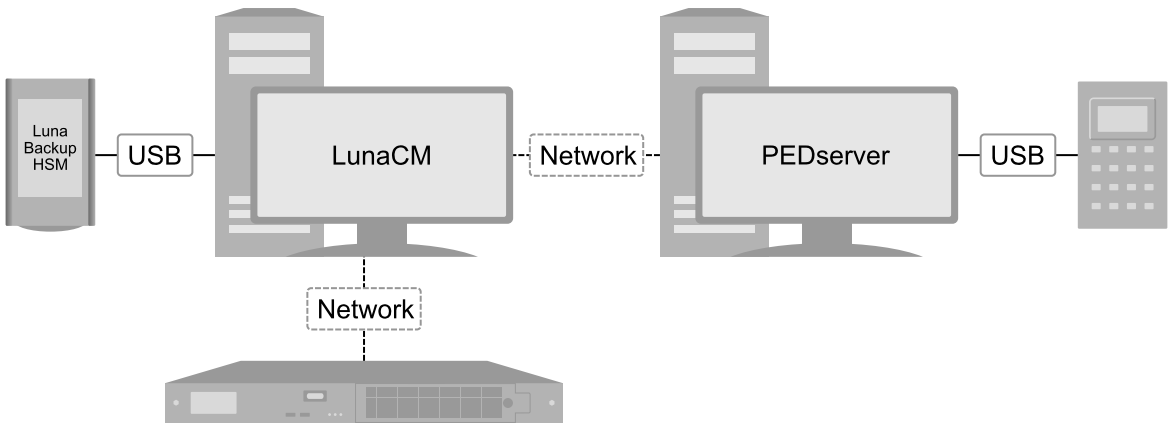
- Activated destination partition:



- Non-activated destination partition:







a. If you have not done so already, install the required client software on the Luna HSM Client workstation. See "Luna HSM Client Software Installation" on page 20 for details.

> **NOTE** If you are installing Luna HSM Client on Windows, the driver may not be installed unless the Luna device is connected to the computer first; refer to Backup/USB/PCIe Drivers Not Installed on Windows 10 or Windows Server 2022 Unless Device is Connected.

b. Connect the backup HSM directly to the Luna HSM Client workstation using the included USB cable.

> **NOTE** On most Luna HSM Client computers, the USB port provides adequate power, and connecting the provided power supply is not recommended. Wait and confirm that the HSM boots properly. If the HSM fails to boot up:
>
> **1.** Disconnect the HSM from the USB port.
>
> **2.** Connect the HSM to power using the provided power supply. Wait for it to boot completely.
>
> **3.** Reconnect the HSM to the USB port on the client.
>
> If the HSM is connected to the USB port before the power supply, you may encounter an issue where the HSM occasionally loses contact with the client, and must be power cycled.

c. [Non-activated source partition] Connect the PED to the Luna HSM Client workstation used to host the remote PED, using the PED USB cable.

> **NOTE** You connect to the remote PED using the IP address of the workstation used to host the PED. This can be the same workstation that hosts the user and backup partition slots, or a different workstation. The workstation used to host the PED must be running PEDserver.

2. [Non-activated source partition] Start the **pedserver** service on the workstation used to host the remote PED:

Windows	C:\Program Files\Safenet\LunaClient> **"pedserver -mode start" on page 316**
Linux	/usr/safenet/lunaclient> **"pedserver -mode start" on page 316**

3. Launch LunaCM on the workstation that hosts the Luna Network HSM 7 and backup partition slots.
4. Identify the slot assignments for:
   - the Luna Network HSM 7 partition you want to restore to.
   - the backup HSM admin partition (where all backups are stored).

   lunacm:> **slot list**

   If you cannot see both slots, check your connections or configure your client as required.
5. Select the Luna Network HSM 7 partition you want to restore from backup:

   lunacm:> **slot set -slot** <slot_id>
6. Log in to the partition as Crypto Officer (CO):
   - If the partition is activated, use the following command and provide the Crypto Officer (CO) challenge secret as prompted:

     lunacm:> **role login -name co**

- If the partition is not activated:

  i. Connect to the Luna HSM Client workstation that hosts the PED. If defaults are not set using lunacm:> **ped set**, specify an IP address (and port if required; 1503 is default).

  lunacm:> **ped connect** [**-ip** <pedserver_host_ip>]

  ii. Log in to the selected Luna Network HSM 7 partition as the Crypto Officer (CO):

  lunacm:> **role login -name co**

  iii. Respond to the prompts on the PED to provide the orange (PED vector) PED key(s) and PIN for the Luna Network HSM 7 and the black (CO) key(s) and PIN for the CO role on the application partition.

  iv. Disconnect the remote PED session. Note that you will remain logged in to the Luna Network HSM 7 partition:

  lunacm:> **ped disconnect**

7. List the available backups on the Backup HSM by specifying the Backup HSM's slot number. You will require the backup partition label to perform the restore operation.

   lunacm:> **partition archive list-slot** <backup_HSM_admin_slot>

8. Initiate the restore operation. Respond to the prompts on the Luna Backup HSM 7 touchscreen to insert the required PED keys.

   lunacm:> **partition archive restore -slot** <backup_HSM_admin_slot> **-partition** <backup_partition_label> [**-smkonly**]

   CAUTION! The **-replace** option is deprecated and has been removed in Luna HSM Client 10.7.0 and newer. If you wish to restore an earlier version of an object, Thales recommends deleting the object(s) manually before restoring the partition from backup.

   Ensure that the target partition can receive objects from the backup HSM before deleting objects or using partition archive restore with the **-replace** option; the cloning protocol may prevent objects from being restored, even if LunaCM states that `X objects will be restored`. This may occur if **HSM policy 55: Enable Restricted Restore** was enabled on the Luna Backup HSM 7 since the original backup was taken. If your partition is on an HSM with firmware older than Luna HSM Firmware 7.7.0, you must update to 7.7.0 or newer to restore objects from this backup.

   NOTE If you are restoring a V1 backup to a V1 partition, include **-smkonly** to restore the SMK only (see "V0 and V1 Partitions" on page 148 for more information). By default, the SMK and any cryptographic material on the backup are restored.

   The restore operation begins once you have completed the authentication process. Objects are restored one at a time.

Luna Backup HSM 7 Connected to Luna HSM Client Using Remote Multifactor Quorum Authentication

In this configuration, you connect the Luna Backup HSM 7 to a USB port on the Luna HSM Client, and insert PED keys into a Remote Luna PED. This allows you to perform backup/restore operations for all application partitions that can be accessed by the client. You can restore a partition backup to the original source partition or to another existing Luna application partition that shares the same cloning domain. To use this method, you require:

- "Luna Backup HSM 7" on page 474 **v1 or v2**
- Luna HSM Client 10.1.0 or newer

This section provides instructions for the following procedures:

- "Initializing the Luna Backup HSM 7 for Multifactor Quorum Authentication" below
- "Configuring the Luna Backup HSM 7 for FIPS Compliance" on page 523
- "Backing Up a Multifactor Quorum-Authenticated Partition" on page 523
- "Restoring To a Multifactor Quorum-Authenticated Partition" on page 528

Initializing the Luna Backup HSM 7 for Multifactor Quorum Authentication

You must initialize the Luna Backup HSM 7 prior to first use. You can initialize the backup HSM by connecting it to a Luna HSM Client and using LunaCM commands to perform the initialization.

Prerequisites

You will need the following PED keys:

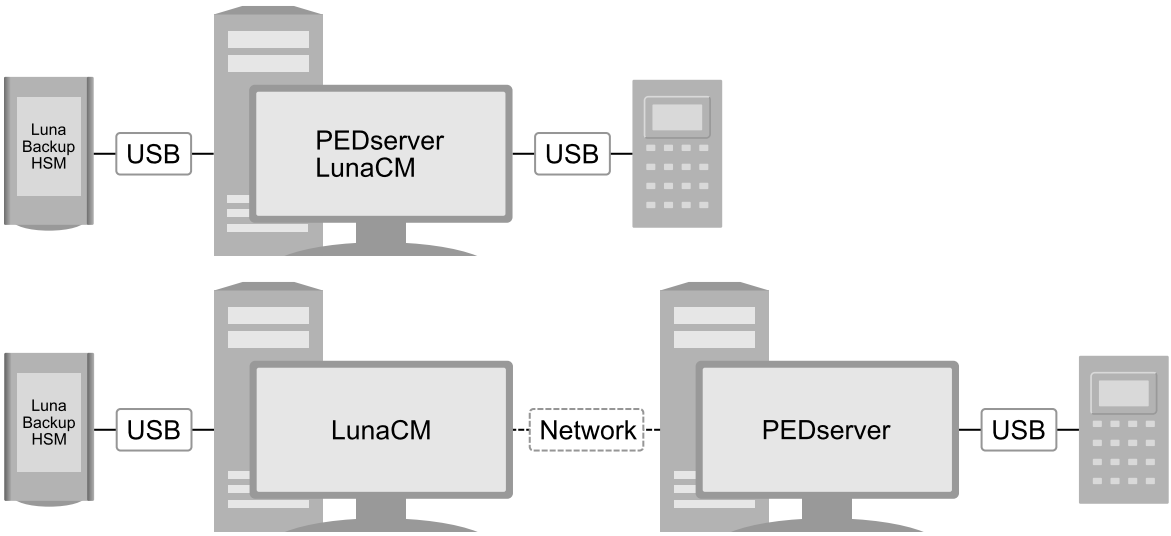
- A blank orange (PED vector) PED key, plus the number required to create duplicate PED keys as necessary.

> **CAUTION!** Always make copies of your orange PED keys, or declare MofN as one-of-several, and store at least one safely. For the Luna Backup HSM 7 v1, *the orange PED key is as important as the HSM SO blue key or the Domain red key*.
>
> The orange PED key is required for all Luna Backup HSM 7 v1 operations. If this key is lost, your backups will become irretrievable. Thales recommends keeping multiple backups of all PED keys stored in a secure location.

- N number of blue (HSM SO) PED keys, as defined by the M of N scheme you choose for the HSM SO role, plus the number required to create duplicate PED keys as necessary.
- Blank or reused red (Domain) PED key(s)
- [Luna Backup HSM 7 Firmware 7.7.1 and newer only] Set the value of **-pedwritedelay** to **2000** to avoid experiencing frequent CKR_CALLBACK_ERRORS, which will prevent you from completing the procedure below. For more information about this error, refer to "Intermittent CKR_CALLBACK_ERROR: PED Cannot Service its USB Data Channel Fast Enough to Communicate with PEDserver" on page 276.

To initialize a Luna Backup HSM 7 for multifactor quorum authentication

1. Configure your Luna HSM Client workstation using one of the following configurations:



a. Install the required client software on the Luna HSM Client workstation. See "Client Software Required to Perform Backup and Restore Operations" on page 469 for details.

> **NOTE** If you are installing Luna HSM Client on Windows, the driver may not be installed unless the Luna device is connected to the computer first; refer to Backup/USB/PCIe Drivers Not Installed on Windows 10 or Windows Server 2022 Unless Device is Connected.

b. Connect the backup HSM directly to the Luna HSM Client workstation using the included USB cable.

> **NOTE** On most Luna HSM Client computers, the USB port provides adequate power, and connecting the provided power supply is not recommended. Wait and confirm that the HSM boots properly. If the HSM fails to boot up:
>
> **1.** Disconnect the HSM from the USB port.
>
> **2.** Connect the HSM to power using the provided power supply. Wait for it to boot completely.
>
> **3.** Reconnect the HSM to the USB port on the client.
>
> If the HSM is connected to the USB port before the power supply, you may encounter an issue where the HSM occasionally loses contact with the client, and must be power cycled.

c. Connect the Luna PED to the Luna HSM Client workstation used to host the remote PED, using the PED USB cable.

> **NOTE** You connect to the remote PED using the IP address of the workstation used to host the PED. This can be the same workstation that hosts the user and backup partition slots, or a different workstation. The workstation used to host the PED must be running pedServer.

2. Start the **pedserver** service on the workstation used to host the remote PED:

Windows	C:\Program Files\Safenet\LunaClient> **"pedserver -mode start" on page 316**
Linux	/usr/safenet/lunaclient> **"pedserver -mode start" on page 316**

3. Launch LunaCM on the workstation that hosts the user and backup partition slots.

4. Select the slot assigned to the backup HSM Admin partition.

   lunacm:> **slot set -slot** <slot_id>

5. If necessary, recover the HSM from Secure Transport Mode. See Secure Transport Mode for more information:

   lunacm:> **stm recover -randomuserstring** <string>

   > **NOTE** Recovering a Luna Backup HSM 7 from secure transport mode may take up to three minutes.

6. Connect to the Luna HSM Client workstation that hosts the PED. If defaults are not set using lunacm:> **ped set**, specify an IP address (and port if required; 1503 is default).

   lunacm:> **ped connect -ip** <ip_address> **-pwd**

   LunaCM generates and displays a one-time password that is used to set up a secure channel between the backup HSM and the PED, allowing you to securely initialize the orange (Remote PED Vector) PED key. Enter the displayed password on the PED when prompted to complete setup of the secure channel.

7. Create an orange (Remote PED vector) PED key for the backup HSM. The PED vector key is required for subsequent multifactor quorum-authenticated sessions to the HSM. Ensure that you label any new PED keys that you create during this process.

   lunacm:> **ped vector init**

   > **CAUTION!** The orange PED key is required for all Luna Backup HSM 7 v1 operations. If this key is lost, your backups will become irretrievable. Thales recommends keeping multiple backups of all PED keys stored in a secure location.

8. Tear down the one-time, password-protected secure channel between the backup HSM and the PED you used to create the orange (Remote PED vector) PED key.

   lunacm:> **ped disconnect**

   You are prompted to enter the one-time password that was generated when you performed **ped connect**. Enter the password and press Enter to proceed.

9. Set up a new secure channel between the backup HSM and the PED. If defaults are not set using lunacm:> **ped set**, specify an IP address (and port if required; 1503 is default). You are prompted to insert the orange PED key you created in step 7.

   lunacm:> **ped connect**

10. Initialize the selected backup HSM in multifactor quorum-authenticated mode. You are prompted by the PED for the red Domain PED key(s) and blue HSM SO PED key(s). Respond to the PED prompts and insert and set the PINs on the required keys when requested. Ensure that you label any new PED keys that you create during this process.

    lunacm:> **hsm init -iped -label** <label>

11. Use the **Duplicate** function on the PED to create and label duplicates of the new PED keys, as required. See "Duplicating Existing PED keys" on page 298 for details.

12. Disconnect the PED when done.

lunacm:> **ped disconnect**

Configuring the Luna Backup HSM 7 for FIPS Compliance

Luna Backup HSM Firmware 7.7.1 and newer uses the same updated cloning protocol as Luna HSM Firmware 7.7.0 and newer. For the Luna Backup HSM 7 to be FIPS-compliant, it must restrict restore operations to application partitions that use the new protocol. This restriction is applied by setting **HSM policy 55:** Enable Restricted Restore to **1** on the backup HSM. The Luna Backup HSM 7 must be initialized and connected to a Luna HSM Client computer to set this policy.

When this policy is enabled on the Luna Backup HSM 7, objects that have been backed up from partitions using firmware older than Luna HSM Firmware 7.7.0 can be restored to Luna HSM Firmware 7.7.0 or newer (V0 or V1) partitions only.

> **CAUTION!** FIPS compliance requires that objects are never cloned or restored to an HSM using less secure firmware, and this includes restoring from Luna Backup HSM 7 firmware.
>
> If you have backups already stored on the Luna Backup HSM 7 that were taken from pre-7.7.0 partitions, turning this policy ON will prevent you from restoring them to the same source partition. You must update the HSM containing the source partition to Luna HSM Firmware 7.7.0 or newer before restoring from backup.

> **NOTE** **HSM policy 12:** Allow non-FIPS algorithms, which is used to set FIPS-compliant mode on other Luna HSMs, does not apply to the Luna Backup HSM 7. Attempts to change this policy will fail with the error `CKR_CANCEL`.

To configure the Luna Backup HSM 7 for FIPS compliance

1. On the Luna HSM Client computer, run LunaCM.
2. Set the active slot to the Luna Backup HSM 7.

   lunacm:> **slot set -slot** <slot_id>
3. Log in as Backup HSM SO.

   lunacm:> **role login -name so**
4. Set **HSM policy 55:** Enable Restricted Restore to **1**.

   lunacm:> **hsm changehsmpolicy -policy 55 -value 1**
5. [Optional] Check that the Luna Backup HSM 7 is now in FIPS approved operation mode.

   lunacm:> **hsm showinfo**

   ```
   *** The HSM is in FIPS 140-2 approved operation mode. ***
   ```

Backing Up a Multifactor Quorum-Authenticated Partition

Backups are created and stored as partitions within the Admin partition on the Luna Backup HSM 7. A new backup partition is created on initial backup. For subsequent backups, you can choose to replace the contents of the existing backup partition with the current source partition objects, or add new objects in the source partition to the existing backup partition. Like all cloning operations, the source and target backup partitions must be initialized with the same domain.

Prerequisites

> You have the required credentials:

If you are creating a new backup partition:

- The Remote PED Vector (orange) PED key(s) for the Backup HSM
- New or reused Partition SO (blue) PED key(s) to initialize the backup partition
- New or reused Crypto Officer (black) PED key(s) to initialize the CO role on the backup partition
- The Domain (red) PED key(s) for the source partition, to initialize the domain on the backup

If you are backing up to an existing backup partition whose domain matches the source partition:

- The Remote PED Vector (orange) PED key(s) for the Backup HSM
- The existing Partition SO (blue) PED key(s) for the backup partition, to log in
- The existing Crypto Officer (black) PED key(s) for the backup partition

> The following policies are set:

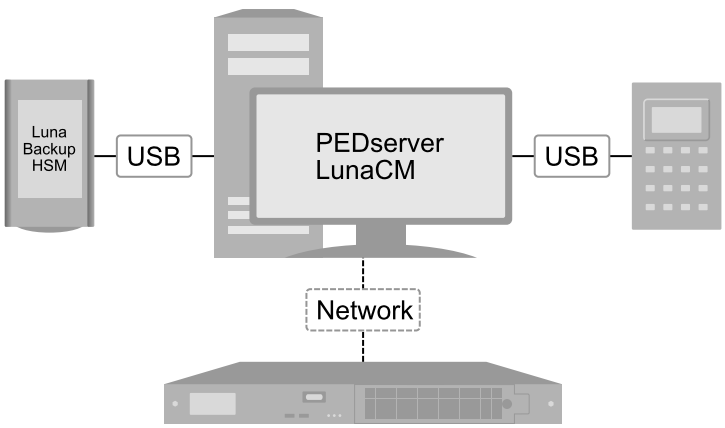
- **HSM policy 16:** Allow network replication must be set to **1** (ON) on the HSM that hosts the user partition.
- [V0 partitions or firmware older than Luna HSM Firmware 7.7.0] **Partition policy 0:** "Allow private key cloning" on page 338 is set to **1** (ON) on the user partition.
- [V0 partitions or firmware older than Luna HSM Firmware 7.7.0] **Partition policy 4:** "Allow secret key cloning" on page 340 is set to **1** (ON) on the user partition.

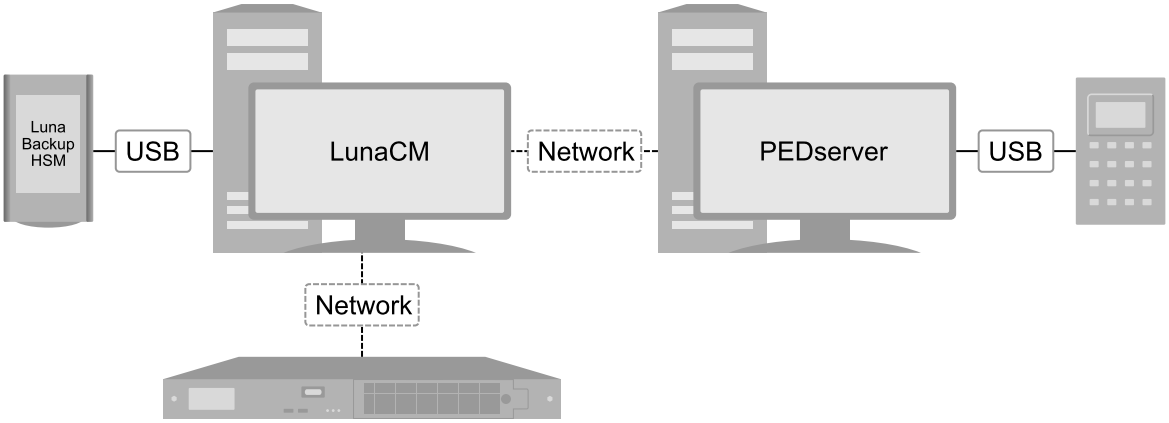
> [Luna Backup HSM 7 Firmware 7.7.1 and newer only] Set the value of **-pedwritedelay** to **2000** to avoid experiencing frequent CKR_CALLBACK_ERRORS, which will prevent you from completing the procedure below. For more information about this error, refer to "Intermittent CKR_CALLBACK_ERROR: PED Cannot Service its USB Data Channel Fast Enough to Communicate with PEDserver" on page 276.

> **NOTE** HSS (PQC) private keys cannot be cloned, due to inherent restrictions of that key type. This means that backup and HA synchronization fail if HSS private keys are encountered in your application partition.

To back up a multifactor quorum-authenticated partition

1. Configure your Luna HSM Client workstation using one of the following configurations:





a. Install the required client software on the Luna HSM Client workstation. See "Client Software Required to Perform Backup and Restore Operations" on page 469 for details.

> **NOTE** If you are installing Luna HSM Client on Windows, the driver may not be installed unless the Luna device is connected to the computer first; refer to Backup/USB/PCIe Drivers Not Installed on Windows 10 or Windows Server 2022 Unless Device is Connected.

b. Connect the backup HSM directly to the Luna HSM Client workstation using the included USB cable.

> **NOTE** On most Luna HSM Client computers, the USB port provides adequate power, and connecting the provided power supply is not recommended. Wait and confirm that the HSM boots properly. If the HSM fails to boot up:
>
> **1.** Disconnect the HSM from the USB port.
>
> **2.** Connect the HSM to power using the provided power supply. Wait for it to boot completely.
>
> **3.** Reconnect the HSM to the USB port on the client.
>
> If the HSM is connected to the USB port before the power supply, you may encounter an issue where the HSM occasionally loses contact with the client, and must be power cycled.

c. Connect the PED to the Luna HSM Client workstation used to host the remote PED, using the PED USB cable.

> **NOTE** You connect to the remote PED using the IP address of the workstation used to host the PED. This can be the same workstation that hosts the user and backup partition slots, or a different workstation. The workstation used to host the PED must be running PEDserver.

2. Start the **pedserver** service on the workstation used to host the remote PED:

Windows	C:\Program Files\Safenet\LunaClient> **"pedserver -mode start" on page 316**
Linux	/usr/safenet/lunaclient> **"pedserver -mode start" on page 316**

3. Launch LunaCM on the workstation that hosts the Luna Network HSM 7 partition slots.

4. Identify the slot assignments for:

- The Luna Network HSM 7 partition you want to backup.

- The Luna Backup HSM 7 admin partition (where all backups are stored).

lunacm:> **slot list**

If you cannot see both slots, check your connections or configure your client as required.

5. Select the Luna Network HSM 7 partition:

   lunacm:> **slot set -slot** <slot_id>

6. Log in to the partition as Crypto Officer (CO):
   - If the partition is activated, use the following command and present the black Crypto Officer PED key(s) to the Luna Network HSM 7 as directed:

     lunacm:> **role login -name co**

   - If the partition is not activated:
     i. Connect to the Luna HSM Client workstation that hosts the PED. If defaults are not set using lunacm:> **ped set**, specify an IP address (and port if required; 1503 is default).

        lunacm:> **ped connect** [**-ip** <pedserver_host_ip>]

     ii. Log in to the selected Luna Network HSM 7 partition as the Crypto Officer (CO):

        lunacm:> **role login -name co**

     iii. Respond to the prompts on the PED to provide the orange (PED vector) PED key(s) and PIN for the Luna Network HSM 7 and the black (CO) key(s) and PIN for the CO role on the application partition.

     iv. Disconnect the remote PED session. Note that you will remain logged in to the Luna Network HSM 7 partition:

        lunacm:> **ped disconnect**

7. Select the backup HSM Admin partition:

   lunacm:> **slot set -slot** <slot_id>

8. Connect to the Luna HSM Client workstation that hosts the PED. If defaults are not set using lunacm:> **ped set**, specify an IP address (and port if required; 1503 is default):

   lunacm:> **ped connect** [**-ip** <pedserver_host_ip>]

9. Select the Luna Network HSM 7 partition:

   lunacm:> **slot set -slot** <slot_id>

10. Initiate the backup:

    lunacm:> **partition archive backup -slot** <backup_HSM_admin_slot> [**-partition** <target_backup_label>] [**-append**] [**-replace**] [**-smkonly**]

    If you omit the **-partition** option when creating a new backup, the partition is assigned a default name (<source_partition_name>_<YYYYMMDD>) based on the source HSM's internally-set time and date.

    If you are backing up a V1 partition, include **-smkonly** to back up the SMK only. By default, the SMK and any encrypted cryptographic material on the partition are backed up.

    The backup begins once you have completed the authentication process. Objects are backed up one at a time. For existing backups, you can use the following options to define how individual objects are backed up:

-append	Add only new objects to an existing backup.
-replace	Delete the existing objects in a target backup partition and replace them with the contents of the source user partition. This is the default.
-append -replace	Add new objects and replace existing objects that have the same OUID but a different fingerprint (such as would occur if any of the object attributes were changed since the previous backup).

> **NOTE** If the backup operation is interrupted (if the Backup HSM is unplugged, or if you fail to respond to PED prompts, for example), the Backup HSM's full available space can become occupied with a single backup partition. If this occurs, delete the backup partition with lunacm:> **partition archive delete** before reattempting the backup operation.

11. You are prompted for the following credentials in the following order:Respond to the prompts on the Luna PED to insert the following PED keys:

If you are creating a new backup partition:

- **i.** The blue HSM SO PED key(s) for the backup HSM.
- **ii.** You are prompted to initialize the backup Partition SO role by creating a new blue PED key or reusing an existing key. After you initialize the role, you are prompted to insert the key again to log in as Partition SO.
- **iii.** The red Domain PED key(s). This must be the same PED key(s) used for the Luna Network HSM 7 partition, otherwise the backup will fail.
- **iv.** The blue Partition SO PED key(s) for the backup partition, to log in again.
- **v.** You are prompted to initialize the Crypto Officer role for the backup by creating a new black PED key or reusing an existing key. After you initialize the role, you are prompted to insert the key again to log in as Crypto Officer.

If you are backing up to an existing backup partition whose domain matches the source partition:

- **i.** The blue HSM SO PED key(s) for the backup HSM.
- **ii.** The blue Partition SO PED key(s) for the backup.
- **iii.** The black Crypto Officer PED key(s) for the backup.

12. Disconnect the PED from the Luna Network HSM 7 and Luna Backup HSM 7:

a. Disconnect the PED from the backup HSM:

lunacm:> **ped disconnect**

b. Select the slot for the Luna Network HSM 7 partition:

lunacm:> **slot set -slot** <slot_id>

c. Disconnect the PED from the Luna Network HSM 7 partition:

lunacm:> **ped disconnect**

13. If this is the first backup to the backup partition, use the **Duplicate** function on the PED to create and label a set of backup keys for the new backup partition PO (blue) and CO (black) PED keys. See "Duplicating Existing PED keys" on page 298 for details.

Restoring To a Multifactor Quorum-Authenticated Partition

You can restore the objects from a multifactor quorum-authenticated backup partition to the same partition that was originally backed up, or to another partition that has been initialized with the same domain (red PED key).

Prerequisites

> The target partition must be initialized using the same domain (red PED key) as the backup partition, the Crypto Officer role must be initialized and the CO role credential changed from its initial value.

> You have the required credentials:

- The Remote PED Vector (orange) PED key(s) for the backup HSM
- The Crypto Officer challenge secret for the target partition
- The Crypto Officer (black) PED key(s) for the backup partition

> **TIP** If the target partition is activated, only the Crypto Officer's challenge secret is required. To simplify the backup process and minimize interactions with the PED, it is recommended that you activate the CO role on the user partitions you want to restore from backup. See "Activation on Multifactor Quorum-Authenticated Partitions" on page 373 for more information.

If the target partition is not activated, you also need:

- The Remote PED Vector (orange) PED key(s) for the target HSM
- The Crypto Officer (black) PED key(s) for the target partition

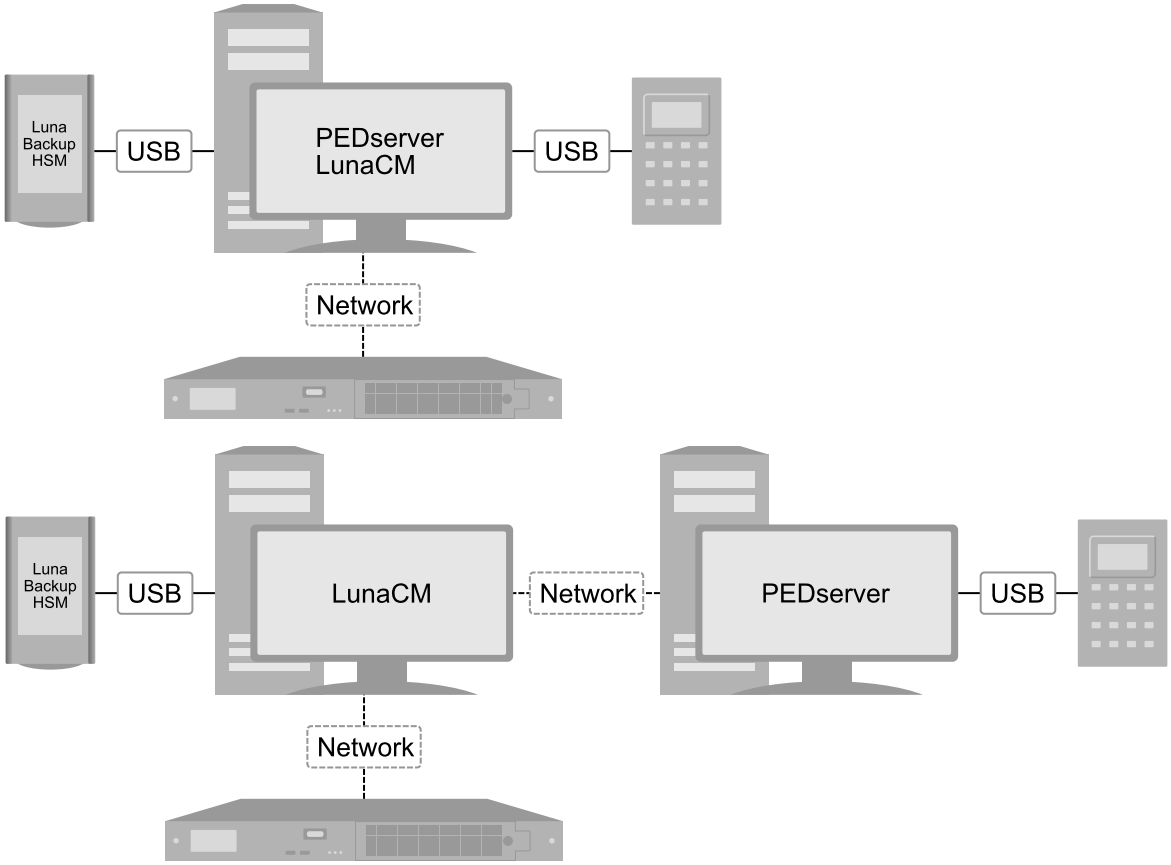
> The following policies are set:

- **HSM policy 16:** Allow network replication must be set to **1** (ON) on the HSM that hosts the user partition you want to restore to.
- [V0 partitions] **Partition policy 0:** "Allow private key cloning" on page 338 is set to **1** (ON) on the user partition you want to restore to.
- [V0 partitions] **Partition policy 4:** "Allow secret key cloning" on page 340 is set to **1** (ON) on the user partition you want to restore to.

> [Luna Backup HSM 7 Firmware 7.7.1 and newer only] Set the value of **-pedwritedelay** to **2000** to avoid experiencing frequent CKR_CALLBACK_ERRORS, which will prevent you from completing the procedure below. For more information about this error, refer to "Intermittent CKR_CALLBACK_ERROR: PED Cannot Service its USB Data Channel Fast Enough to Communicate with PEDserver" on page 276.

To restore a multifactor quorum-authenticated partition

1. Configure your Luna HSM Client workstation using one of the following configurations:



a. Install the required client software on the Luna HSM Client workstation. See "Luna HSM Client Software Installation" on page 20 for details.

> **NOTE** If you are installing Luna HSM Client on Windows, the driver may not be installed unless the Luna device is connected to the computer first; refer to Backup/USB/PCIe Drivers Not Installed on Windows 10 or Windows Server 2022 Unless Device is Connected.

b. Connect the backup HSM directly to the Luna HSM Client workstation using the included USB cable.

> **NOTE** On most Luna HSM Client computers, the USB port provides adequate power, and connecting the provided power supply is not recommended. Wait and confirm that the HSM boots properly. If the HSM fails to boot up:
>
> **1.** Disconnect the HSM from the USB port.
>
> **2.** Connect the HSM to power using the provided power supply. Wait for it to boot completely.
>
> **3.** Reconnect the HSM to the USB port on the client.
>
> If the HSM is connected to the USB port before the power supply, you may encounter an issue where the HSM occasionally loses contact with the client, and must be power cycled.

c. Connect the PED to the Luna HSM Client workstation used to host the remote PED, using the PED USB cable.

> **NOTE** You connect to the remote PED using the IP address of the workstation used to host the PED. This can be the same workstation that hosts the user and backup partition slots, or a different workstation. The workstation used to host the PED must be running **pedserver**.

2. Start the **pedserver** service on the workstation used to host the remote PED:

Windows	C:\Program Files\Safenet\LunaClient> **"pedserver -mode start" on page 316**
Linux	/usr/safenet/lunaclient> **"pedserver -mode start" on page 316**

3. Launch LunaCM on the workstation that hosts the Luna Network HSM 7 and backup partition slots.
4. Identify the slot assignments for:
   - the Luna Network HSM 7 partition you want to restore to.
   - the backup HSM admin partition (where all backups are stored).

   lunacm:> **slot list**

   If you cannot see both slots, check your connections or configure your client as required.
5. Select the Luna Network HSM 7 partition you want to restore from backup:

   lunacm:> **slot set -slot** <slot_id>
6. Log in to the partition as Crypto Officer (CO):
   - If the partition is activated, use the following command and enter the Crypto Officer challenge secret:

     lunacm:> **role login -name co**
   - If the partition is not activated:
     i. Connect to the Luna HSM Client workstation that hosts the PED. If defaults are not set using lunacm:> **ped set**, specify an IP address (and port if required; 1503 is default).

        lunacm:> **ped connect** [**-ip** <pedserver_host_ip>]

     ii. Log in to the selected Luna Network HSM 7 partition as the Crypto Officer (CO):

        lunacm:> **role login -name co**

     iii. Respond to the prompts on the PED to provide the orange (PED vector) PED key(s) and PIN for the Luna Network HSM 7 and the black (CO) key(s) and PIN for the CO role on the application partition.

     iv. Disconnect the remote PED session. Note that you will remain logged in to the Luna Network HSM 7 partition:

        lunacm:> **ped disconnect**
7. Connect the PED to the backup HSM. If defaults are not set using lunacm:> **ped set**, specify an IP address (and port if required; 1503 is default):

   lunacm:> **ped connect** [**-ip** <pedserver_host_ip>]
8. List the available backups on the Backup HSM by specifying the Backup HSM's slot number. You will require the backup partition label to perform the restore operation.

   lunacm:> **partition archive list-slot** <backup_HSM_admin_slot>
9. Initiate the restore operation. Respond to the prompts on the PED to insert the required PED keys.

lunacm:> **partition archive restore** **-slot** <backup_HSM_admin_slot> **-partition** <backup_partition_label> [**-smkonly**]

> **CAUTION!** The **-replace** option is deprecated and has been removed in Luna HSM Client 10.7.0 and newer. If you wish to restore an earlier version of an object, Thales recommends deleting the object(s) manually before restoring the partition from backup.
>
> Ensure that the target partition can receive objects from the backup HSM before deleting objects or using partition archive restore with the **-replace** option; the cloning protocol may prevent objects from being restored, even if LunaCM states that `X objects will be restored`. This may occur if **HSM policy 55: Enable Restricted Restore** was enabled on the Luna Backup HSM 7 since the original backup was taken. If your partition is on an HSM with firmware older than Luna HSM Firmware 7.7.0, you must update to 7.7.0 or newer to restore objects from this backup.

> **NOTE** If you are restoring a V1 backup to a V1 partition, include **-smkonly** to restore the SMK only (see "V0 and V1 Partitions" on page 148 for more information). By default, the SMK and any cryptographic material on the backup are restored.

The restore operation begins once you have completed the authentication process. Objects are restored one at a time.

Luna Backup HSM 7 Connected to Luna HSM Client Using Password Authentication

In this configuration, you connect the Luna Backup HSM 7 to a USB port on the Luna HSM Client, and enter passwords in LunaCM. This configuration allows you to perform backup/restore operations for all application partitions that can be accessed by the client. You can restore a partition backup to the original source partition or to another existing Luna application partition that shares the same cloning domain. To use this method, you require:

- "Luna Backup HSM 7" on page 474 **v1 or v2**
- Luna HSM Client 10.1.0 or newer

This section provides instructions for the following procedures:

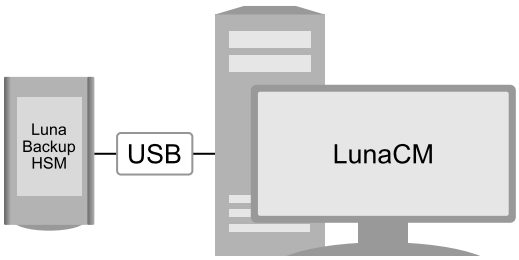
- "Initializing the Luna Backup HSM 7 for Password Authentication" below
- "Configuring the Luna Backup HSM 7 for FIPS Compliance" on page 533
- "Backing Up a Password-Authenticated Partition" on page 534
- "Restoring to a Password-Authenticated Partition" on page 536

Initializing the Luna Backup HSM 7 for Password Authentication

You must initialize the Luna Backup HSM 7 prior to first use. You can initialize the backup HSM by connecting it to a Luna HSM Client and using LunaCM commands to perform the initialization.

To initialize a Luna Backup HSM 7 for password authentication

1. Configure your Luna HSM Client workstation as illustrated below:



   a. Install the required client software on the Luna HSM Client workstation. See "Client Software Required to Perform Backup and Restore Operations" on page 469 for details.

   > **NOTE** If you are installing Luna HSM Client on Windows, the driver may not be installed unless the Luna device is connected to the computer first; refer to Backup/USB/PCIe Drivers Not Installed on Windows 10 or Windows Server 2022 Unless Device is Connected.

   b. Connect the backup HSM directly to the Luna HSM Client workstation using the included USB cable.

   > **NOTE** On most Luna HSM Client computers, the USB port provides adequate power, and connecting the provided power supply is not recommended. Wait and confirm that the HSM boots properly. If the HSM fails to boot up:
   >
   > **1.** Disconnect the HSM from the USB port.
   >
   > **2.** Connect the HSM to power using the provided power supply. Wait for it to boot completely.
   >
   > **3.** Reconnect the HSM to the USB port on the client.
   >
   > If the HSM is connected to the USB port before the power supply, you may encounter an issue where the HSM occasionally loses contact with the client, and must be power cycled.

2. Launch LunaCM on the workstation that hosts the user and backup partition slots.
3. Select the slot assigned to the backup HSM Admin partition:

   lunacm:> **slot set -slot** <slot_id>
4. If necessary, recover the HSM from Secure Transport Mode. See Secure Transport Mode for more information:

   lunacm:> **stm recover**

   > **NOTE** Recovering a Luna Backup HSM 7 from secure transport mode may take up to three minutes.

5. Initialize the selected backup HSM in password-authenticated mode.

   lunacm:> **hsm init -ipwd -label** <label>

   You are prompted for the new HSM SO password and the HSM domain string (existing or new):

Configuring the Luna Backup HSM 7 for FIPS Compliance

Luna Backup HSM Firmware 7.7.1 and newer uses the same updated cloning protocol as Luna HSM Firmware 7.7.0 and newer. For the Luna Backup HSM 7 to be FIPS-compliant, it must restrict restore operations to application partitions that use the new protocol. This restriction is applied by setting **HSM policy 55:** Enable Restricted Restore to **1** on the backup HSM. The Luna Backup HSM 7 must be initialized and connected to a Luna HSM Client computer to set this policy.

When this policy is enabled on the Luna Backup HSM 7, objects that have been backed up from partitions using firmware older than Luna HSM Firmware 7.7.0 can be restored to Luna HSM Firmware 7.7.0 or newer (V0 or V1) partitions only.

> **CAUTION!** FIPS compliance requires that objects are never cloned or restored to an HSM using less secure firmware, and this includes restoring from Luna Backup HSM 7 firmware.
>
> If you have backups already stored on the Luna Backup HSM 7 that were taken from pre-7.7.0 partitions, turning this policy ON will prevent you from restoring them to the same source partition. You must update the HSM containing the source partition to Luna HSM Firmware 7.7.0 or newer before restoring from backup.

> **NOTE** **HSM policy 12:** Allow non-FIPS algorithms, which is used to set FIPS-compliant mode on other Luna HSMs, does not apply to the Luna Backup HSM 7. Attempts to change this policy will fail with the error `CKR_CANCEL`.

To configure the Luna Backup HSM 7 for FIPS compliance

1. On the Luna HSM Client computer, run LunaCM.
2. Set the active slot to the Luna Backup HSM 7.

   lunacm:> **slot set -slot** <slot_id>
3. Log in as Backup HSM SO.

   lunacm:> **role login -name so**
4. Set **HSM policy 55:** Enable Restricted Restore to **1**.

   lunacm:> **hsm changehsmpolicy -policy 55 -value 1**
5. [Optional] Check that the Luna Backup HSM 7 is now in FIPS approved operation mode.

   lunacm:> **hsm showinfo**

   ```
   *** The HSM is in FIPS 140-2 approved operation mode. ***
   ```

> **NOTE** HSS (PQC) private keys cannot be cloned, due to inherent restrictions of that key type. This means that backup and HA synchronization fail if HSS private keys are encountered in your application partition.

Backing Up a Password-Authenticated Partition

Backups are created and stored as partitions within the Admin partition on the Luna Backup HSM 7. A new backup partition is created on initial backup. For subsequent backups, you can choose to replace the contents of the existing backup partition with the current source partition objects, or add new objects in the source partition to the existing backup partition. Like all cloning operations, the source and target backup partitions must be initialized with the same domain.

> **NOTE** Prior to creating a backup, Policy 55 must be OFF on the Backup HSM Device.

Prerequisites

Before you begin, ensure that you have satisfied the following prerequisites:

> You have the required credentials:

If you are creating a new backup:

- The Crypto Officer password and domain string for the source partition
- The HSM SO password for the backup HSM

If you are adding to an existing backup initialized with the same domain string as the source partition:

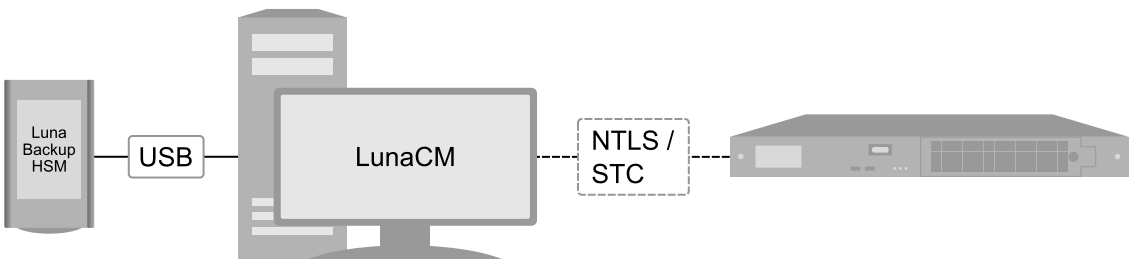
- The Crypto Officer password for the source partition
- The Crypto Officer password for the existing backup
- The HSM SO password for the backup HSM

> The following policies are set:

- **HSM policy 16:** Allow network replication must be set to **1** (ON) on the HSM that hosts the user partition.
- [V0 partitions or firmware older than Luna HSM Firmware 7.7.0] **Partition policy 0:** "Allow private key cloning" on page 338 is set to **1** (ON) on the user partition.
- [V0 partitions or firmware older than Luna HSM Firmware 7.7.0] **Partition policy 4:** "Allow secret key cloning" on page 340 is set to **1** (ON) on the user partition.

To back up a password-authenticated partition

1. Configure your Luna HSM Client workstation as illustrated below:



   a. If you have not already done so, install the required client software on the Luna HSM Client workstation and start LunaCM. See "Client Software Required to Perform Backup and Restore Operations" on page 469 for more information.

> **NOTE** If you are installing Luna HSM Client on Windows, the driver may not be installed unless the Luna device is connected to the computer first; refer to Backup/USB/PCIe Drivers Not Installed on Windows 10 or Windows Server 2022 Unless Device is Connected.

b. Connect the backup HSM directly to the Luna HSM Client workstation using the included USB cable.

> **NOTE** On most Luna HSM Client computers, the USB port provides adequate power, and connecting the provided power supply is not recommended. Wait and confirm that the HSM boots properly. If the HSM fails to boot up:
>
> **1.** Disconnect the HSM from the USB port.
>
> **2.** Connect the HSM to power using the provided power supply. Wait for it to boot completely.
>
> **3.** Reconnect the HSM to the USB port on the client.
>
> If the HSM is connected to the USB port before the power supply, you may encounter an issue where the HSM occasionally loses contact with the client, and must be power cycled.

2. Launch LunaCM on the workstation that hosts the user and backup partition slots.
3. Identify the slots assigned to:
   - The Luna Network HSM 7 partition slot (to be backed up).
   - The Luna Backup HSM 7 admin slot (where all backups are stored).

   lunacm:> **slot list**

   If you cannot see both slots, check your connections or configure your client as required.
4. Select the Luna Network HSM 7 partition:

   lunacm:> **slot set -slot** <slot_id>
5. Log in to the Luna Network HSM 7 partition as the Crypto Officer (CO):

   lunacm:> **role login -name co**
6. Initiate backup of the Luna Network HSM 7 partition to the backup partition:

   lunacm:> **partition archive backup -slot** <backup_hsm_admin_partition_slot_id> [**-partition** <target_backup_partition_label>] [**-append**] [**-replace**] [**-smkonly**]

   If you omit the **-partition** option when creating a new backup, the backup is assigned a default name (<source_partition_name>_<YYYYMMDD>) based on the source HSM's internally-set time and date.

   If you are backing up a V1 partition, include **-smkonly** to back up the SMK only. By default, the SMK and any encrypted cryptographic material on the partition are backed up.

   The backup begins once you have completed the authentication process. Objects are backed up one at a time. For existing backups, you can use the following options to define how individual objects are backed up:

-replace	Delete the target backup partition and replace it with a new backup with the same label, with the contents of the source partition. This is the default.
-append	Add only new objects to the existing backup.
-append -replace	Add new objects and replace existing objects that have the same OUID but a different fingerprint (such as would occur if any of the object attributes were changed since the previous backup).

> **NOTE** If the backup operation is interrupted (if the Backup HSM is unplugged, for example), the Backup HSM's full available space can become occupied with a single backup partition. If this occurs, delete the backup partition with lunacm:> **partition archive delete** before reattempting the backup operation.

7. You are prompted for the following passwords, unless you specified them in the **partition archive backup** options:
   a. The HSM SO password for the backup HSM. This is required to create or access the backup partition in the Admin slot.
   b. The Crypto Officer password for the target partition on the backup HSM (if you specified an existing backup). If you are creating a new backup, you must set its CO password now.
   c. [If creating a new backup] The domain string for the backup partition. The domain must match the domain configured on the source partition.

Restoring to a Password-Authenticated Partition

You can restore the objects from a password-authenticated backup to the same partition that was originally backed up, or to another partition that has been initialized with the same domain string.

Prerequisites

> The backup and the partition you want to restore to must be members of the same domain.

> You need the following credentials:

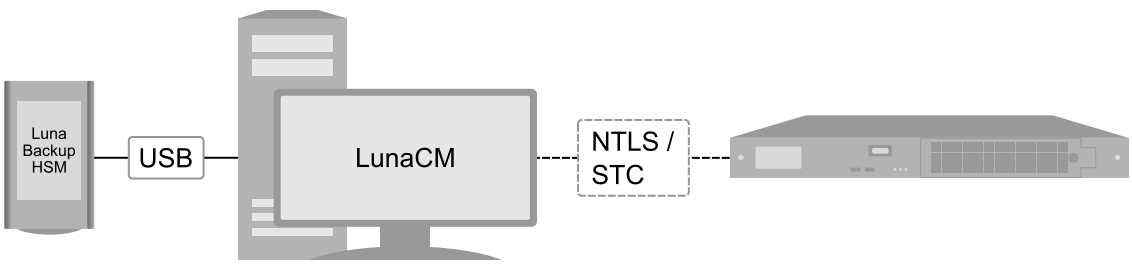
- The Crypto Officer password for the target partition.
- The Crypto Officer password for the backup

> The following policies are set:

- **HSM policy 16:** Allow network replication must be set to **1** (ON) on the HSM that hosts the user partition.
- [V0 partitions or firmware older than Luna HSM Firmware 7.7.0] **Partition policy 0:** "Allow private key cloning" on page 338 is set to **1** (ON) on the partition you want to restore to.
- [V0 partitions or firmware older than Luna HSM Firmware 7.7.0] **Partition policy 4:** "Allow secret key cloning" on page 340 is set to **1** (ON) on the partition you want to restore to.

To restore a password-authenticated partition

1. Configure your Luna HSM Client workstation as illustrated below:



a. Install the required client software on the Luna HSM Client workstation and start LunaCM. See "Client Software Required to Perform Backup and Restore Operations" on page 469 for more information.

> **NOTE** If you are installing Luna HSM Client on Windows, the driver may not be installed unless the Luna device is connected to the computer first; refer to Backup/USB/PCIe Drivers Not Installed on Windows 10 or Windows Server 2022 Unless Device is Connected.

b. Connect the backup HSM directly to the Luna HSM Client workstation using the included USB cable.

> **NOTE** On most Luna HSM Client computers, the USB port provides adequate power, and connecting the provided power supply is not recommended. Wait and confirm that the HSM boots properly. If the HSM fails to boot up:
>
> **1.** Disconnect the HSM from the USB port.
>
> **2.** Connect the HSM to power using the provided power supply. Wait for it to boot completely.
>
> **3.** Reconnect the HSM to the USB port on the client.
>
> If the HSM is connected to the USB port before the power supply, you may encounter an issue where the HSM occasionally loses contact with the client, and must be power cycled.

2. Identify the slots assigned to:
   - The Luna Network HSM 7 partition slot (to be restored).
   - The Luna Backup HSM 7 admin slot (where all backups are stored).

   lunacm:> **slot list**

   If you cannot see both slots, check your connections or configure your client as required.
3. Select the Luna Network HSM 7 partition you want to restore to:

   lunacm:> **slot set -slot** <slot_id>
4. Log in to the partition as Crypto Officer (CO):

   lunacm:> **role login -name co**
5. List the available backups on the Backup HSM by specifying the Backup HSM's slot number. You will require the backup partition label to perform the restore operation.

   lunacm:> **partition archive list -slot** <backup_HSM_slot>
6. Initiate the restore operation. Respond to the prompts to provide the required passwords, as detailed in the summary above.

   lunacm:> **partition archive restore -slot** <backup_HSM_admin_slot> **-partition** <backup_partition_label> [**-smkonly**]

   You are prompted for the Crypto Officer password for the backup. The restore operation begins once you have completed the authentication process. Objects are restored one at a time.

CAUTION! The **-replace** option is deprecated and has been removed in Luna HSM Client 10.7.0 and newer. If you wish to restore an earlier version of an object, Thales recommends deleting the object(s) manually before restoring the partition from backup.

Ensure that the target partition can receive objects from the backup HSM before deleting objects or using partition archive restore with the **-replace** option; the cloning protocol may prevent objects from being restored, even if LunaCM states that `X objects will be restored`. This may occur if **HSM policy 55: Enable Restricted Restore** was enabled on the Luna Backup HSM 7 since the original backup was taken. If your partition is on an HSM with firmware older than Luna HSM Firmware 7.7.0, you must update to 7.7.0 or newer to restore objects from this backup.

NOTE If you are restoring a V1 backup to a V1 partition, include **-smkonly** to restore the SMK only (see "V0 and V1 Partitions" on page 148 for more information). By default, the SMK and any encrypted cryptographic material on the backup are restored.

Luna Backup HSM G5

The Luna Backup HSM G5 allows you to safeguard your important cryptographic objects by making secure backups, and restoring those backups to an application partition.

For setup, management and backup/restore procedures, refer to the following sections:

> "Luna Backup HSM G5 Hardware Installation" on the next page

> "Backup/Restore Using Luna Backup HSM G5 Connected to Luna Network HSM 7" on page 552

> "Backup/Restore Using Luna Backup HSM G5 Connected to Luna HSM Client" on page 557

> "Managing the Luna Backup HSM G5" on page 542

> "Configuring a Remote Backup Server" on page 561

The Luna Backup HSM G5 can be configured to back up either password- or multifactor quorum-authenticated partitions. You must specify the authentication method when you initialize the Luna Backup HSM G5. Once initialized, the backup HSM can only be used with partitions sharing the same authentication type. The only way to change the authentication method is to restore the backup HSM to factory condition and re-initialize it.

The storage capacity and maximum number of backup partitions allowed on the backup HSM is determined by the firmware. You can check the capacity using lunash:> **token backup show -serial** <serialnum> or lunacm:> **hsm showinfo**. To update the backup HSM firmware to a version that allows more backups, see "Updating the Luna Backup HSM G5 Firmware" on page 544.

> **NOTE** Objects stored on a Backup HSM may be smaller than their originals. For example, symmetric keys are 8 bytes smaller when stored on a Backup HSM. This size difference has no effect on backup and restore operations.

Considerations when Performing Cloning and Backup-Restore Operations, when SKS is Involved

If you invoked scalable key storage (SKS) for your applications to create and store large numbers of keys, then the partition is V1. If you perform cloning operations (including HA) or Backup and Restore, see "Cloning or Backup / Restore with SKS" on page 216.

Luna Backup HSM G5 Hardware Installation

This section contains instructions for installing your Luna Backup HSM G5.

Luna Backup HSM G5 Required Items

This section provides a list of the components you should have received with your Luna Backup HSM G5 order.

Qty	Item
1	**Luna Backup HSM G5** 

Qty	Item
1	**External Power Supply** The Luna Backup HSM G5 now ships with an external power supply. Previously, these HSMs relied on an internal power supply, requiring the HSM to be periodically powered on to recharge internal capacitors. Failure to charge the capacitors could result in an inability to power on the HSM. With the introduction of external power supplies, periodically powering on the HSM is no longer required. A failed external power supply can be replaced and there is no need to return the HSM for repair (RMA). **NOTE** External power supplies do contain capacitors which may be affected by extended periods of being unpowered, but they are more easily replaced in the event of failure.
1	**Power Supply Cord** Your order should include one power supply cord for the Luna Backup HSM G5. The actual cord received depends on the country for which you ordered the Luna Backup HSM G5. 
1	**USB cable (USB A to USB mini B)**  Your order should include one USB A to 5-pin (Mini-B) cable.

Optional Items

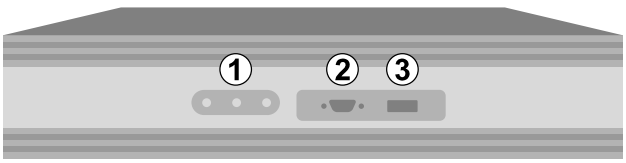
Your order may also include the following optional item

Luna Backup HSM G5 Rack-Mount Shelf

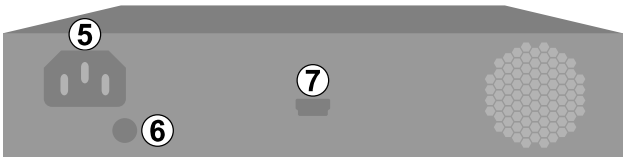
The Luna Backup HSM G5 rack-mount shelf (available by separate order) fits a standard 19-inch equipment rack, allowing you to install up to two Luna Backup HSM G5 units side-by side in server-room racks. For office use, without rack mounting, Luna Backup HSM G5 units can be placed on a desktop and are stackable.

Physical Features

The front panel of the Luna Backup HSM G5 is illustrated below, with important features labeled. In the second image, the front bezel has been removed, exposing the battery enclosure.



The rear panel of the Luna Backup HSM G5 is illustrated below, with important features labeled. The first image depicts a backup HSM with an internal power supply. The second image depicts one that ships with an external power supply.



1	Status LEDs. When illuminated, they indicate: > **Active:** The backup HSM is performing a procedure. Do not disconnect or unplug the device when this light is illuminated. > **Tamper:** The backup HSM is in a tamper state. You must clear the tamper state before backing up or restoring partitions. > **Error:** HSM device driver error. Contact Thales Customer Support (see "Support Contacts" on page 19).
2	Serial port for attaching a local Luna PED using a 9-pin Micro-D to Micro-D cable.
3	USB port. Not applicable to backup/restore functions.
4	Battery enclosure. See "Installing or Replacing the Luna Backup HSM G5 Battery" on page 546.
5	Power connector for a Luna Backup HSM with an internal power supply. See "Storage and Maintenance" on page 543 for more information.
6	Index hole. Engages with the index post on a Luna Backup HSM rack shelf.
7	Mini-USB port for connecting the Luna Backup HSM G5 to a Luna HSM or client workstation. See "Installing the Luna Backup HSM G5" on the next page.
8	Power source connector for a Luna Backup HSM G5 with an external power supply (included).

Installing the Luna Backup HSM G5

You can connect the Luna Backup HSM to a Luna Network HSM, a Luna HSM Client workstation, or a host machine containing a Luna PCIe HSM. Refer to "Planning Your Backup HSM Deployment" on page 469 for detailed descriptions of the configuration options.

To install the Luna Backup HSM G5

1. Connect the Luna Backup HSM G5 to power using the external power source or a standard power cable.
2. If you are connecting the Luna Backup HSM G5 to a client workstation or Luna PCIe HSM 7 host, ensure that you have installed the **Backup** option in the Luna HSM Client installer (see "Luna HSM Client Software Installation" on page 20 for details).
3. [Local PED] If you plan to authenticate the Luna Backup HSM G5 with a local Luna PED, connect the PED using a 9-pin Micro-D to Micro-D cable (see "Physical Features" on the previous page).

   To use the same local PED to authenticate both the Backup HSM and Luna Network HSM 7, connect the PED to the Luna Network HSM 7 using a USB Mini-B to USB cable (see "Physical Features" on the previous page). You can switch between the two using PED modes (see "Modes of Operation" on page 248).
4. Connect the Luna Backup HSM G5 using the included Mini-USB to USB cable. If you are connecting the Backup HSM to:
   - **Luna Network HSM 7:** Connect to one of the USB ports on the front or rear panel of the appliance.
   - **Luna HSM Client:** Connect to a USB port on the client workstation. Run LunaCM on the client workstation to confirm that the Luna Backup HSM G5 appears in a slot.
   - **Luna PCIe HSM 7 host:** Connect to a USB port on the host workstation. Run LunaCM on the host workstation to confirm that the Luna Backup HSM G5 appears in a slot.
5. If your Backup HSM was shipped in Secure Transport Mode, see "Recovering From a Tamper Event or Secure Transport Mode" on page 550.

Managing the Luna Backup HSM G5

This section contains the following procedures for maintaining and using the Luna Backup HSM G5:

Storage and Maintenance

The Luna Backup HSM G5 can be safely stored, containing backups, when not in use. When stored properly, the hardware has a lifetime of 10+ years. Newer Luna Backup HSM G5s ship with an external power supply.

> **CAUTION!** The internal power supply on older Luna Backup HSM G5s uses capacitors that may be affected if they are left unpowered for extended periods of time.If your Luna Backup HSM G5 has an internal power supply, power it on occasionally to recharge the capacitors. If the capacitors lose function, the Luna Backup HSM G5 will no longer receive power.
>
> With the introduction of external power supplies, this is no longer a requirement. If the external power supply fails from being left unpowered, it can be easily replaced.

The Luna Backup HSM G5 Battery

The battery powers the NVRAM and Real-Time-Clock (RTC), and must be installed for use. The battery can be removed for storage, and this is generally good practice. Thales uses high-quality, industrial-grade batteries that are unlikely to leak and damage the HSM hardware, but an externally-stored battery will last longer. The battery must be stored in a clean, dry area (less than 30% Relative Humidity). Temperature should not exceed +30 ºC. When properly stored, the battery has a shelf life of 10 years.

If the battery dies or is removed, and the main power is not connected, NVRAM and the RTC lose power. Battery removal triggers a tamper event. After replacing the battery, the HSM SO must clear the tamper event before operation can resume. The working copy of the Master Tamper Key (MTK) is lost (see "About Luna Backup HSM G5 Secure Transport and Tamper Recovery" on page 548). Backup objects are stored in non-volatile memory, so they are preserved and remain uncorrupted.

There is no low battery indicator, or other provision for checking the battery status. The voltage remains constant until the very end of battery life.

Your stored (backed-up) content is in long-term memory and is not affected by the state of the battery. A failure or removal of the battery does cause a tamper event, but this is intended as an alert to bring the condition to your attention for action, and does not affect the stored content. A situation where battery removal *could* affect your ability to recover archived data from the Luna Backup HSM G5 is where you have previously extracted a portion of the MTK onto an iKey (PED Key) and then have lost/destroyed/overwritten all copies of that key, leaving the MTK unrecoverable.

Initializing the Luna Backup HSM G5 Remote PED Vector

The Remote PED (via PEDserver) authenticates itself to the Luna Backup HSM G5 with a randomly-generated encrypted value stored on an orange PED key. The orange key proves to the HSM that the Remote PED is authorized to perform authentication. The HSM SO can create this key using LunaCM.

If the Luna Backup HSM G5 is already initialized, the HSM SO must log in to complete this procedure.

Prerequisites

- Luna PED with firmware 2.7.1 or newer
- USB mini-B to USB-A connector cable
- Luna PED DC power supply (if included with your Luna PED)

> Blank or reusable orange PED key (or multiple keys, if you plan to make extra copies or use an M of N security scheme). See "Creating PED keys" on page 287 for more information.

> Install the Luna Backup HSM G5 at the client and connect it to power (see "Luna Backup HSM G5 Hardware Installation" on page 539).

> Connect the PED to the Luna Backup HSM G5 using a 9-pin Micro-D to Micro-D cable. Set the PED to **Local PED-SCP** mode (see "Modes of Operation" on page 248).

To initialize the RPV and create the orange PED key

1. Launch LunaCM on the client workstation.
2. Set the active slot to the Luna Backup HSM G5.

   lunacm:> **slot set -slot** <slotnum>
3. If the Luna Backup HSM G5 is initialized, log in as HSM SO. If not, continue to the next step.

   lunacm:> **role login -name so**
4. Ensure that you have the orange PED key(s) ready. Initialize the RPV.

   lunacm:> **ped vector init**
5. Attend to the Luna PED and respond to the on-screen prompts. See "Creating PED keys" on page 287 for a full description of the key-creation process.

   ```
   SLOT
   SETTING RPV...
   Would you like to
   reuse an existing
   keyset?(Y/N)
   ```

   - If you have an orange PED key with an existing RPV that you wish to use for this HSM, press **Yes**.
   - If you are creating a new RPV, press **No**.

   ```
   SLOT
   SETTING RPV...
   Insert a
   RPK / Remote
   PED Key (ORANGE)
   Press ENTER.
   ```

   Continue following the prompts for PED PIN, M of N, and duplication options.

To set up a Remote PED server, see "Configuring a Remote Backup Server" on page 561.

Updating the Luna Backup HSM G5 Firmware

To update Luna Backup HSM G5 firmware, use LunaCM on a client computer that is connected to the Luna Backup HSM G5. You require:

> Luna Backup HSM G5 firmware update file (<filename>**.fuf**)

> the firmware update authentication code file(s) (<filename>**.txt**)

CAUTION! Use an uninterruptible power supply (UPS) to power your HSM. There is a small chance that a power failure during an update could leave your HSM in an unrecoverable condition.

NOTE To perform backup operations on Luna HSM Firmware 7.7.0 or newer (V0 or V1 partitions) you require at minimum:

> Luna Backup HSM 7 Firmware 7.7.1

> Luna Backup HSM G5 Firmware 6.28.0

You can use a Luna Backup HSM with older firmware to restore objects to a V0 or V1 partition, but this is supported for purposes of getting your objects from the older partitions onto the newer V0 or V1 partitions only. V0 and V1 partitions are considered more secure than partitions at earlier firmware versions - any attempt to restore from a higher-security status to lower-security status fails gracefully.

When the Luna Backup HSM is connected directly to the Luna Network HSM 7 appliance, only the SMK can be backed up from or restored to a V1 partition.

To update the Luna Backup HSM G5 firmware

1. Copy the firmware file (<filename>**.fuf**) and the authentication code file (<filename>**.txt**) to the Luna HSM Client root directory.
   - Windows: **C:\Program Files\SafeNet\LunaClient**
   - Linux/AIX: **/usr/safenet/lunaclient/bin**
   - Solaris: **/opt/safenet/lunaclient/bin**

   NOTE On some Windows configurations, you might not have authority to copy or unzip files directly into **C:\Program Files\...**. If this is the case, put the files in a known location that you can reference in a LunaCM command.

2. Launch LunaCM.
3. If more than one HSM is installed, set the active slot to the Admin partition of the HSM you wish to update.

   lunacm:> **slot set -slot** <slot_number>
4. Log in as HSM SO. Depending on the currently-installed firmware version, use one of the following two commands:
   - lunacm:> **role login -name so**
   - lunacm:> **hsm login**
5. Apply the new firmware update by specifying the update file and the authentication code file. If the files are not located in the Luna HSM Client root directory, specify the filepaths.

   lunacm:> **hsm updatefw -fuf** <filename>**.fuf -authcode** <filename>**.txt**

Resetting the Luna Backup HSM G5 to Factory Conditions

These instructions will allow you to restore your Luna Backup HSM G5 to its original factory conditions, erasing its contents. This could be necessary if you have old backups that you do not wish to keep, or if you want to re-initialize the Backup HSM to store backups using a different authentication method (password or multifactor quorum). If you have performed firmware updates, they are unaffected. Factory reset can be performed via LunaCM.

To reset the Luna Backup HSM G5 to factory conditions

1. Launch LunaCM on the Luna Backup HSM G5 workstation.
2. Set the active slot to the Luna Backup HSM G5.

   lunacm:> **slot set -slot** <slotnum>
3. Reset the Backup HSM.

   lunacm:> **hsm factoryreset**

Installing or Replacing the Luna Backup HSM G5 Battery

The Luna Backup HSM G5 must have a functioning battery installed to preserve the NVRAM and RTC in case of primary power loss. You can purchase a replacement battery from any supplier who can match the following specifications:

- 3.6 V Primary lithium-thionyl chloride (Li-SOCl2)
- Fast voltage recovery after long term storage and/or usage
- Low self discharge rate
- 10 years shelf life
- Operating temperature range -55 ºC to +85 ºC
- U.L. Component Recognition, MH 12193

Prerequisites

- Removing the battery causes a tamper event. If you have created a Secure Recovery Vector (purple PED key) and enabled Secure Recovery, you will need this key to clear the tamper after replacing the battery.

To install or replace the Luna Backup HSM G5 battery

1. Remove the front bezel. It is held in place by two spring clips.

2. The battery compartment is spring-loaded and can be removed without much pressure. Use a coin or your fingers to press in the compartment cover and turn counter-clockwise to remove it.

3. If you are replacing the old battery, remove it from the battery compartment.

4. Insert the new battery, negative end first. The positive end should be visible.

5. Use the battery compartment cover to push the battery into the compartment, aligning the tabs on the cover with the compartment slots. Twist the cover clockwise to lock the compartment.

6. Replace the front bezel by aligning the clips with their posts and pushing it into place.

   Removing the battery causes a tamper event on the Luna Backup HSM G5.

7. To clear the tamper, see "Recovering From a Tamper Event or Secure Transport Mode" on page 550.

About Luna Backup HSM G5 Secure Transport and Tamper Recovery

The Luna Backup HSM G5 recognizes a similar list of tamper conditions to the Luna Network HSM 7 (see Tamper Events). When a tamper event occurs, a tamper state is reported in the **HSM Status** field in LunaCM's list of slots.

By default, tamper events are cleared automatically when you reboot the Luna Backup HSM G5 and log in as HSM SO. However, you can choose to prevent any further operations on theLuna Backup HSM G5. The following procedures will allow you to create a purple Secure Recovery Key (SRK) that the Backup HSM SO

must present to unlock the HSM after a tamper event. This key contains part of the Master Tamper Key (MTK), which encrypts all sensitive data stored on the Backup HSM. By splitting the MTK and storing part of it on an SRK (purple PED key), you ensure that none of the stored material can be accessible until the SRK is presented.

You can create the purple SRK even for a Luna Backup HSM G5 that is initialized for password authentication. There is no password-based SRK equivalent; you must have a Luna PED and a purple PED key to use Secure Tamper Recovery and Secure Transport Mode.

Initializing the SRK also allows you to place the Luna Backup HSM G5 in Secure Transport Mode (STM). STM on the Luna Backup HSM G5 functions differently from STM on the Luna Network HSM 7 (see Secure Transport Mode for comparison). When the SRK is initialized and secure recovery enabled, STM on the Backup HSM is effectively a voluntary tamper state, where no operations are possible until you present the purple PED key.

> **CAUTION!** Always keep a securely-stored backup copy of the purple PED key. If you lose this key, the Backup HSM is permanently locked and you will have to obtain an RMA for the Backup HSM.

This section provides directions for the following procedures:

> "Creating a Secure Recovery Key" below

> "Setting Secure Transport Mode" on the next page

> "Recovering From a Tamper Event or Secure Transport Mode" on the next page

> "Disabling Secure Recovery" on page 551

Creating a Secure Recovery Key

To enable secure recovery, you must create the Secure Recovery Key (purple PED key). This procedure will zeroize the SRK split on the Backup HSM, so that you must present the purple PED key to recover from a tamper event or Secure Transport Mode.

Prerequisites

> Install the Backup HSM at the client and connect it to power (see "Luna Backup HSM G5 Hardware Installation" on page 539).

> You require the Backup HSM SO credential (blue PED key).

> Ensure that the Backup HSM can access PED service (Local or Remote PED), and that you have enough blank or rewritable purple PED keys available for your desired authentication scheme (see "Creating PED keys" on page 287).

- [Local PED] Connect the PED using a 9-pin Micro-D to Micro-D cable. Set the PED to **Local PED-SCP** mode (see "Modes of Operation" on page 248).
- [Remote PED] Set up a Remote PED server to authenticate the Backup HSM (see "Configuring a Remote Backup Server" on page 561).
- [Remote PED] Initialize the Backup HSM RPV (see "Initializing the Luna Backup HSM G5 Remote PED Vector" on page 543). You require the orange PED key.

To create a Secure Recovery Key

1. Launch LunaCM on the client workstation.

2. Set the active slot to the Luna Backup HSM.

   lunacm:> **slot set -slot** <slotnum>

3. [Remote PED] Connect the Luna Backup HSM G5 to the Remote PED server.

   lunacm:> **ped connect -ip** <PEDserver_IP> **-port** <portnum>

4. Create a new split of the MTK on the Luna Backup HSM G5.

   lunacm:> **srk generate**

5. Log in as Backup HSM SO.

   lunacm:> **role login -name so**

6. Enable secure recovery.

   lunacm:> **srk enable**

   Attend to the Luna PED prompts to create the purple PED key. Secure Recovery is now enabled on the Luna Backup HSM G5.

Setting Secure Transport Mode

The following procedure will allow you to set Secure Transport Mode on the Luna Backup HSM G5.

Prerequisites

> Ensure the Luna Backup HSM G5 can access PED services.

> Secure Recovery must be enabled on the Backup HSM (see "Creating a Secure Recovery Key" on the previous page). You require the Secure Recovery Key (purple PED key) for the Luna Backup HSM G5.

To set Secure Transport Mode on the Luna Backup HSM G5

1. Launch LunaCM on the client workstation.
2. Set the active slot to the Luna Backup HSM G5.

   lunacm:> **slot set -slot** <slotnum>

3. [Remote PED] Connect the Luna Backup HSM G5 to the Remote PED server.

   lunacm:> **ped connect -ip** <PEDserver_IP> **-port** <portnum>

4. Set Secure Transport Mode.

   lunacm:> **srk transport**

   a. You are prompted for the SRK (purple PED key). This is to ensure that you have the key that matches the SRK split on the HSM.

   b. The Luna PED displays a 16-digit verification code. Write this code down as an additional optional check.

   The SRK is zeroized on the Luna Backup HSM G5 and STM is now active.

Recovering From a Tamper Event or Secure Transport Mode

With Secure Recovery Mode enabled, the procedure to recover from a tamper event or to exit STM is the same.

Prerequisites

> Ensure the Luna Backup HSM G5 can access PED services.

> You require the Secure Recovery Key (purple PED key) for the Luna Backup HSM G5.

> If you are recovering from a tamper event, reboot the Backup HSM and LunaCM before recovering.

lunacm:> **hsm restart**

lunacm:> **clientconfig restart**

To recover from a tamper event or exit STM

1. Launch LunaCM on the client workstation.
2. Set the active slot to the Luna Backup HSM G5.

   lunacm:> **slot set -slot** <slotnum>
3. [Remote PED] Connect the Luna Backup HSM G5 to the Remote PED server.

   lunacm:> **ped connect -ip** <PEDserver_IP> **-port** <portnum>
4. Recover the Luna Backup HSM G5 from the tamper event or STM.

   lunacm:> **srk recover**

   Attend to the Luna PED prompts:

   a. You are prompted for the SRK (purple PED key).

   b. [STM] The Luna PED displays a 16-digit verification code. If this code matches the one that was presented when you set STM, you can be assured that the Luna Backup HSM G5 has remained in STM since then.

   The Luna Backup HSM G5 is recovered from the tamper/STM state and you can resume backup/restore operations.

Disabling Secure Recovery

To disable secure recovery, you must present the Secure Recovery Key (purple PED key) so that it can be stored on the Luna Backup HSM G5. You will no longer need to present the purple key to recover from a tamper event.

Prerequisites

> Ensure the Luna Backup HSM G5 can access PED services.

> You require the Secure Recovery Key (purple PED key) for the Luna Backup HSM G5.

To disable secure recovery

1. Launch LunaCM on the client workstation.
2. Set the active slot to the Luna Backup HSM G5.

   lunacm:> **slot set -slot** <slotnum>
3. [Remote PED] Connect the Luna Backup HSM G5 to the Remote PED server.

   lunacm:> **ped connect -ip** <PEDserver_IP> **-port** <portnum>
4. Log in as Backup HSM SO.

lunacm:> **role login -name so**

5. Disable secure recovery.

   lunacm:> **srk disable**

   You are prompted for the SRK (purple PED key).

Backup/Restore Using Luna Backup HSM G5 Connected to Luna Network HSM 7

You can connect the Luna Backup HSM G5 directly to one of the USB ports on the Luna Network HSM 7 appliance. This configuration allows you to perform backup/restore operations using LunaSH, via a serial or SSH connection to the appliance. It is useful in deployments where backups are kept in the same location as the HSM. The Crypto Officer must have **admin**-level access to LunaSH on the appliance. You can restore a partition backup to the original source partition or to another existing Luna application partition that shares the same cloning domain.

> **NOTE** Please note the following conditions for using an appliance-connected Luna Backup HSM G5:
>
> > If you are backing up or restoring encrypted blobs stored on a V1 partition, the Backup HSM must be connected to the client (see "Backup/Restore Using Luna Backup HSM G5 Connected to Luna HSM Client" on page 557)
>
> Only the SMK can be backed up/restored using an appliance-connected backup HSM.
>
> > If **partition policy 37:** "Force Secure Trusted Channel" on page 347 is enabled on the partition, the backup HSM must be connected to the client (see "Backup/Restore Using Luna Backup HSM G5 Connected to Luna HSM Client" on page 557)
>
> > You can use an appliance-connected Backup HSM with Remote PED only if the source partition is activated ("Activation on Multifactor Quorum-Authenticated Partitions" on page 373) and Luna Appliance Software 7.7.0 or newer is installed.

This section provides instructions for the following procedures using this kind of deployment:

> "Initializing the Luna Backup HSM G5" on the next page
>
> "Backing Up an Application Partition" on page 554
>
> "Restoring an Application Partition from Backup" on page 555

NOTE To perform backup operations on Luna HSM Firmware 7.7.0 or newer (V0 or V1 partitions) you require at minimum:

> Luna Backup HSM 7 Firmware 7.7.1

> Luna Backup HSM G5 Firmware 6.28.0

You can use a Luna Backup HSM with older firmware to restore objects to a V0 or V1 partition, but this is supported for purposes of getting your objects from the older partitions onto the newer V0 or V1 partitions only. V0 and V1 partitions are considered more secure than partitions at earlier firmware versions - any attempt to restore from a higher-security status to lower-security status fails gracefully.

When the Luna Backup HSM is connected directly to the Luna Network HSM 7 appliance, only the SMK can be backed up from or restored to a V1 partition.

NOTE The size of the partition header is different for a Luna Network HSM 7 partition and its equivalent backup partition stored on a Luna Backup HSM G5. As a result, the value displayed in the `Used` column in the output of the **partition list** command (for the backed-up Luna Network HSM 7 partition) is different than the value displayed in the `Used` column in the output of the **token backup partition list** command (for the backup partition on the Backup HSM).

Initializing the Luna Backup HSM G5

Before you can use the Luna Backup HSM G5 to back up your partition objects, it must be initialized. This procedure is analogous to the standard HSM initialization procedure.

Prerequisites

> Install the Luna Backup HSM G5 and connect it to power (see "Installing the Luna Backup HSM G5" on page 542).

> Ensure that the Luna Backup HSM G5 is not in Secure Transport Mode and that any tamper events are cleared (see "Recovering From a Tamper Event or Secure Transport Mode" on page 550).

> [PED Authentication] Ensure that you have enough blank or rewritable blue and red PED keys available for your desired authentication scheme (see "Creating PED keys" on page 287).

> [Local PED] Connect the PED using a 9-pin Micro-D to Micro-D cable. Set the PED to **Local PED-SCP** mode (see "Modes of Operation" on page 248).

To initialize an appliance-connected Luna Backup HSM G5 using LunaSH

1. Log in to LunaSH as **admin**, or an **admin**-level custom user.
2. [Optional] View the Luna Backup HSM G5s currently connected to the appliance and find the correct serial number.

   lunash:> **token backup list**
3. Initialize the Backup HSM by specifying its serial number and a label.

   lunash:> **token backup init -serial** <serialnum> **-label** <label>

   You are prompted to set the HSM SO credential and cloning domain for the Luna Backup HSM G5.

Backing Up an Application Partition

You can use LunaSH to back up the contents of an application partition to the appliance-connected Luna Backup HSM G5. You can use this operation to create a backup on the Backup HSM, or add objects from the source partition to an existing backup.

> **TIP** In the event that the Luna Network HSM appliance is rebooted and a connected Luna Backup HSM G5 does not recover -- symptom: it does not appear in the output of the command
>
> `lunash:>token backup list`
>
> then:
>
> > if working remotely, perform another reboot of the Luna Network HSM appliance to which the Backup HSM is connected
> >
> > if you are local to the equipment, disconnect and reconnect the USB cable between the appliance and the Backup HSM.

Prerequisites

> The Luna Backup HSM G5 must be initialized (see "Initializing the Luna Backup HSM G5" on the previous page).
>
> You must have **admin** or **admin**-level access to LunaSH on the Luna Network HSM 7.
>
> The following policies are set:
>
> - **HSM policy 16:** Allow network replication must be set to **1** (ON) on the HSM that hosts the user partition.
> - [V0 partitions or firmware older than Luna HSM Firmware 7.7.0] **Partition policy 0:** "Allow private key cloning" on page 338 is set to **1** (ON) on the user partition.
> - [V0 partitions or firmware older than Luna HSM Firmware 7.7.0] **Partition policy 4:** "Allow secret key cloning" on page 340 is set to **1** (ON) on the user partition.
>
> You must have the Crypto Officer credential (black PED key) and domain (red PED key) for the source partition.
>
> [Local PED] Connect the PED to the Luna Network HSM 7 using a Mini-B to USB-A cable (see "Local PED Setup" on page 250), and to the Backup HSM using a 9-pin Micro-D to Micro-D cable. Set the PED to **Local PED-USB** mode (see "Modes of Operation" on page 248).
>
> [Remote PED] The source partition must be activated (see "Activation on Multifactor Quorum-Authenticated Partitions" on page 373).
>
> [Remote PED] Set up a Remote PED server to authenticate the Backup HSM (see "About Remote PED" on page 252).
>
> [Remote PED] You require the orange PED key for the Backup HSM, which must be initialized using a local PED connection (see "Initializing the Luna Backup HSM G5 Remote PED Vector" on page 543).

To back up an application partition to an appliance-connected Luna Backup HSM G5 using LunaSH

1. Log in to LunaSH as **admin**, or an **admin**-level custom user.
2. [Remote PED] Connect the Luna Backup HSM G5 to the remote PED server.

   lunash:> **hsm ped connect -ip** <PEDserver_IP> **-serial** <Backup_HSM_serialnum>

3. [Optional] View the Luna Backup HSM G5s currently connected to the appliance and find the correct serial number.

   lunash:> **token backup list**

4. Back up the partition, specifying the source partition label, a label for the backup (either a new or existing label), and the Luna Backup HSM G5 serial number. If you specify an existing backup, use one of the following options:

   - **-add** to keep the existing partition contents and add new objects only
   - **-replace** to erase the contents of the existing backup and replace them with the contents of the source partition

   You do not need to specify these options when backing up a V1 partition, as only the SMK is backed up.

   If you omit the **-tokenpar** option when creating a new backup, the partition is assigned a default name (<source_partition_name>_<YYYYMMDD>) based on the source HSM's internally-set time and date.

   lunash:> **partition backup -partition** <source_label> **-serial** <Backup_HSM_serialnum> [**-tokenpar** <target_label>] [**-add**] [**-replace**]

   You are prompted for the source partition's Crypto Officer credential (black PED key or challenge secret).

   [Remote PED] You are prompted for a Crypto Officer credential for the backup (black PED key) and for the cloning domain that matches the source partition (red PED key). If you are adding to an existing backup, you are not asked for the cloning domain.

5. [Local PED] LunaSH prompts you to connect the Luna PED to the Luna Backup HSM G5. Set the mode on the Luna PED to **Local PED-SCP** (see "Modes of Operation" on page 248). Enter **proceed** in LunaSH.

   You are prompted to set the following credentials:

   - Crypto Officer (password or black PED key) for the backup (can be the same as the source partition)
   - Cloning domain (string or red PED key) for the backup (must be the same as the source partition)

   The partition contents are cloned to the backup.

Restoring an Application Partition from Backup

You can use LunaSH to restore the contents of a backup to the original application partition, or any other Luna application partition that shares the same cloning domain.

Prerequisites

> The target partition must be initialized with the same cloning domain as the backup.

> You must have **admin** or **admin**-level access to LunaSH on the Luna Network HSM 7.

> The following policies are set:

- **HSM policy 16:** Allow network replication must be set to **1** (ON) on the HSM that hosts the user partition you want to restore to.
- [V0 partitions only] **Partition policy 0:** "Allow private key cloning" on page 338 is set to **1** (ON) on the user partition you want to restore to.
- [V0 partitions only] **Partition policy 4:** "Allow secret key cloning" on page 340 is set to **1** (ON) on the user partition you want to restore to.

> You must have the Crypto Officer credentials (black PED key) for the backup and the target partition.

> [Local PED] Connect the PED to the Luna Network HSM 7 using a Mini-B to USB-A cable (see "Local PED Setup" on page 250), and to the Backup HSM using a 9-pin Micro-D to Micro-D cable. Set the PED to **Local PED-USB** mode (see "Modes of Operation" on page 248).

> [Remote PED] The source partition must be activated (see "Activation on Multifactor Quorum-Authenticated Partitions" on page 373).

> [Remote PED] Set up a Remote PED server to authenticate the Backup HSM (see "About Remote PED" on page 252). You require the orange PED key for the Backup HSM.

To restore the contents of a backup to an application partition

1. Log in to LunaSH as **admin**, or an **admin**-level custom user.
2. [Remote PED] Connect the Luna Backup HSM G5 to the remote PED server.

   lunash:> **hsm ped connect -ip** <PEDserver_IP> **-serial** <Backup_HSM_serialnum>
3. [Optional] View the Luna Backup HSM G5s currently connected to the appliance and find the correct serial number.

   lunash:> **token backup list**
4. [Optional] View the backups currently available on the Luna Backup HSM G5.

   lunash:> **token backup partition list -serial** <Backup_HSM_serialnum>
5. Restore the partition contents, specifying the target partition label, the backup label, the Luna Backup HSM G5 serial number, and either:
   - **-add** to keep the existing partition contents and add new objects only
   - **-replace** to erase the contents of the partition and replace them with the contents of the backup

   > **CAUTION!** If you are restoring a V1 backup to a V1 partition, use **-add** to restore the SMK. Use **-replace** only if you wish to erase any existing cryptographic material on the target partition. By default, V1 backups only include the SMK.

   lunash:> **partition restore -partition** <target_label> **-tokenpar** <backup_label> **-serial** <Backup_HSM_serialnum> {**-add** | **-replace**}

   You are prompted for the target partition's Crypto Officer credential (black PED key or challenge secret).
6. [Local PED] LunaSH prompts you to connect the Luna PED to the Luna Backup HSM G5. Change the mode on the Luna PED to **Local PED-SCP** (see "Modes of Operation" on page 248). Enter **proceed** in LunaSH.

   You are prompted for the backup's Crypto Officer credential (black PED key or challenge secret).

   The backup contents are cloned to the application partition.

Backup/Restore Using Luna Backup HSM G5 Connected to Luna HSM Client

You can connect the Luna Backup HSM G5 to a USB port on the client workstation. This configuration allows you to perform backup/restore operations for all application partitions that appear as visible slots in LunaCM. It is useful in deployments where the partition Crypto Officer wants to keep backups at the client. You can restore a partition backup to the original source partition or to another existing Luna application partition that shares the same cloning domain.

This section provides instructions for the following procedures using this kind of deployment:

> "Initializing the Luna Backup HSM G5" below
> "Backing Up an Application Partition" on the next page
> "Restoring an Application Partition from Backup" on page 560

NOTE To perform backup operations on Luna HSM Firmware 7.7.0 or newer (V0 or V1 partitions) you require at minimum:

> Luna Backup HSM 7 Firmware 7.7.1
> Luna Backup HSM G5 Firmware 6.28.0

You can use a Luna Backup HSM with older firmware to restore objects to a V0 or V1 partition, but this is supported for purposes of getting your objects from the older partitions onto the newer V0 or V1 partitions only. V0 and V1 partitions are considered more secure than partitions at earlier firmware versions - any attempt to restore from a higher-security status to lower-security status fails gracefully.

When the Luna Backup HSM is connected directly to the Luna Network HSM 7 appliance, only the SMK can be backed up from or restored to a V1 partition.

NOTE The size of the partition header is different for a Luna Network HSM 7 partition and its equivalent backup partition stored on a Luna Backup HSM G5. As a result, the value displayed in the `Used` column in the output of the **partition list** command (for the backed-up Luna Network HSM 7 partition) is different than the value displayed in the `Used` column in the output of the **token backup partition list** command (for the backup partition on the Backup HSM).

Initializing the Luna Backup HSM G5

Before you can use the Luna Backup HSM G5 to back up your partition objects, it must be initialized. This procedure is analogous to the standard HSM initialization procedure.

Prerequisites

> Install the Luna Backup HSM G5 at the client and connect it to power (see "Installing the Luna Backup HSM G5" on page 542).
> Ensure that the Backup HSM is not in Secure Transport Mode and that any tamper events are cleared (see "Recovering From a Tamper Event or Secure Transport Mode" on page 550).

> [Multifactor Quorum Authentication] Ensure that you have enough blank or rewritable blue and red PED keys available for your desired authentication scheme (see "Creating PED keys" on page 287).
  - [Local PED] Connect the Luna PED using a 9-pin Micro-D to Micro-D cable. Set the PED to **Local PED-SCP** mode (see "Modes of Operation" on page 248).
  - [Remote PED] Initialize the Backup HSM RPV (see "Initializing the Luna Backup HSM G5 Remote PED Vector" on page 543). You require the orange PED key.
  - [Remote PED] Set up a Remote PED server to authenticate the Luna Backup HSM G5 (see "About Remote PED" on page 252).

To initialize a client-connected Luna Backup HSM G5

1. Launch LunaCM on the client workstation.
2. Set the active slot to the Luna Backup HSM G5.

   lunacm:> **slot set -slot** <slotnum>
3. [Remote PED] Connect the Luna Backup HSM G5 to the Remote PED server.

   lunacm:> **ped connect -ip** <PEDserver_IP> **-port** <portnum>
4. Initialize the Luna Backup HSM G5, specifying a label and the method of authentication (**-initwithped** or **-initwithpwd**). You must initialize the HSM with the same authentication method as the partition(s) you plan to back up.

   lunacm:> **hsm init -label** <label> {**-initwithped** |**-initwithpwd**}

   You are prompted to set an HSM SO credential and cloning domain for the Backup HSM.

> **NOTE** After initializing a client-connected Luna Backup HSM G5 to use PED authentication, the HSM erroneously requests a password to log in with any role. This issue occurs when Luna HSM Client 10.3.0 or newer is used with HSM firmware 6.10.9 or older.
>
> **Workaround:** Press ENTER to bypass the password prompt, and present the PED key as usual. Alternatively, use an older client or upgrade to Luna Backup HSM G5 Firmware 6.24.7 or newer to avoid this.

Backing Up an Application Partition

You can use LunaCM to back up the contents of an application partition to the client-connected Luna Backup HSM G5. You can use this operation to create a backup on the Backup HSM, or add objects from the source partition to an existing backup.

> **NOTE** Prior to creating a backup, Policy 55 must be OFF on the Backup HSM Device.

Prerequisites

> The Luna Backup HSM G5 must be initialized (see "Initializing the Luna Backup HSM G5" on the previous page).

> The following policies are set:
  - **HSM policy 16:** Allow network replication must be set to **1** (ON) on the HSM that hosts the user partition.

- [V0 partitions or firmware older than Luna HSM Firmware 7.7.0] **Partition policy 0:** "Allow private key cloning" on page 338 is set to **1** (ON) on the user partition.
- [V0 partitions or firmware older than Luna HSM Firmware 7.7.0] **Partition policy 4:** "Allow secret key cloning" on page 340 is set to **1** (ON) on the user partition.

> You must have the Crypto Officer credential (black PED key) and domain (red PED key) for the source partition.

> You must have the Backup HSM SO credential (blue PED key).

> [Multifactor Quorum Authentication] This procedure is simpler if the source partition is activated (see "Activation on Multifactor Quorum-Authenticated Partitions" on page 373), since you require a Luna PED only for the Backup HSM.

- [Local PED] Connect the PED to the Backup HSM using a 9-pin Micro-D to Micro-D cable. The source partition must be activated. If not, you must use Remote PED.
- [Remote PED] You must have the orange PED key for the Backup HSM (see "Initializing the Luna Backup HSM G5 Remote PED Vector" on page 543). If the source partition is not activated, you may need the orange PED key for the Luna Network HSM 7 as well.
- [Remote PED] Set up Remote PED on the workstation you plan to use for multifactor quorum authentication (see "About Remote PED" on page 252). If the partition is not activated, you must connect to PEDserver with **ped connect** before logging in, and disconnect with **ped disconnect** before initiating the backup.

If you invoked scalable key storage (SKS) for your applications to create and store large numbers of keys, then the partition is V1. If you perform cloning operations (including HA) or Backup and Restore, see "Cloning or Backup / Restore with SKS" on page 216.

To back up an application partition to a client-connected Luna Backup HSM G5

1. Launch LunaCM on the client workstation.
2. Set the active slot to the source partition and log in as Crypto Officer.

   lunacm:> **slot set -slot** <slotnum>

   lunacm:> **role login -name co**
3. [Multifactor Quorum Authentication] Connect the Luna Backup HSM G5 to the Luna PED.
   - [Local PED] Set the mode on the Luna PED to **Local PED-SCP** (see "Modes of Operation" on page 248).
   - [Remote PED] Connect the Luna Backup HSM G5 slot to PEDserver.

     lunacm:> **ped connect -slot** <Backup_HSM_slotnum> **-ip** <PEDserver_IP> **-port** <portnum>
4. Back up the partition, specifying the Luna Backup HSM G5 slot and a label for the backup (either a new or existing label). If you specify an existing backup label, include the **-append** option to add only new objects to the backup (duplicate objects will not be cloned). By default, the existing backup will be overwritten with the current contents of the source partition.

   lunacm:> **partition archive backup -slot** <Backup_HSM_slotnum> [**-partition** <backup_label>] [**-append**] [**-replace**] [**-smkonly**]

   If you omit the **-partition** option when creating a new backup, the partition is assigned a default name (<source_partition_name>_<YYYYMMDD>) based on the source HSM's internally-set time and date.

If you are backing up a V1 partition, include **-smkonly** to back up the SMK only. By default, the SMK and any encrypted cryptographic material on the partition are backed up.

The backup begins once you have completed the authentication process.

Objects are backed up one at a time. For existing backups, you can use the following options to define how individual objects are backed up:

-append	Add only new objects to an existing backup.
-replace	Delete the existing objects in a target backup partition and replace them with the contents of the source user partition. This is the default.
-append -replace	Add new objects and replace existing objects that have the same OUID but a different fingerprint (such as would occur if any of the object attributes were changed since the previous backup).

You are prompted to present or set the following credentials:

- [Remote PED] Backup HSM Remote PED vector (orange PED key)
- Backup HSM SO (password or blue PED key)
- Crypto Officer (password or black PED key) for the backup (can be the same as the source partition)
- Cloning domain (string or red PED key) for the backup (must be the same as the source partition)

The partition contents are cloned to the backup.

5. [Remote PED] Disconnect the Backup HSM from PEDserver.

   lunacm:> **ped disconnect**

Restoring an Application Partition from Backup

You can use LunaCM to restore the contents of a backup to the original application partition, or any other Luna application partition that shares the same cloning domain.

Prerequisites

> The target partition must be initialized with the same cloning domain as the backup partition.

> The following policies are set:

- **HSM policy 16:** Allow network replication must be set to **1** (ON) on the HSM that hosts the user partition you want to restore to.
- [V0 partitions] **Partition policy 0:** "Allow private key cloning" on page 338 is set to **1** (ON) on the user partition you want to restore to.
- [V0 partitions] **Partition policy 4:** "Allow secret key cloning" on page 340 is set to **1** (ON) on the user partition you want to restore to.

> You must have the Crypto Officer credentials for the backup partition and the target partition.

> [Multifactor Quorum Authentication] This procedure is simpler if the application partition is activated (see "Activation on Multifactor Quorum-Authenticated Partitions" on page 373), since you require a Luna PED only for the Backup HSM.

- [Local PED] Connect the PED to the Backup HSM using a 9-pin Micro-D to Micro-D cable. The source partition must be activated. If not, you must use Remote PED.

- [Remote PED] Set up Remote PED on the workstation you plan to use for multifactor quorum authentication (see "About Remote PED" on page 252). If the partition is not activated, you must connect to PEDserver with **ped connect** before logging in, and disconnect with **ped disconnect** before initiating the backup.

To restore the contents of a backup to an application partition

1. Launch LunaCM on the client workstation.
2. Set the active slot to the target partition and log in as Crypto Officer.

   lunacm:> **slot set -slot** <slotnum>

   lunacm:> **role login -name co**
3. [PED Authentication] Connect the Luna Backup HSM G5 to the Luna PED.
   - [Local PED] Set the mode on the Luna PED to **Local PED-SCP** (see "Modes of Operation" on page 248).
   - [Remote PED] Connect the Luna Backup HSM G5 slot to PEDserver.

     lunacm:> **ped connect -slot** <Backup_HSM_slotnum> **-ip** <PEDserver_IP> **-port** <portnum>
4. [Optional] Display the available backups by specifying the Luna Backup HSM G5 slot. Each available backup also appears as a slot in LunaCM.

   lunacm:> **partition archive list -slot** <Backup_HSM_slotnum>
5. [Optional] Display the contents of a backup by specifying the Luna Backup HSM G5 slot and the backup partition label in LunaCM.

   lunacm:> **partition archive contents -slot** <backup_slotnum> **-partition** <backup_label>
6. Restore the partition contents, specifying the Luna Backup HSM G5 slot and the backup you wish to use. By default, duplicate backup objects with the same OUID as objects currently existing on the partition are not restored.

   If you have changed attributes of specific objects since your last backup and you wish to revert these changes, include the **-replace** option.

   If you are restoring a V1 partition and you only want to restore the SMK, include the **-smkonly** option.

   lunacm:> **partition archive restore -slot** <Backup_HSM_slotnum> **-partition** <backup_label> [**-replace**] [**-smkonly**]

   You are prompted for the backup's Crypto Officer credential.

   The backup contents are cloned to the application partition.

Configuring a Remote Backup Server

The Remote Backup Service (RBS) is an optional Luna client component that allows you to connect one or more backup HSMs to a remote Luna HSM Client workstation to back up slots on any local Luna HSM Client workstations that are registered with the RBS server. RBS is useful in deployments where backups are stored in a separate location from the Luna Network HSM 7, to protect against catastrophic loss (fire, flood, etc).

RBS is a utility, included with the Luna HSM Client software, that runs on a workstation hosting one or more Backup HSMs. When RBS is configured and running, other clients or HSMs registered to it can see its Backup HSM(s) as slots in LunaCM.

Installing and Configuring the Remote Backup Service

RBS is installed using the Luna HSM Client installer. You must create a certificate for the RBS workstation and register it on all clients/appliances that will use the remote Backup HSMs. These instructions will allow you to install and configure RBS.

> **NOTE** The Luna HSM Client version installed on the RBS workstation must be the same version installed on the client workstation(s). Ensure that you use a client version that is compatible with your Backup HSM firmware.
>
> RBS with Luna Backup HSM 7 requires minimum Luna HSM Client 10.1.0, or Luna HSM Client 10.3.0 if you are using Luna Backup HSM 7 Firmware 7.7.1 or newer.

Prerequisites

> Install the following Luna HSM Client components on any Luna Network HSM 7 client workstation that hosts slots for the partitions you want to backup using RBS (see "Luna HSM Client Software Installation" on page 20):
>
> - **Network**
> - **Remote PED:** if you are backing up multifactor quorum-authenticated partitions.

> Connect the backup HSM(s) directly to the Luna HSM Client workstation that will host RBS using the included USB cable.

> **NOTE** On most workstations, the USB 3.0 connection provides adequate power to the backup HSM and it will begin the boot sequence. If you are using a low-power workstation, such as a netbook, the USB connection may not provide adequate power, in which case you will also need to connect the external power supply. It is recommended that you use the power supply for all backup HSMs connected to the RBS host workstation. If you are connecting multiple backup HSMs, you can use an external USB 3.0 hub if required.

> Initialize the backup HSMs if necessary.

> **HSM policy 16:** Allow network replication must be set to **1** (ON) on the HSMs that hosts the user partitions to be backed up.

To install and configure RBS

1. On the workstation hosting the Backup HSM(s), install the **Backup** component of the Luna HSM Client (see "Luna HSM Client Software Installation" on page 20). If this workstation will also host a Remote PED, install the **Remote PED** component as well (Windows only).
2. Navigate to the Luna HSM Client home directory (**/usr/safenet/lunaclient/rbs/bin** on Linux/Unix) and generate a certificate for the RBS host.

   > **rbs --genkey**

   You are prompted to enter and confirm an RBS password. The certificate is generated in:

   - Linux/UNIX: <LunaClient_install_directory>/rbs/server/**server.pem**
   - Windows: <LunaClient_install_directory>\cert\server**server.pem**
3. Specify the Backup HSM(s) that RBS will make available to clients.

> **rbs --config**

RBS displays a list of Backup HSMs currently connected to the workstation. Select the ones you want to provide remote backup services. When you have specified your selection, enter **X** to exit the configuration tool.

4. Launch the RBS daemon (Linux/UNIX) or console application (Windows).
   - Linux/UNIX: # **rbs --daemon**
   - Windows: Double-click the **rbs** application. A console window will remain open.

   You are prompted to enter the RBS password.
5. Securely transfer the RBS host certificate (**server.pem**) to your Luna HSM Client workstation using **pscp** or **sftp**.
6. On the client workstation, register the RBS host certificate to the server list.

   > **vtl addServer -n** <Backup_host_IP> **-c server.pem**
7. [Optional] Launch LunaCM on the client to confirm that the Backup HSM appears as an available slot.

> **NOTE** If you encounter issues, try changing the RBS and PEDclient ports from their default values. Check that your firewall is not blocking ports used by the service.

You can now use the Backup HSM(s) as though they were connected to the client workstation locally, using Remote PED.

CHAPTER 15: Slot Numbering and Behavior

Administrative partitions and application partitions are identified as PKCS#11 cryptographic slots in Thales utilities, such as LunaCM and **multitoken**, and for applications that use the Luna library.

Order of Occurrence for Different Luna HSMs

A host computer with Luna HSM Client software and Luna libraries installed can have Luna HSMs connected in any of three ways:

- PCIe embedded/inserted Luna PCIe HSM 7 card (one or multiple HSMs installed - administrative partitions and application partitions are shown separately)
- USB-connected Luna USB HSM 7s (one or multiple - administrative partitions and application partitions are shown separately)
- Luna Network HSM 7 application partitions*, registered and connected via NTLS or STC.

Any connected HSM partitions are shown as numbered slots. Slots are numbered from zero or from one, depending on configuration settings (see "Settings Affecting Slot Order" on the next page, below), and on the firmware version of the HSM(s).

* One or multiple application partitions. Administrative partitions on Luna Network HSM 7s are not visible via LunaCM or other client-side tools. Only registered, connected application partitions are visible. The number of visible partitions (up to 100) depends on your model's capabilities. That is, a remote Luna Network HSM 7 might support 100 application partitions, but your application and LunaCM will see only partitions that have established certificate-exchange NTLS links or STC links with the current Client computer.

In LunaCM, a slot list would normally show:

- Luna Network HSM 7 application partitions for which NTLS links or STC links are established with the current host, followed by
- Luna PCIe HSM 7 cards, followed by
- Luna USB HSM 7s

For Luna Network HSM 7, as seen from a client (via NTLS), only application partitions are visible. The HSM administrative partition of a remote Luna Network HSM 7 is never seen by a Luna HSM Client. The Luna Network HSM 7 slots are listed in the order they are polled, dictated by the entries in the **Luna Network HSM** section of the Crystoki.ini / chrystoki.conf file, like this:

```
ServerName00=192.20.17.200
ServerPort00=1792
ServerName01=192.20.17.220
ServerPort01=1793
```

For Luna PCIe HSM 7 and Luna USB HSM 7, if you have multiple of either HSM type connected on a single host, then the order in which they appear is the hardware slot number, as discovered by the host computer.

For Luna PCIe HSM 7 and Luna USB HSM 7, the HSM administrative slot always appears immediately after the application partition. If no application partition has yet been created, a space is reserved for it, in the slot numbering.

Settings Affecting Slot Order

Settings in the **Presentation** section of the configuration file (Chrystoki.conf for UNIX/Linux, crystoki.ini for Windows) can affect the numbering that the API presents to Luna tools (like LunaCM) or to your application.

[Presentation]
ShowUserSlots=<slot>(<serialnumber>)

- Sets starting slot for the identified partition.
- Default, when ShowUserSlots is not specified, is that all available partitions are visible and appear in default order.
- Can be applied, individually, to multiple partitions, by a single entry containing a comma-separated list (with partition serial numbers in brackets):
  ShowUserSlots=1(351970018022), 2(351970018021), 3(351970018020),....
- If multiple partitions on the same HSM are connected to the Luna HSM Client host computer, redirecting one of those partitions with ShowUserSlots= causes all the others to disappear from the slot list, unless they are also explicitly re-ordered by the same configuration setting.

ShowAdminTokens=yes

- Default is yes. Admin partitions of local HSMs are visible in a slot listing.
- Remotely connected partitions (Luna Network HSM 7) are not affected by this setting, because NTLS connects only application partitions, not HSM SO (Admin) partitions to clients, so a Luna Network HSM SO administrative partition would never be visible in a client-side slot list, regardless.

ShowEmptySlots=1

- Controls how C_GetSlotList - as used by lunacm slot list command, or ckdemo command 14, and by your PKCS#11 application - displays, or does not display unused potential slots, when the number of partitions on an HSM is not at the limit.

OneBaseSlotId=1

- Causes basic slot list to start at slot number 1 (one) instead of default 0 (zero).
  (Any submitted number other than zero is treated as "1". Any letter or other non-numeric character is treated as "0".)

Effects of Settings on Slot List

Say, for example, you have multiple HSMs connected to your host computer (or installed inside), with any combination of Luna HSM Firmware 7.0.1 and newer, and no explicit entries exist for slot order in the config file. The defaults prevail and the slot list would start at zero.

If you set OneBaseSlotId=1 in the configuration file, then the slot list starts at "1" instead of at "0". You could set this for personal preference, or according to how your application might expect slot numbering to occur (or if you have existing scripted solutions that depend on slot numbering starting at zero or starting at one). OneBaseSlotId affects the starting number for all slots, including the HA virtual slot, regardless of firmware.

> **TIP VISIBILITY OF PARTITION SLOTS**
>
> Slot numbering is affected by setting the LunaCM command hagroup haonly (recommended) - with HAonly set, only the virtual slot of an HA group is visible *to your applications* ***see Note below***. This is important if your application relies on stable slot numbers to access partitions or services. HAonly locks the virtual slot number, such that it remains fixed when other slots are added or removed (including the primary for a group).
>
> In situations where you have multiple HA groups configured on a client, and hagroup haonly is applied, each of the several resulting virtual slots remains in its number/position as members of any group are added/removed/dropped/restored. However the numbering *would* change if you explicitly deleted an entire group from the client. In that case, the assumption is that it's a planned activity and you are prepared for movement of other slot number assignments.
>
> If HAonly is not set, then removing or adding a physical slot causes slots to renumber, including the HA virtual slot, which might not be what you want.
>
> Individual partition slots remain visible in LunaCM when HA Only mode is enabled. They are hidden only from *client* applications. Use **CKdemo** (Option **11**) to see the slot numbers to use with client applications.

If you set ShowUserSlots=20(17923506), then the identified token or HSM or application partition would appear at slot 20, regardless of the locations of other HSMs and partitions.

Options for an application to access a partition

Review the other sections on this page, while considering the requirements of your application, to decide how your application will access a partition.

> Identify a partition by reference to the partition label
> For example, the Java keytool utility references by label when using the “tokenlabel:” option in the keystore file [Keytool Usage and Examples]
>
> - unaffected by changes to slot numbering
> - partitions might inadvertently be given the same label.

> Identify a partition by referring to its slot number
> For example, the Java keytool utility references by slot nmber when using the “slot:” option in the keystore file [Keytool Usage and Examples]
>
> - a reliable identifier while slot numbers are stable - if HA is invoked, the hagroup haonly command removes physical slots from view by your application and locks the slot number of the virtual partition.
> - slot numbering can change if physical partitions are exposed and a partition is added to, or removed from, the slot list.

> Identify a partition by its serial number (see Note below)
>
> - always a unique identifier, and is unaffected by changes to slot numbering,
> - your application might not include that ability (example, use of KSP or EKM in an application integration).

> **NOTE** For developers, you could use C_getTokenInfo to get a partition's serial number.

Effects of New Firmware on Slot Login State

Slots retain login state when current-slot focus changes. You can use the LunaCM command **slot set** to shift focus among slots, and whatever login state existed when you were previously focused on a slot is still in effect when you return to that slot.