THALES

Thales Luna PCIe HSM 7

HSM ADMINISTRATION GUIDE

Document Information

Last Updated	2024-09-10 13:42:22 GMT-04:00

Trademarks, Copyrights, and Third-Party Software

Copyright 2001-2024 Thales Group. All rights reserved. Thales and the Thales logo are trademarks and service marks of Thales and/or its subsidiaries and are registered in certain countries. All other trademarks and service marks, whether registered or not in specific countries, are the property of their respective owners.

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.

All intellectual property is protected by copyright. All trademarks and product names used or referred to are the copyright of their respective owners. No part of this document may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, chemical, photocopy, recording or otherwise without the prior written permission of Thales Group.

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 HSM Administration Guide 11
- Customer Release Notes 11
- Audience 12
- Document Conventions 12
- Support Contacts 14

Chapter 1: Luna PCIe HSM 7 Hardware Installation 15
- Server Compatibility 15
  - Compatibility Issue with Intel Ice Lake CPUs 15
- Verifying the Integrity of Your Shipment 17
- Luna PCIe HSM 7 Required Items 19
  - Basic Luna PCIe HSM 7 Order Items 19
  - Multifactor Quorum-Authenticated Luna PCIe HSM 7 Order Items 20
  - Other Required Multifactor Quorum-Authentication Items 21
  - Optional Items 22
- Installing the Luna PCIe HSM 7 Hardware 23
  - Installing the Luna PCIe HSM 7 Card Into the Host Computer 23
  - Connecting a Chassis Intrusion Connector to the Tamper Header 25
  - Connecting a Local PED 26
  - Connecting a Remote PED 26
- Replacing the Luna PCIe HSM 7 Battery 27
  - Required Items 27
  - Prerequisites 28
  - Replacing the Luna PCIe HSM 7 Battery 28

Chapter 3: Luna HSM Client Software Installation 30
- Windows Luna HSM Client Installation 31
  - Command line options overview 31
  - Installing all components and features 34
  - Installing the Luna HSM Client for the Luna Network HSM 7 35
  - Installing the Luna HSM Client for the Luna PCIe HSM 7 35
  - Installing the Luna HSM Client for the Luna USB HSM 7 35
  - Installing the Luna HSM Client for the Luna Backup HSM 36
  - Installing the Luna HSM Client for Remote PED 36
  - Installation Location 36
  - ChrystokiConfigurationPath Environment Variable 37
  - Logging 37
  - Uninstalling the Luna HSM Client 37
- Windows Interactive Luna HSM Client Installation 38
  - Required Client Software 38

Prerequisites ... 38
Installing the Luna HSM Client Software ... 39
Modifying the Installed Windows Luna HSM Client Software ... 42
Java ... 43
Luna CSP and KSP ... 43
USB-powered PED ... 44
Modifying the Number of Luna Backup HSM Slots ... 44
Uninstalling the Luna HSM Client Software ... 45
After Installation ... 47
Troubleshooting ... 47
Linux Luna HSM Client Installation ... 48
Prerequisites ... 48
Where to install, and SELinux ... 49
About Installing the Luna HSM Client Software ... 50
Scripted or Unattended Installation ... 52
Interrupting the Installation ... 53
Controlling User Access to Your Attached HSMs and Partitions ... 56
Uninstalling the Luna HSM Client Software or Removing Components ... 57
Java ... 57
Modifying the Number of Luna Backup HSM Slots ... 57
Effects of Kernel Upgrades ... 58
Troubleshooting ... 58
Adding a Luna Cloud HSM Service ... 58
Initializing a Luna Cloud HSM Service ... 60
Dynamic Partition Loading for Luna Cloud HSM Services ... 61
Configuration File Summary ... 63
Dynamic UserID Loading for Luna Cloud HSM Services ... 86
Updating the Luna HSM Client Software ... 87

Chapter 4: Secure Transport Mode ... 88
When STM is enabled on the HSM ... 88
Recovering an HSM From Secure Transport Mode ... 89
Placing an HSM In Secure Transport Mode ... 90

Chapter 5: Multifactor Quorum Authentication ... 92
Multifactor Quorum Authentication Architecture ... 93
Comparing Password and Multifactor Quorum Authentication ... 93
PED keys ... 94
PED key Types and Roles ... 94
Shared PED key Secrets ... 96
PINs ... 97
Quorum Split Secrets (M of N) ... 97
Updated Luna PED Behavior Notes ... 99
New-series Luna PED Behavior Notes ... 99
Updating or Rolling Back Multifactor Quorum-Authenticated HSM Firmware ... 100
Luna PED Received Items ... 100
Other Required Multifactor Quorum-Authentication Items ... 102

Luna PED Hardware Functions 102
Physical Features 103
Keypad Functions 104
Modes of Operation 104
Luna PED with Newer CPU (External Power Supply Now Optional) 105
Local PED Setup 106
Setting Up a Local PED Connection 106
PED Actions 107
Local PED Troubleshooting 107
Secure Local PED 108
Secure Communication Between the Local PED and Luna PCIe HSM 7s With Firmware 7.7.0 and Newer 108
About Remote PED 108
Remote PED Architecture 109
Remote PED Connections 110
Secure Communication Between the Remote PED and Luna PCIe HSM 7s With Firmware 7.7.0 and Newer 112
Secure Communication Between the Remote PED and Luna PCIe HSM 7s With Firmware 7.4.2 and Older 113
Initializing the Remote PED Vector and Creating an Orange Remote PED key 114
Local RPV Initialization 114
Remote RPV Initialization 115
Rotating or Re-Initializing the Orange Remote PED key 116
Installing PEDserver and Setting Up the Remote Luna PED 117
Opening a Remote PED Connection 118
To open a Remote PED connection 118
Ending or Switching the Remote PED Connection 121
Configuring PED Timeout Settings 121
Configuring PED Inactivity Timeout 121
Configuring PED key Interaction Timeout 122
Configuring Luna PED Operation Timeout 122
Remote PED Troubleshooting 122
Updating External Supply-Powered Luna PED Firmware 126
Files Included in the Upgrade Package 126
Preparing for the Update 127
Updating the Luna PED Firmware 128
Troubleshooting 129
Updating USB-Powered Luna PED Firmware 130
Preparing for the Upgrade 130
Upgrading the Luna PED Firmware to Version 2.9.0 (or newer) 131
Multifactor Quorum PED key Management 132
Creating PED keys 132
Performing Multifactor Quorum Authentication 138
Consequences of Losing PED keys 140
Blue HSM SO PED key 140
Red HSM Domain PED key 141
Orange Remote PED key 141

Blue Partition SO PED key 141
Red Partition Domain PED key 141
Black Crypto Officer PED key 142
Gray Crypto User PED key 142
White Audit User PED key 142
Identifying the PED key Secret 142
Duplicating Existing PED keys 143
Changing the PED key Secret 144
Blue HSM SO PED key 145
Red HSM Domain PED key 145
Orange Remote PED Vector PED key 145
Blue Partition SO PED key 145
Red Partition Domain PED key 146
Black Crypto Officer PED key 146
Gray Crypto User PED key 146
White Audit User PED key 147
PEDserver and PEDclient 147
The PEDserver Utility 147
The PEDclient Utility 148
pedserver 149
pedserver -appliance 150
pedserver -appliance delete 151
pedserver -appliance list 152
pedserver -appliance register 153
pedserver mode 154
pedserver -mode config 155
pedserver -mode connect 157
pedserver -mode disconnect 158
pedserver -mode show 159
pedserver -mode start 161
pedserver -mode stop 163
pedserver -regen 164
pedclient 164
pedclient -mode assignid 166
pedclient -mode config 167
pedclient -mode deleteid 169
pedclient -mode releaseid 170
pedclient -mode setid 171
pedclient -mode show 172
pedclient -mode start 173
pedclient -mode stop 175
pedclient -mode testid 176

Chapter 6: Audit Logging 177
Audit Logging Features 177
Audit limitations and Controlled tamper recovery state 182
The Audit Role 182

Audit Log Secret 183
Audit Log Records 184
Audit Log Message Format 185
Timestamping 186
Log Capacity 186
Audit Logging Stops Working if the Current Log File is Deleted 187
Audit Logging General Advice and Recommendations 187
Logging In as Auditor 189
Configuring and Using Audit Logging 190
Configuring Audit Logging 190
Exporting the Audit Logging Secret and Importing to a Verifying HSM 192
Audit Role Authentication Considerations 193
Audit Log Categories and HSM Events 194
Partition Role IDs 194
HSM Access 194
Log External 196
HSM Management 196
Key Management 197
Key Usage and Key First Usage 199
Per-Key Authorization 200
Audit Log Management 200
Audit Log Troubleshooting 201

Chapter 7: Initializing the HSM 202
Hard versus soft initialization 202
Initializing a New or Factory-reset HSM 202
Re-initializing the HSM 205

Chapter 8: HSM Roles 206
Administrator (AD) 207
Logging In as HSM Security Officer 207
Changing a Role Credential 207
Name, Label, and Password Requirements 208
HSM Labels 209
Cloning Domains 209
Partition Labels 209
Role Passwords or Challenge Secrets 209

Chapter 9: HSM Capabilities and Policies 210
Setting HSM Policies Manually 223
Setting HSM Policies Using a Template 224
Creating an HSM Policy Template 224
Editing an HSM Policy Template 225
Applying an HSM Policy Template 226

Chapter 10: Application Partitions 227
Creating or Deleting an Application Partition 227

Chapter 11: Security in Operation 229
Tamper Events 229
Recovering from a Tamper Event 230
Security Effects of Administrative Actions 231
Overt Security Actions 231
Actions with Security- and Content-Affecting Outcomes 232
Elsewhere 236

Chapter 12: Monitoring the HSM 237
HSM Status Values 237
System Operational and Error Messages 238

Chapter 12: SNMP Monitoring 241
MIB 241
Installing the Luna SNMP Subagent 241
The SafeNet Chrysalis-UTSP MIB 243
The Luna HSM MIB 244
hsmPolicyTable 247
hsmPartitionPolicyTable 248
hsmClientRegistrationTable 248
hsmClientPartitionAssignmentTable 248
SNMP output compared to Luna tools output 248
Frequently Asked Questions 249
Performance Monitoring 250
Partition Utilization Metrics 250
Rules of acquisition 251
Availability of Partition Utilization Metrics 251
Cryptographic Module and Token Return Codes 252
Library Codes 271
Vendor-Defined Return Codes 275
HSM Alarm Codes 281
Alarm Generation and Handling 281
FRAM LOG 282
List of HSM Alarm Codes 283
HSM Alarm Code Samples 288
Temperature - High Warning 288
Temperature – High Soft Tamper 289
Temperature – High Hard Tamper 289
Hard Tampers During Storage 290
Decommission with power on 290
Decommission with power off 291
Chassis open with power on 291
Chassis open with power off 291
Card removal 292
Stored Data Integrity 292

Chapter 13: HSM Updates and Upgrades 294

Updating the Luna PCIe HSM 7 Firmware 294
Changing the Firmware Upgrade Permissions (Linux only) 295
Rolling Back the Luna HSM Firmware 295
Upgrading HSM Capabilities 296

Chapter 14: Functionality Modules 298
FM Deployment Constraints 298
FMs and FIPS Mode 299
FMs and High-Availability (HA) 299
FMs and Backup/Restore/Cloning 300
FMs and HSM Firmware Rollback 300
FM Configuration and Remote PED 300
FM-Enabled HSM Cannot be Verified With CMU 301
Key Attributes 301
No EDDSA or EC_MONTGOMERY Private Keys with C_CreateObject 301
FM Sample Applications Dependent on General Cryptoki Samples 301
Memory for FMs 301
Preparing the Luna PCIe HSM 7 to Use FMs 301
Step 1: Ensure You Have FM-Ready Hardware 302
Step 2: Update Your HSM 302
Step 3: Purchase and Apply the FM Capability License 302
Step 4: Apply HSM Policy Settings 303
Building and Signing an FM 304
Loading an FM Into the HSM Firmware 307
Deleting an FM From the HSM Firmware 308
Recovering the HSM After FM Failure 309
Effects of Administrative Actions on Functionality Modules 310

Chapter 15: Zeroizing or Resetting the HSM to Factory Conditions 312
HSM Zeroization 312
Decommissioning the HSM Card 313
Disabling Decommissioning 314
Resetting the Luna PCIe HSM 7 to Factory Condition 314
Comparing Zeroize, Decommission, and Factory Reset 315
Comparison of Destruction/Denial Actions 315
Stored Data Integrity 317
RMA and Shipping Back to Thales 317
RMA Process for Thales Luna HSM Devices Containing Sensitive Customer Key Material 318
Secure RMA Without Access to Key Material 318
End of Service and Disposal 318

PREFACE: About the HSM Administration Guide

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

> "Luna PCIe HSM 7 Hardware Installation" on page 15

> "Luna HSM Client Software Installation" on page 30

> "Secure Transport Mode" on page 88

> "Multifactor Quorum Authentication" on page 92

> "Audit Logging" on page 177

> "Initializing the HSM" on page 202

> "HSM Roles" on page 206

> "HSM Capabilities and Policies" on page 210

> "Application Partitions" on page 227

> "Security in Operation" on page 229

> "Monitoring the HSM" on page 237

> "HSM Updates and Upgrades" on page 294

> "Functionality Modules" on page 298

> "Zeroizing or Resetting the HSM to Factory Conditions" on page 312

The preface includes the following information about this document:

> "Customer Release Notes" below

> "Audience" on the next page

> "Document Conventions" on the next page

> "Support Contacts" on page 14

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 PCIe HSM 7 Hardware Installation

This chapter describes how to install and connect a Luna PCIe HSM 7. To ensure a successful installation, perform the following tasks in the order indicated:

1. Before unpacking your new hardware, refer to "Verifying the Integrity of Your Shipment" on page 17 for safe unpacking instructions.
2. Ensure that you have all of the required components, as listed in "Luna PCIe HSM 7 Required Items" on page 19
3. Install and connect the hardware, as described in "Installing the Luna PCIe HSM 7 Hardware" on page 23
4. The Luna PCIe HSM 7 uses a 3.6V non-rechargeable lithium battery to provide backup power to its memory. If you need to replace this battery, see "Replacing the Luna PCIe HSM 7 Battery" on page 27.

> **CAUTION!** This product uses semiconductors that can be damaged by electro-static discharge (ESD). When handling the device, avoid contact with exposed components, and always use an anti-static wrist strap connected to an earth ground. In rare cases, ESD can trigger a tamper or decommission event on the HSM. If this happens, all existing roles and cryptographic objects are deleted.

The Luna PCIe HSM 7 has been tested with a variety of representative systems/servers with compliant PCI express slots. If you encounter any issues when installing the Luna PCIe HSM 7 into a new server/host computer, first try a different PCI express slot. The design of certain motherboards or the associated BIOS may prevent proper communication with a Luna PCIe HSM 7. For example, certain PCI express physical slots are intended for use only with a video card or another specific type of hardware, and the Luna PCIe HSM 7 may not work correctly in these slots.

If you encounter further issues, please contact Thales Technical Support.

Server Compatibility

The Luna PCIe HSM 7 conforms to the PCIe 2.0 standard and requires a PCIe x4 or higher slot.

> **NOTE** Do not install the Luna PCIe HSM 7 into a slot reserved for a dedicated function, such as video. If you do, the host system might not boot successfully.

Compatibility Issue with Intel Ice Lake CPUs

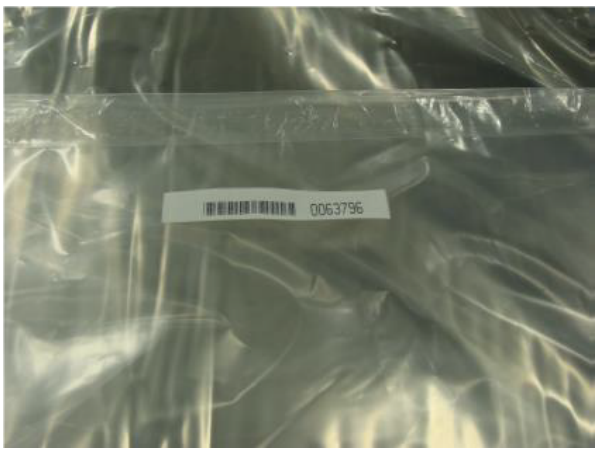
The Luna PCIe HSM 7 conforms to the PCIe 2.0 standard and requires a PCIe x4 or higher slot. When installed in a server using an Intel Ice Lake CPU, the Luna PCIe HSM 7 7 might not boot up after a hot reboot of the server. This issue has been confirmed on Dell, HP, and Fujitsu servers. Please ensure that you install the Luna

PCIe HSM 7 7 in a server with a newer (Rocket Lake, for example) or older (Cascade Lake, for example) CPU.

If you have the Luna PCIe HSM 7 installed in an Ice Lake server, no issues have been found using cold reboots (startup from shutdown). If you choose to use a remote server management system such as iDRAC, ensure that you have assessed the security risks associated with this deployment, and that the system is properly maintained (security patches applied).

Verifying the Integrity of Your Shipment

> **CAUTION!** Thales employs a number of security measures to allow you to verify that your new hardware was not intercepted in transit or otherwise tampered with before you received it. To verify the authenticity and handling history of your received items, review the following checklist before you unpack your new hardware, and then follow the checklist as you unpack each item.

Step	Yes	No
1. Do the items received (individual items, part numbers) match those listed in the enclosed packing list? If yes, go to the next step. If no, contact Thales support.		
2. Before you received the product, did you receive an advanced shipping notification providing details regarding the shipment (part numbers and serial numbers for the product, tamper-evident bags, and tamper seals)? If yes, go to the next step. If no, contact Thales support.		
3. Are all of the tamper-evident bag serial numbers that are listed in the advanced shipping notification present, and do they match the actual bags received? The tamper-evident bag serial numbers appear as shown below.  If yes, go to the next step.If no, contact Thales support.		
4. Did you receive any tamper-evident bags that are *not* listed on the advance shipping notification? If yes, contact Thales support. If no, go to the next step.		

Step	Yes	No
5. Are the correct number of tamper seals affixed to the tamper-evident bag? There should be a tamper seal on each side of the bag and two seals over the bag opening. If no, contact Thales support. If yes, go to the next step.		
6. Are all of the tamper seal serial numbers listed in the advanced shipping notification present, and do they match the actual seals affixed to the device? If yes, go to the next step. If no, contact Thales support.		
7. Are there any tamper seals affixed to the device that are *not* listed on the advance shipping notification? If yes, contact Thales support. If no, go to the next step.		
8. Are there any signs of physical tampering? The tamper seals on the sides indicate tampering if they show the **ALERT** markings as illustrated. If yes, contact Thales support. If no, go to the next step.		
9. Once you have verified all of the received items, you can proceed with the installation.		

Luna PCIe HSM 7 Required Items

This section provides a list of the components you should have received with your Luna PCIe HSM 7 order. The specific items you received depend on whether you ordered a password-authenticated or a multifactor quorum-authenticated Luna PCIe HSM 7, and whether your order included a backup device or other options as described below.

Basic Luna PCIe HSM 7 Order Items

The standard items that you should have received as your basic order for a Luna PCIe HSM 7 are:

Qty	Item
1	**Luna PCIe HSM 7** The HSM comes fitted with a full-height mounting bracket.
1	**Anti-Static Wrist Strap**

Qty	Item
1	**Half-height mounting bracket** Use this bracket if you need to install the Luna PCIe HSM 7 into a half-height slot.

Multifactor Quorum-Authenticated Luna PCIe HSM 7 Order Items

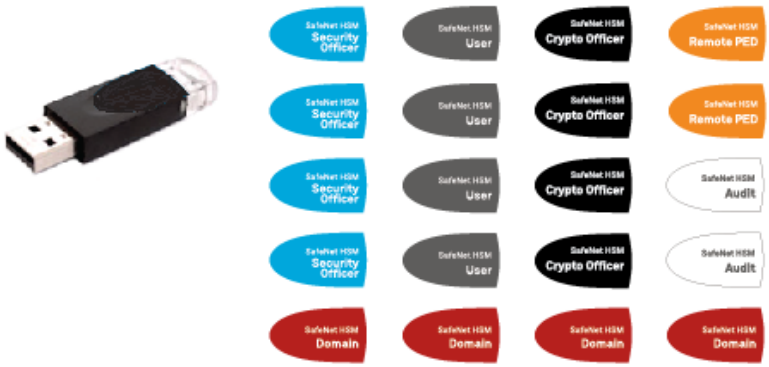
If you ordered a multifactor quorum-authenticated Luna PCIe HSM 7, you should have received some combination of the following items in addition to the items in the basic order.

Qty	Item
1+	**Luna PED** Your order should include at least one Luna PED device. If you intend to back up your Luna PCIe HSM 7 to a Luna Backup HSM, then you require a Luna PED to connect to that Backup HSM. If you intend to combine remote operation and backup, you might prefer to have a second Luna PED. It is possible to use a single Luna PED for both connections, and to simply change between local and remote mode as needed. Note that you can use PED keys that you already own and use with other HSMs if appropriate. You should purchase the number you need for your own convenient operation, and for backup/standby units as your security policies might require.

Qty	Item
1	**Luna PED cable** The Luna PED device connects to your HSM using a Type A to Mini B USB cable.
1	**Luna PED Power Supply Kit *** If you ordered a Luna PED, your order should include a power supply kit with the appropriate connection for your region. The power supply is auto-sensing and includes replaceable mains plug modules for international use. * If you received a USB-powered Luna PED (updated internal hardware, and Luna PED Firmware 2.8.0 or newer), it is powered via the USB connection and does not require a separate, external power supply; none is supplied.

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.

Optional Items

Your order may include a Luna Backup HSM 7.

Qty	Item
1	**Luna Backup HSM 7** You can back up your selected Luna PCIe HSM 7 partition contents (root keys, certificates, other items) to a Luna Backup HSM. The Luna Backup HSM is suitable for off-site storage and for backing up multiple HSM partitions. It can back up contents of password-authenticated or of multifactor quorum-authenticated HSMs. Refer to Luna Backup HSM 7.

Installing the Luna PCIe HSM 7 Hardware

This section describes how to perform the following tasks:

> Install the Luna PCIe HSM 7 card into the host computer. See "Installing the Luna PCIe HSM 7 Card Into the Host Computer" below.

> Connect a chassis intrusion connector to the tamper header on the card, if necessary. See "Connecting a Chassis Intrusion Connector to the Tamper Header" on page 25

> Connect a local PED, if necessary. See "Connecting a Local PED" on page 26

> Connect a remote PED, if necessary. See "Connecting a Remote PED" on page 26

Installing the Luna PCIe HSM 7 Card Into the Host Computer

Install the Luna PCIe HSM 7 card into an open PCIe slot on the host computer.

> **CAUTION!** This product uses semiconductors that can be damaged by electro-static discharge (ESD). When handling the device, avoid contact with exposed components, and always use an anti-static wrist strap connected to an earth ground. In rare cases, ESD can trigger a tamper or decommission event on the HSM. If this happens, all existing roles and cryptographic objects are deleted.

Prerequisites

> Ensure that the PCIe slot is unpowered before you proceed with the installation.

> The Luna PCIe HSM 7 conforms to the PCIe 2.0 standard and requires a PCIe x4 or higher slot. See also "Server Compatibility" on page 15.

> **NOTE** Do not install the Luna PCIe HSM 7 into a slot reserved for a dedicated function, such as video. If you do, the host system might not boot successfully.

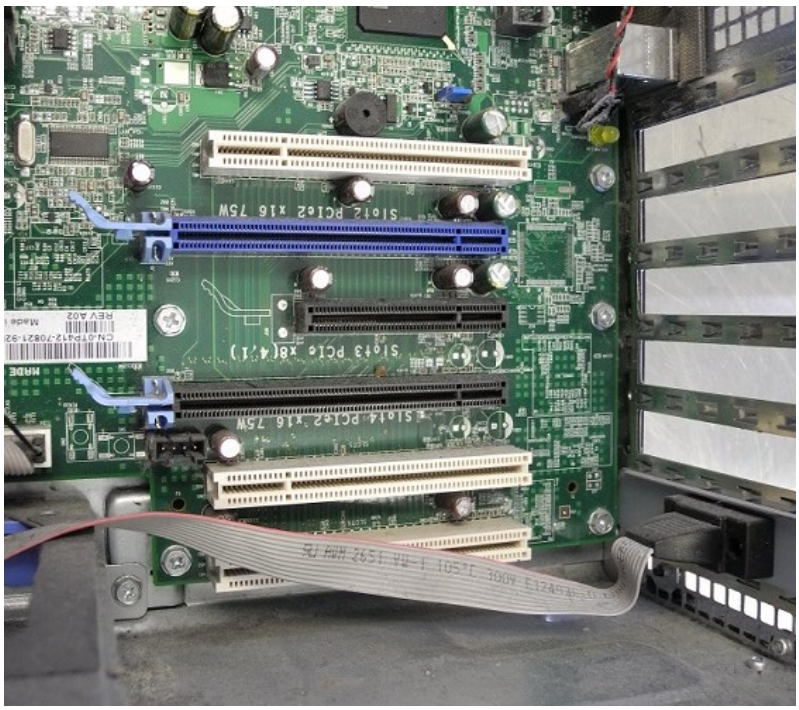
To install the Luna PCIe HSM 7 hardware

1. Open your computer, and remove the slot-cover bracket from an available PCIe slot. If the bracket is secured by a screw, keep that screw.
2. Use the provided anti-static wrist-strap to ground yourself to an exposed metal part of the computer chassis.
3. Remove the Luna PCIe HSM 7 from its anti-static packaging and prepare to insert the card into your computer.

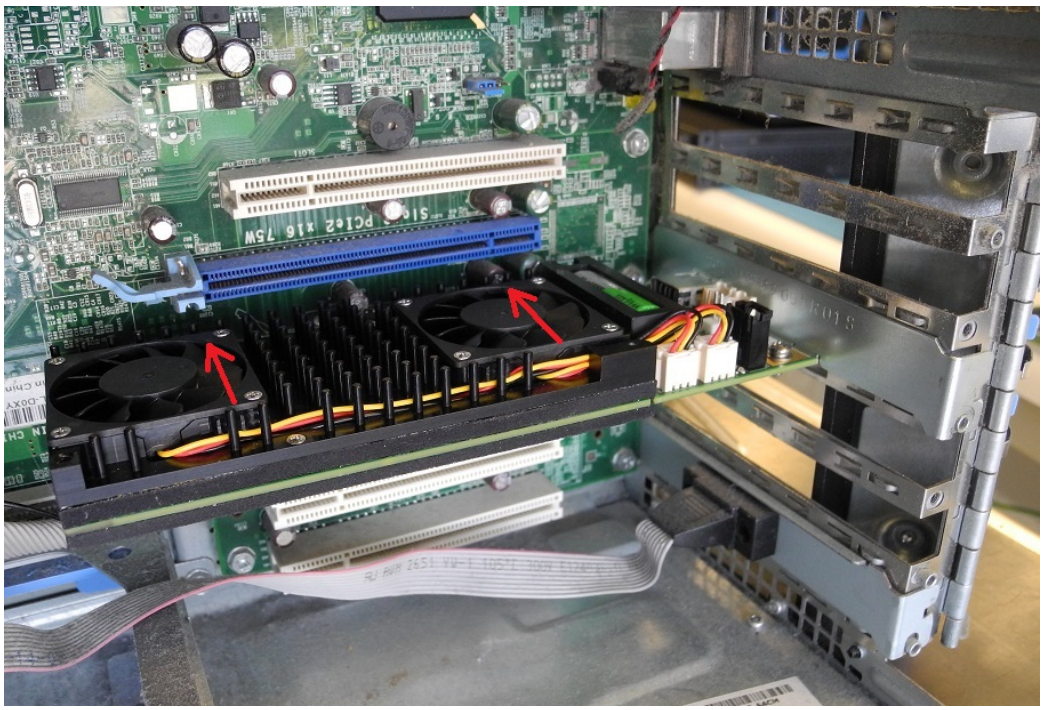
   Your Luna PCIe HSM 7 comes fitted with a full-height mounting bracket, but if you have no full-height slots available, the card can fit into a half-height slot. A half-height mounting bracket is included for this purpose. To install the half-height bracket, remove the two screws connecting the full-height bracket to the card, and use them to mount the half-height bracket in its place.

4. Align the Luna PCIe HSM 7 card with the vacant, unpowered slot. You might need to introduce the tip of the card-hold-down bracket first (the silver-metal part along the back edge of the card), in order to properly align the card with the connector.

   You can use a PCIe X4 or larger slot, as long as it is wired for at least four PCI express channels, and not reserved for a dedicated function. For example, we do not recommend that you use your Luna PCIe HSM 7 card in a designated PCI express video slot - different models of computer and their BIOS firmware can differ in how faithfully they support the PCIe standard.



5. Insert the Luna PCIe HSM 7 card into the connector. It should go straight in – angling the card might cause it to bend. The card is properly seated when no portion of the gold-colored contacts of the card-edge protrudes above the connector socket.



6. Secure the card hold-down bracket with a screw or other restraint, as appropriate in your computer.

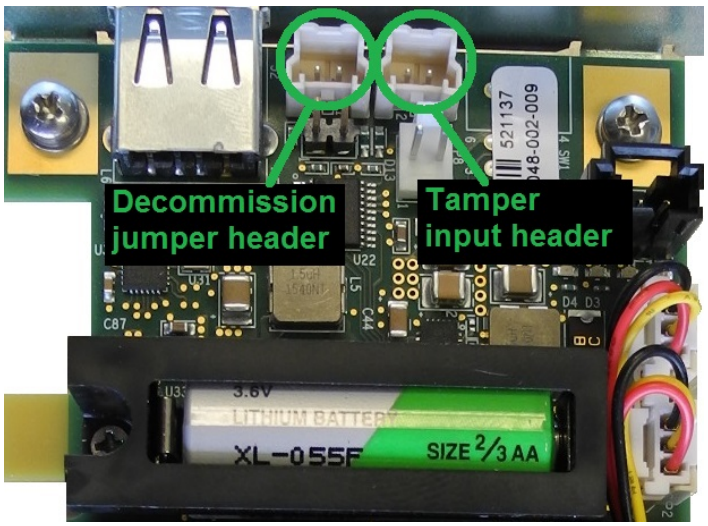
Connecting a Chassis Intrusion Connector to the Tamper Header

The Luna PCIe HSM 7 is equipped with a two-pin tamper header which, when shorted, places the HSM in a tamper state with a status of Chassis Open. If your chassis is so equipped, you can connect the chassis intrusion connector to the tamper header so that the HSM is placed in a tamper state if the chassis is opened. Refer to the documentation provided by your chassis manufacturer for more information.

To connect a chassis intrusion connector to the tamper header

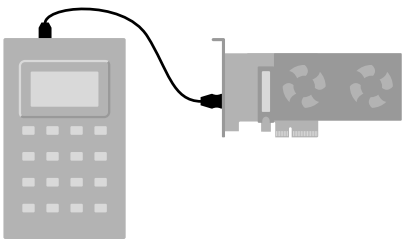
1. Install the card as described in "Installing the Luna PCIe HSM 7 Card Into the Host Computer" on page 23.
2. Connect the chassis intrusion connector to the tamper input header on the card, shown below.

> **NOTE** If used, this pin pair would usually be wired to a chassis switch that is held open when the lid or panel is in place. Opening the lid or panel would allow the switch to close, and tamper the HSM. If you are constructing or ordering a cable for this purpose, the header has 2mm pin pitch and mates with a Molex connector (https://www.molex.com/molex/products/datasheet.jsp?part=active/0355070200_CRIMP_HOUSINGS.xml) or equivalent.



Connecting a Local PED

The local Luna PED connects directly to the USB port on the Luna PCIe HSM 7 card via a USB-to-MiniUSB cable.



To connect a local PED to the Luna PCIe HSM 7

1. Use the Luna PED local cable (mini-USB to USB) to connect the Luna PED to the Luna PCIe HSM 7 card:
   a. Plug the mini-USB connector on the cable into the mini-USB port on the PED.
   b. Plug the USB connector on the cable into the USB port on the card.

Connecting a Remote PED

The Remote Luna PED can be used either locally, connected directly to a Luna HSM (exactly as for the standard PED), or remotely when connected to a suitable workstation and to the electrical main power supply. The normal local use of a PED with Remote PED capability is to use it in local mode to prepare an HSM (imprint an RPK – the orange key with a Remote PED Vector) before shipping it to its remote location. Then you would switch to Remote PED mode.

To prepare an HSM for Remote PED operation you need to connect it locally and imprint the HSM with a Remote PED key (orange). Once the HSM can be reached via remote desktop connection, and the HSM is associated with an orange PED key, all further configuration and administration can be performed remotely.

To connect a remote PED to the Luna PCIe HSM 7:

1. Use the Luna PED local cable to connect the Luna PED to the Luna PCIe HSM 7 card. This step is required to imprint the HSM with a Remote PED Vector (RPV) using the orange PED key (RPK). This should be the only time you need to connect a PED locally to the HSM. Once the orange PED key is imprinted with the same RPV as the HSM, all future PED operations can be performed remotely.
2. Follow the instructions in "About Remote PED" on page 108. Note that you must install at least the Remote PED optional component of the Luna HSM Client software before you can configure the remote PED. See "Luna HSM Client Software Installation" on page 30.

Replacing the Luna PCIe HSM 7 Battery

The Luna PCIe HSM 7 uses a 3.6 V non-rechargeable lithium battery to provide backup power to its memory. This enables the HSM to preserve cryptographic material even when the host system loses power. The battery may need replacement over the course of the HSM's lifetime. To see if your battery needs to be replaced, run **hsm envshow** in LunaCM.

The original battery, when installed in the Luna PCIe HSM 7, produced 3.6 V. A voltage regulator on the HSM card adjusts that voltage lower to be suitable for the tamper circuit. LunaCM reports the *regulated* voltage value. A warning is returned if the battery's regulated voltage drops below 2.75 V. When the battery falls below 2.5 V, the HSM is tampered and you must replace the battery before operations can resume. If **Policy 48: Do Controlled Tamper Recovery** is enabled, you cannot run the hsm tamperclear LunaCM command unless the battery is replaced. For more information about recovery after a tamper event, refer to "Recovering from a Tamper Event" on page 230. If the battery is not replaced, all key material will eventually be lost when the on-board RAM loses power and cannot maintain its memory.

> **CAUTION!** Unless temporary battery power is supplied to the HSM while the main battery is replaced, all cryptographic material will be erased. Use the Luna PCIe HSM 7 Temporary Battery Holder to ensure a continuous power supply.

Required Items

To replace the battery, you will need the following items. Battery manufacturer information is suggested.

> **CAUTION!** You must use the temporary battery specified. Standard AA voltage is too low to power the Luna PCIe HSM 7.

Qty	Item	Description/Specifications	Manufacturer	Part Number
1	Luna PCIe HSM 7 replacement battery	2/3AA, 3.6V, 1650 mAh, Li-$COCl_2$, length 33.5 mm, diameter 14.55 mm	**OmniCel**	ER14335/S
			Xeno Energy	XL-055F

Qty	Item	Description/Specifications	Manufacturer	Part Number
1	Temporary Battery Holder	Used with a temporary battery to maintain power to the Luna PCIe HSM 7 during the replacement process. Can be requested from Thales.	**Thales**	908-000408-001
1	Temporary Battery	AA, 3.6V, 1650 mAh, Li-$COCl_2$, length 50.3 mm, diameter 14.55 mm	**Saft**	LS14500-AA

Prerequisites

To replace the battery, you must first remove the Luna PCIe HSM 7 card from the host system.

> **CAUTION!** Back up any important cryptographic material on the HSM before proceeding. Removing the card from the host system will cause a tamper event. If **HSM policy 40: Decommission on Tamper** is enabled, the application partition and all roles are destroyed, and you must reconfigure the HSM after this procedure.
>
> This product uses semiconductors that can be damaged by electro-static discharge (ESD). When handling the device, avoid contact with exposed components, and always use an anti-static wrist strap connected to an earth ground. In rare cases, ESD can trigger a tamper or decommission event on the HSM. If this happens, all existing roles and cryptographic objects are deleted.

1. Test the temporary and replacement batteries with a voltmeter or multimeter. A measurement of 3.1 V taken directly from the battery terminals with a voltmeter is approximately equivalent to a regulated value of 2.75 V as reported by LunaCM, and indicates that the battery has reached the end of its useful life. Before proceeding with the battery replacement:
   - ensure that the *temporary battery* tests higher than 3.1 V with a voltmeter
   - for best result and battery life, ensure that the *replacement battery* tests near 3.6 V
2. [Optional] If the card will not be in your possession the entire time it is out of service, you can enable "Secure Transport Mode" on page 88. This allows you to know if the card has been tampered with while it was out of your possession.
3. Power off the host machine and disconnect it from power.
4. Use an anti-static wrist strap (provided with your Luna PCIe HSM 7) to ground yourself to an exposed metal part of the computer chassis.
5. Remove the Luna PCIe HSM 7 from its PCIe slot.

Replacing the Luna PCIe HSM 7 Battery

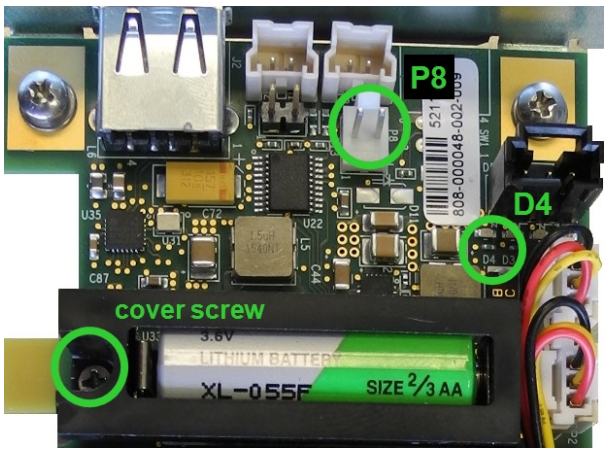
To maintain HSM power, you must connect a temporary battery while replacing the main battery.

To replace the Luna PCIe HSM 7 battery

1. Install the temporary battery in the temporary battery replacement holder.
2. Install the 2-pin plug from the battery holder onto the 2-pin header marked **P8** on the Luna PCIe HSM 7 card.

> **NOTE** The polarity on the **P8** header is not reversible. The jumper will only fit onto the header in the correct direction.

The Luna PCIe HSM 7 card's green **D4** LED is illuminated. This indicates that the card is receiving power from the temporary battery. If the LED appears dim, ensure that the temporary battery's voltage is greater than 3.1 V.



3. If necessary, remove the screw securing the battery cover.
4. Replace the 2/3AA battery on the card. Note the correct polarity.
5. Replace the battery cover and secure it with the screw.
6. Remove the jumper from the **P8** header to disconnect the temporary power.
7. Reinstall the Luna PCIe HSM 7 card.
8. Dispose of the depleted battery according to regional recycling regulations.

CHAPTER 3: 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" on the next page

- "Windows Interactive Luna HSM Client Installation" on page 38

> "Linux Luna HSM Client Installation" on page 48

> "Adding a Luna Cloud HSM Service" on page 58

> "Dynamic Partition Loading for Luna Cloud HSM Services" on page 61

> "Configuration File Summary" on page 63

> "Updating the Luna HSM Client Software" on page 87

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 38) 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:

> "Command line options overview" below

> "Installing the Luna HSM Client for the Luna Network HSM 7" on page 35

> "Installing the Luna HSM Client for the Luna PCIe HSM 7" on page 35

> "Installing the Luna HSM Client for the Luna USB HSM 7" on page 35

> "Installing the Luna HSM Client for the Luna Backup HSM" on page 36

> "Installing the Luna HSM Client for Remote PED" on page 36

> "Installation Location" on page 36

> "ChrystokiConfigurationPath Environment Variable" on page 37

> "Logging" on page 37

> "Uninstalling the Luna HSM Client" on page 37

If you want to perform an interactive installation, using the graphical, interactive installer, see "Windows Interactive Luna HSM Client Installation" on page 38

> **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 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
FMTOOLS	NETWORK, PCIe *	Tools for use when preparing Functionality Modules
JCProv	NETWORK, PCIe, USB	JCPROV PKCS#11

Feature identifier value	Can be installed with these Luna devices	Description
JSP	NETWORK, PCIe, USB	Java Provider component
SDK	NETWORK, PCIe, USB	Software SDK – Java / C++ samples
SNMP	PCIe, USB, Backup	SNMP subagent

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 63 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:

> "Required Client Software" below

> "Prerequisites" below

> "Installing the Luna HSM Client Software" on the next page

> "Modifying the Installed Windows Luna HSM Client Software" on page 42

> "Java" on page 43

> "Luna CSP and KSP" on page 43

> "Modifying the Number of Luna Backup HSM Slots" on page 44

> "Uninstalling the Luna HSM Client Software" on page 45

> "After Installation" on page 47

> "Troubleshooting" on page 47

> "Windows Luna HSM Client Installation" on page 31

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 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 43 for more information.

Interactive (prompted, this page) and non-interactive (no prompts "Windows Luna HSM Client Installation" on page 31) 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 47).
2. Uninstall any previous versions of the Client software before you proceed (see "Uninstalling the Luna HSM Client Software" on page 45).

> **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" on page 241.
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" on page 241.

Luna HSM Client
Version:
gemalto
security to be free
Welcome to the Luna HSM Client setup wizard
This setup wizard will install Luna HSM Client on your computer. Please customize the install below.
Install location:
C:\Program Files\SafeNet\LunaClient
Select folder
Install options:
Luna Devices
Select All
Network
PCIe
USB
Backup
Remote PED
Features
Select All
CSP (CAPI) / KSP (CNG)
JCE / JCA Provider (JSP)
PKCS #11 (JCProv)
Software SDK
SNMP Subagent
FM Tools
FM SDK
I agree to the terms of the Thales Software License Agreement.
INSTALL
QUIT

Luna HSM Client
Version:
gemalto
security to be free
Welcome to the Luna HSM Client setup wizard
This setup wizard will install Luna HSM Client on your computer. Please customize the install below.
Install location:
C:\Program Files\SafeNet\LunaClient
Select folder
Install options:
Luna Devices
Select All
Network
PCIe
USB
Backup
Remote PED
Features
Select All
CSP (CAPI) / KSP (CNG)
JCE / JCA Provider (JSP)
PKCS #11 (JCProv)
Software SDK
SNMP Subagent
FM Tools
FM SDK
I agree to the terms of the Thales Software License Agreement.
INSTALL
QUIT

> **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 63.

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 175) 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):

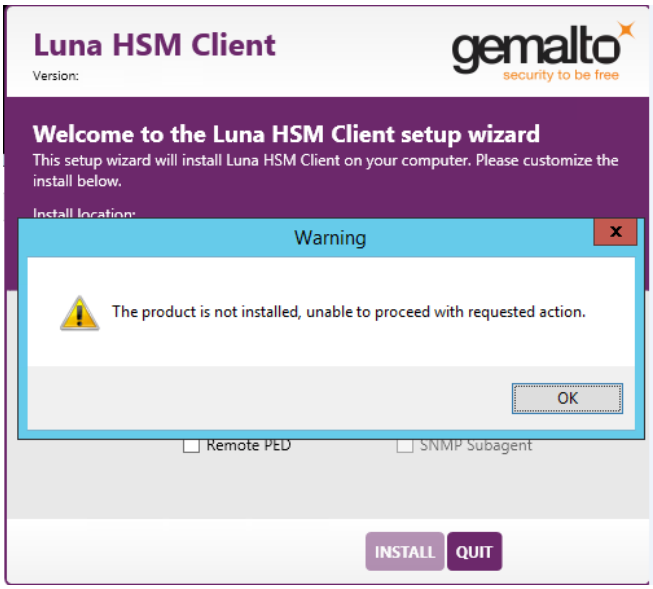
2. Click **Uninstall**. The client software is uninstalled.

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.

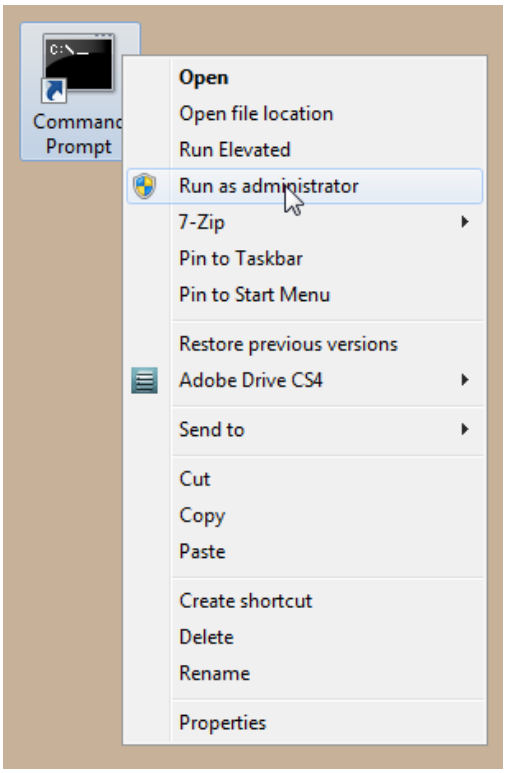
After Installation

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 31 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:

> "Prerequisites" below

> "Where to install, and SELinux" on the next page

> "About Installing the Luna HSM Client Software" on page 50

> "Scripted or Unattended Installation" on page 52

> "Controlling User Access to Your Attached HSMs and Partitions" on page 56

> "Uninstalling the Luna HSM Client Software or Removing Components" on page 57

> "Java" on page 57

> "Interrupting the Installation" on page 53

> "Modifying the Number of Luna Backup HSM Slots" on page 57

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 51). 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 52.
   - 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

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 63.

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.

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 31
- "Windows Interactive Luna HSM Client Installation" on page 38
- "Linux Luna HSM Client Installation" on page 48

> For more information on Luna/Luna Cloud HSM service compatibility, refer to Cloning Keys Between Luna 6, Luna 7, and Luna Cloud HSM.

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

> Initializing the Crypto Officer and Crypto User Roles

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.

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 42 (**Remote PED** package)

Linux: "About Installing the Luna HSM Client Software" on page 50 (**[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 104).

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

"pedserver -mode stop" on page 163

> 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 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

> Setting Up an HA Group

> Partition Backup and Restore

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 page 63.

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 58 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 38
  - "Linux Luna HSM Client Installation" on page 48
- 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" on the next page.

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**

Section/Setting	Description
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**
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**

Section/Setting	Description
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**
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**

Section/Setting	Description
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.	
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**
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. 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

Section/Setting	Description
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.
LNHServerClientCert##	The location and filename of the Luna HSM Client certificate used to access this Luna Network HSM 7 cluster.
LNHServerClientKey##	The location and filename of the Luna HSM Client private key used to access this Luna Network HSM 7 cluster.
LNHServerCAFile##	The location and filename of the Luna Network HSM 7 cluster server certificate.
LNHServerCN##	The Common Name for the cluster certificate.
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.
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.
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.
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.

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.
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
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). 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.
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.

Section/Setting	Description
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.
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)

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 an FM-enabled partition > **1**: the client allows login to an FM-enabled partition This entry is added to the configuration file the first time you initialize or log in to an FM-enabled partition using LunaCM. You are prompted to confirm that you want to allow login.

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**

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. **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**

Section/Setting	Description
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.
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/**

Section/Setting	Description
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.
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.
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.

Section/Setting	Description
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.
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.

Section/Setting	Description
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.
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 58 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 58 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.

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 30).

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 4: Secure Transport Mode

Luna HSM 7 units are shipped from the factory in Secure Transport Mode (STM). The purpose of STM is to provide a logical check on the HSM 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.

The Secure Transport Mode capability provides an additional layer of protection beyond the physical security controls provided by tamper-evident shipping bags.

Thales sends customers control validation information in two separate emails prior to shipment:

> **Physical security control validation** - an email containing the serial number of the HSM and the serial number of the associated tamper evident bag that encloses the HSM.

> **Logical control validation** - an email containing the serial number of each HSM in the shipment, along with the STM Random User String and the STM Verification String associated with each HSM.

Customers can use the logical and physical HSM controls to verify that HSMs shipped from the factory have not been modified in transit. The Thales shipping procedures are designed to prevent a possible man-in-the-middle attack, as attackers would need unobserved direct access to the HSM while in transit, along with simultaneous possession of both the STM Random User String and the STM Verification String for that HSM.

Thales customers can also implement STM when shipping pre-configured HSMs between their office locations or when pre-configured HSMs are to be put into storage. Customers implementing STM have added protection because only the HSM Security Officer can place an initialized HSM into STM, or recover the HSM from STM, further increasing the difficulty of man-in-the-middle attacks.

> **CAUTION!** Do not place the HSM into secure transport mode (STM) when there is already an active tamper. Such action would cause a mismatch of the verification string when the HSM is brought out of transport mode. Use hsm tamperclear to clear a tamper, if one is present, before proceeding with STM.

When STM is enabled on the HSM

1. The HSM generates a random string of 16 characters and presents that as the "Random User String" (suitable for copying and pasting into an e-mail).
2. The HSM gathers several sources of internal information reflecting the state of the HSM at that time, including a random nonce value generated for this purpose; the nonce value is not displayed, and never exists outside the HSM.
3. The HSM combines these items (the generated Random User String, the HSM state information, and the random nonce value), and produces the Verification String (suitable for copying and pasting into an e-mail).
4. The HSM then enters Secure Transport Mode, such that only limited operations are allowed until the HSM is brought out of STM.

5. The HSM can now be shipped from the factory to customers, or customers can place the HSM into storage or ship securely to another location. The HSM and the STM strings should not come together until they are in the possession of the intended recipient.

STM verification email

As part of the delivery process for your new HSM, Thales Client Services will send you an email containing two 16-digit strings: a **Random User String** and a **Verification String**. You require these strings to verify that your HSM has not been altered while in transit.

> **NOTE** If the STM verification process fails due to a lost or incorrect verification string, customers do have the option of proceeding with the recovery of the HSM from STM mode. If the STM verification process fails due to a tamper, customers can also choose to factory-reset the HSM to bring it back to a Factory state, and then re-initialize.
>
> Refer to the **CAUTION** notes below to avoid inadvertently causing a spurious STM recovery failure that would mask whether a real event had occurred.

For information about the various tamper events, see "Tamper Events" on page 229.

Recovering an HSM From Secure Transport Mode

Only the HSM SO can recover an initialized HSM that has been placed into STM. When the HSM is zeroized, HSM SO log in is not required.

New HSMs

New HSMs are shipped from the factory in Secure Transport Mode (STM). You must recover from STM before you can initialize the HSM. As part of the delivery of your new HSM, you should have received an email from Thales Client Services containing two 16-digit strings:

> Random User String: XXXX-XXXX-XXXX-XXXX

> Verification String: XXXX-XXXX-XXXX-XXXX

To recover an HSM from STM

1. Ensure that you have the two strings that were presented when the HSM was placed into STM, or that were emailed to you if this is a new HSM.
2. If the HSM is initialized, log in as the HSM SO (see "Logging In as HSM Security Officer" on page 207). If this is a new or zeroized HSM, skip to the next step.

> **CAUTION!** Be very careful entering the HSM SO authentication. A single failed attempt increments a counter that results in a change of the generated comparison string, which will cause STM verification to fail during Secure Transport Mode recovery.

3. Recover from STM, specifying the random user string that was displayed when the HSM was placed in STM, or that was emailed to you if this is a new HSM:

   lunacm:> **stm recover -randomuserstring** <XXXX-XXXX-XXXX-XXXX>

> **NOTE** The random user string is for verification purposes only. If you do not require STM validation, or you wish to bypass the STM validation, you can enter a different string to proceed with the recovery of the HSM from STM mode.

4. You are presented with a verification string. **Visually compare** the string with the original verification string that was sent via e-mail (or other means).

   If the string matches the original verification string, the HSM has not been used or otherwise altered since STM was enabled, and can be safely re-deployed.

   Enter **proceed** to recover from STM.

If the verification strings do not match

1. Reconfirm that you have entered the correct random user string for your HSM. Enter **quit** if you want to enter the string again.
2. If the verification strings still do not match:
   - If this is a new HSM, enter **quit** to leave the HSM in Secure Transport Mode, and contact Thales Technical Support.
   - Otherwise, if you feel that the verification failure was benign, enter **proceed** to release the HSM from Secure Transport Mode, and decide to either:
     - proceed with using the HSM
     - perform a factory reset and re-initialize the HSM as a safety precaution before proceeding further.

Placing an HSM In Secure Transport Mode

Only the HSM SO can place an initialized HSM into STM. When the HSM is zeroized, HSM SO log in is not required.

> **CAUTION!** Using Luna HSM Firmware 7.7.1-20 or older, before placing a multifactor quorum-authenticated HSM in Secure Transport Mode, ensure that CO, LCO and CU roles are deactivated, using role deactivate with each role name. For Luna Network HSM 7s, roles must be deactivated for all partitions, from LunaCM in a connected client. Failure to do so can result in mismatch when the generated strings are later compared during Secure Transport Mode recovery.
>
> Using Luna HSM Firmware 7.7.2 or newer, this is not necessary, because placing the HSM in STM logs out and deactivates those roles, and prevents auto-reactivation. The sessions can be logged in and reactivated manually.

To place an HSM into Secure Transport Mode

1. Log in as the HSM SO (see "Logging In as HSM Security Officer" on page 207).
2. Back up the contents of all application partitions.

   See Partition Backup and Restore for details.
3. Enter the following command to place the HSM into STM:

   lunacm:> **stm transport**

4. After confirming the action, you are presented with:
   - **Verification String:** <XXXX-XXXX-XXXX-XXXX>
   - **Random User String:** <XXXX-XXXX-XXXX-XXXX>

   Record both strings. They are required to verify that the HSM has not been altered while in STM.

> **CAUTION!** Transmit the verification string and random user string to the receiver of the HSM using a secure method, distinct from the transport of the physical HSM, so that it is not possible for an attacker to have access to both the HSM and the verification codes while the HSM is in STM.
>
> This product uses semiconductors that can be damaged by electro-static discharge (ESD). When handling the device, avoid contact with exposed components, and always use an anti-static wrist strap connected to an earth ground. In rare cases, ESD can trigger a tamper or decommission event on the HSM. If this happens, all existing roles and cryptographic objects are deleted.

CHAPTER 5: 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 PCIe HSM 7 7.x requires Luna PED Firmware 2.7.1 or newer. This firmware is backward-compatible with Luna PCIe HSM 7 6.x.

This chapter contains the following sections about multifactor quorum authentication:

> "Multifactor Quorum Authentication Architecture" on the next page
- "Comparing Password and Multifactor Quorum Authentication" on the next page

> "PED keys" on page 94
- "PED key Types and Roles" on page 94
- "Shared PED key Secrets" on page 96
- "Domain PED keys" on page 97
- "PINs" on page 97
- "Quorum Split Secrets (M of N)" on page 97

> "Luna PED Received Items" on page 100

> "Luna PED Hardware Functions" on page 102

> "Updating External Supply-Powered Luna PED Firmware" on page 126

> "Local PED Setup" on page 106

> "About Remote PED" on page 108

> "Multifactor Quorum PED key Management" on page 132

> "PEDserver and PEDclient" on page 147

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 102).

> **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 96.
- **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 97.
- **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 97.

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 132.

> **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" on page 206. 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 Luna HSM Domain	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 97. **Mandatory**
	Orange Luna HSM Remote PED	Remote PED Vector	Establishes a connection to a Remote PED server. See * below table. **Optional**
HSM Auditing	**White** Luna HSM Audit	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 Luna HSM Domain	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 97. **Mandatory**

Lifecycle	PED key	Authentication Secret	Function
Partition Operation	**Black** Luna HSM Crypto Officer	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 Luna HSM Crypto User	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 Luna HSM Crypto User	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.

> **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 132). Copies made later are true copies with the same PIN, intended as backups for one person (see "Duplicating Existing PED keys" on page 143). 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 140.

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). 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 108.

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 PCIe HSM 7s With Firmware 7.7.0 and Newer" on page 108
  - "Secure Communication Between the Remote PED and Luna PCIe HSM 7s With Firmware 7.7.0 and Newer" on page 112

> 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.

> 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 PCIe 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 92.

Basic Luna PED Order Items

The following items are included with your Luna PED. All are required for a successful installation.

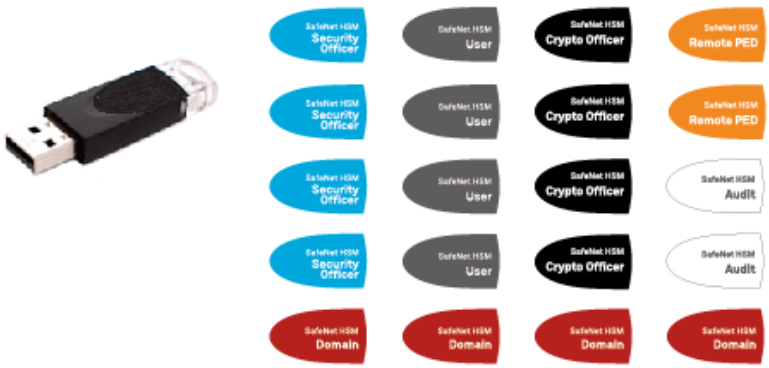
Qty	Item
1	**Luna PED** (with Luna PED Firmware 2.7.1 or newer)

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 104

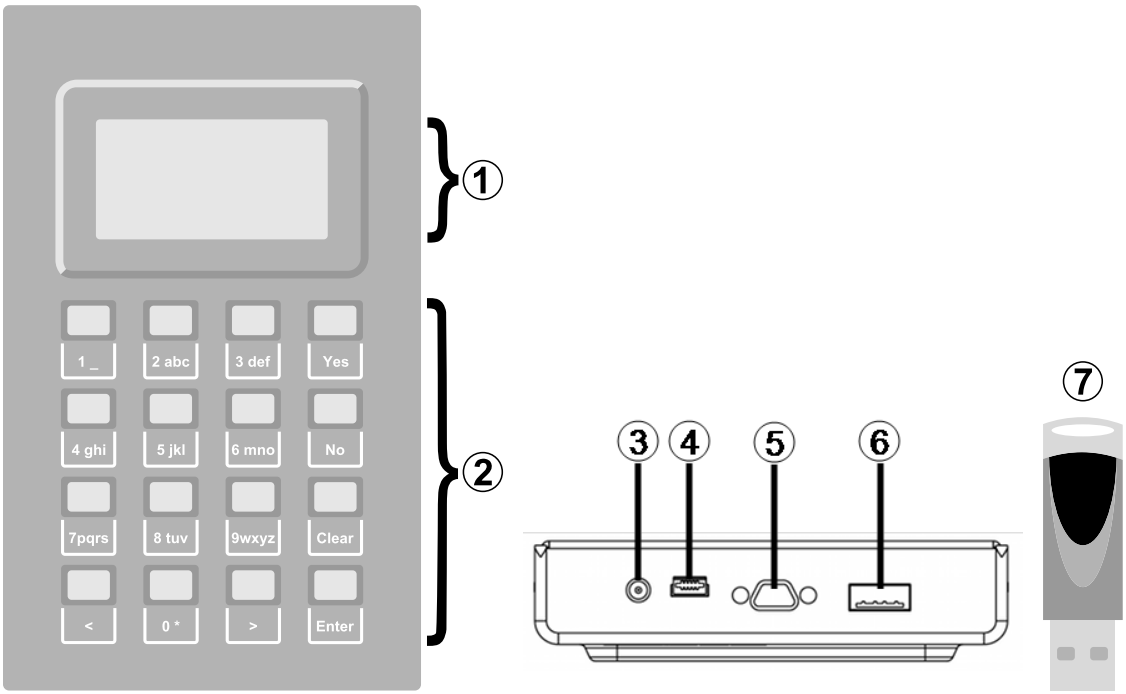
> "Modes of Operation" on page 104

> "Admin Mode Functions" on page 105

> "Luna PED with Newer CPU (External Power Supply Now Optional)" on page 105

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 106 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 PCIe HSM 7 to be located in a data center or other location restricting physical access. See "About Remote PED" on page 108 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.

```
Select Mode
1 Local PED-SCP
4 Admin
7 Remote PED
0 Local PED-USB

      PED V.2.7.1-5
```

   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 PCIe HSM 7 7 card, or 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.

```
Admin mode...
1 PED Key
5 Backup Devices
7 Software Update
9 Self Test

< EXIT
```

- **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 PCIe 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

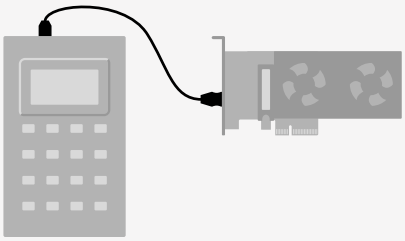
> USB mini-B to USB-A connector cable

> 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 host system.



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 104.

The Luna PED is now ready to perform authentication for the HSM. You may proceed with setting up or deploying your Luna PCIe 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 LunaCM command (see "Performing Multifactor Quorum Authentication" on page 138)

> Create copies of your PED keys (see "Duplicating Existing PED keys" on page 143)

> Change to the Admin Mode to run tests or update PED software (see "Changing Modes" on page 104)

> Prepare to set up a Remote PED server (see "About Remote PED" on the next page)

Local PED Troubleshooting

If you encounter problems with Local PED, refer to this section.

CKR_PED_UNPLUGGED error after hsm restart

After running **hsm restart**, LunaCM returns a CKR_PED_UNPLUGGED error when authentication is attempted.

```
lunacm:>role login -n so

        Please attend to the PED.

Caution: You have only 3 so login attempts left. If you fail 3
         more consecutive login attempts (i.e. with no successful
         logins in between) the HSM will be ZEROIZED!!!

Error in execution: CKR_PED_UNPLUGGED.

Command Result : 0x8000002e (CKR_PED_UNPLUGGED)
```

If you receive this error, disconnect the Luna PED from the HSM's USB port and reconnect it before issuing the login command again.

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 PCIe 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 PCIe HSM 7. All exchanged data is protected in the following way:

> All CSPs exchanged between the Local PED and the Luna PCIe 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 PCIe 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 110

> "Secure Communication Between the Remote PED and Luna PCIe HSM 7s With Firmware 7.7.0 and Newer" on page 112

> "Initializing the Remote PED Vector and Creating an Orange Remote PED key" on page 114

> "Installing PEDserver and Setting Up the Remote Luna PED" on page 117

> "Opening a Remote PED Connection" on page 118

> "Ending or Switching the Remote PED Connection" on page 121

> "Configuring PED Timeout Settings" on page 121

> "Remote PED Troubleshooting" on page 122

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

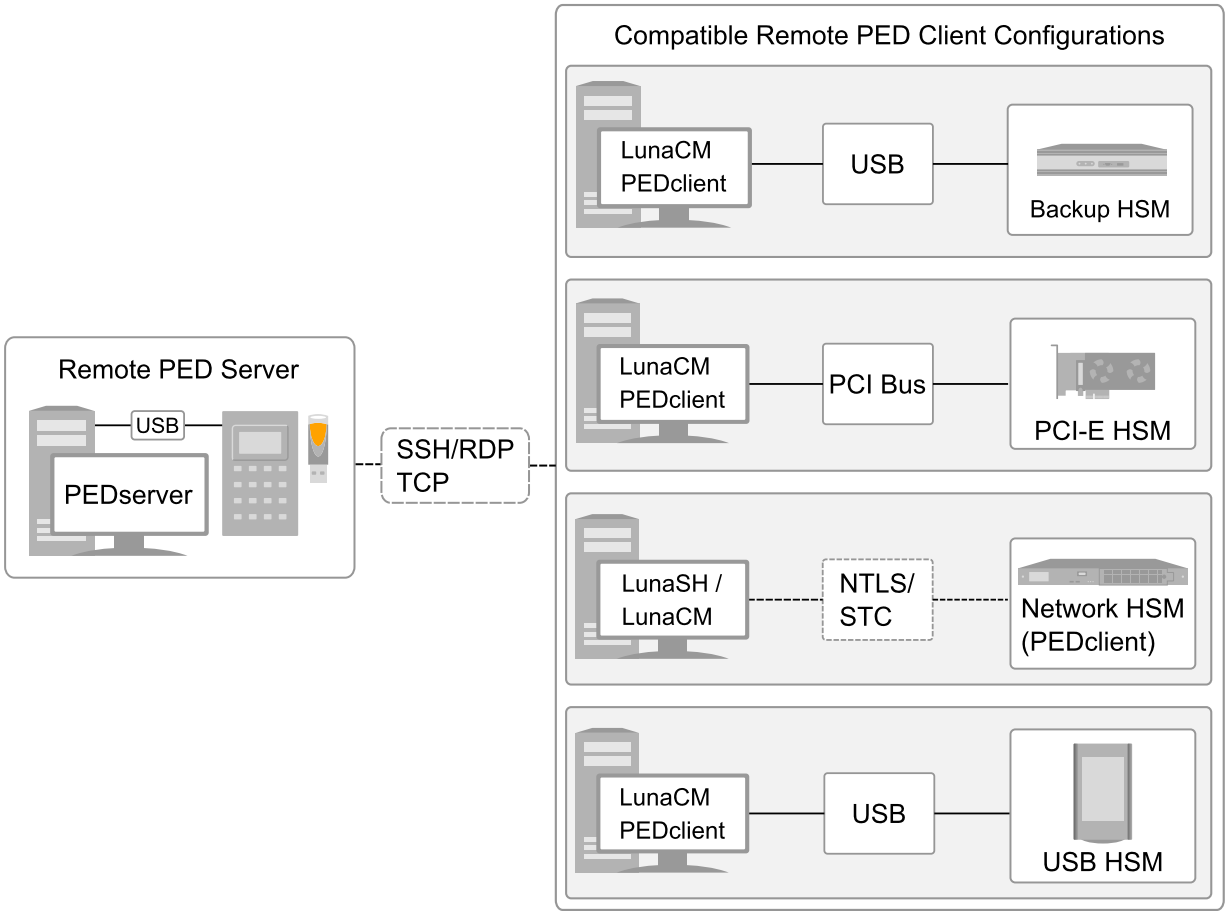
> **Remote PED Vector (RPV):** a randomly generated, encrypted value used to authenticate between a Remote PED (via PEDserver) and a Luna HSM (via PEDclient).

> **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



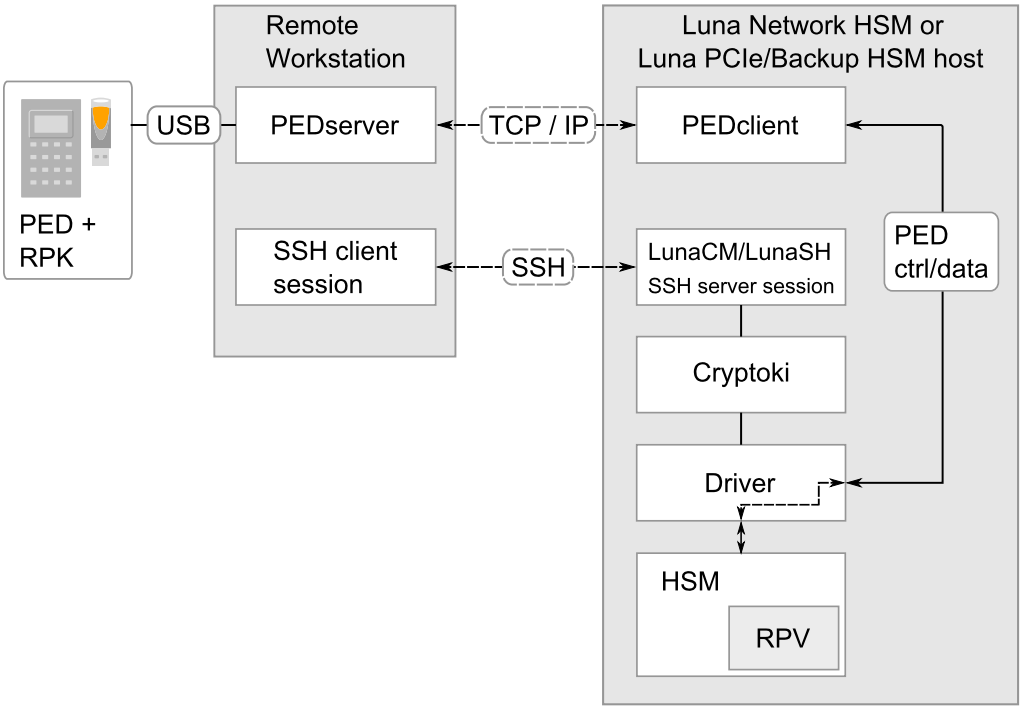
Remote PED Connections

A Luna PCIe HSM 7 on a host computer running PEDclient 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



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 121.

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 121.

Timeout

Remote PED connections have configurable timeout settings. For more information, refer to "Configuring PED Timeout Settings" on page 121.

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 **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 PCIe HSM 7s With Firmware 7.7.0 and Newer

Communication between the Remote PED and Luna PCIe 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 PCIe HSM 7 is protected using the same method that is used to secure communication between a Local PED and Luna PCIe HSM 7 (see "Secure Communication Between the Local PED and Luna PCIe HSM 7s With Firmware 7.7.0 and Newer" on page 108). 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 PCIe 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 PCIe 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 PCIe 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 PCIe 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.

Initializing the Remote PED Vector and Creating an Orange Remote PED key

The Remote PED (via PEDserver) authenticates itself to the Luna PCIe 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 PCIe 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. 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 116.

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 132 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 106).
2. Launch LunaCM on the Luna PCIe HSM 7 host workstation.

3. If the HSM is initialized, log in as HSM SO (see "Logging In as HSM Security Officer" on page 207). If not, skip 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 132 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 117.

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 HSM Firmware 7.7.0 and Luna HSM Client 10.3.0.

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 132 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.

lunacm:> **ped connect -ip** <PEDserver_IP>

The Remote PED connection command prepares to secure the connection and LunaCM 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:

```
SLOT
COMPUTE SESSION KEY.

Enter PED Password.

********
```

2. Enter the numeric password on the PIN pad, exactly as displayed in LunaCM, and press **Enter**.
3. Set the active slot to the Luna PCIe HSM 7 admin partition.
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 132 for a full description of the key-creation process.

   When the initialization is complete, the HSM establishes a Remote PED connection using the newly-created RPV.

   You may now initialize the HSM. See "Initializing the HSM" on page 202 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 132 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 106 or "Opening a Remote PED Connection" on the next page).
2. Launch LunaCM on the Luna PCIe HSM 7 host workstation.
3. [Remote PED only] Open a Remote PED connection using the original orange PED key(s).

   lunacm:> **ped connect -ip** <PEDserver_IP>
4. Log in as HSM SO (see "Logging In as HSM Security Officer" on page 207).

   lunacm:> **role login-name so**
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.

   lunacm:> **ped vector**
6. Attend to the Luna PED and respond to the on-screen prompts. See "Creating PED keys" on page 132 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 30, 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 104.
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" below.

Opening a Remote PED Connection

The HSM/client administrator can use this procedure to establish an HSM-initiated Remote PED 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 the previous page)

> Administrative access to the Luna PCIe HSM 7 host via SSH

> Orange PED key with the HSM's RPV (see "Initializing the Remote PED Vector and Creating an Orange Remote PED key" on page 114)

If you encounter issues, see "Remote PED Troubleshooting" on page 122.

To open a Remote PED connection

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 161** [**-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 159**

   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.

6. Via SSH, launch LunaCM on the Luna PCIe HSM 7 host.

7. Initiate the Remote PED connection.

   lunacm:> **ped connect -ip** <PEDserver_IP> **-port** <PEDserver_port> **-slot** <slot>

> **NOTE** The **-slot** option may be required if you have multiple Luna PCIe HSM 7s installed in one server. If you do not include this option, the currently-active slot is used.

```
lunacm:>ped connect -ip 192.124.106.100 -port 1503

Command Result : No Error
```

8. Issue the first command that requires authentication.
   - If the HSM is already initialized and you have the blue HSM SO key, log in.

     lunacm:> **role login -name so**
   - If the HSM is uninitialized, you can initialize it now (see "Initializing the HSM" on page 202). Have blank or reusable blue and red PED keys ready (or multiple blue and red keys in case of M of N or if making multiple copies). See "Creating PED keys" on page 132 for more information.

     lunacm:> **hsm init -label** <label>
9. 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.
```

10. TheLuna 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 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 the next page.

11. [OPTIONAL] Set a default IP address and/or port for the Luna PCIe HSM 7 to look for a Remote PED host with PEDserver running.

    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** to initiate the Remote PED connection. The orange PED key may be required if the RPK has been invalidated since you last used it.

Ending or Switching the Remote PED Connection

PEDserver runs on the Remote PED host until explicitly stopped. PEDclient (running on the Luna PCIe HSM 7 host) has a default timeout period of 1800 seconds. 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 connection

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>.

Configuring PED Timeout Settings

You can configure the PED timeout settings for your Remote PED connection. This is useful in the following situations:

> You would like to improve workflows for your HSM roles or enhance the security of your multifactor quorum-authenticated Luna PCIe HSM 7 deployment by increasing or decreasing the duration of PED inactivity that can elapse until the PED connection breaks.

> You are using a quorum (M of N split-secret) authentication scheme for your HSM roles and need to increase the time that is available for each required user to present their PED key. For more information about this authentication scheme, refer to "Quorum Split Secrets (M of N)" on page 97.

> You are updating to Luna HSM Firmware 7.7.0 or newer and need to increase the time that is available migrate all your pre-existing orange PED keys. For more information about this migration procedure, refer to Migrating Existing Orange Remote PED Keys.

Configuring PED Inactivity Timeout

You can increase or decrease the number of seconds of PED inactivity that can elapse before the PED connection breaks. PEDserver and PEDclient both have configurable timeout settings, but the utility that uses the briefer value determines the actual timeout duration.

To configure PED inactivity timeout, run **"pedserver -mode config" on page 155 -set -idletimeout** <seconds>or **"pedclient -mode config" on page 167 -set -idletimeout** <seconds>.

After configuration, you can verify that the PED inactivity timeout duration has changed by running **"pedserver -mode show" on page 159** or **"pedclient -mode show" on page 172** and verifying the value of the **Idle Timeout Value:** entry.

Configuring PED key Interaction Timeout

You can set the amount of time that can elapse without completing PED key requests, before the PED key request ends and must be repeated.

Estimate your actual settings based on the number of keys you are migrating.

To configure PED key interaction timeout, set PEDTimeout2 under the **Luna** section of the Luna HSM Client configuration file to the desired value. For more information about editing this entry in the configuration file, refer to "Configuration File Summary" on page 63.

Configuring Luna PED Operation Timeout

You can set the amount of time that can elapse without completing a Luna PED operation request, before the PED operation ends and must be repeated.

To configure PED key interaction timeout, set CommandTimeOutPedSet under the **Luna** section of the Luna HSM Client configuration file to the desired value. For more information about editing this entry in the configuration file, refer to "Configuration File Summary" on page 63.

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:

> "Intermittent CKR_CALLBACK_ERROR: PED Cannot Service its USB Data Channel Fast Enough to Communicate with PEDserver" below

> "No Menu Appears on Luna PED Display: Ensure Driver is Properly Installed" on the next page

> "RC_SOCKET_ERROR: PEDserver Requires Administrator Privileges" on the next page

> "CKR_PED_UNPLUGGED: Reconnect Remote PED Before Issuing Commands" on the next page

> "Remote PED Firewall Blocking" on page 124

> "Remote PED Blocked Port Access" on page 125

> "ped connect Fails if IP is Not Accessible" on page 126

> "PEDserver on VPN fails" on page 126

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 PCIe 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 155.

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 117.

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.

CKR_PED_UNPLUGGED: Reconnect Remote PED Before Issuing Commands

As described in the connection procedures, 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:

```
lunacm:> role login -n so

        Please attend to the PED.

Error in execution: CKR_PED_UNPLUGGED.

Command Result : 0x8000002e (CKR_PED_UNPLUGGED)
```

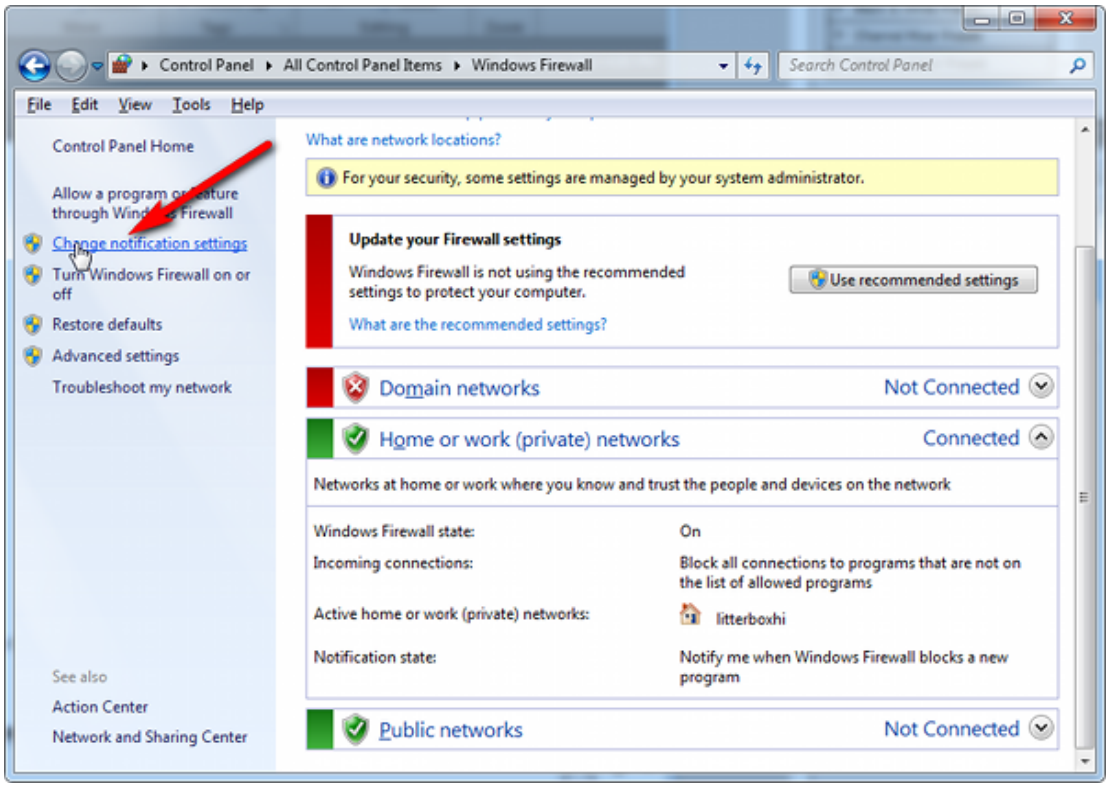
To avoid this error, re-initiate the connection before issuing any commands requiring authentication via PED:

lunacm:> **ped connect -ip** <PEDserver_IP> **-port** <PEDserver_port>

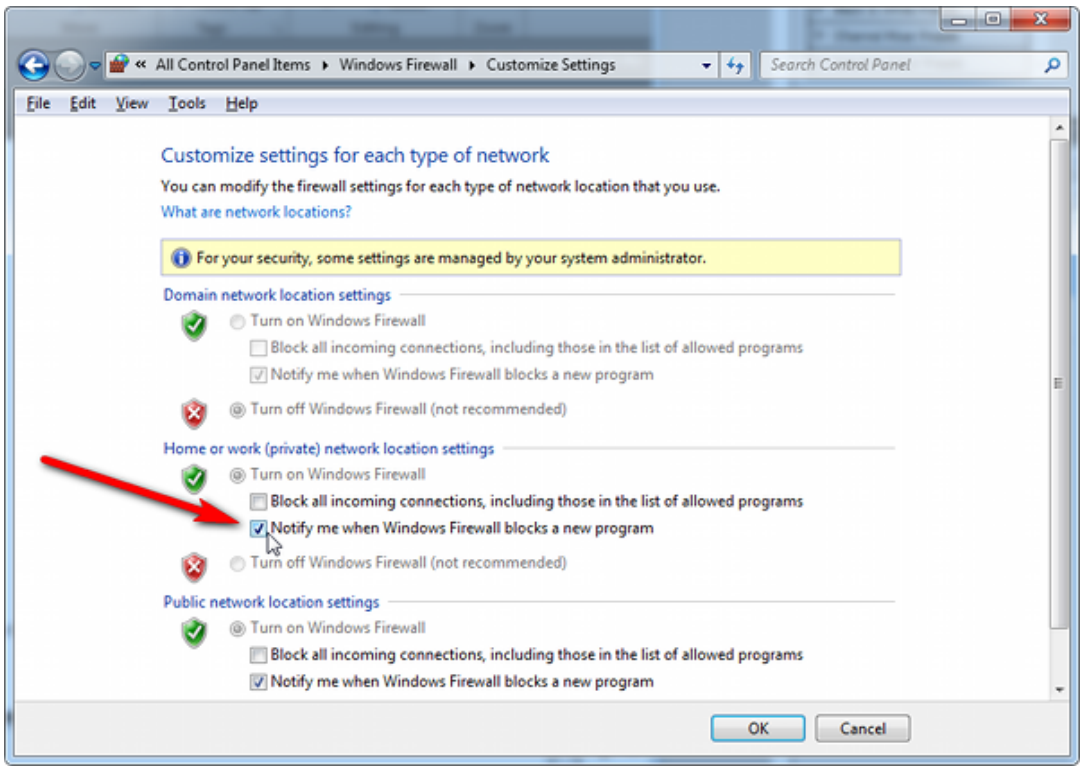
Remote PED Firewall Blocking

If you experience problems while attempting to configure a Luna Remote PED session over VPN, you might need to adjust Windows Firewall settings.

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** 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 PCIe 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. Launch LunaCM on the Luna PCIe HSM 7 host, and open the HSM-initiated connection.

   lunacm:> **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 PCIe 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 161** **-ip** <new_IP> **-port** <port>

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 130.

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 page 130), 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 126 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 126), 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 126 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" below
- "Stage 1: Reusing Existing PED keys" on page 134
- "Stage 2: Defining M of N" on page 135
- "Stage 3: Setting a PIN" on page 136
- "Stage 4: Duplicating New PED keys" on page 137

> "Performing Multifactor Quorum Authentication" on page 138

> "Consequences of Losing PED keys" on page 140

> "Identifying the PED key Secret" on page 142

> "Duplicating Existing PED keys" on page 143

> "Changing the PED key Secret" on page 144

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:

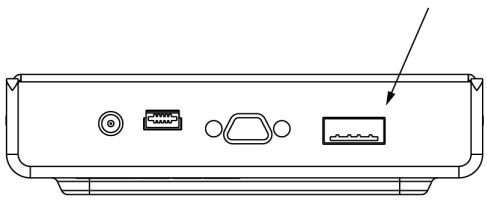
> If you are reusing an existing PED key or keyset, the owners of those keys must be present with their keys and PINs ready.

> 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 LunaCM commands to initialize the applicable role, domain, or vector.
   - **Blue HSM SO and Red HSM Domain PED key**:

     lunacm:> **hsm init**
   - **Orange Remote PED Vector PED key**:

     lunacm:> **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**:

     lunacm:> **role init -name au**

   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 96 and "Domain PED keys" on page 97 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.

```
SLOT
SETTING SO PIN...
Would you like to
reuse an existing
keyset?(Y/N)
```

- 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**.

```
SLOT
SETTING SO PIN...
Insert a
SO / HSM Admin
PED Key (BLUE)
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 97 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 97 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 140.

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.

```
SLOT
SETTING SO PIN...
Enter new PED PIN:
*********
Confirm new PED PIN:
*********
```

   - 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.

```
SLOT
SETTING SO PIN...
Enter new PED PIN:

Confirm new PED PIN:
```

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 143.

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.

```
SLOT
SETTING SO PIN...
Are you duplicating
this keyset?(Y/N)
```

```
SLOT
SETTING SO PIN...
Would you like to
make another
duplicate set?(Y/N)
```

   - If you select **No**, the key creation process is complete.

- If you select **Yes**, complete "Stage 3: Setting a PIN" on page 136 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 page 136 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 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:

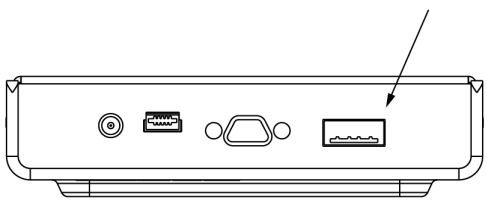
```
lunacm:>role login -name po

        Please attend to the PED.
```

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:



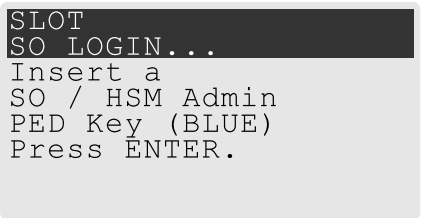
> **CAUTION!** Multiple failed authentication attempts result in zeroization of the HSM or partition, or role lockout, depending on the role. This is a security measure designed to thwart repeated, unauthorized attempts to access cryptographic material. For details, see "Logging In as HSM Security Officer" on page 207 or Logging In to the Application Partition.

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.

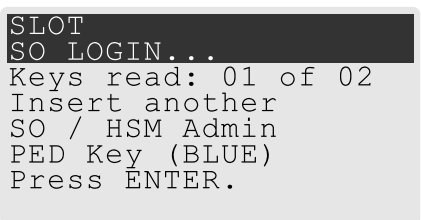
```
Command Result : No Error
```

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 PCIe 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 page 142

> "Gray Crypto User PED key" on page 142

> "White Audit User PED key" on page 142

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 Admin partition 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 Admin partition 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 Admin partition(s). If the HSM is factory-reset, the contents of the HSM Admin partition are unrecoverable. Follow the same procedure as you would if you lost the blue HSM SO key, but you cannot restore the HSM Admin partition 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 116.

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 104)

> 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 104)

> 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 97.

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 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" on the next page

> "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 page 146

> "Black Crypto Officer PED key" on page 146

> "Gray Crypto User PED key" on page 146

> "White Audit User PED key" on page 147

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 LunaCM, set the active slot to the Admin partition and login as HSM SO.

   lunacm:> **role login -name so**

2. Initiate the PED key change.

   lunacm:> **role changepw -name so**

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 132.

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 Admin partition 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 LunaCM, set the active slot to the Admin partition and login as HSM SO.

   lunacm:> **role login -name so**

2. Initialize the RPV.

   lunacm:> **ped vector init**

   You are prompted to create a new Remote PED key. See "Creating PED keys" on page 132.

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 132.

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.

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 132.

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 132.

> **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. In LunaCM, set the active slot to the Admin partition and login as Auditor.

   lunacm:> **role login -name au**

2. Initiate the PED key change.

   lunacm:> **role changepw -name au**

3. 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 132.

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.

See "pedserver" on page 149.

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 for more information.

> It acts as the logging daemon for HSM audit logs.

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 164.

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 154.
-regen	Regenerates the client certificate. Applies to server-initiated (peer-to-peer) mode only. See "pedserver -regen" on page 164.

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 152.
register	Registers an appliance. See "pedserver -appliance register" on page 153

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 157.
disconnect	Disconnects from the appliance. See "pedserver -mode disconnect" on page 158.
show	Queries if PEDserver is currently running, and gets details about PEDserver. See "pedserver -mode show" on page 159.
start	Starts PEDserver. See "pedserver -mode start" on page 161.
stop	Shuts down PEDserver. See "pedserver -mode stop" on page 163

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 .

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 167.
deleteid	Deletes a PED ID mapping. See "pedclient -mode deleteid" on page 169.
releaseid	Releases a PED ID mapping from an HSM. See "pedclient -mode releaseid" on page 170.
setid	Creates a PED ID mapping. See "pedclient -mode setid" on page 171.
show	Queries if PEDclient is currently running and gets details about PEDclient. See "pedclient -mode show" on page 172.
start	Starts up PEDclient. See "pedclient -mode start" on page 173.
stop	Shuts down PEDclient. See "pedclient -mode stop" on page 175.
testid	Tests a PED ID mapping. See "pedclient -mode testid" on page 176.

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 6: Audit Logging

Each event that occurs on the HSM can be recorded in the HSM event log, allowing you to audit your HSM usage. The HSM event log is viewable and configurable only by the **audit** user role. This **audit** role is disabled by default and must be explicitly enabled.

This chapter describes how to use audit logging to provide security audits of HSM activity. It contains the following sections:

> "Audit Logging General Advice and Recommendations" on page 187

> "Logging In as Auditor" on page 189

> "Configuring and Using Audit Logging" on page 190

> "Audit Log Categories and HSM Events" on page 194

> "Audit Log Troubleshooting" on page 201

Audit Logging Features

The following list summarizes the functionality of the audit logging feature:

> Log entries originate from the Luna PCIe HSM 7 (cryptographic module - the feature is implemented via HSM firmware (rather than in the library), for maximum security.

> Log origin is assured.

> Logs and individual records can be validated by any Luna PCIe HSM 7 that is a member of the same domain.

> Audit Logging can be performed on password-authenticated and multifactor quorum-authenticated (both FIPS 140-3 level 3) configurations, but these configurations may not validate each other's logs - see the "same domain" requirement, above.

> **NOTE** The "same domain" requirement still applies, but with the introduction of Extended Domain Management [see Universal Cloning and Domain Planning] (Partition Policy 44) in firmware version 7.8.0, you can change/add domains, such that the verifying HSM could be given the same domain as the original logging HSM.
>
> Thus, from Luna HSM firmware version 7.8.0 onward, it is possible
>
> - for a Multifactor Quorum HSM log to be validated by a Password-authenticated HSM or
> - for a Password-authenticated HSM log to be validated by a Multifactor Quorum HSM.

> Each entry includes the following:

- When the event occurred
- Who initiated the event (the authenticated entity)
- What the event was
- The result of the logging event (success, error, etc.)

> Multiple categories of audit logging are supported, configured by the audit role.

> Audit management is a separate role - the role creation does not require the presence or co-operation of the Luna PCIe HSM 7 SO.

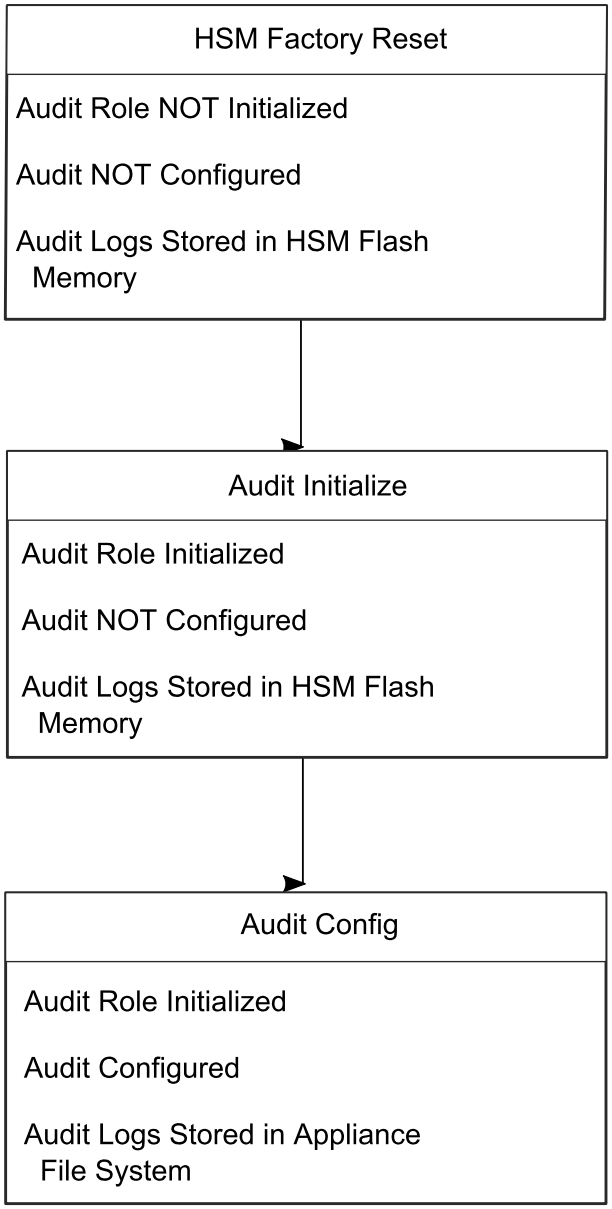
> The category of audit logging is configurable by (and only by) the audit role.

> Audit log integrity is ensured against the following:

- Truncation - erasing part of a log record
- Modification - modifying a log record
- Deletion - erasing of the entire log record
- Addition - writing of a fake log record

> The following critical events are logged unconditionally, regardless of the state of the audit role (initialized or not):

- Tamper
- Decommission
- Zeroization
- SO creation
- Audit role creation



Note:

Logs are exported from the HSM's memory to the appliance's hard drive. Only an authenticated Auditor role is allowed to configure or initiate the export function. Therefore, an HSM in the Factory Reset state is ***not*** allowed to export log files from HSM memory to the appliance file system.

Note:

"audit log clear" clears logs only from the appliance file system. It does ***not*** affect logs stored in the HSM memory.
Logs move out of HSM memory to the host file system, only when audit log rotation has been configured by the Auditor - so initialize and configure early to avoid log-entry build-up on the HSM.

Types of events included in the logs

The events that are included in the log is configurable by the audit role. The types of events that can be logged include the following:

- log access attempts (logins)
- log HSM management (init/reset/etc)
- key management events (key create/delete)
- asymmetric key usage (sig/ver)
- first asymmetric key usage only (sig/ver)

- symmetric key usage (enc/dec)
- first symmetric key usage only (enc/dec)
- log messages from CA_LogExternal
- log events relating to log configuration

Each of these events can be logged if they fail, succeed, or both.

Event log storage

When the HSM logs an event, the log is stored on the HSM. The audit user cannot view these log entries. Before a log can be viewed, it must be rotated. Log rotation saves the log entries on the HSM to the local file system, where they can be viewed. Log records are HMACed using an audit log secret to ensure their authenticity. The audit log secret is unique to the HSM where the log was created, and is required to view the HSM event logs. The secret can be exported, allowing you to view and verify the logs on another HSM.

TIP Log entries are stored in the cryptographic module (HSM) until they are rotated off. Log entries are not rotated out of the cryptographic module *until* the audit user is initialized and audit logging is configured. By default, even if there is no audit user or configuration, the cryptographic module logs unconditional events within its own memory, like:

> zeroize
> decommission
> hardware tamper
> card removal
> etc.

If the crypto module internal space ever fills completely with log records,

> whether slowly from unconditional logs, or
> quickly from more voluble high-volume event recording,

...the HSM / cryptographic module would *stop all operations* that were notrole init **-name au** and audit config. The HSM would resume providing service only after the audit user cleared the logs.

To avoid that ever happening, configure audit logging to organize log parameters and handling, being sure to set sufficient frequency of rotation for the volume of record generation that you enable.

Best practice is to:

> initialize the audit role as soon as the HSM is first powered on for production role init **-name au**
> configure the log storage path on the external file system, along with the types of events to log, the rotation interval, etc. audit config
> then, initialize the HSM Security Officer (this helps ensure that all messages, demanded by your auditing authority, are captured) hsm init
> then, proceed with partition initialization and usage with your application(s)
> then, *revisit* audit log configuration at regular intervals to tune the balance between
  - desired message types,
  - volume of audited actions normally performed (*),
  - and so on
> when you change behavior of the crypto module or change the types of events to audit, be sure to *revisit also* the rotation interval.

[* Example, you might always want to record the generation of keys, but if usage of those keys is very high-volume (like in some signature use-cases), and thus would generate a high volume of log entries, it might be permissible, and prudent, to log only first-use of any key. Check with the relevant authority.]

Event logging impacts HSM performance

Each audit log record generated requires HSM resources. Configuring event logging to record most, or all, events may have an impact on HSM performance. You may need to adjust your logging configuration to provide adequate logging without significantly affecting performance. By default, only critical events are logged, imposing virtually no load on the HSM.

Audit limitations and Controlled tamper recovery state

The following conditions apply when HSM Policy "48: Do controlled tamper recovery" is enabled (default setting).

- Auditor (the Audit role) cannot verify the integrity of audit logs until after recovery from tamper.
- Auditor cannot be initialized when the HSM is in controlled tamper recovery state.
- Existing Audit role can login when in controlled tamper recovery state.
- Existing Audit role cannot make audit config changes when in controlled tamper recovery state.
- Existing Audit role cannot export the audit secret when in controlled tamper recovery state.

The Audit Role

A Luna PCIe HSM 7 Audit role allows complete separation of Audit responsibilities from the HSM Security Officer (HSM SO), the Crypto Officer(or User), and other HSM roles. If the Audit role is initialized, the HSM and Partition SOs are prevented from working with the log files, and auditors are unable to perform administrative tasks on the HSM. As a general rule, the Audit role should be created before the HSM Security Officer role, to ensure that all important HSM operations (including those that occur during initialization), are captured.

Use the LunaCM command **role init -name Auditor** to initialize the audit role, as described in role init.

Password-authenticated HSMs

For Luna PCIe HSM 7s with Password Authentication, the auditor role logs into the HSM to perform their activities using a password. After initializing the Audit role on a password-authenticated HSM, log in as the Auditor and set the domain (see role setdomain). This step is required before setting logging parameters or the log filepath, or importing/exporting audit logs.

Multifactor Quorum-authenticated HSMs

For Luna PCIe HSM 7s with multifactor quorum authentication, the auditor role logs into the HSM to perform their activities using the Audit (white) PED key.

Role Initialization

Creating the Audit role (and imprinting the white PED key for multifactor quorum-authenticated HSMs) does not require the presence or cooperation of the HSM SO.

Audit Role Available Commands

In LunaCM, all commands are visible to the person who launches the utility, and some can be used without specific authentication to the HSM, such as view/show/list commands, which might be classified as "monitoring" functions. Attempts to run operational or administrative commands that need role-specific authentication, without that authentication, result in an error message. The Audit role has a limited set of operations available to it, on the HSM, which constitutes any of the generally accessible monitoring commands, plus everything under the "audit" heading.

```
lunacm:>audit


  The following sub commands are available:

  Command         Short    Description
  ------------------------------------
```

```
  verify          v        Verify a block of log messages
  config          c        Configure audit parameters
  export          e        Read the wrapped log secret from the HSM
  import          m        Import the wrapped log secret to the HSM
  time            t        Sync HSM time to host, or get HSM time
  status          s        Show status of logging subsystem
  logmsg          logm     Write a message to the HSM's log

  Syntax: audit <sub command>


Command Result : No Error
```

Anyone accessing the computer and running LunaCM can see the above commands, but cannot run them if they do not have the "audit" role authentication (password or PED key, as appropriate).

What is important is not the role you can access on the computer (a named user, admin, operator), but the role you can access within the HSM.

Audit Log Secret

The HSM creates a log secret unique to the HSM, computed during the first initialization after manufacture. The log secret resides in flash memory (permanent, non-volatile memory), and is used to create log records that are sent to a log file. Later, the log secret is used to prove that a log record originated from a legitimate HSM and has not been tampered with.

Log Secret and Log Verification

The 256-bit log secret which is used to compute the HMACs is stored in the parameter area on the HSM. It is set the first time an event is logged. It can be exported from one HSM to another so that a particular sequence of log messages can be verified by the other HSM. Conversely, it can be imported from other HSMs for verification purpose.

To accomplish cross-HSM verification, the HSM generates a key-cloning vector (KCV, a.k.a. the Domain key) for the audit role when it is initialized. The KCV can then be used to encrypt the log secret for export to the HOST.

To verify a log that was generated on another HSM, assuming it is in the same domain, we simply import the wrapped secret, which the HSM subsequently decrypts; any records that are submitted to the host for verification will use this secret thereafter.

When the HSM exports the secret, it calculates a 32-bit checksum which is appended to the secret before it is encrypted with the KCV.

When the HSM imports the wrapped secret, it is decrypted, and the 32-bit checksum is calculated over the decrypted secret. If this doesn’t match the decrypted checksum, then the secret that the HSM is trying to import comes from a system on a different domain, and an error is returned.

To verify a log generated on another HSM, in the same domain, the host passes to the target HSM the wrapped secret, which the target HSM subsequently decrypts; any records submitted to the target HSM for verification use this secret thereafter.

Importing a log secret from another HSM does not overwrite the target log secret because the operation writes the foreign log secret only to a separate parameter area for the wrapped log secret.

> **CAUTION!** Once an HSM has imported a wrapped log secret from another HSM, it must export and then re-import its own log secret in order to verify its own logs again.

Audit Log Records

A log record consists of two fields – the log message and the HMAC for the previous record. When the HSM creates a log record, it uses the log secret to compute the SHA256-HMAC of all data contained in that log message, plus the HMAC of the previous log entry. The HMAC is stored in HSM flash memory. The log message is then transmitted, along with the HMAC of the previous record, to the host. The host has a logging daemon to receive and store the log data on the host hard drive.

For the first log message ever returned from the HSM to the host there is no previous record and, therefore, no HMAC in flash. In this case, the previous HMAC is set to zero and the first HMAC is computed over the first log message concatenated with 32 zero-bytes. The first record in the log file then consists of the first log message plus 32 zero-bytes. The second record consists of the second message plus HMAC1 = HMAC (message1 || 0x0000). This results in the organization shown below.

MSG 1	HMAC 0
	. . .
MSG n-1	HMAC n-2
MSG n	HMAC n-1
. . .	
MSG n+m	HMAC n+m-1
MSG n+m+1	HMAC n+m
. . .	
MSG end	HMAC n+m-1

Recent HMAC in NVRAM	HMAC end

To verify a sequence of *m* log records which is a subset of the complete log, starting at index *n*, the host must submit the data illustrated above. The HSM calculates the HMAC for each record the same way as it did when the record was originally generated, and compares this HMAC to the value it received. If all of the calculated HMACs match the received HMACs, then the entire sequence verifies. If an HMAC doesn't match, then the associated record and all following records can be considered suspect. Because the HMAC of each message depends on the HMAC of the previous one, inserting or altering messages would cause the calculated HMAC to be invalid.

The HSM always stores the HMAC of the most-recently generated log message in flash memory. When checking truncation, the host would send the newest record in its log to the HSM; and, the HSM would compute the HMAC and compare it to the one in flash. If it does not match, then truncation has occurred.

Audit Log Message Format

Each message is a fixed-length, comma delimited, and newline-terminated string. The table below shows the width and meaning of the fields in a message.

Offset	Length (Chars)	Description
0	10	Sequence number
10	1	Comma
11	17	Timestamp
28	1	Comma
29	256	Message text, interpreted from raw data
285	1	Comma
286	64	HMAC of previous record as ASCII-HEX
350	1	Comma
351	96	Data for this record as ASCII-HEX (raw data)
447	1	Newline '\n'

The raw data for the message is stored in ASCII-HEX form, along with a human-readable version. Although this format makes the messages larger, it simplifies the verification process, as the HSM expects to receive raw data records.

Example

The following example shows a sample log record. It is separated into multiple lines for readability even though it is a single record. Some white spaces are also omitted.

```
38,12/08/13 15:30:50,session 1 Access 2147483651:22621 operation LUNA_CREATE_CONTAINER
returned LUNA_RET_SM_UNKNOWN_TOSM_STATE(0x00300014) (using PIN (entry=LUNA_ENTRY_DATA_AREA)),
29C51014B6F131EC67CF48734101BBE301335C25F43EDF8828745C40755ABE25,
2600001003600B00EA552950140030005D580000030000800100000000000000000000000000000000000000
```

The sequence number is “`38`”. The time is “`12/08/13 15:30:50`”.

The log message is “`session 1 Access 2147483651:22621 operation LUNA_CREATE_CONTAINER returned LUNA_RET_SM_UNKNOWN_TOSM_STATE(0x00300014) (using PIN (entry=LUNA_ENTRY_DATA_AREA))`”.

In the message text, the “who” is the session identified by “`session 1 Access 2147483651:22621`” (the application is identified by the access ID major = 2147483651, minor = 22621).

The “what” is “`LUNA_CREATE_CONTAINER`”.

The operation status is "`LUNA_RET_SM_UNKNOWN_TOSM_STATE(0x00300014)`".

The HMAC of previous record is "`29C51014B6F131EC67CF48734101BBE301335C25F43EDF8828745C40755ABE25`".

The remainder is the raw data for this record as ASCII-HEX.

> The "who" is LunaSH session "session 1 Access 2147483651:22621" (identified by the lunash access ID major = 2147483651, minor = 22621).

> The "what" is "LUNA_CREATE_CONTAINER".

> The operation status is "LUNA_RET_SM_UNKNOWN_TOSM_STATE(0x00300014)".

Timestamping

The HSM has an internal real-time clock (RTC). The RTC does not have a relevant time value until it is synchronized with the HOST system time. Because the HSM and the host time could drift apart over time, periodic re-synchronization is necessary. Only an authenticated Auditor is allowed to synchronize the time.

Time Reported in Log

When you perform **audit time get**, you might see a variance of a few seconds between the reported HSM time and the Host time. Any difference up to five seconds should be considered normal, as the HSM reads new values from its internal clock on a five-second interval. So, typically, Host time would show as slightly ahead.

Log Capacity

The Luna PCIe HSM 7 has approximately 16 MB available on the card for audit logging. The normal function of Audit logging is to export log entries constantly to the host file system. Short-term, within-the-HSM log storage capacity becomes important only in the rare situations where the HSM remains functioning but the file system is unreachable from the HSM.

LOG FULL condition

If you receive CKR_LOG_FULL, the log capacity has been reached, and all HSM operations will stop. This is to prevent the HSM from performing unlogged operations. In this condition, most HSM commands will not work; only commands that allow the Auditor to log in, clear the log storage, set the logging configuration, or reset the HSM to factory conditions are permitted.

Ensure that you have set **audit config** correctly to rotate logs to the file system periodically in order to avoid this situation. In particular:

> filepath points to an existing location (no typos or other errors in specifying the filepath for log files)

> writing to that location is permitted (check the folder/directory permissions)

> the indicated location has sufficient space available to write log files (make some room if necessary).

See the later steps in "Configuring Audit Logging" on page 190 for the procedure.

Configuration Persists Unless Factory Reset is Performed

Audit logging configuration is not removed or reset upon HSM re-initialization or a tamper event. Factory reset or HSM decommission will remove the Audit user and configuration. Logs must be cleared by specific command. Therefore, if your security regime requires decommission at end-of-life, or prior to shipping an HSM, then explicit clearing of HSM logs should be part of that procedure.

This is by design, as part of separation of roles in the HSM. When the Audit role exists, the HSM SO cannot modify the logging configuration, and therefore cannot hide any activity from auditors.

Audit Logging Stops Working if the Current Log File is Deleted

As a general rule, you should not delete a file while it is open and in use by an application. In Linux, deletion of a file is deletion of an inode, but the actual file itself, while now invisible, remains on the file system until the space is cleaned up or overwritten. If a file is in use by an application - such as audit logging, in this case - the application can continue using and updating that file, unaware that it is now in deleted status.

If you delete the current audit log file, the audit logging feature does not detect that and does not create a new file, so you might lose log entries.

The workaround is to restart the **pedclient** daemon, which creates a new log file.

Example

1. You have configured audit logging, and the entire audit path is deleted. In Linux, the file isn't actually deleted until the last reference to the file has been destroyed. Since the pedclient has the file open, logging continues, because technically the log file still exists. Applications, including the pedclient, have no idea that anything is wrong.
2. Upon stopping the pedclient, the log file is deleted. When the pedclient gets started again, the HSM tries to tell the pedclient to use the old path. This path doesn't exist anymore, so log messages are not offloaded. At this point, the HSM starts storing log messages internally. With 16 MB of Flash memory dedicated to this purpose, that works out to 198,120 messages, maximum. This space can actually fill up very quickly, in as little as a few minutes under heavy load.
3. At this point the user must set the audit log path to a valid value. and the HSM offloads all stored log messages to the host. This takes a couple of minutes, during which time the HSM is unresponsive.
4. Once all messages have been offloaded, normal operation resumes with messages being sent to the host and not being stored locally inside the HSM.

Audit Logging General Advice and Recommendations

The Security Audit Logging feature can produce a significant volume of data. It is expected, however, that Audit Officers will configure it properly for their specific operating environments. The data produced when the feature has been properly configured might be used for a number of reasons, such as:

> Reconstructing a particular action or set of actions (forensics)

> Tracing the actions of an application or individual user (accounting)

> Holding a specific individual accountable for their actions (non-repudiation)

That last point represents the ultimate conclusion of any audit trail – to establish an irrefutable record of the chain of events leading up to a particular incident for the purpose of identifying and holding accountable the individual responsible. Not every organization will want to use security audit to meet the strict requirements of establishing such a chain of events. However, all security audit users will want to have an accurate representation of a particular sequence of events. To ensure that the audit log does contain an accurate representation of events and that it can be readily interpreted when it is reviewed, these basic guidelines should be followed after the audit logging feature has been properly configured:

> Use a shell script to execute the lunacm:> **audit time sync** command at least once every 24 hours, provided the host has maintained its connection(s) to its configured NTP server(s).
For newer firmware versions, that have HSM Policy **57 - Allow sync with host time**, you can initialize the time on the HSM, then set the policy on, to automatically sync the HSM with the local host every 24 hours.

> Do not allow synchronization with the host's clock if the host has lost connectivity to NTP. This ensures that the HSM's internal clock is not set to a less accurate time than it has maintained internally. In general, the HSM's RTC will drift much less than the host's RTC and will, therefore, be significantly more accurate than the host in the absence of NTP.

> Review logs at least daily and adjust configuration settings if necessary. It is important that any anomalies be identified as soon as possible and that the logging configuration that has been set is effective.

> The audit log records are comma-delimited. We recommend that full use be made of the CSV formatting to import records into a database system or spreadsheet tool for analysis, if an SIEM system is not available.

> The ASCII hex data representing the command and returned values and error code should be examined if an anomaly is detected in log review/analysis. It may be possible to match this data to the HSM's dual-port data. The dual-port, if it is available, will contain additional data that could be helpful in establishing the context surrounding the anomalous event. For example, if an unexpected error occurs it could be possible to identify the trace through the firmware subsystems associated with the error condition. This information would be needed to help in determining if the error was unexpected but legitimate or if it was forced in an attempt to exploit a potential weakness.

An important element of the security audit logging feature is the 'Log External' function. See Audit Logging for more information. For applications that cannot add this function call, it is possible to use lunacm:> **audit logmsg** within a startup script to insert a text record at the time the application is started.

> **NOTE** Audit log and syslog entries are timestamped in UTC format.

Disk Full

In the event that all the audit disk space is used up, audit logs are written to the HSM's small persistent memory. When the HSM's persistent memory is full, normal crypto commands will fail with "disk full" error.

To resolve that situation, the audit user must:

1. Archive the audit logs on the host side.
2. Move the audit logs to some other location for safe storage.
3. Clear the audit log directory.
4. Restart the logger daemon (**PEDclient**).

   > **"pedclient -mode stop" on page 175**

   > **"pedclient -mode start" on page 173**

To prevent the "disk full" situation, we recommend that the audit user routinely archive the audit logs and clear the audit log directory.

TIP Log entries are stored in the cryptographic module (HSM) until they are rotated off. Log entries are not rotated out of the cryptographic module *until* the audit user is initialized and audit logging is configured. By default, even if there is no audit user or configuration, the cryptographic module logs unconditional events within its own memory, like:

> zeroize
> decommission
> hardware tamper
> card removal
> etc.

If the crypto module internal space ever fills completely with log records,

> whether slowly from unconditional logs, or
> quickly from more voluble high-volume event recording,

...the HSM / cryptographic module would *stop all operations* that were notrole init **-name au** and audit config. The HSM would resume providing service only after the audit user cleared the logs.

To avoid that ever happening, configure audit logging to organize log parameters and handling, being sure to set sufficient frequency of rotation for the volume of record generation that you enable.

Best practice is to:

> initialize the audit role as soon as the HSM is first powered on for production role init **-name au**
> configure the log storage path on the external file system, along with the types of events to log, the rotation interval, etc. audit config
> then, initialize the HSM Security Officer (this helps ensure that all messages, demanded by your auditing authority, are captured) hsm init
> then, proceed with partition initialization and usage with your application(s)
> then, *revisit* audit log configuration at regular intervals to tune the balance between
  - desired message types,
  - volume of audited actions normally performed (*),
  - and so on
> when you change behavior of the crypto module or change the types of events to audit, be sure to *revisit also* the rotation interval.

[* Example, you might always want to record the generation of keys, but if usage of those keys is very high-volume (like in some signature use-cases), and thus would generate a high volume of log entries, it might be permissible, and prudent, to log only first-use of any key. Check with the relevant authority.]

Logging In as Auditor

Before you can change the audit logging configuration, archive audit logs, or verify audit logs from another HSM, you must log in to the Luna PCIe HSM 7's Admin partition as Auditor (AU), or relevant commands will fail.

To log in as Auditor

1. Launch LunaCM on the Luna PCIe HSM 7 host workstation.
2. Set the active slot to the HSM Admin partition.

   lunacm:> **slot set -slot** <slotnum>
3. Log in as Auditor.

   lunacm:> **role login -name au**

   You are prompted for the Auditor credential.

Failed Auditor Login Attempts

If you fail three (3) consecutive Auditor login attempts, the Auditor role is locked out for ten minutes.

> **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.

Configuring and Using Audit Logging

This section describes the procedures required to enable audit logging, configure it to specify what is logged and how often the logs are rotated, and how to copy, verify and read the audit logs. It contains the following information:

> "Configuring Audit Logging" below

> "Exporting the Audit Logging Secret and Importing to a Verifying HSM" on page 192

> "Audit Role Authentication Considerations" on page 193

Configuring Audit Logging

Configure audit logging using the LunaCM **audit** commands.

To configure audit logging:

1. Configure the Luna PCIe HSM 7 host computer to use network time protocol (NTP).
2. Ensure that the PEDclient service is running:

   See **"pedclient -mode show" on page 172** and **"pedclient -mode start" on page 173**.
3. Set the slot focus to the HSM administrative partition of the desired HSM:

   lunacm:> **slot set -slot** <slotnum>
4. Initialize the Auditor role (you can also use the shortcut **au**):

   lunacm:> **role init -name Auditor**

   - On password-authenticated HSMs, you are prompted for a password.

- On multifactor quorum-authenticated HSMs, you are referred to Luna PED, which prompts for a white PED key.

5. Now that the Auditor role exists on the HSM, the auditing function must be configured. However, before you can configure you must log in as the Auditor user (you can also use the shortcut **au**):

   lunacm:> **role login -name au**

   - On password-authenticated HSMs, you are prompted to enter the password for the Auditor user.
   - On multifactor quorum-authenticated HSMs, you are referred to the Luna PED, which prompts for the white PED key for the Auditor user.

6. Set the domain for the Audit role:

   lunacm:> **role setdomain**

7. Synchronize the HSM's clock with the host time (which should also be synchronized with the NTP server) so that all subsequent log records will have a valid and accurate timestamp.

   lunacm:> **audit time sync**

8. Set the filepath where log files are to be saved. You must complete this step before you can start event logging.

   lunacm:> **audit config path** <filepath>

   If you previously configured logging on the HSM and then made changes to your configuration that made that path invalid (such as deleting the path outside of LunaCM or reinstalling the HSM in a different host system), set a valid log path by running **audit config path** before restarting event logging. If the log path is set incorrectly, logs will be stored in the HSM's limited memory and not exported to the file system. Event logs may be lost if the HSM's memory runs out.

9. Configure audit logging to specify what you want to log. You can specify the level of audit appropriate for needs of the organization's policy and the nature of the application(s) using the HSM:

   lunacm:> **audit config evmask** <event_value>

   > **NOTE** Before you configure audit logging, we suggest using **audit config ?** to see all the available options in the configuration process.

   Security audits can generate a very large amount of data, which consumes HSM processing resources, host storage resources, and makes the job of the Auditor quite difficult when it comes time to review the logs. For this reason, ensure that you configure audit logging such that you capture only relevant data, and no more.

   For example, the **First Symmetric Key Usage Only** or **First Asymmetric Key Usage Only** category is intended to assist Auditors to capture the relevant data in a space-efficient manner for high processing volume applications. On the other hand, a top-level Certificate Authority would likely be required, by policy, to capture all operations performed on the HSM but, since it is typically not an application that would see high volumes, configuring the HSM to audit all events would not impose a significant space and/or performance premium in that situation.

   As a further example, the command **audit config evmask all** will log everything the HSM does. This might be useful in some circumstances, but will quickly fill up log files.

10. Configure audit logging to specify how often you want to rotate the logs. Log entries are made within the HSM, and are written to the currently active log file. When a log file reaches the rotation trigger, it is closed,

and a new file gets the next log entry. The number of log files grows according to the logging settings and the rotation schedule that you configured. At any time, you can copy files to a remote computer and then clear the originals from the HSM, if you wish to free the space.

a. Specify the rotation interval. You can rotate the logs hourly, daily, weekly, monthly, or never.

lunacm:> **audit config interval** <value>

b. Specify the maximum log file size. When the log reaches the maximum size, it is automatically rotated, regardless of rotation interval:

lunacm:> **audit config size** <size>

For example, the commands **audit config interval daily** and **audit config size 4m** would rotate the logs every day, unless they reached a size of 4 Mb first, in which case they would be rotated automatically. The daily rotation would still occur.

> **CAUTION!** This step is very important. If you do not configure the log rotation correctly, logs are stored on the HSM and have nowhere to go. If the logs fill up all available space on the HSM, most operations will fail with CKR_LOG_FULL, and cryptographic services will be interrupted.

See **audit config** for additional examples.

Exporting the Audit Logging Secret and Importing to a Verifying HSM

You can export the audit log secret from one HSM and import it to another to allow the first HSM's logs to be viewed and verified on the second. The HSMs must share the same authentication method and Audit cloning domain (password string or red PED key). You can verify logs from a Luna PCIe HSM 7 using a Luna Network HSM 7, and vice-versa.

To export the Audit Logging secret from the HSM and import to the verifying HSM

1. Export the audit logging secret to the user local directory. The file is written to the subdirectory specified by a previous **audit config path** command.

lunacm:> **audit export file** <filename>

2. Exit LunaCM and list the contents of the **lunalogs** directory to see the filename of the wrapped log secret:

Linux	**ls** <client_install_dir>**/lunalog** `123456 7001347 123456.lws`

Windows	**dir** <client_install_dir>**\lunalog**
	`04/12/2017 03:56 PM <DIR> 123456`
	`04/05/2017 02:35 PM <DIR> 7001347`
	`04/05/2017 02:35 PM 48 123456.lws`

3. Transfer the logging secret to the HSM that will verify the logs. If you are verifying the logs with another locally-installed Luna PCIe HSM 7, skip this step.

- If you are planning to verify logs with a Luna PCIe HSM 7, use **pscp** or **scp** to transfer the logging secret to the appliance. Provide the audit user's credentials when prompted.

  <client_install_dir>:>**pscp** <log_secret_file> **audit@**<hostname_or_IP>**: .**

- If you are planning to verify logs with a Luna PCIe HSM 7 installed in a different host computer, you can use **scp**, **pscp**, or other secure means to transfer the logging secret.

  <client_install_dir>:>**pscp** <log_secret_file> <user>**@**<hostname_or_IP>**: .**

4. Log in to the verifying HSM appliance as the **audit** user. For this example, we will assume that you have already initialized the HSM audit user role, using the same domain/secret as is associated with the source HSM.
   - If you are using a Luna Network HSM 7, connect via SSH and log in to LunaSH as the **audit** user:

     lunash:> **audit login**
   - If you are using a Luna PCIe HSM 7 or Luna USB HSM 7, open LunaCM and log in using the Auditor role:

     lunacm:> **role login -name au**
5. Import the audit logging secret to the HSM.
   - Luna Network HSM 7 (LunaSH):

     lunash:> **audit secret import -serialtarget** <target_HSM_SN> **-serialsource** <source_HSM_SN> **-file** <log_secret_file>
   - Luna PCIe HSM 7 or Luna USB HSM 7 (LunaCM):

     lunacm:> **audit import file** <log_secret_file>
6. You can now verify audit log files from the source HSM.
   - Luna Network HSM 7 (LunaSH):

     lunash:> **audit log verify -file** <audit_log_filename>**.log**
   - Luna PCIe HSM 7 or Luna USB HSM 7 (LunaCM):

     lunacm:> **audit verify file** <audit_log_filename>**.log**

     You might need to provide the full path to the file, depending upon your current environment settings.

> **NOTE** Linux users, if you notice that audit log messages are going to more than one location on your file system, you can edit the /etc/rsyslog.conf file to prevent reporting local3.info messages in /var/log/messages as follows:
>
> ```
> //Log anything (except local3 and mail) of level info or higher.
> *.info;local3.none;mail.none;authpriv.none;cron.none /var/log/messages
> ```
>
> The portion highlighted in red stops the duplication of output. This change is optional.

Audit Role Authentication Considerations

> The audit role PED key or password is a critical property to manage the audit logs. If that authentication secret is lost, the HSM must be factory reset (that is, zeroize the HSM) in order to initialize the audit role again.

> Multiple bad logins produce different results for the HSM SO and for the audit role, as follows:
>
> - After 3 bad SO logins, the LUNA_RET_SO_LOGIN_FAILURE_THRESHOLD error is returned and the HSM is zeroized.

- After 3 bad audit logins, the LUNA_RET_AUDIT_LOGIN_FAILURE_THRESHOLD error is returned, but the HSM is unaffected. If a subsequent login attempt is executed within 30 seconds, the LUNA_RET_AUDIT_LOGIN_TIMEOUT_IN_PROGRESS error is returned. If you wait for more than 30 seconds and try login again with the correct password, the login is successful.

Audit Log Categories and HSM Events

This section provides a summary of the audit log categories and their associated HSM events.

Partition Role IDs

If you are using a Luna PCIe HSM 7 with Luna HSM Firmware 7.7.0 or newer and Luna HSM Client 10.3.0 or newer, the HSM event log reports events with the following IDs assigned to each partition role:

Administrative Partition Role IDs

Partition Role	Role ID
Administrator	0
HSM Security Officer	1
Auditor	8

Application Partition Role IDs

Partition Role	Role ID
Partition Security Officer	1
Crypto Officer	0
Limited Crypto Officer	9
Crypto User	5

HSM Access

HSM Event	Description
LUNA_LOGIN	C_Login. This event must be allowed to proceed even if the result should be logged but cannot (for example, due to a log full condition).

HSM Event	Description
LUNA_LOGOUT	C_Logout. This event must be allowed to proceed even if the result should be logged but cannot (for example, due to a log full condition).
LUNA_LOGOUT_OTHER	C_LogoutOther. This event must be allowed to proceed even if the result should be logged but cannot (for example, due to a log full condition).
LUNA_MODIFY_OBJECT	C_SetAttributeValue
LUNA_OPEN_SESSION	C_OpenSession. This event must be allowed to proceed even if the result should be logged but cannot (for example, due to a log full condition).
LUNA_CLOSE_ALL_ SESSIONS	C_CloseAllSessions
LUNA_CLOSE_SESSION	C_CloseSession This event must be allowed to proceed even if the result should be logged but cannot (for example, due to a log full condition).
LUNA_OPEN_ACCESS	CA_OpenApplicationID
LUNA_CLEAN_ACCESS	CA_Restart, CA_RestartForContainer
LUNA_CLOSE_ACCESS	CA_CloseApplicationID
LUNA_LOAD_CUSTOM_ MODULE	CA_LoadModule
LUNA_LOAD_ENCRYPTED_ CUSTOM_MODULE	CA_LoadEncryptedModule
LUNA_UNLOAD_CUSTOM_ MODULE	CA_UnloadModule
LUNA_EXECUTE_CUSTOM_ COMMAND	CA_PerformModuleCall
LUNA_HA_LOGIN	CA_HAGetLoginChallenge, CA_HAAnswerLoginChallenge, CA_HALogin, CA_HAAnswerMofNChallenge, HAActivateMofN

Log External

HSM Event	Description
LUNA_LOG_EXTERNAL	CA_LogExternal

HSM Management

HSM Event	Description
LUNA_ZEROIZE	CA_FactoryReset This event is logged unconditionally.
LUNA_INIT_TOKEN	C_InitToken This event is logged unconditionally.
LUNA_SET_PIN	C_SetPIN
LUNA_INIT_PIN	C_InitPIN
LUNA_CREATE_CONTAINER	CA_CreateContainer
LUNA_DELETE_CONTAINER	CA_DeleteContainer, CA_DeleteContainerWithHandle
LUNA_SEED_RANDOM	C_SeedRandom
LUNA_EXTRACT_CONTEXTS	C_GetOperationState
LUNA_INSERT_CONTEXTS	C_SetOperationState
LUNA_SELF_TEST	C_PerformSelfTest
LUNA_LOAD_CERT	CA_SetTokenCertificateSignature
LUNA_HA_INIT	CA_HAInit
LUNA_SET_HSM_POLICY	CA_SetHSMPolicy
LUNA_SET_DESTRUCTIVE_HSM_POLICY	CA_SetDestructiveHSMPolicy
LUNA_SET_CONTAINER_POLICY	CA_SetContainerPolicy
LUNA_SET_CAPABILITY	Internal, for capability update
LUNA_CREATE_LOGIN_CHALLENGE	CA_CreateLoginChallenge

HSM Event	Description
LUNA_REQUEST_CHALLENGE	CA_SIMInsert, CA_SIMMultiSign
LUNA_PED_INIT_RPV	CA_InitializeRemotePEDVector
LUNA_PED_DELETE_RPV	CA_DeleteRemotePEDVector
LUNA_MTK_LOCK	Internal, for manufacturing
LUNA_MTK_UNLOCK_CHALLENGE	Internal, for manufacturing
LUNA_MTK_UNLOCK_RESPONSE	Internal, for manufacturing
LUNA_MTK_RESTORE	CA_MTKRestore
LUNA_MTK_RESPLIT	CA_MTKResplit
LUNA_MTK_ZEROIZE	CA_MTKZeroize
LUNA_FW_UPGRADE_INIT	CA_FirmwareUpdate
LUNA_FW_UPGRADE_UPDATE	CA_FirmwareUpdate
LUNA_FW_UPGRADE_FINAL	CA_FirmwareUpdate
LUNA_FW_ROLLBACK	CA_FirmwareRollback
LUNA_MTK_SET_STORAGE	CA_MTKSetStorage
LUNA_SET_CONTAINER_SIZE	CA_SetContainerSize

Key Management

HSM Event	Description
LUNA_CREATE_OBJECT	C_CreateObject
LUNA_COPY_OBJECT	C_CopyObject
LUNA_DESTROY_OBJECT	C_DestroyObject
LUNA_DESTROY_MULTIPLE_OBJECTS	CA_DestroyMultipleObjects
LUNA_GENERATE_KEY	C_GenerateKey

HSM Event	Description
LUNA_GENERATE_KEY_PAIR	C_GenerateKeyPair
LUNA_WRAP_KEY	C_WrapKey
LUNA_UNWRAP_KEY	C_UnwrapKey
LUNA_DERIVE_KEY	C_DeriveKey
LUNA_GET_RANDOM	C_GenerateRandom
LUNA_CLONE_AS_SOURCE, LUNA_REPLICATE_AS_SOURCE	CA_CloneAsSource
LUNA_CLONE_AS_TARGET_INIT, LUNA_REPLICATE_AS_TARGET_INIT	CA_CloneAsTargetInit
LUNA_CLONE_AS_TARGET, LUNA_REPLICATE_AS_TARGET	CA_CloneAsTarget
LUNA_GEN_TKN_KEYS	CA_GenerateTokenKeys
LUNA_GEN_KCV	CA_ManualKCV, C_InitPIN, C_InitToken, CA_InitAudit
LUNA_SET_LKCV	CA_SetLKCV
LUNA_M_OF_N_GENERATE	CA_GenerateMofN_Common, CA_GenerateMofN
LUNA_M_OF_N_ACTIVATE	CA_ActivateMofN
LUNA_M_OF_N_MODIFY	CA_ActivateMofN
LUNA_EXTRACT	CA_Extract
LUNA_INSERT	CA_Insert
LUNA_LKM_COMMAND	CA_LKMInitiatorChallenge, CA_LKMReceiverResponse, CA_LKMInitiatorComplete, CA_LKMReceiverComplete.
LUNA_MODIFY_USAGE_COUNT	CA_ModifyUsageCount

Key Usage and Key First Usage

HSM Event	Description
LUNA_ENCRYPT_INIT	C_EncryptInit
LUNA_ENCRYPT	C_Encrypt
LUNA_ENCRYPT_END	C_EncryptFinal
LUNA_DECRYPT_INIT	C_DecryptInit
LUNA_DECRYPT	C_Decrypt
LUNA_DECRYPT_END	C_DecryptFinal
LUNA_DIGEST_INIT	C_DigestInit
LUNA_DIGEST	C_Digest
LUNA_DIGEST_KEY	C_DigestKey
LUNA_DIGEST_END	C_DigestFinal
LUNA_SIGN_INIT	C_SignInit
LUNA_SIGN	C_Sign
LUNA_SIGN_END	C_SignFinal
LUNA_VERIFY_INIT	C_VerifyInit
LUNA_VERIFY	C_Verify
LUNA_VERIFY_END	C_VerifyFinal
LUNA_SIGN_SINGLEPART	C_Sign
LUNA_VERIFY_SINGLEPART	C_Verify
LUNA_WRAP_CSP	CA_CloneMofN_Common
LUNA_M_OF_N_DUPLICATE	CA_DuplicateMofN
LUNA_ENCRYPT_SINGLEPART	C_Encrypt
LUNA_DECRYPT_SINGLEPART	C_Decrypt

Per-Key Authorization

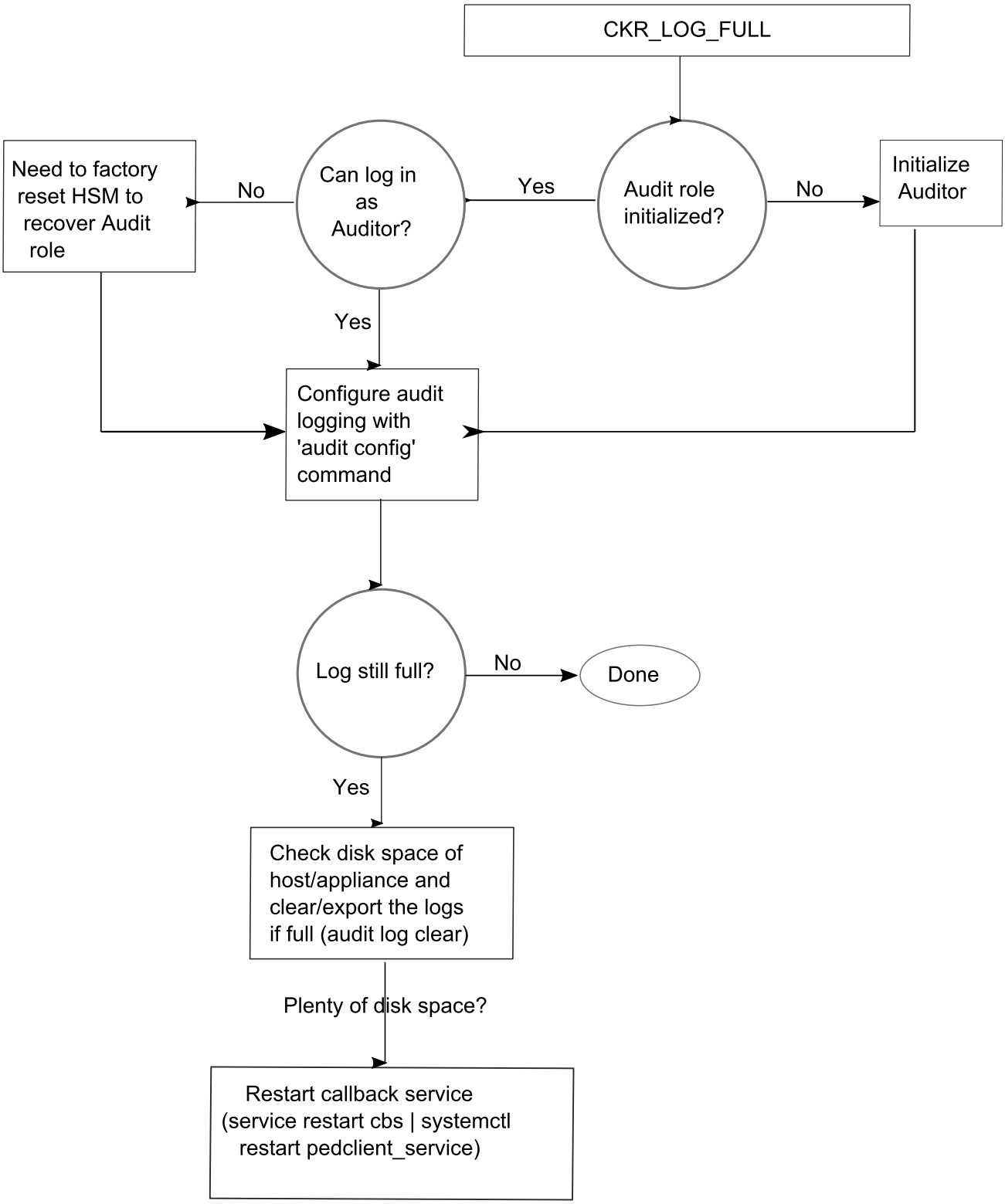
HSM Event	Description
LUNA_AUTHORIZE_KEY	CA_AuthorizeKey
LUNA_SET_AUTHORIZATION_DATA	CA_SetAuthorizationData
LUNA_RESET_AUTHORIZATION_DATA	CA_ResetAuthorizationData
LUNA_ASSIGN_KEY	CA_AssignKey
LUNA_INCREMENT_FAILED_AUTH_COUNT	CA_IncrementFailedAuthCount

Audit Log Management

HSM Event	Description
LUNA_LOG_SET_TIME	CA_TimeSync
LUNA_LOG_GET_TIME	CA_GetTime
LUNA_LOG_SET_ CONFIG	CA_LogSetConfig This event must be allowed to proceed even if the result should be logged but cannot (for example, due to a log full condition).
LUNA_LOG_GET_ CONFIG	CA_LogGetConfig This event must be allowed to proceed even if the result should be logged but cannot (for example, due to a log full condition).
LUNA_LOG_VERIFY	CA_LogVerify
LUNA_CREATE_AUDIT_ CONTAINER **	CA_ InitAudit The event is logged unconditionally.
LUNA_LOG_IMPORT_ SECRET	CA_LogImportSecret
LUNA_LOG_EXPORT_ SECRET	CA_LogExportSecret

Audit Log Troubleshooting

The following sequence might help for problems with audit logging, like "log full."



CHAPTER 7: Initializing the HSM

Initialization prepares a new HSM for use, or an existing HSM for reuse. You must initialize the HSM before you can generate or store objects, allow clients to connect, or perform cryptographic operations:

> On a new or factory-reset HSM, initialization sets the HSM SO credentials, the HSM label, and the cloning domain of the HSM Admin partition. This is often referred to as a 'hard' initialization. See "Initializing a New or Factory-reset HSM" below.

> On an initialized HSM, re-initialization destroys all existing partitions and objects, but retains the SO credentials and cloning domain. You have the option to change or retain the existing label. This is often referred to as a 'soft' initialization. See "Re-initializing the HSM" on page 205.

> **NOTE** To ensure accurate auditing, perform initialization only after you have set the system time parameters (time, date, time zone, use of NTP (Network Time Protocol). You can use the **-authtimeconfig** option when initializing the HSM to require HSM SO authorization of any time-related changes once the HSM is initialized.

Hard versus soft initialization

The following table summarizes the differences between a hard and soft initialization.

Condition/Effect	Soft init	Hard init
HSM SO authentication required	Yes	No
Can set new HSM label	Yes	Yes
Creates new HSM SO identity	No	Yes
Creates new Domain	No	Yes
Destroys partitions	Yes	No (none exist to destroy)
Destroys objects	Yes	No (none exist to destroy)

Initializing a New or Factory-reset HSM

> **NOTE** New HSMs are shipped in Secure Transport Mode (STM). You must recover the HSM from STM before you can initialize the HSM. See "Secure Transport Mode" on page 88 for details.

On a new, or factory-reset HSM (using **hsm factoryreset**), the following attributes are set during a hard initialization:

HSM Label	The label is a string that uniquely identifies this HSM. 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. The following characters are allowed: `abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789 !@#$%^&*()-_=+[]{}\\|/;:'",.<>?`~` Spaces are allowed; enclose the label in double quotes if it includes spaces. Including both spaces and quotation marks in a label may cause unexpected labeling behavior. For more information, refer to "Name, Label, and Password Requirements" on page 208.
HSM SO credentials	For multifactor quorum-authenticated HSMs, you create a new HSM SO (blue) PED key(set) or re-use an existing PED key(set) from an HSM you want to share credentials with. If you are using multifactor quorum authentication, ensure that you have an authentication strategy before beginning. See "Multifactor Quorum Authentication" on page 92. For password-authenticated HSMs, you specify the HSM SO password. For proper security, it should be different from the appliance admin password, and employ standard password-security characteristics. In LunaCM, passwords and activation challenge secrets must be 8-255 characters in length (**NOTE:** If you are using firmware version 7.0.x, 7.3.3, or 7.4.2, activation challenge secrets must be 7-16 characters in length). The following characters are allowed: `abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789 !@#$%^&*()-_=+[]{}\\|/;:',.<>?`~` Double quotation marks (`"`) are problematic and should not be used within passwords. Spaces are allowed; to specify a password with spaces using the **-password** or **-newpw** option of a command, enclose the password in double quotation marks.

Cloning domain for the HSM Admin partition	The cloning domain is a shared identifier that makes cloning possible among a group of HSM partitions. It specifies the security domain (group of HSM partitions) within which the HSM Admin partition can share cryptographic objects though cloning, backup/restore, or in high availability configurations. Note that the HSM Admin partition cloning domain is independent of the cloning domain specified when creating application partitions on the HSM.
	For multifactor quorum-authenticated HSMs, you create a new Domain (red) PED key(set) or re-use an existing PED key(set) from an HSM you want to be able to clone with.
	For password-authenticated HSMs, you create a new domain string or re-use an existing string from an HSM you want to be able to clone 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.

To initialize a new or factory-reset HSM

1. Open a LunaCM session and set the active slot to the HSM Admin partition.
2. If Secure Transport Mode is set, you must unlock the HSM before proceeding. New Luna HSMs are shipped from the factory in Secure Transport Mode (STM). STM allows you to verify whether or not an HSM has been tampered while it is not in your possession, such as when it is shipped to another location, or placed into storage. See "Secure Transport Mode" on page 88 for more information.

   To recover your HSM from Secure Transport Mode, proceed as follows:

   a. As part of the delivery process for your new HSM, you should have received an email from Thales Client Services, containing two 16-digit strings, as follows. You will need both of these strings to recover the HSM from STM:

      Random User String: XXXX-XXXX-XXXX-XXXX

      Verification String: XXXX-XXXX-XXXX-XXXX

   b. Ensure that you have the Random User String and Verification String that were emailed to you for your new HSM.

   c. Enter the following command to recover from STM, specifying the Random User String that was emailed to you for your new HSM:

      lunacm:> **stm recover -randomuserstring** <XXXX-XXXX-XXXX-XXXX>

   d. You are presented with a verification string. If the verification string matches the original verification string emailed to you for your new HSM, the HSM has not been tampered, and can be safely deployed. If the verification string does not match the original verification string emailed to you for your new HSM, the HSM has been tampered while in STM. If the verification strings do not match, contact Thales Technical Support immediately.

   e. Enter **proceed** to recover from STM (regardless of whether the strings match or not), or enter **quit** to remain in STM.
3. If you are initializing a multifactor quorum-authentication HSM, have the Luna PED connected and ready (via USB, in Local PED-USB mode). If your PED is not in USB mode, see "Changing Modes" on page 104. Alternatively, have a Remote PED instance set up, see "About Remote PED" on page 108.
4. Run the **hsm init** command, specifying a label for your Luna PCIe HSM 7:

   lunacm:> **hsm init -label** <label>
5. Respond to the prompts to complete the initialization process:
   - on a password-authenticated HSM, you are prompted for the HSM password and for the HSM Admin partition cloning domain string (cloning domains for application partitions are set when the application partitions are initialized).
   - on a multifactor quorum-authenticated HSM, you are prompted to attend to the Luna PED to create a new HSM SO (blue) PED key for this HSM, re-use an HSM SO PED key from an existing HSM so that you can also use it to log in to this HSM, or overwrite an existing key with a new authentication secret for use with this HSM. You are also prompted to create, re-use, or overwrite the Domain (red) PED key. You can create MofN quorum keysets and duplicate keys as required. See "Multifactor Quorum Authentication" on page 92 for more information.

Re-initializing the HSM

On an existing, non-factory-reset HSM, re-initialization clears all existing partitions and objects, but retains the SO credentials and cloning domain. You have the option to change or retain the existing label. Re-initialization is also referred to as a soft init. If you do not want to do a soft init, and also change the SO credentials and cloning domain, you need to use the **hsm factoryreset** command to factory reset the HSM, and then perform the procedure described in "Initializing a New or Factory-reset HSM" on page 202.

> **CAUTION!** Ensure you have backups for any partitions and objects you want to keep, before reinitializing the HSM.

To re-initialize the HSM (soft init)

1. Open a LunaCM session and set the slot to the HSM Admin partition.
2. Log in as the HSM SO.
3. If Secure Transport Mode is set, you must unlock the HSM before proceeding. See "Secure Transport Mode" on page 88.
4. If you are initializing a multifactor quorum-authenticated HSM, have the Luna PED connected and ready (via USB, in Local PED-USB mode). If your PED is not in USB mode, see "Changing Modes" on page 104.
5. Re-initialize the HSM, specifying a label for your Luna PCIe HSM 7:

   lunacm:> **hsm init -label** <label>

CHAPTER 8: HSM 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 host system, 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.

Luna PCIe HSM 7 divides roles on the HSM according to an enhanced version of the PKCS#11 standard. Configuration, administration, and auditing of the HSM itself is the responsibility of the roles described below. Cryptographic functions take place on the application partition, which has a different set of independent roles (see Partition Roles).

Personnel holding the HSM roles described below access HSM functions by logging in to the Admin partition on the HSM using LunaCM. They must therefore have the appropriate Administrator access to the workstation hosting the Luna PCIe HSM 7.

The HSM-level roles are as follows:

HSM Security Officer (SO)

The HSM SO handles all administrative and configuration tasks on the HSM, including:

> Initializing the HSM and setting the SO credential (see "Initializing the HSM" on page 202)

> Setting and changing global HSM policies (see "HSM Capabilities and Policies" on page 210)

> Creating/deleting the application partition (see "Creating or Deleting an Application Partition" on page 227)

> Updating the HSM firmware (see"Updating the Luna PCIe HSM 7 Firmware" on page 294)

Managing the HSM Security Officer Role

Refer also to the following procedures to manage the HSM SO role:

> "Logging In as HSM Security Officer" on the next page

> "Changing a Role Credential" on the next page

Auditor (AU)

The Auditor is responsible for managing HSM audit logging. These responsibilities have been separated from the other roles on the HSM and application partition so that the Auditor can provide independent oversight of all HSM processes, and no other user, including the HSM SO, can clear those logs. The Auditor's tasks include:

> Initializing the Auditor role

> Setting up audit logging on the HSM

> Configuring the maximum size of audit log files and the time interval for log rotation

> Archiving the audit logs

Managing the Auditor Role

Refer to "Configuring and Using Audit Logging" on page 190 for procedures involving the Auditor role. See also:

> "Logging In as Auditor" on page 189

> "Changing a Role Credential" below

Administrator (AD)

The HSM Administrator is a deprecated role on the Admin partition whose functions are now served by the application partition roles (see Partition Roles). Initializing this role is not recommended.

Logging In as HSM Security Officer

Before you can create an application partition or perform other administrative functions on the HSM, you must log in to the Luna PCIe HSM 7's Admin partition as HSM Security Officer (SO), or administrative commands will fail.

To log in as HSM SO

1. Launch LunaCM on the Luna PCIe HSM 7 host workstation.
2. Set the active slot to the HSM Admin partition.

   lunacm:> **slot set -slot** <slotnum>
3. Log in as HSM SO.

   lunacm:> **role login -name so**

   You are prompted for the HSM SO credential.

Failed HSM SO Login Attempts

If you fail three (3) consecutive HSM SO login attempts, application partitions are destroyed, the HSM is zeroized and all of its contents are rendered unrecoverable. The number is not adjustable. As soon as you authenticate successfully, the counter is reset to zero.

> **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.

Changing a 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 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 role (HSM SO, Auditor, 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" on page 214** 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 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).

   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 132 for details on creating PED keys.

   In LunaCM, passwords and activation challenge secrets must be 8-255 characters in length (**NOTE:** If you are using firmware version 7.0.x, 7.3.3, or 7.4.2, activation challenge secrets must be 7-16 characters in length). The following characters are allowed:
   `abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789 !@#$%^&*()-_=+[]{}\|/;:',.<>?`~`
   Double quotation marks (`"`) are problematic and should not be used within passwords.
   Spaces are allowed; to specify a password with spaces using the **-password** or **-newpw** option of a command, enclose the password in double quotation marks.

   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.

Name, Label, and Password Requirements

This page describes length and character requirements for setting labels, domains, passwords, and challenge secrets on the Luna PCIe HSM 7. This information can also be found in relevant sections throughout the documentation. Refer to the applicable section below:

> "HSM Labels" below
> "Cloning Domains" below
> "Partition Labels" below
> "Role Passwords or Challenge Secrets" below

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. The following characters are allowed:

```
abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789 !@#$%^&*()-_=+[]{}\|/;:'",.<>?`~
```

Spaces are allowed; enclose the label in double quotes if it includes spaces. Including both spaces and quotation marks in a label may cause unexpected labeling behavior.

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 Labels

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.

Role Passwords or Challenge Secrets

In LunaCM, passwords and activation challenge secrets must be 8-255 characters in length (**NOTE:** If you are using firmware version 7.0.x, 7.3.3, or 7.4.2, activation challenge secrets must be 7-16 characters in length). The following characters are allowed:

```
abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789 !@#$%^&*()-_=+[]{}\|/;:',.<>?`~
```

Double quotation marks (`"`) are problematic and should not be used within passwords.

Spaces are allowed; to specify a password with spaces using the **-password** or **-newpw** option of a command, enclose the password in double quotation marks.

CHAPTER 9: HSM Capabilities and Policies

The HSM can be configured to suit the cryptographic needs of your organization. Configurable functions are governed by the following settings:

> **HSM Capabilities** are features of HSM functionality, set at manufacture based on the HSM model you selected at time of purchase. You can add new capabilities to the HSM by purchasing and applying capability licenses from Thales (see "Upgrading HSM Capabilities" on page 296). Some capabilities have corresponding modifiable HSM policies.

> **HSM Policies** are configurable settings that allow the HSM Security Officer to modify the function of their corresponding capabilities. Some policies affect HSM-wide functionality, and others allow further customization of individual partitions by the Partition Security Officer.

The table below describes all Luna PCIe HSM 7 capabilities, their corresponding policies, and the results of changing their settings. This section contains the following procedures:

> "Setting HSM Policies Manually" on page 223

> "Setting HSM Policies Using a Template" on page 224

To zeroize the HSM and revert policies to their default values, see "Resetting the Luna PCIe HSM 7 to Factory Condition" on page 314.

To zeroize the HSM and keep the existing policy settings, use lunacm:> **hsm zeroize**

Destructive Policies

Some policies affect the security of the HSM. As a security measure, changing those security-affecting policies results in application partitions, or the entire HSM, being zeroized. Among those listed below, such policies are marked as **Destructive**.

#	HSM Capability	HSM Policy
0	**Enable PIN-based authentication** > **1**: The HSM authenticates all users with keyboard-entered passwords. > **0**: See HSM capability 1 below.	N/A

#	HSM Capability	HSM Policy
1	**Enable PED-based authentication** > **1**: The HSM authenticates users with secrets stored on physical PED keys, read by a Luna PED. The Crypto Officer and Crypto User roles may also be configured with a secondary, keyboard-entered challenge secret. > **0**: See HSM capability 0 above.	N/A
2	**Performance level** Numerical value indicates the HSM's performance level, determined by the model you selected at time of purchase: > **4: Standard** performance > **8: Enterprise** performance > **15: Maximum** performance	N/A
4	**Enable domestic mechanisms & key sizes** Always **1**. All Luna PCIe HSM 7s are capable of full-strength cryptography with no US export restrictions.	N/A
6	**Enable masking** Always **0** for HSMs with older firmware. SKS (which uses masking) was not available before Luna HSM Firmware 7.7.0. **1** for Luna PCIe HSM 7s at Luna HSM Firmware 7.7.0 and newer, to support SKS.	**Allow masking** **Destructive** If this policy is allowed, see **partition policy 3:** Allow private key masking and **partition policy 7:** Allow secret key masking.
7	**Enable cloning** Always **1**. All current Luna PCIe HSM 7s can clone cryptographic objects from one partition to another.	**Allow cloning** **Destructive** > **1** (default): The HSM may clone cryptographic objects from one partition to another. This is required to back up partitions or include them in HA groups. Partition SOs can enable/disable cloning on individual partitions. > **0**: No partition on the HSM may clone cryptographic objects. Partition SOs cannot change this.

#	HSM Capability	HSM Policy
9	**Enable full (non-backup) functionality** > **1**: The HSM is capable of full cryptographic functions. > **0**: The HSM is capable of backup functions only.	N/A
12	**Enable non-FIPS algorithms** Always **1**. The HSM can use all cryptographic algorithms described in Supported Mechanisms.	**Allow non-FIPS algorithms** **Destructive *** > **1** (default): The HSM may use all available cryptographic algorithms, meaning all the FIPS-approved algorithms as well as all the non-FIPS algorithms. > **0**: Only algorithms sanctioned by the FIPS 140-2 standard are permitted. The following is displayed in the output from lunacm:> **hsm showinfo**: `The HSM is in FIPS 140-2 approved operation mode.` **NOTE** When C_GetMechanismInfo is called and the HSM policy "Allow NonFIPS Algorithms" is disabled: > If a mechanism has the WRAP flag set and MPE_NO_WRAP, the WRAP flag is *not* returned by the HSM as part of the mechanism info. > If a mechanism has the SIGN flag set and MPE_NO_SIGN, the SIGN flag is *not* returned by the HSM as part of the mechanism info. When the policy is enabled, the HSM returns all the flags that are applicable to the requested mechanism. This policy must be ON for the HSM, in order to allow the non-FIPS choice to be made on a per-partition basis (Luna HSM Firmware 7.7.1 and newer) using partition policy 43. If this HSM policy is OFF, then non-FIPS algorithms cannot be permitted on a per-partition basis and partition policy 43 is not available for use.

#	HSM Capability	HSM Policy
15	**Enable SO reset of partition PIN** Always **1**. This capability enables: > the Partition SO to reset the password or PED key secret of the Crypto Officer. > the Crypto Officer to reset the password or PED key secret of the Crypto User.	**SO can reset partition PIN** **Destructive** > **1**: Partition SO may reset the password or PED key secret of a Crypto Officer who has been locked out after too many failed login attempts. > **0** (default): The CO lockout is permanent and the partition contents are no longer accessible. The partition must be re-initialized, and key material restored from a backup device. See Resetting the Crypto Officer or Crypto User Credential.
16	**Enable network replication** Always **1**. This capability enables cloning of cryptographic objects over a network. This is required for HA groups, and for partition backup to a remote Luna Backup HSM. **NOTE** Luna HSM Firmware 7.8.0 and newer ignores this setting where CPv4 is invoked; that cloning option is always allowed, when the corresponding cloning policies are enabled.	**Allow network replication** > **1** (default): Cloning of cryptographic objects is permitted over a network. Remote backup is allowed, and the partition may be used in an HA group. > **0**: Cloning over a network is not permitted. Partition backup is possible to a locally-connected Luna Backup HSM only.
17	**Enable Korean Algorithms** > **1**: if you have purchased and applied a license for the Korea-specific algorithm set. See "Upgrading HSM Capabilities" on page 296 to purchase this capability. > **0** if you have not applied this license.	N/A

#	HSM Capability	HSM Policy
18	**FIPS evaluated** Always **0** - deprecated capability. All Luna PCIe HSM 7s are capable of operating in FIPS Mode. **NOTE** This capability is visible (not used) in previous HSM firmware versions, but is removed from Luna HSM Firmware 7.7.0 and newer.	N/A
19	**Manufacturing Token** Always **0**. For Thales internal use only.	N/A
21	**Enable forcing user PIN change** Always **1**. This capability forces the Crypto Officer or Crypto User to change the initial role credential created by the Partition SO.	**Force user PIN change after set/reset** > **1** (default): After the Partition SO initializes or resets the Crypto Officer credential, the CO must change the credential before any other actions are permitted. This also applies when the CO initializes/resets the Crypto User role. This policy is intended to enforce the separation of roles on the partition. > **0**: The CO/CU may continue to use the credential assigned by the Partition SO. They can change / rotate the credential at any time, but the change is not forced at first use of the role after role initialization or reset. See "Changing a Role Credential" on page 207.
22	**Enable offboard storage** Always **1** - deprecated capability with no application to Luna 7.	**Allow offboard storage** **Destructive** Deprecated policy with no application to Luna 7. Default: **1**
23	**Enable partition groups** Always **0** - deprecated capability.	N/A
25	**Enable Remote PED usage** Always **1** on multifactor quorum-authenticated HSMs. Always **0** on password-authenticated HSMs.	**Allow Remote PED usage** > **1** (default): The HSM may authenticate roles using a remotely-located Luna PED server or a locally-installed Luna PED. > **0**: The HSM must use a locally-installed Luna PED to authenticate roles.

#	HSM Capability	HSM Policy
27	**HSM non-volatile storage space** Displays the maximum non-volatile storage space (in bytes) on the HSM, determined by the Luna PCIe HSM 7 model you selected at time of purchase.	N/A
30	**Enable Unmasking** Always **1**. This capability enables migration from legacy Luna HSMs that used SIM.	**Allow unmasking** > **1** (default): Cryptographic objects may be migrated from legacy Luna HSMs that used SIM. > **0**: Migration from legacy HSMs using SIM is not possible.
33	**Maximum number of partitions** Displays the maximum number of application partitions that can be created on the HSM.	**Current maximum number of partitions** You can change HSM policy 33 to lower the effective maximum number of partitions below the actual licensed maximum. You cannot, however, lower the maximum below the number of partitions currently existing on the HSM.
35	**Enable Single Domain** Always **0**.	N/A
36	**Enable Unified PED Key** Always **0**.	N/A
37	**Enable MofN** Always **1** on multifactor quorum-authenticated HSMs. Always **0** on password-authenticated HSMs.	**Allow MofN** > **1** (default): During PED key creation, you have the option to require a quorum to authenticate the role, by splitting the authentication secret among multiple PED keys (see "Quorum Split Secrets (M of N)" on page 97) > **0**: Users do not have the option to split PED key secrets (M and N are automatically set to **1**).
38	**Enable small form factor backup/restore** Always **0**.	N/A
39	**Enable Secure Trusted Channel** Always **1**.	**Allow Secure Trusted Channel** Secure Trusted Channel is a Luna Network HSM 7 feature, and has no function on Luna PCIe HSM 7. Thales does not recommend turning this policy on at any time.

#	HSM Capability	HSM Policy
40	**Enable decommission on tamper** Always **1**. This enables the HSM to be automatically decommissioned if a tamper event occurs (see "Comparing Zeroize, Decommission, and Factory Reset" on page 315).	**Decommission on tamper** **Destructive** > **1**: The HSM is decommissioned if a tamper event occurs (see "Tamper Events" on page 229). > **0** (default): The contents of the HSM are not affected by a tamper event.
42	**Enable partition re-initialize** Always **0**.	N/A
43	**Enable low level math acceleration** Always **1**. This capability enables acceleration of cryptographic functionality for maximum HSM performance.	**Allow low-level math acceleration** > **1** (default): Provides maximum HSM performance. > **0**: Do not turn this policy off unless instructed by Thales Technical Support.
46	**Allow Disabling Decommission** Always **1**. This capability enables the HSM SO to disable decommissioning of the HSM.	**Disable Decommission** **Destructive** > **1**: Decommission is disabled on the HSM, even if the PCIe battery is removed (see "Replacing the Luna PCIe HSM 7 Battery" on page 27). > **0** (default): Decommission works as described in "Decommissioning the HSM Card" on page 313. **CAUTION!** Changing this policy will destroy partitions on the HSM, and they must be recreated. If HSM policy 40 is enabled, you cannot enable this policy (fails with error: CKR_CONFIG_FAILS_DEPENDENCIES). However, attempting to enable it will still destroy HSM partitions.
47	**Enable Tunnel Slot** Always **0**.	N/A
48	**Enable Controlled Tamper Recovery** Always **1**. This capability enables the HSM SO to require tamper events to be explicitly cleared before normal operations can resume.	**Do Controlled Tamper Recovery** > **1** (default): After a tamper event, the HSM SO must explicitly clear the tamper before the HSM can resume normal operations. > **0**: The HSM must be restarted before it can resume normal operations. See "Tamper Events" on page 229 for more information.

#	HSM Capability	HSM Policy
49	**Enable Partition Utilization Metrics** Always **1**. This capability enables the HSM SO to view (or export to a named file) counters that record how many times specific cryptographic operations have been performed in application partitions since the last counter-reset event. This provides a picture of operational utilization that can be used to guide the (re-)allocation and balancing of partitions and applications, for better service to all users of your partitions.	**Allow Partition Utilization Metrics** > **1**: The HSM SO can view Partition Utilization Metrics. > **0** (default): Partition Utilization Metrics are not available. See "Partition Utilization Metrics" on page 250 for more information.

#	HSM Capability	HSM Policy
50	**Enable Functionality Modules** This capability enables Functionality Modules (FMs) to be loaded to the HSM (see "Functionality Modules" on page 298). > **1** on FM-ready HSMs with Luna HSM Firmware 7.4.0 or newer, with the FM capability license installed (see "Preparing the Luna PCIe HSM 7 to Use FMs" on page 301). > **0** on FM-ready HSMs with Luna HSM Firmware 7.4.0 or newer, without the FM capability license. Does not appear on HSMs that are not FM-ready or are running firmware older than Luna HSM Firmware 7.4.0.	**Allow Functionality Modules** **Destructive** > **1**: With this policy enabled, Functionality Modules may be loaded to the HSM, permitting custom cryptographic operations. Allows use of the **ctfm** utility and FM-related commands, and the use of Functionality Modules in general with this HSM. **NOTE** FIPS compliance requires that objects are never cloned or restored to an HSM using less secure firmware. FIPS 140 validation is performed against the HSM hardware with a specific firmware version. Since the introduction of a Functionality Module changes the firmware, allowing FMs in the HSM removes the HSM from FIPS compliance. For purposes of cloning, an HSM where FMs have *ever* been allowed is considered less secure than one where FMs have *never* been allowed. See the Caution below. You can subsequently disable FMs, but future cloning operations will work only with other FM-HOC HSMs. > **0** (default): FMs may not be loaded to the HSM. **CAUTION!** Enabling FMs (**HSM policy 50**) introduces changes to Luna HSM functionality, some of which are permanent; they cannot be removed by disabling the policy. FM-enabled status is **not** reversible by Factory Reset. Refer to "FM Deployment Constraints" on page 298 for details before enabling. If you are using Crypto Command Center, ensure that your CCC version supports FM-enabled HSMs before you enable **HSM policy 50**. Refer to the CCC CRN for details.

#	HSM Capability	HSM Policy
51	**Enable SMFS Auto Activation** This capability enables the Secure Memory File System (SMFS) to be activated automatically on startup. > **1** on FM-ready HSMs with Luna HSM Firmware 7.4.0 or newer, with the FM capability license installed (see "Preparing the Luna PCIe HSM 7 to Use FMs" on page 301). > **0** on FM-ready HSMs with Luna HSM Firmware 7.4.0 or newer, without the FM capability license. Does not appear on HSMs that are not FM-ready or are running firmware older than Luna HSM Firmware 7.4.0.	**Allow SMFS Auto Activation** **Destructive** > **1**: With this policy enabled, the Secure Memory File System (SMFS) is automatically activated on startup, providing a secure, tamper-enabled location in the HSM memory where Functionality Modules can load keys and parameters. Auto-activation for SMFS, like auto-activation for multifactor quorum-authenticated partitions in general, persists through a power outage of up to 2 hours duration. > **0** (default): If disabled, the HSM SO must manually activate the SMFS each time the HSM reboots or loses power.
52	**Allow Restricting FM Privilege Level** This capability enables the HSM SO to restrict the sensitive key attributes of partition objects from FMs. > **1** on FM-ready HSMs with Luna HSM Firmware 7.4.0 or newer, with the FM capability license installed (see "Preparing the Luna PCIe HSM 7 to Use FMs" on page 301). > **0** on FM-ready HSMs with Luna HSM Firmware 7.4.0 or newer, without the FM capability license. Does not appear on HSMs that are not FM-ready or are running firmware older than Luna HSM Firmware 7.4.0.	**Restrict FM Privilege Level** **Destructive** > **1**: FM privilege is restricted. > **0** (default): FM privilege permits FMs to see the sensitive key attributes (including key values) of cryptographic objects on application partitions. This privilege is necessary for most FMs, so that the Crypto Officer (CO) and Crypto User (CU) roles can use partition objects with the FM. However, some FMs might not require this privilege and it can be restricted to satisfy some certification requirements (such as Common Criteria).

#	HSM Capability	HSM Policy
53	**Allow Encrypting of Keys from FM to HSM** This capability enables key encryption between the FM and the Functionality Module Crypto Engine interface (FMCE). > **1** on FM-ready HSMs with Luna HSM Firmware 7.4.0 or newer, with the FM capability license installed (see "Preparing the Luna PCIe HSM 7 to Use FMs" on page 301). > **0** on FM-ready HSMs with Luna HSM Firmware 7.4.0 or newer, without the FM capability license. Does not appear on HSMs that are not FM-ready or are running firmware older than Luna HSM Firmware 7.4.0.	**Encrypt Keys Passing from FM to HSM** **Destructive** > **1**: With this policy enabled, keys created by an FM are encrypted before crossing from the FM to the Functionality Module Crypto Engine interface (FMCE). This internal encryption may be required to satisfy some certification requirements (such as Common Criteria). > **0** (default): Keys are not encrypted before crossing to the FMCE.

#	HSM Capability	HSM Policy
55	**Enable Restricted Restore** This capability allows the HSM SO to restrict a Luna Backup HSM 7 from being used with firmware older than Luna HSM Firmware 7.7.0, for any purpose other than to migrate cryptographic objects to Luna HSM Firmware 7.7.0 or newer. See Behavior of Pre-Firmware 7.7, V0, and V1 Partitions for more information. Appears on Luna Backup HSM 7 running Luna Backup HSM 7 Firmware 7.7.1 or newer. **NOTE** Not visible on Luna Network HSM 7 via lunash commands. Look for this Capability/Policy via **lunacm** when a Luna Backup HSM 7 is the current slot.	**Enable Restricted Restore** **ON-to-OFF Destructive** > **1**: Objects backed up from pre-7.7.0 firmware partitions *can only be* restored to V0 or V1 partitions (Luna HSM Firmware 7.7.0 or newer). Enable this policy to ensure FIPS compliance. > **0** (default): Objects backed up from pre-7.7.0 firmware partitions *can be* restored to pre-7.7.0 firmware partitions. Do not use this setting if you require FIPS compliance. **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.

#	HSM Capability	HSM Policy
56	**Enable User Defined ECC Curves** This capability allows the HSM SO to restrict or allow the use of user-defined ECC curves. The state of the associated policy is preserved through firmware update.	**Allow User Defined ECC Curves** **Destructive** > **1**: User-defined ECC curves can be used, without restriction. > **0** (default): Named curves (that we have verified) can still be used, as can user-defined ECC curves where the named-curve parameters are provided. User-defined ECC curves that cannot map to built-in named curves during key-pair generation, public key creation, private key unwrapping, cloning or SKS, and key derivation, return the error ECC_CURVE_NOT_ALLOWED. > Named-curve samples are provided when you include the SDK option while installing the Client. The files must be unmodified. /usr/safenet/lunaclient/samples/ecc_examples • bpP160r1.txt • bpP512t1.txt • x962_char2_163v1.txt • bpP192r1.txt • secp384r1.txt • x962_char2_359V1.txt • bpP224r1.txt • bpP384R1.txt • sm2p256v1.txt **NOTE** For FIPS compliance, NIST requires us to make security claims with respect to the curves that we support. It is impossible to test and report on all possible user-defined ECC curves. Therefore, commonly-used, named curves are explicitly tested, documented to comply with FIPS requirements, and allowed in FIPS mode.

#	HSM Capability	HSM Policy

57	**Enable Sync with Host Time**
	This capability enables the HSM SO to automatically synchronize the HSM's time to the host system time every 24 hours.
		Allow Sync with Host Time
		> **1**: The HSM's time is synchronized to the host system time once every 24 hours.
		The maximum drift that is allowed to be synchronized by this policy is 3 seconds. If the HSM time and the host time have drifted by more than 3 seconds in the last 24 hours, a log entry is created instead:
		`[HSM] LOG(INFO): Hsm clock(1647624503) drifts from host clock(1647628144) > threshold, stop sync clock!`
		`[HSM] ALM2029: HSM clock drift allowed threshold exceeded`
		This applies to the first synchronization as well -- set the time manually using hsm time get and hsm time sync before setting this policy to ON. .
		> **0** (default): HSM time is not automatically synchronized to host time. The HSM SO can still synchronize the clocks manually.

* The Backup HSM performs only backup and restore operations and is not a general-purpose HSM. It has no information about the origin of keys or objects. In the case of FIPS-mode or non-FIPS the status of a source HSM (Policy 12) is not noticed, and a target HSM decides what to do with keys from a restore operation. However, the actions of a Backup HSM can be affected by the cloning protocol that is used - see Policy 55.

Setting HSM Policies Manually

The HSM SO can change available policies to customize HSM functionality. Some policies apply to all partitions on the HSM; others enable the Partition SO to customize functionality at the partition level. Refer to "HSM Capabilities and Policies" on page 210 for a complete list of HSM policies and their effects.

In most cases, HSM policies are either enabled (**1**) or disabled (**0**), but some allow a range of values.

To change multiple policy settings during HSM initialization, see "Setting HSM Policies Using a Template" on the next page.

Prerequisites

> The HSM must be initialized (see "Initializing the HSM" on page 202).

> If you are changing a destructive policy and you have partitions existing on the HSM, back up any important cryptographic objects (see Partition Backup and Restore).

To manually set or change an HSM policy

1. Launch LunaCM and set the active slot to the HSM Admin partition.

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

2. [Optional] Display the existing HSM policy settings.

lunacm:> **hsm showpolicies**

3. Log in as HSM SO (see "Logging In as HSM Security Officer" on page 207).

   lunacm:> **role login -name so**

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).

   lunacm:> **hsm changehsmpolicy -policy** <policy_ID> **-value** <value>

   If you are changing a destructive policy, you are prompted to enter **proceed** to continue the operation.

Setting HSM Policies Using a Template

An HSM policy template is a file containing a set of preferred HSM policy settings, used to initialize HSMs with those settings. You can use the same file to initialize multiple HSMs, rather than changing policies manually after initialization. This can save time and effort when initializing multiple HSMs that are to function together (such as in 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.

See also Setting Partition Policies Using a Policy Template.

> **NOTE** This feature requires minimum Luna HSM Firmware 7.1.0.

You can create a policy template file from an initialized or uninitialized HSM, and edit it using a standard text editor.

HSM policy templates cannot be used to alter settings for an initialized HSM. Once an HSM has been initialized, the SO must change individual policy values manually (see "Setting HSM Policies Manually" on the previous page).

To zeroize the HSM and revert policies to their default values, see "Resetting the Luna PCIe HSM 7 to Factory Condition" on page 314.

To zeroize the HSM and keep the existing policy settings, use lunacm:> **hsm zeroize**

This section provides instructions for the following procedures, and some general guidelines and restrictions:

> "Creating an HSM Policy Template" below

> "Editing an HSM Policy Template" on the next page

> "Applying an HSM Policy Template" on page 226

Creating an HSM Policy Template

The following procedures describe how to generate an HSM policy template from the HSM. This can be done optionally at two points in the HSM setup process:

> before the HSM is initialized: this produces a template file containing the default policy settings, which can then be edited

> after initializing and setting the HSM policies manually: this produces a template file with the current HSM policy settings, which can then be used to initialize other HSMs with the same settings. The HSM SO must complete the procedure.

To create an HSM policy template

1. Launch LunaCM and set the active slot to the Admin partition. If you are creating a template from an initialized HSM, you must log in as HSM SO.

   lunacm:> **slot set slot** <admin_slotnum>

   lunacm:> **role login -name so**

2. Create the HSM policy template file with an original filename. Specify the path to the location where you wish to save the template. No file extension is required. If a template file with the same name exists in the specified directory, it is overwritten.

   lunacm:> **hsm showpolicies -exporttemplate** <filepath/filename>

```
lunacm:>hsm showpolicies -exporttemplate /usr/safenet/lunaclient/templates/HSMPT

HSM policies for HSM: myPCIeHSM written to /usr/safenet/lunaclient/templates/HSMPT

Command Result : No Error
```

3. Customize the template file with a standard text editor (see "Editing an HSM Policy Template" below).

Editing an HSM Policy Template

Use a standard text editor to manually edit HSM policy templates for custom configurations. This section provides template examples and customization guidelines.

HSM Policy Template Example

This example shows the contents of an HSM policy template created using the factory default policy settings. Use a standard text editor to change the policy values (0=OFF, 1=ON, or the desired value 0-255). You cannot edit the destructiveness of HSM policies. See "HSM Capabilities and Policies" on page 210 for more information.

If you export a policy template from an uninitialized HSM, the **Sourced from HSM** 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 7.1.0
# Field format - Policy ID:Policy Description:Policy Value
# Sourced from HSM: myLunaHSM, SN: 66331


6:"Allow masking":0
7:"Allow cloning":1
12:"Allow non-FIPS algorithms":1
15:"SO can reset partition PIN":0
16:"Allow network replication":1
21:"Force user PIN change after set/reset":1
22:"Allow offboard storage":1
23:"Allow partition groups":0
25:"Allow remote PED usage":0
30:"Allow unmasking":1
33:"Current maximum number of partitions":100
35:"Force Single Domain":0
36:"Allow Unified PED Key":0
37:"Allow MofN":0
```

```
38:"Allow small form factor backup/restore":0
39:"Allow Secure Trusted Channel":0
40:"Decommission on tamper":0
42:"Allow partition re-initialize":0
43:"Allow low level math acceleration":0
46:"Disable Decommission":1
47:"Allow Tunnel Slot":0
48:"Do Controlled Tamper Recovery":1
49:"Allow Partition Utilization Metrics":0
56:"Allow User Defined ECC Curves":0
```

Editing Guidelines and Restrictions

When creating or editing 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 HSM will use the default value for that policy.

> You may not use invalid policy values (outside the acceptable range), or values that conflict with your HSM's capabilities. For example, **HSM capability 6: Enable Masking** is always **Disallowed**, so you cannot set the corresponding HSM policy to **1**. If you attempt to initialize an HSM with a template containing invalid policy values, an error is returned and initialization fails.

Applying an HSM Policy Template

The following procedure describes how to initialize the HSM using a policy template.

To apply a policy template to a new HSM

1. Ensure that the template file is saved on the workstation hosting the destination HSM.
2. Launch LunaCM and initialize the destination HSM using the policy template file. If the template file is not in the same directory as LunaCM, include the correct filepath.

   lunacm:> **hsm init -label** <label> **-applytemplate** <filepath/filename>
3. Verify that the template has been applied correctly by checking the partition's policy settings.

   lunacm:> **hsm showpolicies**

CHAPTER 10: Application Partitions

The Luna PCIe HSM 7 has two partitions:

> one administrative partition, created when you initialize the HSM. The administrative partition is owned by the HSM Security Officer (SO). This partition is used by the HSM SO and the Auditor, and is not used to store cryptographic objects.

> one application partition, created by the HSM SO. The application partition is owned by its Partition Security Officer (PO), and has its own access controls and security policies independent from the administrative partition. Its function is to store cryptographic objects used by your applications.

An application partition is like a safe deposit box that resides within a bank's vault. The HSM (vault) itself offers an extremely high level of security for its contents. An application partition (safe deposit box) on the HSM has its own security and access controls, so that even though the HSM SO has access to the vault, they still cannot access the contents of the individual partitions. Only the Partition Security Officer holds the partition's administrative credentials.

This chapter contains the following procedures for managing application partitions:

> "Creating or Deleting an Application Partition" below

Creating or Deleting an Application Partition

The HSM Security Officer (SO) is responsible for creating the application partition. The HSM SO can delete the partition at any time, destroying all partition roles and stored cryptographic objects.

Prerequisites

> The HSM must be initialized (see "Initializing the HSM" on page 202).

> You require the HSM SO credential (blue PED key).

To create an application partition

1. Log in as HSM SO (see "Logging In as HSM Security Officer" on page 207).

   lunacm:> **role login -name so**

2. Create the application partition.

   lunacm:> **partition create**

3. [Optional] Confirm that the partition was created.

   lunacm:> **slot list**

To delete an application partition

1. Log in as HSM SO (see "Logging In as HSM Security Officer" on page 207).

   lunacm:> **role login -name so**

2. Delete the application partition by specifying the slot number.

   lunacm:> **partition delete -slot** <slot>

CHAPTER 11: Security in Operation

This section addresses actions and settings with security-related implications.

> "Tamper Events" below

> "Security Effects of Administrative Actions" on page 231

Refer also to Security of Your Partition Challenge.

Tamper Events

Luna PCIe HSM 7 detects hardware anomalies (such as card over-temperature) and physical events (such as card removal or chassis intrusion), and registers them as tamper events. A tamper event is considered a security breach, and effectively locks the HSM.

If **Policy 48: Do Controlled Tamper Recovery** is enabled (the default), the HSM SO must clear the tamper condition before the HSM is reset, to return the HSM to normal operation (see "HSM Capabilities and Policies" on page 210). While the HSM is in the tamper condition, only the subset of LunaCM commands required to view the HSM status or clear the tamper condition are available. For multifactor quorum-authenticated HSMs, the cached PED key data that allows activation is zeroized, and activation is disabled. When an HSM is in the tamper state, only the HSM SO is able to log in to the HSM.

You can enable **Policy 40: Decommission on Tamper** to decommission the HSM when a tamper event occurs, so that partitions and roles are deleted from the HSM. By default, **Policy 40: Decommission on Tamper** is disabled, and the contents of the HSM are not affected by the tamper event.

If both policies are disabled, the HSM sends a warning when a tamper event occurs but does not make partition data inaccessible. We do not recommend disabling both policies.

If both policies are enabled, the HSM SO role is deleted when a tamper event occurs, so you do not need to log in this role to clear the tamper condition.

There are several conditions that can result in a tamper event. The type of tamper event is indicated by the **HSM Status** field in the output of lunacm:> **slot list**. The status also indicates whether the tamper event requires an HSM reset in addition to a tamper clear.

> **NOTE** A tamper event resets the HSM hardware, including the PCIe logic. This prevents the HSM from reporting any statuses, including the cause of the tamper condition. The only thing which is detected in this case is `k7pf0: ALM0015: PCIe Link Failure`. The HSM must be rebooted before the cause of the tamper event can be reported.

Tamper event	Response
Chassis intrusion (requires chassis connector to card tamper header)	Halt the HSM. Deactivate activated partitions. Decommission the HSM if **policy 40: Decommission on Tamper** is enabled.
Card removal	Halt the HSM. Deactivate activated partitions. Decommission the HSM if **policy 40: Decommission on Tamper** is enabled.
Over/under temperature	Halt the HSM. Deactivate activated partitions. Decommission the HSM if **policy 40: Decommission on Tamper** is enabled. Warnings are logged for mild over/under temperature events. Warnings are self-clearing if the condition is resolved.
Over/under voltage	Halt the HSM. Deactivate activated partitions. Decommission the HSM if **policy 40: Decommission on Tamper** is enabled. Warnings are logged for mild over/under voltage events. Warnings are self-clearing if the condition is resolved.
Battery removal/depletion	Halt the HSM. Deactivate activated partitions. Decommission the HSM. Warnings are logged for low battery conditions.

Recovering from a Tamper Event

How you recover from a tamper event depends on how the following HSM policies are set. See "HSM Capabilities and Policies" on page 210 for more information:

Policy 40: Decommission on tamper	If enabled, the HSM is decommissioned when a tamper event occurs. You must clear the tamper condition before you can re-initialize the HSM SO, re-create your partitions, restore the partition contents from backup, and re-initialize the partition roles (Partition SO, Crypto Officer, and Crypto User, and Audit, as relevant).
Policy 48: Do Controlled Tamper Recovery	If enabled, the tamper condition that halted the HSM must be cleared by the HSM SO (by issuing the **tamper clear** command), before the HSM can be reset to resume normal operations.

Activation and auto-activation is disabled on tamper

If you are using activation or auto-activation on your multifactor quorum-authenticated partitions, it is disabled when a tamper is detected, or if any uncleared tamper conditions are detected on reboot. See Activation on Multifactor Quorum-Authenticated Partitions and Partition Capabilities and Policies for more information.

To recover from a tamper

1. View the output of lunacm:> **slot list** (displayed by default on login). The reason for the tamper is indicated by the **HSM Status** field. You can also use lunacm:> **hsm tampershow** to display the last tamper event.

   > **NOTE** The **slot list** and **hsm tampershow** commands only show the last tamper event, even if several tampers have occurred. To view a complete list of the tamper events that have occurred on the HSM, use the **lunadiag** utility.

2. Resolve the issue(s) that caused the tamper event.
3. If **Policy 48: Do Controlled Tamper Recovery** is enabled, clear the tamper condition. Otherwise, go to the next step:

   lunacm:> **hsm tamperclear**

4. If the tamper message indicates that a reset is required, exit LunaCM and use the **lunareset** utility to reset the HSM.

   lunareset <device>

5. Verify that all tampers have been cleared:

   lunacm:> **hsm tampershow**

6. If the HSM was decommissioned as a result of the tamper, you must re-create your partitions, re-initialize the partition roles (Partition SO, Crypto Officer, and Crypto User, and Audit as relevant), and restore the partition contents from backup. Refer to the following procedures:

   a. To re-create your partitions, see "Creating or Deleting an Application Partition" on page 227.

   b. Re-initialize the partition roles. See Initializing an Application Partition.

   c. To restore the partition contents from backup, see Partition Backup and Restore.

7. If the **Policy 22: Allow Activation** and/or **Policy 23: Allow AutoActivation** are enabled on your multifactor quorum-authenticated partitions, the CO and CU (if enabled) must log in to reactivate those roles:

   lunacm:> **role login -name** <role>

Security Effects of Administrative Actions

Actions that you take, in the course of administering your Luna HSM, can have effects, including destruction, on the roles, the spaces, and the contents of your HSM and its application partition(s). It is important to be aware of such consequences before taking action.

Overt Security Actions

Some actions in the administration of the HSM, or of an application partition, are explicitly intended to adjust specific security aspects of the HSM or partition. Examples are:

> Changing a password

> Modifying a policy to make a password or other attribute more stringent than the original setting

Those are discussed in their own sections.

Actions with Security- and Content-Affecting Outcomes

Other administrative events have security repercussions as included effects of the primary action, which could have other intent. Some examples are:

- HSM factory reset
- HSM zeroization
- Change of a destructive policy
- HSM initialization
- HSM firmware rollback
- Application partition initialization

This table lists some major administrative actions that can be performed on the HSM, and compares relevant security-related effects. Use the information in this table to help decide if your contemplated action is appropriate in current circumstances, or if additional preparation (such as backup of partition content, collection of audit data) would be prudent before continuing.

Factory Reset HSM

Domain	Destroyed
HSM SO Role	Destroyed
Partition SO Role	Destroyed
Auditor Role	Destroyed
Partition Roles	Destroyed
HSM or Partition/Contents	HSM/Destroyed
HSM Policies	Reset
RPV	Destroyed
Messaging	You are about to factory reset the HSM. All contents of the HSM will be destroyed. HSM policies will be reset and the remote PED vector will be erased.

Zeroize HSM

Domain	Destroyed
HSM SO Role	Destroyed
Partition SO Role	Destroyed

Auditor Role	Unchanged
Partition Roles	Destroyed
HSM or Partition/Contents	HSM/Destroyed
HSM Policies	Unchanged
RPV	Unchanged
Messaging	You are about to zeroize the HSM. All contents of the HSM will be destroyed. HSM policies, remote PED vector and Auditor left unchanged.

Change Destructive HSM Policy

Domain	Unchanged
HSM SO Role	Unchanged
Partition SO Role	Destroyed
Auditor Role	Unchanged
Partition Roles	Destroyed
HSM or Partition/Contents	HSM/Destroyed
HSM Policies	Unchanged except for new policy
RPV	Unchanged
Messaging	You are about to change a destructive HSM policy. All partitions of the HSM will be destroyed.

HSM Initialize When Zeroized (hard init)

Domain	Destroyed
HSM SO Role	Destroyed
Partition SO Role	Destroyed
Auditor Role	Unchanged
Partition Roles	Destroyed
HSM or Partition/Contents	HSM/Destroyed

HSM Policies	Unchanged
RPV	Unchanged
Messaging	You are about to initialize the HSM. All contents of the HSM will be destroyed.

HSM Initialize From Non-Zeroized State (soft init)

Domain	Unchanged
HSM SO Role	Unchanged
Partition SO Role	Destroyed
Auditor Role	Unchanged
Partition Roles	Destroyed
HSM or Partition/Contents	HSM/Destroyed
HSM Policies	Unchanged
RPV	Unchanged
Messaging	You are about to initialize the HSM that is already initialized. All partitions of the HSM will be destroyed. You are required to provide the current SO password.

HSM Firmware Rollback

Domain	Destroyed
HSM SO Role	Destroyed
Partition SO Role	Destroyed
Auditor Role	Destroyed
Partition Roles	Destroyed
HSM or Partition/Contents	HSM/Destroyed
HSM Policies	Unchanged
RPV	Unchanged

Messaging	WARNING: This operation will rollback your HSM to the previous firmware version !!! (1) This is a destructive operation. (2) You will lose all your partitions. (3) You may lose some capabilities. (4) You must re-initialize the HSM. (5) If the PED use is remote, you must re-connect it.

Partition Initialize When Zeroized (hard init)

Domain	Unchanged
HSM SO Role	Unchanged
Partition SO Role	Destroyed
Auditor Role	Unchanged
Partition Roles	Destroyed
HSM or Partition/Contents	Partition/Destroyed
HSM Policies	Unchanged
RPV	Unchanged
Messaging	You are about to initialize the partition. All contents of the partition will be destroyed.

Partition Initialize From Non-Zeroized State (soft init)

Domain	Unchanged
HSM SO Role	Unchanged
Partition SO Role	Destroyed
Auditor Role	Unchanged
Partition Roles	Destroyed
HSM or Partition/Contents	Partition/Destroyed
HSM Policies	Unchanged

RPV	Unchanged
Messaging	You are about to initialize the partition that is already initialized. All contents of the partition will be destroyed. You are required to provide the current Partition SO password.

Elsewhere

Certain other actions can sometimes cause collateral changes to the HSM, like firmware update. They usually do not affect contents, unless a partition is full and the action changes the size of partitions or changes the amount of space-per-partition that is taken by overhead/infrastructure. These are discussed elsewhere.

CHAPTER 12: Monitoring the HSM

Thales provides different methods of monitoring activity on the HSM. This chapter contains the following sections:

> "HSM Status Values" below
> "System Operational and Error Messages" on the next page
> "SNMP Monitoring" on page 241
> "Performance Monitoring" on page 250
> "Partition Utilization Metrics" on page 250
> "Cryptographic Module and Token Return Codes" on page 252
> "Library Codes" on page 271
> "Vendor-Defined Return Codes" on page 275
> "HSM Alarm Codes" on page 281

HSM Status Values

Each HSM administrative slot shown in a LunaCM slot listing includes an HSM status. Here are the possible values and what they mean, and what is required to recover from each one.

Indicated Status of HSM	Meaning	Recovery
OK	The HSM is in a good state, working properly.	n/a
Zeroized	The HSM is in zeroized state. All objects and roles are unusable.	HSM initialization is required before the HSM can be used again. "Hard init" - HSM SO and domain are gone, no authentication required. (see Note1)
Decommissioned	The HSM has been decommissioned.	HSM initialization is required before the HSM can be used again. "Hard init" - HSM SO and domain are gone, no authentication required. (see Note1)
Transport Mode	The HSM is in Secure Transport Mode.	STM must be disabled before the HSM can be used.

Indicated Status of HSM	Meaning	Recovery
Transport Mode, zeroized	The HSM is in Secure Transport Mode, and is also zeroized.	STM must be disabled, and then HSM initialization is required before the HSM can be used.
Transport Mode, Decommissioned	The HSM is in Secure Transport Mode, and has been decommissioned.	STM must be disabled, and then HSM initialization is required before the HSM can be used.
Hardware Tamper	The HSM has been tampered. (MTK is destroyed and must be rebuilt from recovery vectors.)	Reboot the host or restart the HSM. The event is logged
Hardware Tamper, Zeroized	The HSM has been tampered. (MTK is destroyed and must be rebuilt from recovery vectors.) The HSM is also in zeroized state. All objects and roles are unusable.	Reboot the host or restart the HSM. The event is logged. HSM initialization is required before the HSM can be used again. HSM SO and domain are gone, no authentication required. (see Note1)
HSM Tamper, Decommissioned	The HSM has been tampered. (MTK is destroyed and must be rebuilt from recovery vectors.) The HSM has also been decommissioned.	Reboot the host or restart the HSM. The event is logged. HSM initialization is required before the HSM can be used again. HSM SO and domain are gone, no authentication required. (see Note1)

NOTE1: A condition, not reported above, preserves the HSM SO and the associated Domain, while SO objects and all application partitions and contents are destroyed. In this case, HSM SO login is required to perform a "soft init". See "Initializing the HSM" on page 202 for more information.

For a comparison of various destruction or denial actions on the HSM, see "Comparison of Destruction/Denial Actions" on page 315.

System Operational and Error Messages

Extra slots that say "token not present"

This happens for two reasons:

> PKCS#11 originated in a world of software cryptography, which only later acknowledged the existence of Hardware Security Modules, so initially it did not have the concept of physically removable crypto slots. PKCS#11 requires a static list of slots when an application starts. The cryptographic "token" can be inserted into, or removed from a slot dynamically (by a user), for the duration of the application.

> When the token is inserted, the running application must be able to detect that token. When the token is removed, the running application gets "token not present". Because we allow for the possibility of backup, we routinely declare 'place-holder' slots that might later be filled by a physical Luna USB HSM 7 or a Luna Backup HSM.

In the Chrystoki.conf file (or the Windows crystoki.ini file), for Luna USB HSM 7, you can remove the empty slots by modifying the CardReader entry, like this:

```
CardReader = {
 LunaG5Slots=0;
}
```

For Luna Network HSM 7, which has its configuration file internal to the appliance, and not directly accessible for modification, you cannot change the default cryptographic slot allotments.

Error: 'hsm update firmware' failed. (10A0B : LUNA_RET_OPERATION_RESTRICTED) when attempting to perform hsm update firmware

You must ensure that STM is disabled before you run the firmware update.

Also, as with any update, you should backup any important HSM contents before proceeding.

LUNA_RET_OPERATION_RESTRICTED when attempting to perform a restore operation

Did you perform the backup before Functionality Modules (FMs) were enabled on the HSM? Enabling FMs allows injection of software into the HSM, which makes it inherently less secure than an HSM that has never had FMs enabled. The cloning operation (used to perform backup and restore) recognizes the state of the source and target devices, and can refuse to transfer objects from a more secure device to a less-secure device.

In general, to ensure that you will be able to backup and restore (with partition archive commands) or clone directly with clone commands or include partitions in HA groups (which also uses cloning), ensure that partition policies are set the same on the involved partitions.

KR_ECC_POINT_INVALID Error when decrypting a file encrypted from BSAFE through ECIES using ECC key with any of the curves from the x9_t2 section

As indicated on the BSAFE web site, they support only the NIST-approved curves (prime, Binary, and Koblitz). That includes most/all the curves from test items 0 through 37 in CK Demo: the "secp", "X9_62_prime", and "sect" curves.

The X9.62 curves that are failing in this task are X9.62 binary/char2 curves which do not appear to be supported by BSAFE. So, you appear to be encountering a BSAFE limitation and not a Luna HSM problem.

Slow/interrupted response from the HSM, and the "hsm show" command shows LUNA_RET_SM_SESSION_REALLOC_ERROR

```
   Appliance Details:
   ==================
   Software Version:                7.0.0
Error:  'hsm show' failed. (310102 : LUNA_RET_SM_SESSION_REALLOC_ERROR)

Command Result : 65535 (Luna Shell execution)
```

The error LUNA_RET_SM_SESSION_REALLOC_ERROR means the HSM cannot expand the session table.

The HSM maintains a table for all of the open sessions. For performance reasons, the table is quite small initially. As sessions are opened (and not closed) the table fills up. When the table gets full, the HSM tries to expand the table. If there is not enough available RAM to grow the table, this error is returned.

RAM can be used up by an application that creates and does not delete a large number of session objects, as well as by an application that opens and fails to close a large number of sessions.

The obvious solution is proper housekeeping. Your applications must clean up after themselves, by closing sessions that are no longer in use - this deletes session objects associated with those sessions. If your application practice is to have long-lived sessions, and to open many objects in a given session, then your application should explicitly delete those session objects as soon as each one is no longer necessary.

By far, we see more of the former problem - abandoned sessions - and very often in conjunction with Java-based applications. Proper garbage collection includes deleting session objects when they are no longer useful, or simply closing sessions as soon as they are not required. Formally closing a session (or stopping/restarting the HSM) deletes all session objects within each affected session. These actions keep the session table small, so it uses the least possible HSM volatile memory.

Low Battery Message

The HSM card used in the Luna Network HSM 7 and Luna PCIe HSM 7 products, is equipped with a non-replaceable battery that is expected to last the life of the product. If you notice a log message or other warning about ‘battery low’, or similar, contact Technical Support.

CHAPTER 12: SNMP Monitoring

This chapter describes Simple Network Management Protocol (SNMP v3) support for remote monitoring of conditions on a local HSM that might require administrative attention. It contains the following sections:

> "Installing the Luna SNMP Subagent" below

> "The SafeNet Chrysalis-UTSP MIB" on page 243

> "The Luna HSM MIB" on page 244

> "Frequently Asked Questions" on page 249

MIB

Thales Group provides the following MIBs (management information base) in the Luna HSM Client installation package:

MIB Name	Description
CHRYSALIS-UTSP-MIB.txt	Defines SNMP access to information about the Luna HSM operations.
SAFENET-HSM-MIB.txt	Defines SNMP access to information about the Luna HSM.
SAFENET-GLOBAL-MIB.txt	This MIB defines the structure of the object identifier tree for the other MIBs. Must be found in your system path so that symbols can be resolved.

Copy all MIBs in **<Luna_HSM_Client_install_dir>/snmp** to the MIB directory on your system. Only the MIBs necessary for Luna PCIe HSM 7 and Luna USB HSM 7 are included in a client installation.

> **NOTE** Your SNMP application also requires the following standard SNMP MIBs:
> > **SNMPv2-SMI.txt** -- defined in RFC 2578, Section 2
> > **SNMPv2-TC.txt** -- defined in RFC 2579, Section 2

Installing the Luna SNMP Subagent

We find that most customers choosing to use SNMP already have an SNMP infrastructure in place. Therefore, we provide a subagent that you can install on your managed workstations, and which can point to your agent via the socket created by the agent. This applies to Luna USB HSM 7 and Luna PCIe HSM 7 - for Luna Network HSM 7, the subagent is already on the appliance.

The SNMP subagent (luna-snmp) is an AgentX SNMP module that extends an existing SNMP agent with support for Luna HSM monitoring. It is an optional component of the Luna HSM Client installation. The subagent has been tested against net-snmp, but should work with any SNMP agent that supports the AgentX protocol.

To install the SNMP subagent

After selecting one or more products from the main Luna HSM Client installation menu, you are presented with a list of optional components, including the SNMP subagent. It is not selected by default, but can be installed with any product except the Luna Network HSM 7 client installed in isolation.

1. In the installation media, go to the appropriate folder for your operating system.
2. Run the installer (install.sh for Linux and UNIX, LunaHSMClient.exe for Windows).
3. Choose the Luna products that you wish to install, and include SNMP among your selections. The subagent is installed for any Luna product except Luna Network HSM 7 in isolation.
4. Proceed to Post-installation configuration.

Post-installation configuration

After the Luna HSM Client is installed, complete the following steps to configure the SNMP subagent:

1. Copy the SafeNet MIBs from **<install dir>/snmp** to the main SNMP agent's MIB directory. Or copy to another computer (your SNMP computer) if you are not running SNMP from the same computer where Luna HSM Client software is installed.
2. If running on Windows, configure the subagent via the file <install dir>/snmp/luna-snmp.conf to point to the AgentX port where the main SNMP agent is listening. The file must then be copied to the same directory as snmpd.conf. (This assumes net-snmp is installed; the setup might differ if you have another agent.)

   If running on a UNIX-based platform, the subagent should work without extra configuration assuming that the primary SNMP agent is listening on the default local socket (/var/agentx/master). You still have the option of editing and using luna-snmp.conf.
3. After configuration is complete, start the agent. Then start the subagent via the service tool applicable to your platform (for example, **service luna-snmp start** on Linux, or start Luna SNMP Subagent Service from the services in Windows).

Normally the agent is started first. However, the subagent periodically attempts to connect to the agent until it is successful. The defaults controlling this behavior are listed below. They can be overridden by changing the appropriate entries in **luna-snmp.conf**.

Troubleshooting

If you encounter the following warning:

Warning: Failed to connect to the agentx master agent ([NIL]):

you must enable AgentX support by adding **master agentx** to your SNMPD configuration file. Refer to the man page for **snmpd.conf** for more information.

Configuration Options In the luna-snmp.conf File

Option	Description	Default
agentXSocket [<transport-specifier>:]<transport-address>[,...]	Defines the address to which the subagent should connect. The default on UNIX-based systems is the Unix Domain socket "/var/agentx/master". Another common alternative is tcp:localhost:705. See the section LISTENING ADDRESSES in the snmpd man page for more information about the format of addresses (http://www.net-snmp.org/docs/man/snmpd.html).	The default, for Linux, is "/var/agentx/master". In the file, you can choose to un-comment "tcp:localhost:705" which is most commonly used with Windows.
agentXPingInterval <NUM>	Makes the subagent try to reconnect every <NUM> seconds to the master if it ever becomes (or starts) disconnected.	15
agentXTimeout <NUM>	Defines the timeout period (NUM seconds) for an AgentX request.	1
agentXRetries <NUM>	Defines the number of retries for an AgentX request.	5

The SafeNet Chrysalis-UTSP MIB

> **NOTE** The Chrysalis MIB is the SafeNet MIB for all Luna HSM products - the Chrysalis name is retained for historical continuity.

To illustrate accessing data, the command "snmpwalk -v 3 -u admin -l authPriv -a SHA1 -A 12345678 -x AES -X 87654321 myLuna19 private" produced this output:

> CHRYSALIS-UTSP-MIB::hsmOperationRequests.0 = Counter64: 3858380

> CHRYSALIS-UTSP-MIB::hsmOperationErrors.0 = Counter64: 385838

> CHRYSALIS-UTSP-MIB::hsmCriticalEvents.0 = Counter64: 0

> CHRYSALIS-UTSP-MIB::hsmNonCriticalEvents.0 = Counter64: 5

> CHRYSALIS-UTSP-MIB::ntlsOperStatus.0 = INTEGER: up(1)

> CHRYSALIS-UTSP-MIB::ntlsConnectedClients.0 = Gauge32: 0

> CHRYSALIS-UTSP-MIB::ntlsLinks.0 = Gauge32: 0

> CHRYSALIS-UTSP-MIB::ntlsSuccessfulClientConnections.0 = Counter64: 16571615927115620

> CHRYSALIS-UTSP-MIB::ntlsFailedClientConnections.0 = Counter64: 1657161592711562

The various counts are recorded since the last restart.

Item	Description
hsmOperationRequests	The total number of HSM operations that have been requested.
hsmOperationErrors	The total number of HSM operations that have been requested, that have resulted in errors.
hsmCriticalEvents	The total number of critical HSM events that have been detected (Tamper, Decommission, Zeroization, SO creation, or Audit role creation). **NOTE** Not implemented in this release. hsmCriticalEvents always reports 0.
hsmNonCriticalEvents	The total number of NON-critical HSM events that have been detected (any that are not among the critical list, above). **NOTE** Not implemented in this release. hsmNonCriticalEvents always reports 0.
ntlsOperStatus	The current operational status of the NTL service, where the options are: 1 = up, 2 = not running, and 3 = status cannot be determined.
ntlsConnectedClients	The current number of connected clients using NTLS.
ntlsLinks	The current number of links in NTLS - can be multiple per client, depending on processes.
ntlsSuccessfulClientConnections	The total number of successful client connections.
ntlsFailedClientConnections	The total number of UNsuccessful client connections.

The Luna HSM MIB

The SAFENET-HSM-MIB defines HSM status information and HSM Partition information that can be viewed via SNMP.

To access tables, use a command like:

```
snmptable  -a SHA  -A snmppass  -u snmpuser -x AES -X snmppass -l authPriv -v 3 192.20.11.59
SAFENET-HSM-MIB::hsmTable
```

The information is defined in tables, as detailed in the following sections.

SNMP Table Updates

The SNMP tables are updated and cached every 60 seconds. Any changes made on the HSM may therefore take up to 60 seconds to be included in the tables. When a query is received to view the tables, the most recent cached version is displayed. If a change you were expecting is not displayed, wait 60 seconds and try again.

> **NOTE** Some values may not get updated automatically, such as the HSM firmware version (hsmFirmwareVersion) following a firmware upgrade. To force an update, restart the SNMP agent.

hsmTable

This table provides a list of all the HSM information on the managed element.

Item	Type	Description	Values
hsmSerialNumber	DisplayString	Serial number of the HSM - used as an index into the tables.	From factory
hsmFirmwareVersion	DisplayString	Version of firmware executing on the HSM.	As found
hsmLabel	DisplayString	Label associated with the HSM.	Provided by SO at init time
hsmModel	DisplayString	Model identifier for the HSM.	From factory
hsmAuthenticationMethod	INTEGER	Authentication mode of the HSM.	unknown(1), -- not known password(2), -- requires passwords pedKeys(3) -- requires PED
hsmRpvInitialized	INTEGER	Remote ped vector initialized flag of the HSM.	notSupported (1), -- rpv not supported uninitialized(2), -- rpv not initialized initialized(3) -- rpv initialized
hsmFipsMode	TruthValue	FIPS 140-2 operation mode enabled flag of the HSM.	Factory set
hsmPerformance	INTEGER	Performance level of the HSM.	
hsmStorageTotalBytes	Unsigned32	Total storage capacity in bytes of the HSM	Factory set
hsmStorageAllocatedBytes	Unsigned32	Number of allocated bytes on the HSM	Calculated

Item	Type	Description	Values
hsmStorageAvailableBytes	Unsigned32	Number of available bytes on the HSM	Calculated
hsmMaximumPartitions	Unsigned32	Maximum number of partitions allowed on the HSM	2, 5, 10, 15, or 20, per license
hsmPartitionsCreated	Unsigned32	Number of partitions created on the HSM	As found
hsmPartitionsFree	Unsigned32	Number of partitions that can still be created on the HSM	Calculated
hsmBackupProtocol	INTEGER	Backup protocol used on the HSM	unknown(1), none(2), cloning(3), keyExport(4)
hsmAdminLoginAttempts	Counter32	Number of failed Administrator login attempts left before HSM zeroized	As found, calculated
hsmAuditRoleInitialized	INTEGER	Audit role is initialized flag	notSupported (0), yes(1), no(2)
hsmManuallyZeroized	TruthValue	Was HSM manually zeroized flag	As found
hsmUpTime	Counter64	Up time in seconds since last HSM reset	Counted
hsmBusyTime	Counter64	Busy time in seconds since the last HSM reset	Calculated
hsmCommandCount	Counter64	HSM commands processed since last HSM reset	Counted

The hsmPartitionTable

This table provides a list of all the partition information on the managed element.

Item	Type	Description	Values
hsmPartitionSerialNumber	DisplayString	Serial number for the partition	Generated
hsmPartitionLabel	DisplayString	Label assigned to the partition	Provided at partition creation
hsmPartitionActivated	TruthValue	Partition activation flag	Set by policy

Item	Type	Description	Values
hsmPartitionStorageTotalBytes	Unsigned32	Total storage capacity in bytes of the partition	Set or calculated at partition creation or re-size
hsmPartitionStorageAllocatedBytes	Unsigned32	Number of allocated (in use) bytes on the partition	Calculated
hsmPartitionStorageAvailableBytes	Unsigned32	Number of avalailable (unused) bytes on the partition	Calculated
hsmPartitionObjectCount	Unsigned32	Number of objects in the partition	Counted

hsmLicenseTable

This table provides a list of all the license information on the managed element. More than one HSM might be connected to a Host, so they are accessed with two indices; the first index identifies the HSM for which the license entry corresponds (hsmSerialNumber), the second is the index for the corresponding license (hsmLicenseID).

Item	Type	Description	Values
hsmLicenseID	DisplayString	License identifier	Set at factory or at capability update
hsmLicenseDescription	DisplayString	License description	Set at factory or at capability update

hsmPolicyTable

This table provides a list of all the HSM policy information on the managed element.

Item	Type	Description	Values
hsmPolicyType	INTEGER	Type of policy	capability(1), policy(2)
hsmPolicyID	Unsigned32	Policy identifier	Numeric value identifies policy and is used as a index into the policy table
hsmPolicyDescription	DisplayString	Description of the policy	Brief text description of what the policy does
hsmPolicyValue	DisplayString	Current value of the policy	Brief text description to show current state/value of policy

hsmPartitionPolicyTable

This table provides a list of all the partition policy information on the managed element.

Item	Type	Description	Values
hsmPartitionPolicyType	INTEGER	Capability or policy	capability(1), policy(2)
hsmPartitionPolicyID	Unsigned32	Policy identifier	Numeric value identifies policy and is used as a index into the policy table
hsmPartitionPolicyDescription	DisplayString	Description of the policy	Brief text description of what the policy does
hsmPartitionPolicyValue	DisplayString	Current value of the policy	Brief text description to show current state/value of policy

hsmClientRegistrationTable

This table provides a list of registered clients.

Item	Type	Description	Values
hsmClientName	DisplayString	Name of the client	Name provided on client cert
hsmClientAddress	DisplayString	Address of the client	IP address of the client

hsmClientPartitionAssignmentTable

This table provides a list of assigned partitions for a given client.

Item	Type	Description	Values
hsmClientHsmSerialNumber	DisplayString	Index into the HSM table	--
hsmClientPartitionSerialNumber DisplayString	DisplayString	Index into the Partition Table	--

SNMP output compared to Luna tools output

For comparison, the following shows LunaCM or LunaSH command outputs that provide HSM information equivalent to the SNMP information depicted in the tables above (from the HSM MIB).

HSM Information

At the HSM level the information in the outputs of **hsm show** and **hsm showpolicies** and **hsm displaylicenses** includes the following:

> SW Version

- FW Version
- HSM label
- Serial #
- HW Model
- Authentication Method
- RPV state
- FIPS mode
- HSM total storage space (bytes)
- HSM used storage space (bytes)
- HSM free storage space (bytes)
- Performance level
- Max # of partitions
- # of partitions created
- # of free partitions
- HSM policies and their settings

Partition Information
At the application partition level, the information in the outputs of **partition show** and **partition showpolicies** includes the following:

- Partition Name
- Partition Serial #
- Activation State
- AutoActivation State
- Partition total storage space (bytes)
- Partition used storage space (bytes)
- Partition free storage space (bytes)
- Partition Object Count
- Partition policies and their settings

Frequently Asked Questions

This section provides additional information by answering questions that are frequently asked by our customers.

We want to use SNMP to remotely monitor and manage our installation – why do you not support such standard SNMP traps as CPU and Memory exhaustion?
Those sorts of traps were specifically excluded because they can be used to establish a covert channel (an illicit signaling channel that can be used to communicate from a high assurance “area” to a lower assurance one in an effort to circumvent the security policy). Resource exhaustion events/alerts are the oldest known form of covert

channel signaling. Exercise care with any HSM product that does allow such traps - what other basic security holes might be present?

Performance Monitoring

An HSM administrator might find it helpful to know how busy the HSM is and at what percentage of its capacity it has been running.

The HSM Information Monitor is a use counter that provides an indication of momentary and cumulative resource usage on the HSM, in the form of a percentage. The HSM firmware tracks the overall time elapsed since the last reset (Up-Time), and the overall time during which the processor was not performing useful work (Idle-Time).

On request, the HSM calculates "Busy-time" over an interval, by subtracting Idle-time for that interval from Up-time for the interval. Then, the load on the processor is calculated as the Busy-time divided by the Up-time, and expressed as a percentage.

You can use the available commands for a single, one-off query, which actually takes an initial reading and then another, five seconds later (the default setting), in order to calculate and show the one-time difference.

You can specify a sampling interval (five seconds is the shortest) and a number of repetitions for an extended view of processor activity/resource usage. The resulting records, showing the time of each measurement, the percentage value at that time, and the difference from the previous measurement, can be output to a file that you import into other tools to analyze and graph the trends.

By watching trends and correlating with what your application is doing, you can:

> Determine the kinds of loads you are placing on the HSM.

> Seek efficiencies in how your applications are coded and configured.

> Plan for expansion or upgrades of your existing HSM infrastructure.

> Plan for upgrades of electrical capacity and HVAC capacity.

Notes about Monitor/Counter Behavior

When performing certain operations the HSM reaches its maximum performance capability before the counter reaches 100%. This occurs because the counter measures the load on the HSM's CPU and the CPU is able to saturate the asymmetric engines and still have capacity to perform other actions.

Also, symmetric cryptographic operations cause the counter to quickly rise to 90% even though there is significant remaining capacity. This behavior occurs because, as the HSM receives more concurrent symmetric commands, its CPU is able to handle them more efficiently (by performing them in bulk) – thus achieving more throughput from the same number of CPU cycles.

See lunacm:> **hsm monitor**.

Partition Utilization Metrics

In order to ensure the quality of service (QoS) that you provide to applications that make use of HSM partitions, it is first necessary to know how the users and applications are making use of the HSM resources - that is, the distribution of demand.

For an HSM with a single application partition, it can be helpful to know what type of load is being imposed on the HSM and the enumeration and categorization of operations that are being performed. Application developers might have a good idea of the expected ratio of operations, but the operations team managing the application servers would like to know the real-world utilization, for their planning and management purposes.

> **NOTE** Utilization metrics are based on *utilization counters* that track operations by category. This is not to be confused with *usage counters*, that track and limit the number of times a key or certificate is allowed to be used.
>
> This feature requires minimum Luna HSM Firmware 7.3.0 and Luna HSM Client 7.3.0.

Rules of acquisition

Utilization Metrics count these operations within category "bins" per partition:

- Sign
- Verify
- Encrypt
- Decrypt
- Key generate
- Key derive

Operations not in that list do not increment any counter. That is, an operation request to the HSM increments counters in 0 or more bins. The list might expand in future releases. Each bin has a single counter that counts how many requests have been received from the host, since the last counter-reset order or power cycle. Counters for a partition can be read and reset as a single operation, or as two separate operations.

The utilization counters count *requests* to the HSM, because, while successful requests are expected and are counted, unsuccessful requests also consume resources and therefore need to be counted as well. Any request that fails on the host - meaning it does not reach the HSM - is not counted, because it did not use any HSM resources.

Utilization counters are volatile, and therefore are lost in the event of a power failure. If they are valued, they should be polled regularly and the results kept in non-volatile storage on the host.

Availability of Partition Utilization Metrics

Utilization metrics are supported by Luna HSM Firmware 7.3.0 and newer, using HSM-level policy **49: Allow Partition Utilization Metrics**. That policy is off (value 0) by default, as it is not required in all use-cases, and is most useful where multiple applications use the HSM.

> **NOTE** The Utilization Metrics feature allows the HSM SO to know which operations are being performed on the HSM. This information is normally available only to the Auditor when audit logging is turned on. However, while the SO can see a record of cryptographic operations, there is no visibility as to which keys are being used.

Setting the policy on (value 1) enables utilization metrics for all partitions including the Admin partition. Changing the policy is not destructive in either direction (off-to-on or on-to-off).

The **hsm showUtilization** command allows you to view the current utilization counter values for all partitions, and overall counts for the entire HSM, without resetting the counters.

The **hsm resetUtilization** command allows you to reset to zero the current utilization counter values for all partitions.

To access the Partition Utilization Metrics feature

1. Ensure that your HSM is at Luna HSM Firmware 7.3.0 or newer (if needed, upgrade to a suitable version; see "Updating the Luna PCIe HSM 7 Firmware" on page 294).
2. Log in as HSM SO (see "Logging In as HSM Security Officer" on page 207).

   lunacm:> **role login -name so**
3. Enable HSM policy **49: Allow Partition Utilization Metrics**.

   lunacm:> **hsm changehsmpolicy -policy 49 -value 1**

To view or save Partition Utilization Metrics without resetting

lunacm:> **hsm showUtilization -serial** <partition_SN)

To reset the Partition Utilization Metrics counters to zero

Metrics are reset whenever power is lost to the HSM or the HSM is reset, or the HSM is initialized. These events do not save the metrics.

To display the metrics since the last reset (making them available to be captured manually or by script) and then immediately reset the metrics:

lunacm:> **hsm resetUtilization**

Cryptographic Module and Token Return Codes

The following table summarizes HSM error codes:

HSM Error	Hex Code	PKCS#11 or SFNT Defined CKR Error
LUNA_RET_OK	0x00000000	CKR_OK
LUNA_RET_CANCEL	0x00010000	CKR_CANCEL
LUNA_RET_FLAGS_INVALID	0x00040000	CKR_FLAGS_INVALID, removed from v2.0
LUNA_RET_TOKEN_NOT_PRESENT	0x00E00000	CKR_TOKEN_NOT_PRESENT
LUNA_RET_FORMER_INVALID_ENTRY_TYPE	0x00300130	CKR_DEVICE_ERROR
LUNA_RET_SP_TX_ERROR	0x00300131	CKR_DEVICE_ERROR

HSM Error	Hex Code	PKCS#11 or SFNT Defined CKR Error
LUNA_RET_SP_RX_ERROR	0x00300132	CKR_DEVICE_ERROR
LUNA_RET_PED_ID_INVALID	0x00300140	CKR_DEVICE_ERROR
LUNA_RET_PED_UNSUPPORTED_PROTOCOL	0x00300141	CKR_DEVICE_ERROR
LUNA_RET_PED_UNPLUGGED	0x00300142	CKR_PED_UNPLUGGED
LUNA_RET_PED_ERROR	0x00300144	CKR_DEVICE_ERROR
LUNA_RET_PED_UNSUPPORTED_CRYPTO_PROTOCOL	0x00300145	CKR_DEVICE_ERROR
LUNA_RET_PED_DEK_INVALID	0x00300146	CKR_DEVICE_ERROR
LUNA_RET_PED_CLIENT_NOT_RUNNING	0x00300147	CKR_PED_CLIENT_NOT_RUNNING
LUNA_RET_CL_ALIGNMENT_ERROR	0x00300200	CKR_DEVICE_ERROR
LUNA_RET_CL_QUEUE_LOCATION_ERROR	0x00300201	CKR_DEVICE_ERROR
LUNA_RET_CL_QUEUE_OVERLAP_ERROR	0x00300202	CKR_DEVICE_ERROR
LUNA_RET_CL_TRANSMISSION_ERROR	0x00300203	CKR_DEVICE_ERROR
LUNA_RET_CL_NO_TRANSMISSION	0x00300204	CKR_DEVICE_ERROR
LUNA_RET_CL_COMMAND_MALFORMED	0x00300205	CKR_DEVICE_ERROR
LUNA_RET_CL_MAILBOXES_NOT_AVAILABLE	0x00300206	CKR_DEVICE_ERROR
LUNA_RET_MM_NOT_ENOUGH_MEMORY	0x00310000	CKR_DEVICE_MEMORY †
LUNA_RET_MM_INVALID_HANDLE	0x00310001	CKR_DEVICE_MEMORY †
LUNA_RET_MM_USAGE_ALREADY_SET	0x00310002	CKR_DEVICE_MEMORY †
LUNA_RET_MM_ACCESS_OUTSIDE_ALLOCATION_RANGE	0x00310003	CKR_DEVICE_MEMORY †
LUNA_RET_MM_INVALID_USAGE	0x00310004	CKR_DEVICE_MEMORY †
LUNA_RET_MM_ITERATOR_PAST_END	0x00310005	CKR_DEVICE_MEMORY †

HSM Error	Hex Code	PKCS#11 or SFNT Defined CKR Error
LUNA_RET_MM_FATAL_ERROR	0x00310006	CKR_DEVICE_MEMORY †
LUNA_RET_MEMORY_ALLOCATION_FAILED	0x00310007	CKR_DEVICE_MEMORY †
LUNA_RET_TEMPLATE_INCOMPLETE	0x00D00000	CKR_TEMPLATE_INCOMPLETE
LUNA_RET_TEMPLATE_INCONSISTENT	0x00D10000	CKR_TEMPLATE_INCONSISTENT*
LUNA_RET_ATTRIBUTE_TYPE_INVALID	0x00120000	CKR_ATTRIBUTE_TYPE_INVALID
LUNA_RET_ATTRIBUTE_VALUE_INVALID	0x00130000	CKR_ATTRIBUTE_VALUE_INVALID
LUNA_RET_ATTRIBUTE_READ_ONLY	0x00100000	CKR_ATTRIBUTE_READ_ONLY
LUNA_RET_ATTRIBUTE_SENSITIVE	0x00110000	CKR_ATTRIBUTE_SENSITIVE
LUNA_RET_OBJECT_HANDLE_INVALID	0x00820000	CKR_OBJECT_HANDLE_INVALID
LUNA_RET_MAX_OBJECT_COUNT	0x00820001	CKR_MAX_OBJECT_COUNT_ EXCEEDED
LUNA_RET_ATTRIBUTE_NOT_FOUND	0x00120010	CKR_ATTRIBUTE_TYPE_INVALID
LUNA_RET_CAN_NOT_CREATE_SECRET_KEY	0x00D10011	CKR_TEMPLATE_INCONSISTENT
LUNA_RET_CAN_NOT_CREATE_PRIVATE_KEY	0x00D10012	CKR_TEMPLATE_INCONSISTENT
LUNA_RET_SECRET_KEY_MUST_BE_SENSITIVE	0x00130013	CKR_ATTRIBUTE_VALUE_INVALID
LUNA_RET_SECRET_KEY_MUST_HAVE_ SENSITIVE_ATTRIBUTE	0x00D00014	CKR_TEMPLATE_INCOMPLETE
LUNA_RET_PRIVATE_KEY_MUST_BE_ SENSITIVE	0x00130015	CKR_ATTRIBUTE_VALUE_INVALID
LUNA_RET_PRIVATE_KEY_MUST_HAVE_ SENSITIVE_ATTRIBUTE	0x00D00016	CKR_TEMPLATE_INCOMPLETE
LUNA_RET_SIGNING_KEY_MUST_BE_LOCAL	0x00680001	CKR_KEY_FUNCTION_NOT_ PERMITTED
LUNA_RET_MULTI_FUNCTION_KEYS_NOT_ ALLOWED	0x00D10018	CKR_TEMPLATE_INCONSISTENT

HSM Error	Hex Code	PKCS#11 or SFNT Defined CKR Error
LUNA_RET_CAN_NOT_CHANGE_KEY_ FUNCTION	0x00100019	CKR_ATTRIBUTE_READ_ONLY
LUNA_RET_KEY_SIZE_RANGE	0x00620000	CKR_KEY_SIZE_RANGE
LUNA_RET_KEY_TYPE_INCONSISTENT	0x00630000	CKR_KEY_TYPE_INCONSISTENT
LUNA_RET_KEY_INVALID_FOR_OPERATION	0x00630001	CKR_KEY_TYPE_INCONSISTENT
LUNA_RET_KEY_PARITY	0x00630002	CKR_KEY_TYPE_INCONSISTENT
LUNA_RET_KEY_UNEXTRACTABLE	0x006a0000	CKR_KEY_UNEXTRACTABLE
LUNA_RET_KEY_EXTRACTABLE	0x006a0001	KR_KEY_UNEXTRACTABLE
LUNA_RET_KEY_INDIGESTIBLE	0x00670000	CKR_KEY_INDIGESTIBLE
LUNA_RET_KEY_NOT_WRAPPABLE	0x00690000	CKR_KEY_NOT_WRAPPABLE
LUNA_RET_KEY_NOT_UNWRAPPABLE	0x00690001	CKR_KEY_NOT_WRAPPABLE
LUNA_RET_ARGUMENTS_BAD	0x00070000	CKR_ARGUMENTS_BAD
LUNA_RET_INVALID_ENTRY_TYPE	0x00070001	CKR_INVALID_ENTRY_TYPE
LUNA_RET_DATA_INVALID	0x00200000	CKR_DATA_INVALID
LUNA_RET_SM_DATA_INVALID	0x00200002	CKR_DATA_INVALID
LUNA_RET_NO_RNG_SEED	0x00200015	CKR_DATA_INVALID
LUNA_RET_FUNCTION_NOT_SUPPORTED	0x00540000	CKR_FUNCTION_NOT_ SUPPORTED
LUNA_RET_NO_OFFBOARD_STORAGE	0x00540001	CKR_FUNCTION_NOT_ SUPPORTED
LUNA_RET_CL_COMMAND_NON_BACKUP	0x00540002	CKR_FUNCTION_NOT_ SUPPORTED
LUNA_RET_BUFFER_TOO_SMALL	0x01500000	CKR_BUFFER_TOO_SMALL
LUNA_RET_DATA_LEN_RANGE	0x00210000	CKR_DATA_LEN_RANGE

HSM Error	Hex Code	PKCS#11 or SFNT Defined CKR Error
LUNA_RET_GENERAL_ERROR	0x00050000	CKR_GENERAL_ERROR
LUNA_RET_DEVICE_ERROR	0x00300000	CKR_DEVICE_ERROR
LUNA_RET_UNKNOWN_COMMAND	0x00300001	CKR_FUNCTION_NOT_ SUPPORTED
LUNA_RET_TOKEN_LOCKED_OUT	0x00300002	CKR_PIN_LOCKED
LUNA_RET_RNG_ERROR	0x00300003	CKR_DEVICE_ERROR
LUNA_RET_DES_SELF_TEST_FAILURE	0x00300004	CKR_DEVICE_ERROR
LUNA_RET_CAST_SELF_TEST_FAILURE	0x00300005	CKR_DEVICE_ERROR
LUNA_RET_CAST3_SELF_TEST_FAILURE	0x00300006	CKR_DEVICE_ERROR
LUNA_RET_CAST5_SELF_TEST_FAILURE	0x00300007	CKR_DEVICE_ERROR
LUNA_RET_MD2_SELF_TEST_FAILURE	0x00300008	CKR_DEVICE_ERROR
LUNA_RET_MD5_SELF_TEST_FAILURE	0x00300009	CKR_DEVICE_ERROR
LUNA_RET_SHA_SELF_TEST_FAILURE	0x0030000a	CKR_DEVICE_ERROR
LUNA_RET_RSA_SELF_TEST_FAILURE	0x0030000b	CKR_DEVICE_ERROR
LUNA_RET_RC2_SELF_TEST_FAILURE	0x0030000c	CKR_DEVICE_ERROR
LUNA_RET_RC4_SELF_TEST_FAILURE	0x0030000d	CKR_DEVICE_ERROR
LUNA_RET_RC5_SELF_TEST_FAILURE	0x0030000e	CKR_DEVICE_ERROR
LUNA_RET_SO_LOGIN_FAILURE_THRESHOLD	0x0030000f	CKR_SO_LOGIN_FAILURE_ THRESHOLD
LUNA_RET_RNG_SELF_TEST_FAILURE	0x00300010	CKR_DEVICE_ERROR
LUNA_RET_SM_UNKNOWN_COMMAND	0x00300011	CKR_DEVICE_ERROR
LUNA_RET_UM_TSN_MISSING	0x00300012	CKR_DEVICE_ERROR
LUNA_RET_SM_TSV_MISSING	0x00300013	CKR_DEVICE_ERROR

HSM Error	Hex Code	PKCS#11 or SFNT Defined CKR Error
LUNA_RET_SM_UNKNOWN_TOSM_STATE	0x00300014	CKR_DEVICE_ERROR
LUNA_RET_DSA_PARAM_GEN_FAILURE	0x00300015	CKR_DEVICE_ERROR
LUNA_RET_DSA_SELF_TEST_FAILURE	0x00300016	CKR_DEVICE_ERROR
LUNA_RET_SEED_SELF_TEST_FAILURE	0x00300017	CKR_DEVICE_ERROR
LUNA_RET_AES_SELF_TEST_FAILURE	0x00300018	CKR_DEVICE_ERROR
LUNA_RET_FUNCTION_NOT_SUPPORTED_BY_HARDWARE	0x00300019	CKR_DEVICE_ERROR
LUNA_RET_HAS160_SELF_TEST_FAILURE	0x0030001a	CKR_DEVICE_ERROR
LUNA_RET_KCDSA_PARAM_GEN_FAILURE	0x0030001b	CKR_DEVICE_ERROR
LUNA_RET_KCDSA_SELF_TEST_FAILURE	0x0030001c	CKR_DEVICE_ERROR
LUNA_RET_HSM_INTERNAL_BUFFER_TOO_SMALL	0x0030001d	CKR_DEVICE_ERROR
LUNA_RET_COUNTER_WRAPAROUND	0x0030001e	CKR_DEVICE_ERROR
LUNA_RET_TIMEOUT	0x0030001f	CKR_TIMEOUT
LUNA_RET_NOT_READY	0x00300020	CKR_DEVICE_ERROR
LUNA_RET_RETRY	0x00300021	CKR_DEVICE_ERROR
LUNA_RET_SHA1_RSA_SELF_TEST_FAILURE	0x00300022	CKR_DEVICE_ERROR
LUNA_RET_SELF_TEST_FAILURE	0x00300023	CKR_DEVICE_ERROR
LUNA_RET_INCOMPATIBLE	0x00300024	CKR_DEVICE_ERROR
LUNA_RET_RIPEMD160_SELF_TEST_FAILURE	0x00300034	CKR_DEVICE_ERROR
LUNA_RET_TOKEN_LOCKED_OUT_CL	0x00300100	CKR_DEVICE_ERROR
LUNA_RET_TOKEN_LOCKED_OUT_MM	0x00300101	CKR_DEVICE_ERROR
LUNA_RET_TOKEN_LOCKED_OUT_UM	0x00300102	CKR_DEVICE_ERROR

HSM Error	Hex Code	PKCS#11 or SFNT Defined CKR Error
LUNA_RET_TOKEN_LOCKED_OUT_SM	0x00300103	CKR_DEVICE_ERROR
LUNA_RET_TOKEN_LOCKED_OUT_RN	0x00300104	CKR_DEVICE_ERROR
LUNA_RET_TOKEN_LOCKED_OUT_CA	0x00300105	CKR_DEVICE_ERROR
LUNA_RET_TOKEN_LOCKED_OUT_PM	0x00300106	CKR_DEVICE_ERROR
LUNA_RET_TOKEN_LOCKED_OUT_OH	0x00300107	CKR_DEVICE_ERROR
LUNA_RET_TOKEN_LOCKED_OUT_CCM	0x00300108	CKR_DEVICE_ERROR
LUNA_RET_TOKEN_LOCKED_OUT_SHA_DIGEST	0x00300109	CKR_DEVICE_ERROR
LUNA_RET_SM_ACCESS_REALLOC_ERROR	0x00310101	CKR_DEVICE_ERROR
LUNA_RET_SM_SESSION_REALLOC_ERROR	0x00310102	CKR_DEVICE_ERROR
LUNA_RET_SM_MEMORY_ALLOCATION_ERROR	0x00310103	CKR_DEVICE_ERROR
LUNA_RET_ENCRYPTED_DATA_INVALID	0x00400000	CKR_ENCRYPTED_DATA_INVALID
LUNA_RET_ENCRYPTED_DATA_LEN_RANGE	0x00410000	CKR_ENCRYPTED_DATA_LEN_ RANGE
LUNA_RET_FUNCTION_CANCELED	0x00500000	CKR_FUNCTION_CANCELED
LUNA_RET_KEY_HANDLE_INVALID	0x00600000	CKR_KEY_HANDLE_INVALID
LUNA_RET_MECHANISM_INVALID	0x00700000	CKR_MECHANISM_INVALID
LUNA_RET_MECHANISM_PARAM_INVALID	0x00710000	CKR_MECHANISM_PARAM_ INVALID
LUNA_RET_OPERATION_ACTIVE	0x00900000	CKR_OPERATION_ACTIVE
LUNA_RET_OPERATION_NOT_INITIALIZED	0x00910000	CKR_OPERATION_NOT_ INITIALIZED
LUNA_RET_UM_PIN_INCORRECT	0x00a00000	CKR_PIN_INCORRECT
LUNA_RET_UM_PIN_INCORRECT_CONTAINER_ ZEROIZED	0x00a00001	CKR_PIN_INCORRECT

HSM Error	Hex Code	PKCS#11 or SFNT Defined CKR Error
LUNA_RET_UM_PIN_INCORRECT_CONTAINER_LOCKED	0x00a00002	CKR_PIN_INCORRECT
LUNA_RET_UM_PIN_LEN_RANGE	0x00a20000	CKR_PIN_LEN_RANGE
LUNA_RET_SM_PIN_EXPIRED	0x00a30000	CKR_PIN_EXPIRED
LUNA_RET_SM_EXCLUSIVE_SESSION_EXISTS	0x00b20000	CKR_SESSION_EXCLUSIVE_EXISTS
LUNA_RET_SM_SESSION_HANDLE_INVALID	0x00b30000	CKR_SESSION_HANDLE_INVALID
LUNA_RET_SIGNATURE_INVALID	0x00c00000	CKR_SIGNATURE_INVALID
LUNA_RET_SIGNATURE_LEN_RANGE	0x00c10000	CKR_SIGNATURE_LEN_RANGE
LUNA_RET_UNWRAPPING_KEY_HANDLE_INVALID	0x00f00000	CKR_UNWRAPPING_KEY_HANDLE_INVALID
LUNA_RET_UNWRAPPING_KEY_SIZE_RANGE	0x00f10000	CKR_UNWRAPPING_KEY_SIZE_RANGE
LUNA_RET_UNWRAPPING_KEY_TYPE_INCONSISTENT	0x00f20000	CKR_UNWRAPPING_KEY_TYPE_INCONSISTENT
LUNA_RET_USER_ALREADY_LOGGED_IN	0x01000000	CKR_USER_ALREADY_LOGGED_IN
LUNA_RET_SM_OTHER_USER_LOGGED_IN	0x01000001	CKR_USER_ALREADY_LOGGED_IN
LUNA_RET_USER_NOT_LOGGED_IN	0x01010000	CKR_USER_NOT_LOGGED_IN
LUNA_RET_SM_NOT_LOGGED_IN	0x01010001	CKR_USER_NOT_LOGGED_IN
LUNA_RET_USER_PIN_NOT_INITIALIZED	0x01020000	CKR_USER_PIN_NOT_INITIALIZED
LUNA_RET_USER_TYPE_INVALID	0x01030000	CKR_USER_TYPE_INVALID
LUNA_RET_WRAPPED_KEY_INVALID	0x01100000	CKR_WRAPPED_KEY_INVALID
LUNA_RET_WRAPPED_KEY_LEN_RANGE	0x01120000	CKR_WRAPPED_KEY_LEN_RANGE

HSM Error	Hex Code	PKCS#11 or SFNT Defined CKR Error
LUNA_RET_WRAPPING_KEY_HANDLE_INVALID	0x01130000	CKR_WRAPPING_KEY_HANDLE_ INVALID
LUNA_RET_WRAPPING_KEY_SIZE_RANGE	0x01140000	CKR_WRAPPING_KEY_SIZE_ RANGE
LUNA_RET_WRAPPING_KEY_TYPE_ INCONSISTENT	0x01150000	CKR_WRAPPING_KEY_TYPE_ INCONSISTENT
LUNA_RET_CERT_VERSION_NOT_SUPPORTED	0x00300300	CKR_DEVICE_ERROR
LUNA_RET_SIM_AUTHFORM_INVALID	0x0020011e	CKR_SIM_AUTHFORM_INVALID
LUNA_RET_CCM_TOO_LARGE	0x00210001	CKR_DATA_LEN_RANGE
LUNA_RET_TEST_VS_BSAFE_FAILED	0x00300820	CKR_DEVICE_ERROR
LUNA_RET_SFNT3120_ERROR	0x00300821	CKR_DEVICE_ERROR
LUNA_RET_SFNT3120_SELFTEST_FAILED	0x00300822	CKR_DEVICE_ERROR
LUNA_RET_SFNT3120_CRC	0x00300823	CKR_DEVICE_ERROR
LUNA_RET_SFNT3120_ALG_NO_SOFTWARE_ SUPPORT	0x00300824	CKR_DEVICE_ERROR
LUNA_RET_ISES_ERROR	0x00300880	CKR_DEVICE_ERROR
LUNA_RET_ISES_INIT_FAILED	0x00300881	CKR_DEVICE_ERROR
LUNA_RET_ISES_LNAU_TEST_FAILED	0x00300882	CKR_DEVICE_ERROR
LUNA_RET_ISES_RNG_TEST_FAILED	0x00300883	CKR_DEVICE_ERROR
LUNA_RET_ISES_CMD_FAILED	0x00300884	CKR_DEVICE_ERROR
LUNA_RET_ISES_CMD_PARAMETER_INVALID	0x00300885	CKR_DEVICE_ERROR
LUNA_RET_ISES_TEST_VS_BSAFE_FAILED	0x00300886	CKR_DEVICE_ERROR
LUNA_RET_RM_ELEMENT_VALUE_INVALID	0x00200a00	CKR_DATA_INVALID
LUNA_RET_RM_ELEMENT_ID_INVALID	0x00200a01	CKR_DATA_INVALID

HSM Error	Hex Code	PKCS#11 or SFNT Defined CKR Error
LUNA_RET_RM_NO_MEMORY	0x00310a02	CKR_DEVICE_MEMORY
LUNA_RET_RM_BAD_HSM_PARAMS	0x00300a03	CKR_DEVICE_ERROR
LUNA_RET_RM_POLICY_ELEMENT_ DESTRUCTIVE	0x00200a04	CKR_DATA_INVALID
LUNA_RET_RM_POLICY_ELEMENT_NOT_ DESTRUCTIVE	0x00200a05	CKR_DATA_INVALID
LUNA_RET_RM_CONFIG_CHANGE_ILLEGAL	0x00010a06	CKR_CANCEL
LUNA_RET_RM_CONFIG_CHANGE_FAILS_ DEPENDENCIES	0x00010a07	CKR_CANCEL
LUNA_RET_LICENSE_ID_UNKNOWN	0x00200a08	CKR_DATA_INVALID
LUNA_RET_LICENSE_CAPACITY_EXCEEDED	0x00010a09	CKR_LICENSE_CAPACITY_ EXCEEDED
LUNA_RET_RM_POLICY_WRITE_RESTRICTED	0x00010a0a	CKR_CANCEL
LUNA_RET_OPERATION_RESTRICTED	0x00010a0b	CKR_OPERATION_NOT_ ALLOWED
LUNA_RET_CANNOT_PERFORM_OPERATION_ TWICE	0x00010a0c	CKR_CANCEL
LUNA_RET_BAD_PPID	0x00200a0d	CKR_DATA_INVALID
LUNA_RET_BAD_FW_VERSION	0x00200a0e	CKR_DATA_INVALID
LUNA_RET_OPERATION_SHOULD_BE_ DESTRUCTIVE	0x00200a0f	CKR_DATA_INVALID
LUNA_RET_RM_CONFIG_ILLEGAL	0x00200a10	CKR_DATA_INVALID
LUNA_RET_BAD_SN	0x00200a11	CKR_DATA_INVALID
LUNA_RET_CHALLENGE_TYPE_INVALID	0x00200b00	CKR_DATA_INVALID
LUNA_RET_CHALLENGE_REQUIRES_PED	0x00010b01	CKR_CANCEL
LUNA_RET_CHALLENGE_NOT_REQUIRED	0x00010b02	CKR_CANCEL

HSM Error	Hex Code	PKCS#11 or SFNT Defined CKR Error
LUNA_RET_CHALLENGE_RESPONSE_ INCORRECT	0x00a00b03	CKR_PIN_INCORRECT
LUNA_RET_OH_OBJECT_VERSION_INVALID	0x00300c00	CKR_DEVICE_ERROR
LUNA_RET_OH_OBJECT_TYPE_INVALID	0x00300c01	CKR_DEVICE_ERROR
LUNA_RET_OH_OBJECT_ALREADY_EXISTS	0x00010c02	CKR_CANCEL
LUNA_RET_OH_OBJECT_OWNER_DOES_NOT_ EXIST	0x00200c03	CKR_DATA_INVALID
LUNA_RET_STORAGE_TYPE_INCONSISTENT	0x00200c04	CKR_DATA_INVALID
LUNA_RET_CONTAINER_CAN_NOT_HAVE_ MEMBERS	0x00200c05	CKR_DATA_INVALID
LUNA_RET_SAVED_STATE_INVALID	0x01600000	CKR_SAVED_STATE_INVALID
LUNA_RET_STATE_UNSAVEABLE	0x01800000	CKR_STATE_UNSAVEABLE
LUNA_RET_ERROR	0x80000000	CKR_GENERAL_ERROR
LUNA_RET_CONTAINER_HANDLE_INVALID	0x80000001	CKR_CONTAINER_HANDLE_ INVALID
LUNA_RET_INVALID_PADDING_TYPE	0x80000002	CKR_DATA_INVALID
LUNA_RET_NOT_FOUND	0x80000007	CKR_FUNCTION_FAILED
LUNA_RET_TOO_MANY_CONTAINERS	0x80000008	CKR_TOO_MANY_CONTAINERS
LUNA_RET_CONTAINER_LOCKED	0x80000009	CKR_PIN_LOCKED
LUNA_RET_CONTAINER_IS_DISABLED	0x8000000a	CKR_PARTITION_DISABLED
LUNA_RET_SECURITY_PARAMETER_MISSING	0x8000000b	CKR_SECURITY_PARAMETER_ MISSING
LUNA_RET_DEVICE_TIMEOUT	0x8000000c	CKR_DEVICE_TIMEOUT
LUNA_RET_OBJECT_DELETED	0x8000000d	HSM Internal ONLY
LUNA_RET_INVALID_FUF_TARGET	0x8000000e	CKR_INVALID_FUF_TARGET

HSM Error	Hex Code	PKCS#11 or SFNT Defined CKR Error
LUNA_RET_INVALID_FUF_HEADER	0x8000000f	CKR_INVALID_FUF_HEADER
LUNA_RET_INVALID_FUF_VERSION	0x80000010	CKR_INVALID_FUF_VERSION
LUNA_RET_KCV_PARAMETER_ALREADY_ EXISTS	0x80000100	CKR_CLONING_PARAMETER_ ALREADY_EXISTS
LUNA_RET_KCV_PARAMETER_COULD_NOT_ BE_ADDED	0x80000101	CKR_DEVICE_MEMORY
LUNA_RET_INVALID_CERTIFICATE_DATA	0x80000102	CKR_CERTIFICATE_DATA_ INVALID
LUNA_RET_INVALID_CERTIFICATE_TYPE	0x80000103	CKR_CERTIFICATE_DATA_ INVALID
LUNA_RET_INVALID_CERTIFICATE_VERSION	0x80000104	CKR_CERTIFICATE_DATA_ INVALID
LUNA_RET_INVALID_MODULUS_SIZE	0x80000105	CKR_ATTRIBUTE_VALUE_INVALID
LUNA_RET_WRAPPING_ERROR	0x80000107	CKR_WRAPPING_ERROR
LUNA_RET_UNWRAPPING_ERROR	0x80000108	CKR_UNWRAPPING_ERROR
LUNA_RET_INVALID_PRIVATE_KEY_TYPE	0x80000109	CKR_DATA_INVALID
LUNA_RET_TSN_MISMATCH	0x8000010a	CKR_DATA_INVALID
LUNA_RET_KCV_PARAMETER_MISSING	0x8000010b	CKR_CLONING_PARAMETER_ MISSING
LUNA_RET_TWC_PARAMETER_MISSING	0x8000010c	CKR_CERTIFICATE_DATA_ MISSING
LUNA_RET_TUK_PARAMETER_MISSING	0x8000010d	CKR_CERTIFICATE_DATA_ MISSING
LUNA_RET_CPK_PARAMETER_MISSING	0x8000010e	CKR_KEY_NEEDED
LUNA_RET_MASKING_NOT_SUPPORTED	0x8000010f	CKR_FUNCTION_NOT_ SUPPORTED
LUNA_RET_INVALID_ACCESS_LEVEL	0x80000110	CKR_ARGUMENTS_BAD

HSM Error	Hex Code	PKCS#11 or SFNT Defined CKR Error
LUNA_RET_MAC_MISSING	0x80000111	CKR_MAC_MISSING
LUNA_RET_DAC_POLICY_PID_MISMATCH	0x80000112	CKR_DAC_POLICY_PID_ MISMATCH
LUNA_RET_DAC_MISSING	0x80000113	CKR_DAC_MISSING
LUNA_RET_BAD_DAC	0x80000114	CKR_BAD_DAC
LUNA_RET_SSK_MISSING	0x80000115	CKR_SSK_MISSING
LUNA_RET_BAD_MAC	0x80000116	CKR_BAD_MAC
LUNA_RET_DAK_MISSING	0x80000117	CKR_DAK_MISSING
LUNA_RET_BAD_DAK	0x80000118	CKR_BAD_DAK
LUNA_RET_HOK_MISSING	0x80000119	CKR_CERTIFICATE_DATA_ MISSING
LUNA_RET_CITS_DAK_MISSING	0x8000011a	CKR_CITS_DAK_MISSING
LUNA_RET_SIM_AUTHORIZATION_FAILED	0x8000011b	CKR_SIM_AUTHORIZATION_ FAILED
LUNA_RET_SIM_VERSION_UNSUPPORTED	0x8000011c	CKR_SIM_VERSION_ UNSUPPORTED
LUNA_RET_SIM_CORRUPT_DATA	0x8000011d	CKR_SIM_CORRUPT_DATA
LUNA_RET_ECC_MIC_MISSING	0x8000011e	CKR_CERTIFICATE_DATA_ MISSING
LUNA_RET_ECC_HOK_MISSING	0x8000011f	CKR_CERTIFICATE_DATA_ MISSING
LUNA_RET_ECC_HOC_MISSING	0x80000120	CKR_CERTIFICATE_DATA_ MISSING
LUNA_RET_ECC_DAK_MISSING	0x80000121	CKR_CERTIFICATE_DATA_ MISSING
LUNA_RET_ECC_DAC_MISSING	0x80000122	CKR_CERTIFICATE_DATA_ MISSING

HSM Error	Hex Code	PKCS#11 or SFNT Defined CKR Error
LUNA_RET_ROOT_CERT_MISSING	0x80000123	CKR_CERTIFICATE_DATA_ MISSING
LUNA_RET_HOC_MISSING	0x80000124	CKR_CERTIFICATE_DATA_ MISSING
LUNA_RET_INVALID_CERTIFICATE_FUNCTION	0x80000125	CKR_CERTIFICATE_DATA_ INVALID
LUNA_RET_N_TOO_LARGE	0x80000200	CKR_ARGUMENTS_BAD
LUNA_RET_N_TOO_SMALL	0x80000201	CKR_ARGUMENTS_BAD
LUNA_RET_M_TOO_LARGE	0x80000202	CKR_ARGUMENTS_BAD
LUNA_RET_M_TOO_SMALL	0x80000203	CKR_ARGUMENTS_BAD
LUNA_RET_WEIGHT_TOO_LARGE	0x80000204	CKR_ARGUMENTS_BAD
LUNA_RET_WEIGHT_TOO_SMALL	0x80000205	CKR_ARGUMENTS_BAD
LUNA_RET_TOTAL_WEIGHT_INVALID	0x80000206	CKR_ARGUMENTS_BAD
LUNA_RET_MISSING_SPLITS	0x80000207	CKR_ARGUMENTS_BAD
LUNA_RET_SPLIT_DATA_INVALID	0x80000208	CKR_ARGUMENTS_BAD
LUNA_RET_SPLIT_ID_INVALID	0x80000209	CKR_ARGUMENTS_BAD
LUNA_RET_M_OF_N_PARAMETER_NOT_ AVAILABLE	0x8000020a	CKR_OPERATION_NOT_ INITIALIZED
LUNA_RET_M_OF_N_ACTIVATION_REQUIRED	0x8000020b	CKR_OPERATION_NOT_ INITIALIZED
LUNA_RET_TOO_MANY_WEIGHTS	0x8000020e	CKR_ARGUMENTS_BAD
LUNA_RET_MISSING_WEIGHT_VALUE	0x8000020f	CKR_ARGUMENTS_BAD
LUNA_RET_MISSING_VALUE_FOR_M	0x80000210	CKR_ARGUMENTS_BAD
LUNA_RET_MISSING_VALUE_FOR_N	0x80000211	CKR_ARGUMENTS_BAD
LUNA_RET_MISSING_NUMBER_OF_VECTORS	0x80000212	CKR_ARGUMENTS_BAD

HSM Error	Hex Code	PKCS#11 or SFNT Defined CKR Error
LUNA_RET_MISSING_VECTOR	0x80000213	CKR_ARGUMENTS_BAD
LUNA_RET_VECTOR_TOO_LARGE	0x80000214	CKR_ARGUMENTS_BAD
LUNA_RET_VECTOR_TOO_SMALL	0x80000215	CKR_ARGUMENTS_BAD
LUNA_RET_TOO_MANY_VECTORS_PROVIDED	0x80000216	CKR_ARGUMENTS_BAD
LUNA_RET_INVALID_VECTOR_SIZE	0x80000217	CKR_ARGUMENTS_BAD
LUNA_RET_M_OF_N_PARAMETER_EXIST	0x80000218	CKR_FUNCTION_FAILED
LUNA_RET_VECTOR_VERSION_INVALID	0x80000219	CKR_DATA_INVALID
LUNA_RET_VECTOR_OF_DIFFERENT_SET	0x8000021a	CKR_ARGUMENTS_BAD
LUNA_RET_VECTOR_DUPLICATE	0x8000021b	CKR_ARGUMENTS_BAD
LUNA_RET_VECTOR_TYPE_INVALID	0x8000021c	CKR_ARGUMENTS_BAD
LUNA_RET_MISSING_COMMAND_PARAMETER	0x8000021d	CKR_ARGUMENTS_BAD
LUNA_RET_M_OF_N_CLONING_IS_NOT_ ALLOWED	0x8000021e	CKR_FUNCTION_NOT_ SUPPORTED
LUNA_RET_M_OF_N_IS_NOT_REQUIRED	0x8000021f	CKR_OPERATION_NOT_ INITIALIZED
LUNA_RET_M_OF_N_IS_NOT_INITIALZED	0x80000220	CKR_OPERATION_NOT_ INITIALIZED
LUNA_RET_M_OF_N_SECRET_INVALID	0x80000221	CKR_GENERAL_ERROR
LUNA_RET_CCM_NOT_PRESENT	0x80000300	CKR_FUNCTION_NOT_ SUPPORTED
LUNA_RET_CCM_NOT_SUPPORTED	0x80000301	CKR_FUNCTION_NOT_ SUPPORTED
LUNA_RET_CCM_UNREMOVABLE	0x80000302	CKR_DATA_INVALID
LUNA_RET_CCM_CERT_INVALID	0x80000303	CKR_DATA_INVALID
LUNA_RET_CCM_SIGN_INVALID	0x80000304	CKR_DATA_INVALID

HSM Error	Hex Code	PKCS#11 or SFNT Defined CKR Error
LUNA_RET_CCM_UPDATE_DENIED	0x80000305	CKR_DATA_INVALID
LUNA_RET_CCM_FWUPDATE_DENIED	0x80000306	CKR_DATA_INVALID
LUNA_RET_SM_ACCESS_ID_INVALID	0x80000400	CKR_DATA_INVALID
LUNA_RET_SM_ACCESS_ALREADY_EXISTS	0x80000401	CKR_DATA_INVALID
LUNA_RET_SM_MULTIPLE_ACCESS_DISABLED	0x80000402	CKR_FUNCTION_NOT_ SUPPORTED
LUNA_RET_SM_UNKNOWN_ACCESS_TYPE	0x80000403	CKR_ARGUMENTS_BAD
LUNA_RET_SM_BAD_ACCESS_HANDLE	0x80000404	CKR_DATA_INVALID
LUNA_RET_SM_BAD_CONTEXT_NUMBER	0x80000405	CKR_DATA_INVALID
LUNA_RET_SM_UNKNOWN_SESSION_TYPE	0x80000406	CKR_DATA_INVALID
LUNA_RET_SM_CONTEXT_ALREADY_ ALLOCATED	0x80000407	CKR_DATA_INVALID
LUNA_RET_SM_CONTEXT_NOT_ALLOCATED	0x80000408	CKR_DEVICE_MEMORY
LUNA_RET_SM_CONTEXT_BUFFER_OVERFLOW	0x80000409	CKR_DEVICE_MEMORY
LUNA_RET_SM_TOSM_DOES_NOT_VALIDATE	0x8000040A	CKR_USER_NOT_LOGGED_IN
LUNA_RET_SM_ACCESS_DOES_NOT_VALIDATE	0x8000040B	CKR_USER_NOT_AUTHORIZED
LUNA_RET_MTK_ZEROIZED	0x80000531	CKR_MTK_ZEROIZED
LUNA_RET_MTK_STATE_INVALID	0x80000532	CKR_MTK_STATE_INVALID
LUNA_RET_MTK_SPLIT_INVALID	0x80000533	CKR_MTK_SPLIT_INVALID
LUNA_RET_INVALID_IP_PACKET	0x80000600	CKR_DEVICE_ERROR
LUNA_RET_INVALID_BOARD_TYPE	0x80000700	CKR_DEVICE_ERROR
LUNA_RET_ECC_NOT_SUPPORTED	0x80000601	CKR_FUNCTION_NOT_ SUPPORTED
LUNA_RET_ECC_BUFFER_OVERFLOW	0x80000602	CKR_DEVICE_ERROR

HSM Error	Hex Code	PKCS#11 or SFNT Defined CKR Error
LUNA_RET_ECC_POINT_INVALID	0x80000603	CKR_ECC_POINT_INVALID**
LUNA_RET_ECC_SELF_TEST_FAILURE	0x80000604	CKR_DEVICE_ERROR
LUNA_RET_ECC_UNKNOWN_CURVE	0x80000605	CKR_ECC_UNKNOWN_CURVE
LUNA_RET_HA_NOT_SUPPORTED	0x80000900	CKR_FUNCTION_NOT_ SUPPORTED
LUNA_RET_HA_USER_NOT_INITIALIZED	0x80000901	CKR_OPERATION_NOT_ INITIALIZED
LUNA_RET_HSM_STORAGE_FULL	0x80000902	CKR_HSM_STORAGE_FULL
LUNA_RET_CONTAINER_OBJECT_STORAGE_ FULL	0x80000903	CKR_CONTAINER_OBJECT_ STORAGE_FULL
LUNA_RET_KEY_NOT_ACTIVE	0x80000904	CKR_KEY_NOT_ACTIVE
LUNA_RET_CB_NOT_SUPPORTED	0x80000a01	CKR_FUNCTION_NOT_ SUPPORTED
LUNA_RET_CB_PARAM_INVALID	0x80000a02	CKR_CALLBACK_ERROR
LUNA_RET_CB_NO_MEMORY	0x80000a03	CKR_DEVICE_MEMORY
LUNA_RET_CB_TIMEOUT	0x80000a04	CKR_CALLBACK_ERROR
LUNA_RET_CB_RETRY	0x80000a05	CKR_CALLBACK_ERROR
LUNA_RET_CB_ABORTED	0x80000a06	CKR_CALLBACK_ERROR
LUNA_RET_CB_SYS_ERROR	0x80000a07	CKR_CALLBACK_ERROR
LUNA_RET_CB_HIOS_HANDLE_INVALID	0x80000a10	CKR_CALLBACK_ERROR
LUNA_RET_CB_HIOS_ID_INVALID	0x80000a11	CKR_CALLBACK_ERROR
LUNA_RET_CB_HIOS_CLOSED	0x80000a12	CKR_CALLBACK_ERROR
LUNA_RET_CB_HIOS_CANCELED	0x80000a13	CKR_CALLBACK_ERROR
LUNA_RET_CB_HIOS_IO_ERROR	0x80000a14	CKR_CALLBACK_ERROR

HSM Error	Hex Code	PKCS#11 or SFNT Defined CKR Error
LUNA_RET_CB_HIOS_SEND_TIMEOUT	0x80000a15	CKR_CALLBACK_ERROR
LUNA_RET_CB_HIOS_RECV_TIMEOUT	0x80000a16	CKR_CALLBACK_ERROR
LUNA_RET_CB_HIOS_STATE_INVALID	0x80000a17	CKR_CALLBACK_ERROR
LUNA_RET_CB_HIOS_OUTPUT_BUFFER_TOO_SMALL	0x80000a18	CKR_CALLBACK_ERROR
LUNA_RET_CB_HIOS_INPUT_BUFFER_TOO_SMALL	0x80000a19	CKR_CALLBACK_ERROR
LUNA_RET_CB_HANDLE_INVALID	0x80000a20	CKR_CALLBACK_ERROR
LUNA_RET_CB_ID_INVALID	0x80000a21	CKR_CALLBACK_ERROR
LUNA_RET_CB_REMOTE_ABORT	0x80000a22	CKR_CALLBACK_ERROR
LUNA_RET_CB_REMOTE_CLOSED	0x80000a23	CKR_CALLBACK_ERROR
LUNA_RET_CB_REMOTE_ABANDONED	0x80000a24	CKR_CALLBACK_ERROR
LUNA_RET_CB_MUST_READ	0x80000a25	CKR_CALLBACK_ERROR
LUNA_RET_CB_MUST_WRITE	0x80000a26	CKR_CALLBACK_ERROR
LUNA_RET_CB_INVALID_CALL_FOR_THE_STATE	0x80000a27	CKR_CALLBACK_ERROR
LUNA_RET_CB_SYNC_ERROR	0x80000a28	CKR_CALLBACK_ERROR
LUNA_RET_CB_PROT_DATA_INVALID	0x80000a29	CKR_CALLBACK_ERROR
LUNA_RET_LOG_FILE_NOT_OPEN	0x80000d00	CKR_LOG_FILE_NOT_OPEN
LUNA_RET_LOG_FILE_WRITE_ERROR	0x80000d01	CKR_LOG_FILE_WRITE_ERROR
LUNA_RET_LOG_BAD_FILE_NAME	0x80000d02	CKR_LOG_BAD_FILE_NAME
LUNA_RET_LOG_FULL	0x80000d03	CKR_LOG_FULL
LUNA_RET_LOG_NO_KCV	0x80000d04	CKR_LOG_NO_KCV
LUNA_RET_LOG_BAD_RECORD_HMAC	0x80000d05	CKR_LOG_BAD_RECORD_HMAC

HSM Error	Hex Code	PKCS#11 or SFNT Defined CKR Error
LUNA_RET_LOG_BAD_TIME	0x80000d06	CKR_LOG_BAD_TIME
LUNA_RET_LOG_AUDIT_NOT_INITIALIZED	0x80000d07	CKR_LOG_AUDIT_NOT_ INITIALIZED
LUNA_RET_LOG_RESYNC_NEEDED	0x80000d08	CKR_LOG_RESYNC_NEEDED
LUNA_RET_AUDIT_LOGIN_TIMEOUT_IN_ PROGRESS	0x80000d09	CKR_AUDIT_LOGIN_TIMEOUT_IN_ PROGRESS
LUNA_RET_AUDIT_LOGIN_FAILURE_ THRESHOLD	0x80000d0a	CKR_AUDIT_LOGIN_FAILURE_ THRESHOLD
LUNA_RET_XTC_ERROR	0x80001600	CKR_XTC_ERROR
LUNA_RET_CONTEXT_INVALID	0x80001601	CKR_CONTEXT_INVALID
LUNA_RET_SESSION_COUNT	0x80001603	CKR_MAX_SESSION_COUNT
LUNA_RET_BUSY	0x80001604	CKR_BUSY

* This error (CKR_TEMPLATE_INCONSISTENT) might be encountered when using CKDemo in a new client with firmware older than version 6.22.0. Try CKDemo option 98, sub-option 16. If it is set to "enhanced roles", try selecting it to set it to "legacy Luna roles". The setting is a toggle, and flips every time you select it.

** This error, or "unable to read public key", might be encountered when using BSAFE to encrypt data with ECC public key using curves from the Brainpool suite. As indicated on the BSAFE website (May 2012) they do not appear to support Brainpool curves. Therefore, your own applications should not attempt that combination, and you should avoid attempting to specify Brainpool curves with BSAFE ECC when using the Luna CKDemo utility.

† These errors (0x00310000 through 0x00310007) are all considered as the same generic CKR_DEVICE_ MEMORY error *by a host application*. They are visible using Luna HSM Firmware 7.8.1 or newer. Your application is responsible for closing crypto sessions and releasing resources when they are no longer being used. Failure to do so allows accumulating orphan sessions to eventually consume all available HSM memory. The HSM has no way to know that any given session is no longer needed, unless your application explicitly closes the session. If a solution/application mixes Java and C/C++ operations, those might not agree on when sessions are to close.

Workaround: If you are seeing device memory errors in the logs, consider:

1. Stopping your application (so that it does not continue attempting cryptographic operations against the HSM)
2. Restarting the HSM to clear any orphan sessions or processes (use hsm restart to restart the HSM)
3. Restarting your application.

> **NOTE** This is a temporary measure to clear the effects of application behavior that should be corrected, or the problem can recur. Contact your application vendor to report the memory issue.

Library Codes

Hex value	Decimal value	Return code/error description
0	0	OKAY, NO ERROR
0xC0000000	3221225472	PROGRAMMING ERROR: RETURN CODE
0xC0000001	3221225473	OUT OF MEMORY
0xC0000002	3221225474	NON-SPECIFIC ERROR
0xC0000003	3221225475	UNEXPECTED NULL POINTER
0xC0000004	3221225476	PROGRAMMING ERROR: LOGIC
0xC0000005	3221225477	OPERATION WOULD BLOCK IF ATTEMPTED
0xC0000006	3221225478	BUFFER IS TOO SMALL
0xC0000100	3221225728	OPERATION CANCEL
0xC0000101	3221225729	INVALID SLOT IDENTIFIER
0xC0000102	3221225730	INVALID DATA
0xC0000103	3221225731	INVALID PIN
0xC0000104	3221225732	NO TOKEN PRESENT
0xC0000105	3221225733	FUNCTION IS NOT SUPPORTED
0xC0000106	3221225734	NON-CRYPTOKI ELEMENT CLONE
0xC0000107	3221225735	INVALID BUFFER SIZE FOR CHALLENGE
0xC0000108	3221225736	PIN IS LOCKED
0xC0000109	3221225737	INVALID VERSION
0xC000010a	3221225738	NEEDED KEY NOT PROVIDED

Hex value	Decimal value	Return code/error description
0xC000010b	3221225739	USER NAME IS IN USE
0xC0000200	3221225984	INVALID DISTINGUISHED ENCODING RULES CLASS
0xC0000303	3221226243	OPERATION TIMED OUT
0xC0000304	3221226244	RESET FAILED
0xC0000400	3221226496	INVALID TOKEN STATE
0xC0000401	3221226497	DATA APPEARS CORRUPTED
0xC0000402	3221226498	INVALID FILENAME
0xC0000403	3221226499	FILE IS READ-ONLY
0xC0000404	3221226500	FILE ERROR
0xC0000405	3221226501	INVALID OBJECT IDENTIFIER
0xC0000406	3221226502	INVALID SOCKET ADDRESS
0xC0000407	3221226503	INVALID LISTEN SOCKET
0xC0000408	3221226504	CACHE IS NOT CURRENT
0xC0000409	3221226505	CACHE IS NOT MAPPED
0xC000040a	3221226506	OBJECT IS NOT IN LIST
0xC000040b	3221226507	INVALID INDEX
0xC000040c	3221226508	OBJECT ALREADY EXISTS
0xC000040d	3221226509	SEMAPHORE ERROR
0xC000040e	3221226510	END OF LIST ENCOUNTERED
0xC000040f	3221226511	WOULD ASSIGN SAME VALUE
0xC0000410	3221226512	INVALID GROUP NAME
0xC0000411	3221226513	NOT HSM BACKUP TOKEN

Hex value	Decimal value	Return code/error description
0xC0000412	3221226514	NOT PARTITION BACKUP TOKEN
0xC0000413	3221226515	SIM NOT SUPPORTED
0xC0000500	3221226752	SOCKET ERROR
0xC0000501	3221226753	SOCKET WRITE ERROR
0xC0000502	3221226754	SOCKET READ ERROR
0xC0000503	3221226755	CLIENT MESSAGE ERROR
0xC0000504	3221226756	SERVER DISCONNECTED
0xC0000505	3221226757	CLIENT DISCONNECTED
0xC0000506	3221226758	SOCKET WOULD BLOCK
0xC0000507	3221226759	SOCKET ADDRESS IS IN USE
0xC0000508	3221226760	SOCKET BAD FILE DESCRIPTOR
0xC0000509	3221226761	HOST RESOLUTION ERROR
0xC000050a	3221226762	INVALID HOST CERTIFICATE
0xC0000600	3221227008	NO BUFFER AVAILABLE
0xC0000601	3221227009	INVALID ENUMERATION OPTION
0xC0000700	3221227264	SSL ERROR
0xC0000701	3221227265	SSL CTX ERROR
0xC0000702	3221227266	SSL CIPHER LIST ERROR
0xC0000703	3221227267	SSL CERT VERIFICATION LOCATION ERROR
0xC0000704	3221227268	SSL LOAD SERVER CERT ERROR
0xC0000705	3221227269	SSL LOAD SERVER PRIVATE KEY ERROR
0xC0000706	3221227270	SSL VALIDATE SERVER PRIVATE KEY ERROR

Hex value	Decimal value	Return code/error description
0xC0000707	3221227271	SSL CREATE SSL ERROR
0xC0000708	3221227272	SSL LOAD CLIENT CERT ERROR
0xC0000709	3221227273	SSL GET CERTIFICATE ERROR
0xC000070a	3221227274	SSL INVALID CERT STRUCTURE
0xC000070b	3221227275	SSL LOAD CLIENT PRIVATE KEY ERROR
0xC000070c	3221227276	SSL GET PEER CERT ERROR
0xC000070d	3221227277	SSL WANT READ ERROR
0xC000070e	3221227278	SSL WANT WRITE ERROR
0xC000070f	3221227279	SSL WANT X509 LOOKUP ERROR
0xC0000710	3221227280	SSL SYSCALL ERROR
0xC0000711	3221227281	SSL FAILED HANDSHAKE
0xC0000800	3221227520	INVALID CERTIFICATE TYPE
0xC0000900	3221227776	INVALID PORT
0xC0000901	3221227777	SESSION SCRIPT EXISTS
0xC0001000	3221229568	PARTITION LOCKED
0xC0001001	3221229569	PARTITION NOT ACTIVATED
0xc0002000	3221233664	FAILED TO CREATE THREAD
0xc0002001	3221233665	CALLBACK ERROR
0xc0002002	3221233666	UNKNOWN CALLBACK COMMAND
0xc0002003	3221233667	SHUTTING DOWN
0xc0002004	3221233668	REMOTE SIDE DISCONNECTED
0xc0002005	3221233669	SOCKET CLOSED
0xC0002006	3221233670	INVALID COMMAND

Hex value	Decimal value	Return code/error description
0xC0002007	3221233671	UNKNOWN COMMAND
0xC0002008	3221233672	UNKNOWN COMMAND VERSION
0xC0002009	3221233673	FILE LOCK FAILED
0xC0002010	3221233680	FILE LOCK ERROR
0xc0002011	3221233681	FAILED TO CREATE PROCESS
0xc0002012	3221233682	USB PED NOT FOUND
0xc0002013	3221233683	USB PED NOT RESPONDING
0xc0002014	3221233684	USB PED OPERATION CANCELLED
0xc0002015	3221233685	USB PED TOO MANY CONNECTED
0xc0002016	3221233686	USB PED OUT OF SYNC
0xC0001100	3221229824	UNABLE TO CONNECT

Vendor-Defined Return Codes

Code	Name
0x80000004	CKR_RC_ERROR
0x80000005	CKR_CONTAINER_HANDLE_INVALID
0x80000006	CKR_TOO_MANY_CONTAINERS
0x80000007	CKR_USER_LOCKED_OUT
0x80000008	CKR_CLONING_PARAMETER_ALREADY_EXISTS
0x80000009	CKR_CLONING_PARAMETER_MISSING
0x8000000a	CKR_CERTIFICATE_DATA_MISSING
0x8000000b	CKR_CERTIFICATE_DATA_INVALID
0x8000000c	CKR_ACCEL_DEVICE_ERROR

Code	Name
0x8000000d	CKR_WRAPPING_ERROR
0x8000000e	CKR_UNWRAPPING_ERROR
0x8000000f	CKR_MAC_MISSING
0x80000010	CKR_DAC_POLICY_PID_MISMATCH
0x80000011	CKR_DAC_MISSING
0x80000012	CKR_BAD_DAC
0x80000013	CKR_SSK_MISSING
0x80000014	CKR_BAD_MAC
0x80000015	CKR_DAK_MISSING
0x80000016	CKR_BAD_DAK
0x80000017	CKR_SIM_AUTHORIZATION_FAILED
0x80000018	CKR_SIM_VERSION_UNSUPPORTED
0x80000019	CKR_SIM_CORRUPT_DATA
0x8000001a	CKR_USER_NOT_AUTHORIZED
0x8000001b	CKR_MAX_OBJECT_COUNT_EXCEEDED
0x8000001c	CKR_SO_LOGIN_FAILURE_THRESHOLD
0x8000001d	CKR_SIM_AUTHFORM_INVALID
0x8000001e	CKR_CITS_DAK_MISSING
0x8000001f	CKR_UNABLE_TO_CONNECT
0x80000020	CKR_PARTITION_DISABLED
0x80000021	CKR_CALLBACK_ERROR
0x80000022	CKR_SECURITY_PARAMETER_MISSING
0x80000023	CKR_SP_TIMEOUT

Code	Name
0x80000024	CKR_TIMEOUT
0x80000025	CKR_ECC_UNKNOWN_CURVE
0x80000026	CKR_MTK_ZEROIZED
0x80000027	CKR_MTK_STATE_INVALID
0x80000028	CKR_INVALID_ENTRY_TYPE
0x80000029	CKR_MTK_SPLIT_INVALID
0x8000002a	CKR_HSM_STORAGE_FULL
0x8000002b	CKR_DEVICE_TIMEOUT
0x8000002c	CKR_CONTAINER_OBJECT_STORAGE_FULL
0x8000002d	CKR_PED_CLIENT_NOT_RUNNING
0x8000002e	CKR_PED_UNPLUGGED
0x8000002f	CKR_ECC_POINT_INVALID
0x80000030	CKR_OPERATION_NOT_ALLOWED
0x80000031	CKR_LICENSE_CAPACITY_EXCEEDED
0x80000032	CKR_LOG_FILE_NOT_OPEN
0x80000033	CKR_LOG_FILE_WRITE_ERROR
0x80000034	CKR_LOG_BAD_FILE_NAME
0x80000035	CKR_LOG_FULL
0x80000036	CKR_LOG_NO_KCV
0x80000037	CKR_LOG_BAD_RECORD_HMAC
0x80000038	CKR_LOG_BAD_TIME
0x80000039	CKR_LOG_AUDIT_NOT_INITIALIZED
0x8000003A	CKR_LOG_RESYNC_NEEDED

Code	Name
0x8000003B	CKR_AUDIT_LOGIN_TIMEOUT_IN_PROGRESS
0x8000003C	CKR_AUDIT_LOGIN_FAILURE_THRESHOLD
0x8000003D	CKR_INVALID_FUF_TARGET
0x8000003E	CKR_INVALID_FUF_HEADER
0x8000003F	CKR_INVALID_FUF_VERSION
0x80000040	CKR_ECC_ECC_RESULT_AT_INF
0x80000041	CKR_AGAIN
0x80000042	CKR_TOKEN_COPIED
0x80000043	CKR_SLOT_NOT_EMPTY
0x80000044	CKR_USER_ALREADY_ACTIVATED
0x80000045	CKR_STC_NO_CONTEXT
0x80000046	CKR_STC_CLIENT_IDENTITY_NOT_CONFIGURED
0x80000047	CKR_STC_PARTITION_IDENTITY_NOT_CONFIGURED
0x80000048	CKR_STC_DH_KEYGEN_ERROR
0x80000049	CKR_STC_CIPHER_SUITE_REJECTED
0x8000004a	CKR_STC_DH_KEY_NOT_FROM_SAME_GROUP
0x8000004b	CKR_STC_COMPUTE_DH_KEY_ERROR
0x8000004c	CKR_STC_FIRST_PHASE_KDF_ERROR
0x8000004d	CKR_STC_SECOND_PHASE_KDF_ERROR
0x8000004e	CKR_STC_KEY_CONFIRMATION_FAILED
0x8000004f	CKR_STC_NO_SESSION_KEY
0x80000050	CKR_STC_RESPONSE_BAD_MAC
0x80000051	CKR_STC_NOT_ENABLED

Code	Name
0x80000052	CKR_STC_CLIENT_HANDLE_INVALID
0x80000053	CKR_STC_SESSION_INVALID
0x80000054	CKR_STC_CONTAINER_INVALID
0x80000055	CKR_STC_SEQUENCE_NUM_INVALID
0x80000056	CKR_STC_NO_CHANNEL
0x80000057	CKR_STC_RESPONSE_DECRYPT_ERROR
0x80000058	CKR_STC_RESPONSE_REPLAYED
0x80000059	CKR_STC_REKEY_CHANNEL_MISMATCH
0x8000005a	CKR_STC_RSA_ENCRYPT_ERROR
0x8000005b	CKR_STC_RSA_SIGN_ERROR
0x8000005c	CKR_STC_RSA_DECRYPT_ERROR
0x8000005d	CKR_STC_RESPONSE_UNEXPECTED_KEY
0x8000005e	CKR_STC_UNEXPECTED_NONCE_PAYLOAD_SIZE
0x8000005f	CKR_STC_UNEXPECTED_DH_DATA_SIZE
0x80000060	CKR_STC_OPEN_CIPHER_MISMATCH
0x80000061	CKR_STC_OPEN_DHNIST_PUBKEY_ERROR
0x80000062	CKR_STC_OPEN_KEY_MATERIAL_GEN_FAIL
0x80000063	CKR_STC_OPEN_RESP_GEN_FAIL
0x80000064	CKR_STC_ACTIVATE_MACTAG_U_VERIFY_FAIL
0x80000065	CKR_STC_ACTIVATE_MACTAG_V_GEN_FAIL
0x80000066	CKR_STC_ACTIVATE_RESP_GEN_FAIL
0x80000067	CKR_CHALLENGE_INCORRECT
0x80000068	CKR_ACCESS_ID_INVALID

Code	Name
0x80000069	CKR_ACCESS_ID_ALREADY_EXISTS
0x8000006a	CKR_KEY_NOT_KEKABLE
0x8000006b	CKR_MECHANISM_INVALID_FOR_FP
0x8000006c	CKR_OPERATION_INVALID_FOR_FP
0x8000006d	CKR_SESSION_HANDLE_INVALID_FOR_FP
0x8000006e	CKR_CMD_NOT_ALLOWED_HSM_IN_TRANSPORT
0x8000006f	CKR_OBJECT_ALREADY_EXISTS
0x80000070	CKR_PARTITION_ROLE_DESC_VERSION_INVALID
0x80000071	CKR_PARTITION_ROLE_POLICY_VERSION_INVALID
0x80000072	CKR_PARTITION_ROLE_POLICY_SET_VERSION_INVALID
0x80000073	CKR_REKEK_KEY
0x80000074	CKR_KEK_RETRY_FAILURE
0x80000075	CKR_RNG_RESEED_TOO_EARLY
0x80000076	CKR_HSM_TAMPERED
0x80000077	CKR_CONFIG_CHANGE_ILLEGAL
0x80000078	CKR_SESSION_CONTEXT_NOT_ALLOCATED
0x80000079	CKR_SESSION_CONTEXT_ALREADY_ALLOCATED
0x8000007a	CKR_INVALID_BL_ITB_AUTH_HEADER
0x80000114	CKR_OBJECT_READ_ONLY
0x80000136	CKR_KEY_NOT_ACTIVE
0x80000400	CKR_ACCESS_ID_INVALID
0x80001600	CKR_XTC_ERROR
0x80001601	CKR_CONTEXT_INVALID

Code	Name
0x80001603	CKR_MAX_SESSION_COUNT
0x80001604	CKR_BUSY
0x80001606	CKR_SERVICE_UNAVAILABLE

HSM Alarm Codes

The Luna PCIe HSM 7 alarm messages indicate error conditions on the HSM card that might require user intervention. The alarms apply to a Luna HSM, compliant with security level FIPS 140-2 Level 3 . The alarm messages provide appropriate detail to alert HSM users of important events. Each alarm message has a unique character string for the message ID that allows higher level tools on the host system to parse for the alarm message IDs and generate notifications.

Messages are saved to the system log file in Linux host systems, allowing host application software like SNMP to parse the log file, and to the Windows Event Viewer in Windows host systems

Messages can be retrieved with the "dmesg" utility, to read messages from the driver log, which collects messages from the bootloader (BL), the firmware (FW), or from the Host Driver itself.

This section contains the following information:

> "Alarm Generation and Handling" below

> "List of HSM Alarm Codes" on page 283

> "HSM Alarm Code Samples" on page 288

> "Stored Data Integrity" on page 292

Alarm Generation and Handling

Alarm messages can be generated due to the HSM BootLoader, Firmware, and Host Driver Software detecting unexpected conditions. Other alarm messages are generated after unexpected interrupts or tamper events. For each of these problems detailed error information and an alarm message is output to notify the user that something special has happened.

At least one alarm message is output as a result of each tamper event by BL, FW, or Host Driver. Depending on the type of tamper all of them may report an alarm message related to the same tamper event. The message timestamps assist you to identify which alarm messages are for the same tamper event. Tamper alarm messages from BL, FW, and Host Driver have the same text description for the same tamper event. A specific type of tamper event is not reported again until FW clears the tamper information in the tamper circuit. If the tamper event happens after that, then either a new tamper condition has been detected or the same tamper event is still active and cannot be cleared.

Alarm Handling for Special Situations

Alarm messages are still generated during rare occurrences where BL, FW, or Host Driver might be in an abnormal state.

As long as the Host Driver is running, the BL and FW are able to output their alarm messages to the DLOG (driver log), which can be parsed to notify the user. If either BL or FW stops execution due to error detection, they output an alarm message to the Host Driver, which stores it in DLOG. All BL and FW checking for alarm conditions is stopped but all HW tamper event monitoring (soft and hard tampers) is still enabled including Host Driver monitoring. The card reset caused by these tampers restarts BL and possibly FW and the alarm messages are output. The following situations are also handled:

> **BL starts before Host Driver is loaded (System power-up):** Without Host Driver available, BL outputs all alarms only to an internal HSM log. When the Host Driver loads it resets the HSM card, causing BL to start again. BL can then send any new alarms to the host driver and either stop or proceed to FW, as the situation allows.

- For an L3 card if FW is started it will output alarm messages for any existing tamper conditions. Any tamper event alarm messages including those not sent out while the Host Driver was not loaded can be fetched from the FRAM Log.

> **NOTE** If needed, use the **lunadiag** utility to output the FRAM Log in order to determine the tamper information, or to pass on to Thales Technical Support.

> **FW halted due to internal error:** In order to get to FW the Host Driver must be running so the FW halted alarm message will be stored in DLOG. No further BL or FW alarm messages are generated in this state until the next card reset.

> **FW in locked state (tamper clear required):** An alarm message is generated to signal locked state is active. FW is still doing periodic checks and FW alarm messages are still possible. Only a small subset of FW commands is available.

> **FW in Secure Transport Mode (STM):** An alarm message is generated to signal STM is active. FW is still doing periodic checks and FW alarm messages are still possible. Only a small subset of FW commands are available.

> **Host Driver loses communications with the HSM card**: If the Host Driver has any errors communicating with the K7 (BL or FW) it will generate alarm messages. The Host Driver also periodically checks that the Luna PCIe HSM 7 card is still present on the PCIe bus (i.e. chassis open causes a cold reset of the HSM) and if there is no response for a pre-determined period of time an alarm message is generated.

FRAM LOG

The Boot Loader and firmware also store all alarm event information in the FRAM Log in the non-volatile FRAM device on the K7. There is no specific FRAM Log partition for DLOG or alarm messages. Use LUNADIAG to retrieve the FRAM Log contents and return it to Thales Customer Support for further analysis. In the event the Host Driver is unavailable to receive this information, it is still present in the FRAM Log and can be retrieved long after the alarm event has finished.

List of HSM Alarm Codes

ALM ID	Alarm Message	Description	Info
Host Driver			**Tamper Flag**
0001	Soft tamper - over voltage	HSM voltage is above the operating range. HSM will stay in reset until voltage goes back in range.	HCCSR: VST
0002	Soft tamper - temperature (nnC)	HSM temperature (nn degrees Celsius) is outside the range (-2C to 80C). HSM will stay in reset until temperature goes back in range.	HRCSR: TST
0003	Soft tamper - indeterminate cause	A soft tamper occurred but cannot determine the cause.	
0004	Hard tamper - high temperature	HSM temperature is higher than 88C.	HT_T
0005	Hard tamper - low temperature	HSM temperature is lower than -40C	LT_T
0006	Hard tamper - over voltage	HSM voltage is higher than the maximum allowed.	OV_T, TC3_T
0009	Hard tamper - oscillator failure	HSM tamper clock oscillator has failed	OSC_T
0010	Decommission signal triggered	Decommission button (connector P9) has been pressed.	TC2_T
0011	Hard tamper - indeterminate cause	A hard tamper occurred but cannot determine the cause.	
0012	Hardware Error	Error detected in device hardware	
0013	High Temperature - nnC	HSM has reached nn degrees Celsius and needs to be cooled to avoid tampering	
0014	Low Battery	HSM battery voltage is below 2.75V and needs to be replaced soon.	

ALM ID	Alarm Message	Description	Info
0015	PCIe Link Failure	HSM no longer appears on PCIe bus. Chassis may have been opened.	
0016	Device Error	Internal error detected during communications with HSM	
0017	Request Timed Out	Request to HSM took too long	

Boot Loader			Tamper Flag
1000	Unknown alarm ID xx in boot loader	Illegal alarm ID used in Boot Loader.	
1001	HSM restart required	Soft or hard tamper occurred. HSM needs to be restarted (reset) before firmware is allowed to run.	
1003	HSM halted - internal boot loader error	Boot Loader detected an error during diagnostics and did not jump to FW.	
1004	Warning - boot loader diagnostic error	Boot Loader detected an error during diagnostics that does not stop execution but needs to be investigated (i.e. fan, VPD, or RTC problems).	
1005	HSM FW signature check failed	The FW image on the HSM failed authentication and will not be executed.	
1006	Soft tamper temperature/voltage	HSM voltage or temperature is outside the acceptable range. HSM will stay in reset until back in range.	PORSM status reg.
1007	Hard tamper - high temperature	HSM voltage or temperature is outside the acceptable range. HSM will stay in reset until back in range.	HT_T
1008	Hard tamper - low temperature	HSM temperature is lower than -40C.	LT_T

ALM ID	Alarm Message	Description	Info
1009	Hard tamper - over voltage	HSM voltage is higher than the maximum allowed.	OV_T, TC3_T
1012	Hard tamper - oscillator failure	HSM tamper clock oscillator has failed	OSC_T
1013	Hard tamper - tamper configuration invalid	HSM tamper configuration lost (set to defaults) due to power loss.	FS_T
1014	Chassis opened	Chassis open switch (connector P7) has been triggered.	TC1_T
1015	HSM removed from chassis	HSM was removed from host chassis then re-inserted	CS
1016	Decommission signal triggered	Decommission button (connector P9) has been pressed.	TC2_T

Firmware			
2000	Unknown alarm ID xx in firmware	Illegal alarm ID used in firmware.	
2001	High temperature warning activated	HSM temperature is above 75C (FW checks every 2 minutes). This warning will not re-appear unless temperature drops below 75C and goes back up again.	
2002	High temperature warning deactivated	HSM temperature has dropped below 75C.	
2003	Battery low voltage warning	Battery voltage is below 2.75V (FW checks every hour). This warning will not re-appear unless voltage goes above 2.75V then back down. Battery should to be replaced soon.	
2004	Battery depleted	Battery voltage is below 2.5V (FW checks every hour). HSM FW will be halted. Battery must to be replaced.	
2005	HSM deactivated	Auto-activation data has been cleared	

ALM ID	Alarm Message	Description	Info
2006	HSM decommissioned by FW	All user crypto material has been invalidated due to KEK CRC failure, decommission signal, or tamper (if decommission on tamper enabled).	
2007	HSM zeroized	All user crypto material has been erased. HSM product credentials still exist. This can occur for a variety of reasons including manual zeroization.	
2008	Internal data corruption	Settings to control tamper monitoring are incorrect or Critical Security Parameter data (MTK) is invalid (the tamper monitoring settings if incorrect are corrected.). Otherwise there was an unexpected tamper security write protection change.	
2009	HSM halted - internal firmware error	FW detected an error which caused it to halt itself. Can also be errors generated by the kernel such as: bad exception, out of memory, unrecoverable errors.	
2010	HSM locked - tamper clear required	Limited set of FW commands available due to an HSM tamper condition. Tamper needs to be cleared before proceeding. Controlled tamper recovery must be enabled for this message to appear.	
2011	HSM unlocked - tamper clear done	Tamper was cleared when in controlled tamper recovery mode.	
2012	HSM in secure transport mode	Checked on every FW start-up to remind the user to do a recovery operation. Limited set of FW commands available.	
2013	HSM recovered from secure transport mode	HSM in secure transport mode was recovered back to normal mode.	

ALM ID	Alarm Message	Description	Info
2014	Auto-activation data invalid – HSM deactivated	FW checked auto-activation data validity and failed. Re-activation required.	
2015	Hard tamper - high temperature	(L3 only) HSM temperature was higher than 88C.	HT_T
2016	Hard tamper - low temperature	(L3 only) HSM temperature was lower than -40C.	LT_T
2017	Hard tamper - over voltage	(L3 only) HSM voltage was higher than the maximum allowed.	OV_T, TC3_T
2018	Hard tamper - oscillator failure	(L3 only) HSM tamper clock oscillator has failed	OSC_T
2019	Hard tamper - tamper configuration invalid	(L3 only) HSM tamper configuration lost (set to defaults) due to power loss.	FS_T
2020	Chassis opened	Chassis open switch (connector P7) has been triggered.	TC1_T
2021	HSM was removed from chassis	HSM was removed from host chassis just before this FW execution. HSM will be deactivated.	CS
2022	Decommission signal triggered	Decommission button (connector P9) has been pressed.	TC2_T
2023	HSM fan x failure	Fault detected in HSM on-board fan (fan 1 or fan 2).	
2024	Stored data integrity verify error	Integrity of an object or CSP did not verify correctly. See "Stored Data Integrity" on page 292.	
2025	Firmware update in progress	A firmware update procedure is in progress. Recorded in the logs, but not shown onscreen. [Added with firmware 7.7.0]	

ALM ID	Alarm Message	Description	Info
2026	Firmware update canceled	A firmware update procedure was halted due to insufficient memory to continue - the HSM rolls back to the previous f/w. [Added with firmware 7.7.0]	
2027	HSM storage exceeded	Attempt to use storage beyond the size of a partition (which was doubled with firmware 7.7.0) - the update proceeds to completion, but some restrictions apply to the affected partition -- see "Memory and partition and object interactions during and after firmware update " on page 1. This is recorded only in the logs, not onscreen, but a message "HSM storage is currently over capacity" is shown onscreen. [Added with firmware 7.7.0]	
2028	HSM capacity exceeded	Attempt to exceed the total memory size of the HSM cancels the operation. Refer to your backups. [Added with firmware 7.7.0]	

HSM Alarm Code Samples

This section shows the details of some of the alarm event scenarios.

ALM = alarm message.

Temperature - High Warning

If HSM temperature reaches 75 degrees Celsius and then drops back below 75C the following actions occur:

> Temperature >= 75C

- After 5 minutes at this temperature or higher, the Host Driver receives a 'High Temperature Warning' interrupt and issues an ALM
- Firmware checks temperature at start-up and once per hour
- Firmware issues ALM for high temperature warning activated

> Temperature < 75C

- Firmware issues ALM for high temperature warning deactivated

Temperature – High Soft Tamper

When the temperature starts below 75C and reaches the high soft tamper limit of 80C and then drops back below 75C the following actions occur:

> Temperature >= 75C
  - After 5 minutes at this temperature or higher, the Host Driver receives a High Temperature Warning interrupt and issues an ALM
  - Firmware issues ALM for activation of high temperature warning

> Temperature >= 80C
  - Soft Tamper reset – card put into reset. Stays in reset until temperature lowers.
  - Host Driver receives soft tamper interrupt and issues ALM (only one when soft tamper condition starts).

> Temperature < 80C
  - Bootloader issues soft tamper ALM, then an ALM that HSM restart is required and waits for host reset.
  - User receives ALM and goes to LunaCM/Lunash to do an “hsm restart” command.
  - Bootloader starts – jumps to firmware.
  - Firmware starts – no actions taken for the soft tamper. If temperature >= 75C, firmware re-issues ALM for activation of high temperature warning.

> Temperature < 75C
  - Firmware issues ALM for deactivation of high temperature warning.

Temperature – High Hard Tamper

When the temperature starts below 75C and reaches high hard tamper limit of 88C and then drops back below 75C the following actions occur:

> Same as soft tamper described above up to when card is held in soft tamper reset

> Temperature > 88C
  - Hard Tamper reset – Card in hard tamper reset for 5 seconds then returns to soft tamper reset. K7 HW does erase/reset of all internal temporary memory. Tamper chip latches time and type of tamper. Host driver receives hard tamper interrupt and issues ALM.
  - HSM also erases auto-activation and STM data in tamper chip
  - If decommission on tamper is enabled then key encryption data is erased in tamper chip as well

> Temperature < 80C
  - Bootloader starts – issues hard tamper ALM and logs it in FRAM Log
  - Bootloader issues ALM that HSM restart is required and waits for host reset.
  - User receives ALM and goes to LunaCM/Lunash to perform an **hsm restart** command.
  - Bootloader starts – jumps to firmware.
  - Firmware starts – saves hard tamper latches. If controlled tamper recovery is enabled, firmware locks HSM commands to a minimal subset only, and issues ALM for HSM locked. User must go to

LunaCM/Lunash and perform a “tamper clear” command to get a full HSM command set. When tamper clear is issued, firmware outputs an ALM for HSM unlocked.

- Firmware – issues deactivation and decommission (if enabled for tamper) ALMs
- Firmware - temperature >= 75C, firmware re-issues ALM for activation of high temperature warning

> Temperature < 75C

- Firmware issues ALM for deactivation of high temperature warning

> Temperature < 80C

- Bootloader starts – issues hard tamper ALM
- Bootloader erases all of flash except for Boot Loader area and issues ALM for 'HSM permanently tampered'
- Bootloader issues ALM that 'HSM restart is required' and waits for host reset.
- User receives ALM and goes to LunaCM/Lunash to do an “hsm restart” command.
- Bootloader starts – Only bootloader commands are available. Bootloader again issues 'ALM for HSM permanently tampered'. User can dump the FRAM Log using LUNADIAG.

Hard Tampers During Storage

When the HSM is powered off its tamper detection is powered by the on-card battery. Some hard tampers can occur when main power is not applied. The condition that caused the tamper might not be present (for example high or low temperature) when the HSM is powered back on, while others might never turn off (for example enclosure penetration, oscillator failure). If they occur while in storage, then after the HSM is powered up, the bootloader runs and logs the tamper events in FRAM Log and the serial port. Since the host K7 driver has not started yet, none of the messages from the bootloader are sent to the host, but other alarm messages are output later to notify the user.

- Bootloader waits for the host driver to be loaded
- When the host driver starts up it immediately resets the HSM causing the bootloader to run again
- Bootloader does not re-log the same tamper events
- Bootloader jumps to firmware which outputs the ALM for the tamper event. If controlled tamper recovery is enabled firmware also outputs an ALM for the 'HSM is locked and a tamper clear is required'. The user can then use LunaCM or Lunash to clear the tamper

> **NOTE** If needed, use the **lunadiag** utility to output the FRAM Log in order to determine the tamper information, or to pass on to Thales Technical Support.

Decommission with power on

If the HSM is powered on and a decommission is triggered either by the decommission switch or by a tamper (if decommission on tamper is enabled) then the HSM goes into reset for 5 seconds. The following alarm messages are output to FRAM Log, serial port, and host driver:

> The host driver immediately receives an interrupt and outputs an 'ALM for decommission triggered'

> After 5 seconds lapses, the bootloader starts running and also outputs an 'ALM for decommission triggered'

> Bootloader outputs an ALM for 'HSM restart required' and then waits

> User gets alarm notification and performs an HSM restart
> Bootloader restarts and jumps to firmware which finishes the decommission operations and firmware outputs an ALM for 'HSM decommissioned by firmware' and an ALM for 'HSM locked' (if enabled)

Decommission with power off

If the HSM is powered off and a decommission is triggered either by the decommission switch or by a tamper (if decommission on tamper is enabled) then the decommission is latched in the tamper chip. When the HSM is powered on the following alarm messages are output:

> Bootloader starts running and outputs an ALM for 'Decommission triggered' only to FRAM Log and serial port since the host driver is not loaded yet
> Bootloader waits for the driver to be loaded which then forces a host reset
> Bootloader restarts and jumps to firmware which finishes the decommission operations and firmware outputs an ALM for 'HSM decommissioned by firmware' and an ALM for 'HSM locked' (if enabled)

NOTE If needed, use the **lunadiag** utility to output the FRAM Log in order to determine the tamper information, or to pass on to Thales Technical Support.

Chassis open with power on

If the HSM is powered on and the chassis open switch triggered then a cold reset is performed on the HSM which effectively removes the HSM from the PCIe bus. After about 10 seconds the HSM is released from reset and the following alarm messages are output:

> Host Driver notices the device is no longer present on the PCIe bus and outputs an ALM for 'HSM missing from PCIe bus'
> Bootloader starts running and outputs an ALM for 'HSM chassis opened' only to FRAM Log and serial port
> Bootloader waits for the driver to be loaded
> User gets notification of missing HSM and powers off then on the host system
> Bootloader starts running and does not re-log the same tamper events
> Bootloader waits for the host driver to be loaded
> When the host driver starts up it immediately resets the HSM causing Bootloader to run again
> Bootloader jumps to firmware which finishes the chassis opened operations and firmware outputs an ALM for 'HSM chassis opened' and an ALM for 'HSM locked' (if enabled).

NOTE If the chassis is still open then the HSM performs a cold reset after the tampers are cleared by firmware.

If needed, use the **lunadiag** utility to output the FRAM Log in order to determine the tamper information, or to pass on to Thales Technical Support.

Chassis open with power off

If the HSM is powered off and the chassis open switch triggered then the chassis open is latched in the tamper chip. When the HSM is powered on the following alarm messages are output:

> Bootloader starts running and outputs an ALM for 'HSM chassis opened' only to FRAM Log and serial port
> Bootloader waits for the driver to be loaded which then forces a host reset
> Bootloader starts running and does not re-log the same tamper events
> Bootloader jumps to firmware which finishes the chassis opened operations and firmware outputs an ALM for 'HSM chassis opened' and an ALM for 'HSM locked' (if enabled)

NOTE If the chassis is still open then the HSM performs a cold reset after the tampers are cleared by firmware.

Card removal

When an HSM is powered off and removed from the chassis a card removal latch is saved in the tamper chip. When the HSM is powered on the following alarm messages are output:

> Bootloader starts running and outputs an ALM for 'card removal' only to FRAM Log and serial port
> Bootloader waits for the driver to be loaded which then forces a host reset
> Bootloader starts running and does not re-log the same tamper events
> Bootloader restarts and jumps to firmware which outputs an ALM for 'HSM was removed from the chassis' and an ALM for 'HSM locked' (if enabled)

NOTE If needed, use the **lunadiag** utility to output the FRAM Log in order to determine the tamper information, or to pass on to Thales Technical Support.

Stored Data Integrity

The HSM performs data integrity checks at startup and during runtime.

Startup

If a check fails during startup, meaning that an object stored in flash memory was corrupted, then ALM 2024 is generated, along with additional log messages, and the HSM firmware halts:

```
k7pf0: [HSM] ALM2024: Stored data integrity verify error
... additional messages that might include "LOG (SEVERE)" and "LOG (CRITICAL)", "Fatal
error", and possibly also
k7pf0: [HSM] ALM2009: HSM halted - internal firmware error
```

What to do

1. Restart the HSM.
2. If the ALM persists, cycle the power to the HSM.
3. If the ALM persists, zeroize the HSM.
4. If the ALM persists, contact Support.

Runtime

If a check fails during runtime, meaning that an object stored in volatile memory was corrupted, then ALM 2024 is generated, along with log messages, and the HSM is unable to perform any actions that involve the corrupted object:

```
k7pf0: [HSM] ALM2024: Stored data integrity verify error
... additional messages that might include "LOG (SEVERE)"
```

What to do

1. Try restarting the HSM.
2. If an SDI alarm occurs during startup, see the section about "Startup", above.
3. If no SDI alarm occurs during startup, but an SDI alarm occurs later, contact Support.

CHAPTER 13: HSM Updates and Upgrades

Thales releases periodic updates to the Luna PCIe HSM 7 firmware, as well as updated versions of the Luna HSM Client software. If you have recently purchased a new Luna PCIe HSM 7 and your organization requires FIPS certification, you can download and install a FIPS-validated version of the HSM firmware. You can download these updates as they become available from the Thales Customer Support Portal: https://supportportal.thalesgroup.com.

Depending on the model of Luna PCIe HSM 7 you selected at time of purchase, you may also be able to purchase upgrades to the HSM's capabilities.

The Customer Release Notes (CRN) contain important information on updates.

The following chapter provides tested update paths and procedures for installing update packages, as well as a list of the version dependencies for certain features. It contains the following sections:

> "Updating the Luna PCIe HSM 7 Firmware" below

> "Updating the Luna HSM Client Software" on page 87

> Updating the Luna Backup HSM 7 Firmware

> Updating the Luna Backup HSM G5 Firmware

> "Rolling Back the Luna HSM Firmware" on the next page

> "Upgrading HSM Capabilities" on page 296

Updating the Luna PCIe HSM 7 Firmware

To update the firmware on a Luna PCIe HSM 7, download the desired firmware version from the Thales Support Portal. Use LunaCM on the host workstation to apply the update. You require:

> Luna PCIe HSM 7 firmware update file (<filename>**.fuf**) and

> the firmware update authentication code file (<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 If you are updating to Luna HSM Firmware 7.7.0 or newer, refer to Special Considerations for Updating to Luna PCIe HSM 7 Firmware 7.7.0 or Higher before proceeding with the firmware update.

To update the Luna PCIe HSM 7 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.

   lunacm:> **role login -name so**
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 full filepaths.

   lunacm:> **hsm updatefw -fuf** <filename>**.fuf -authcode** <filename>**.txt**

Changing the Firmware Upgrade Permissions (Linux only)

By default, the root user and any user who is part of the **hsmusers** group can perform a firmware update. You can use this procedure to restrict firmware update operations to root only (that is, disable firmware update for members of the **hsmusers** group).

To restrict firmware update operations to the root user only

1. Open the the **/etc/modprobe.d/k7.conf** file for editing:

   sudoedit /etc/modprobe.d/k7.conf
2. Change the **k7_rootonly_reset** option from **0** to **1**. Save the file and exit the editor.
3. Stop any processes that are using the K7 driver. Typically this means stopping the **pedclient** service, and the **luna-snmp** service, if you are using SNMP.

   sudo systemctl stop pedclient_service

   sudo systemctl stop luna-snmp
4. Reload the driver:

   sudo systemctl reload k7

Rolling Back the Luna HSM Firmware

When updating the HSM firmware, the Luna PCIe HSM 7 saves the previously-installed firmware version on the HSM. If required, you can roll back to this previously-installed version. Rollback allows you to try firmware without permanently committing to the new version.

Rollback does not create a new rollback target; a single rollback target is preserved when a firmware update is performed. After a rollback operation, no further rollback is possible until the next firmware update saves the pre-update version as the new rollback target.

> **CAUTION!** Firmware rollback is destructive; earlier firmware versions might have older mechanisms and security vulnerabilities that a new version does not. Back up any important materials before rolling back the firmware. This procedure zeroizes the HSM and all cryptographic objects are erased.

> **NOTE** Firmware rollback is not supported on HSMs that use Functionality Modules. If you have ever enabled **HSM policy 50: Allow Functionality Modules**, even if the policy is currently disabled, you cannot roll back the HSM firmware. See "FM Deployment Constraints" on page 298 for details.

To roll back the Luna HSM firmware to the previous version

1. Check the previous firmware version that is available on the HSM.

   lunacm:> **hsm showinfo**

2. Back up any important cryptographic objects currently stored on the HSM (see Partition Backup and Restore).
3. At the LunaCM prompt, login as HSM SO.

   lunacm:> **role login -name so**

4. Roll back the HSM firmware.

   lunacm:> **hsm rollbackfw**

   LunaCM performs an automatic restart following the rollback procedure.

5. Re-initialize the HSM and restore your partition from backup.

Upgrading HSM Capabilities

A Secure Capability Upgrade for Luna PCIe HSM 7 is delivered to you as a downloaded file set. Follow the FTP instructions in the email you received from Thales Technical Support and unzip the files to the host workstation. The update procedure is similar to the procedure for firmware updates.

> **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.

You require:

> the Luna PCIe HSM 7 capability upgrade file (<filename>**.cuf**)

> the capability update authentication code file (<filename>**.txt**)

Installing the Capability Upgrade

Once the files are unpacked and available on the host workstation, open a command-prompt session.

To install the upgrade package

1. Navigate to the Luna HSM Client directory and launch LunaCM.
2. Log in as HSM SO.

   lunacm:> **role login -name so**
3. Apply the new capability by specifying the upgrade file and the file containing the authorization code. If the files are not located in the Luna Network HSM Client directory, specify the filepaths.

   lunacm:> **hsm updatecap -cuf** <upgrade_file> **-authcode** <authcode_file>
4. Check that the new capability is in place.

   lunacm:> **hsm showpolicies**

CHAPTER 14: Functionality Modules

Functionality Modules (FMs) consist of your own custom-developed code, loaded and operating within the logical and physical security of a Luna PCIe HSM 7 as part of the HSM firmware. FMs allow you to customize your Luna PCIe HSM 7's functionality to suit the needs of your organization. Custom functionality provided by your own FMs can include:

> new cryptographic algorithms

> security-sensitive code, isolated from the rest of the HSM environment

> keys and critical parameters managed by the FM, independent from standard PKCS#11 objects, held in tamper-protected persistent storage

To create FMs, you will need the Functionality Module Software Development Kit (SDK), which is included with the Luna HSM Client software. Applications that use FM functions are supported on Windows and Linux.

This chapter describes how to prepare the Luna PCIe HSM 7 to use FMs, and manage FMs on the HSM. For detailed information on the FM architecture and how to use FMs with your applications, refer to About the FM SDK Programming Guide.

> **NOTE** This feature requires minimum Luna HSM Firmware 7.4.0 and Luna HSM Client 7.4.0.
> This feature has hardware dependencies described in "Preparing the Luna PCIe HSM 7 to Use FMs" on page 301.

This chapter contains the following sections:

> "FM Deployment Constraints" below

> "Preparing the Luna PCIe HSM 7 to Use FMs" on page 301

> "Building and Signing an FM" on page 304

> "Loading an FM Into the HSM Firmware" on page 307

> "Deleting an FM From the HSM Firmware" on page 308

> "Recovering the HSM After FM Failure" on page 309

FM Deployment Constraints

This section describes important considerations and constraints associated with deploying your Functionality Modules (FMs). Your Luna PCIe HSM 7 must meet all the criteria described in "Preparing the Luna PCIe HSM 7 to Use FMs" on page 301.

Introducing FMs into your Luna PCIe HSM 7 deployment will change the functionality of certain HSM features. Please take the following constraints into consideration before using FMs:

> "FMs and FIPS Mode" on the next page

> "FMs and High-Availability (HA)" on the next page

> "FMs and Backup/Restore/Cloning" on the next page
> "FMs and HSM Firmware Rollback" on the next page
> "FM Configuration and Remote PED" on the next page
> "FM-Enabled HSM Cannot be Verified With CMU" on page 301
> "Key Attributes" on page 301
> "No EDDSA or EC_MONTGOMERY Private Keys with C_CreateObject" on page 301
> "FM Sample Applications Dependent on General Cryptoki Samples" on page 301
> "Memory for FMs" on page 301

CAUTION! Enabling FMs (**HSM policy 50**) introduces changes to Luna HSM functionality, some of which are permanent; they cannot be removed by disabling the policy. FM-enabled status is **not** reversible by Factory Reset.

If you are using Crypto Command Center, ensure that your CCC version supports FM-enabled HSMs before you enable **HSM policy 50**. Refer to the CCC CRN for details.

FMs and FIPS Mode

FMs change the abilities of the HSM firmware, adding new cryptographic algorithms or other functions. Since the new functionality is not certified by NIST, be sure that your FM does not break FIPS compliance. To be certain that your organization is meeting FIPS requirements, ensure that you are using a FIPS-certified version of the Luna HSM firmware, and that your Luna PCIe HSM 7 has the following HSM policy settings:

> **HSM policy 12: "Allow non-FIPS algorithms" on page 212**: **0**
> **HSM policy 50: "Allow Functionality Modules" on page 218**: **0**

NOTE Using Luna HSM Firmware 7.4.2 and older, this restriction is enforced; it is not possible to set **HSM policy 50: "Allow Functionality Modules" on page 218** to **1** while **HSM policy 12: "Allow non-FIPS algorithms" on page 212** is **0**. Using newer firmware versions, it is possible to enable FMs in FIPS mode, but your FM functionality may not be FIPS-compliant; refer to NIST standards to ensure compliance.

If FIPS compliance is not required, then enabling FMs does not present an issue for you. Enabling Functionality Modules (setting **HSM policy 50: "Allow Functionality Modules" on page 218** to **1**) is not reversible. For more information about HSM policies, see "HSM Capabilities and Policies" on page 210.

FMs and High-Availability (HA)

FM-specific functions must specify the exact HSM that will handle the operations. Therefore, the Luna HSM Client's HA implementation currently cannot accommodate FM functionality. If you want your FM-specific operations to be load-balanced across multiple HSMs, you must program this functionality into your applications yourself.

HA will still work with standard Luna operations.

For HA to function with Functionality Modules, all HSMs with application partitions in the HA group must have the same algorithms and functionality available. If one member partition does not have a required algorithm available in HSM firmware, cryptographic objects using that algorithm cannot be cloned to that partition, and this will disrupt HA functions.

Therefore, all HSMs containing HA group members must have FMs enabled (as described in "Preparing the Luna PCIe HSM 7 to Use FMs" on the next page), and they must all have the same FM(s) loaded. HA login requires two FM-enabled HSMs.

For more information about HA, see High-Availability Groups.

FMs and Backup/Restore/Cloning

To back up and restore objects on FM-enabled partitions, you require the following minimum Luna Backup HSM firmware versions:

> Luna Backup HSM 7 Firmware 7.7.1

> Luna Backup HSM G5 Firmware 6.28.0

As a general rule, cryptographic objects can be cloned from a partition with less-secure settings to one with identical or more secure settings. Therefore, it is not possible to clone objects from a standard partition to an FM-enabled partition.

To back up keys stored in the SMFS, your application must provide all the functions to back up and restore these keys.

FMs and HSM Firmware Rollback

Enabling **HSM Policy 50** permanently disables the ability to roll back the HSM firmware to a version older than Luna HSM Firmware 7.4.0. Attempting to roll back the firmware once **HSM policy 50** has been enabled will return the following error:

```
Error in execution: CKR_OPERATION_NOT_ALLOWED.

Command Result : 0x80000030 (CKR_OPERATION_NOT_ALLOWED)
```

FM Configuration and Remote PED

Various FM functions require HSM resets (for example, creating a partition or enabling an FM).

If you are configuring FMs while authenticating with Remote PED, the Remote PED connection is broken with each reset. LunaCM continues to show an active Remote PED connection until you restart LunaCM. You must close that apparent connection with lunacm:> ped disconnect and then open it again with lunacm:> ped connect before you can resume remote configuration.

This might be required several times during Luna PCIe HSM 7 setup for FMs. To prevent this, enable **HSM Policy 51:** "Allow SMFS Auto Activation" on page 219. If SMFS is not auto-activated, then the SMFS will require further individual PED prompts during the configuration process (SMFS is deactivated upon HSM reset if SMFS auto-activation is off).

> **NOTE** Thales recommends that first time configuration of FM's be done locally, to minimize the issues mentioned above.

FM-Enabled HSM Cannot be Verified With CMU

The FM-enabled Luna PCIe HSM 7 does not currently support confirming the HSM's authenticity using cmu verifyhsm, as described in Verifying the HSM's Authenticity, or retrieving and confirming a Public Key Confirmation from the HSM using cmu getpkc and cmu verifypkc.

Key Attributes

On an HSM with FMs enabled, keys that are derived or generated have the "always-sensitive" and the "never-extractable" attributes set to "false".

No EDDSA or EC_MONTGOMERY Private Keys with C_CreateObject

This release of the Luna PCIe HSM 7 firmware does not allow FMs to use C_CreateObject to create EDDSA or EC_MONTGOMERY private keys. Use C_GenerateKeyPair to create these types of key.

FM Sample Applications Dependent on General Cryptoki Samples

When you install the FM SDK, the installation script ensures that the general Luna (PKCS) SDK and samples are also installed (first). This satisfies source dependencies for the FM samples. If you later delete or remove the Luna SDK, you might break those dependencies, and the FM samples will not build. You can manually correct this by performing a manual **rpm -i** of the cksample package.

Memory for FMs

Multiple FMs can be loaded into the FM space of the HSM, with a total memory limit of:

> 8 megabytes for FMs

> 4 megabytes of SMFS

Unused FMs can be deleted, to free some memory space.

Preparing the Luna PCIe HSM 7 to Use FMs

This section provides information on how to prepare your Luna PCIe HSM 7 to accept Functionality Modules (FMs). FMs require a specific factory configuration, the correct firmware version, a license upgrade, and the correct policy settings, as described below:

> "Step 1: Ensure You Have FM-Ready Hardware" on the next page

> "Step 2: Update Your HSM" on the next page

> "Step 3: Purchase and Apply the FM Capability License" on the next page

> "Step 4: Apply HSM Policy Settings" on page 303

> **CAUTION!** Enabling FMs (**HSM policy 50**) introduces changes to Luna HSM functionality, some of which are permanent; they cannot be removed by disabling the policy. FM-enabled status is **not** reversible by Factory Reset. Refer to "FM Deployment Constraints" on page 298 for details before enabling.
>
> If you are using Crypto Command Center, ensure that your CCC version supports FM-enabled HSMs before you enable **HSM policy 50**. Refer to the CCC CRN for details.

Step 1: Ensure You Have FM-Ready Hardware

The FM feature requires a specific Luna PCIe HSM 7 hardware configuration that must be created by Thales at the factory. Luna PCIe HSM 7s that have this configuration are "FM-ready". If your Luna PCIe HSM 7 is not FM-ready, contact your Thales representative or Thales Customer Support for further guidance.

Determining Whether the HSM is FM-Ready

Currently, all Luna PCIe HSM 7s are FM-ready from the factory. HSMs older than this factory update are not. To determine if your HSM is FM-ready, check the Product Part # on the PCIe card label:

If the last 3-digit section of the Product Part # is **003** or higher, your HSM is FM-ready. If **002** or lower, contact your Thales representative or Thales Customer Support for guidance on how to obtain FM-ready hardware.

Step 2: Update Your HSM

To use FMs, you require Luna HSM Firmware 7.4.0 or newer. You can download the latest software/firmware packages from the Thales Support Portal (see "Updating the Luna PCIe HSM 7 Firmware" on page 294).

When you have completed the upgrade, you can check the output from lunacm:>hsm **showinfo** to ensure that the HSM is FM-ready:

```
FM HW Status ->         FM Ready
Firmware Version -> 7.4.0
```

Step 3: Purchase and Apply the FM Capability License

To use FMs, contact your Thales sales representative to purchase the FM capability license. The FM license is delivered as a **.cuf** file that is specific to your HSM serial number. Refer to "Upgrading HSM Capabilities" on page 296 for the procedure.

When you have activated your license on the HSM, you can use lunacm:>**hsm showinfo** to check that it is installed:

```
License Count -> 8
        1. 621000068-000 K7 Base
        2. 621010185-003 Key backup via cloning protocol
        3. 621000134-002 Enable 32 megabytes of object storage
        4. 621000135-002 Enable allow decommissioning
        5. 621000021-002 Maximum performance
        6. 621000138-001 Controlled tamper recovery
        7. 621000154-001 Enable decommission on tamper with policy off
        8. 621000074-001 Enable Functionality Modules
```

Step 4: Apply HSM Policy Settings

Applying the FM capability license allows you to set 4 new HSM policies that affect FMs on the Luna PCIe HSM 7 (see "HSM Capabilities and Policies" on page 210). Use lunacm:>**hsm showpolicies** to list HSM policies.

```
50: Allow Functionality Modules : 0
51: Allow SMFS Auto Activation : 0
52: Restrict FM Privilege Level : 0
53: Encrypt keys passing from FM to HSM : 0
```

HSM Policy 50: Allow Functionality Modules

With this policy enabled, Functionality Modules may be loaded to the HSM, permitting custom cryptographic operations. Allows use of the **ctfm** utility and FM-related commands, and the use of Functionality Modules in general with this HSM.

The HSM SO must set HSM policy 50 to 1 (ON) to use FMs on the Luna PCIe HSM 7. Changing this policy (OFF-to-ON or ON-to-OFF) will zeroize the HSM and it must be re-initialized.

> **CAUTION!** Enabling FMs (**HSM policy 50**) introduces changes to Luna HSM functionality, some of which are permanent; they cannot be removed by disabling the policy. FM-enabled status is **not** reversible by Factory Reset. Refer to "FM Deployment Constraints" on page 298 for details before enabling.
>
> If you are using Crypto Command Center, ensure that your CCC version supports FM-enabled HSMs before you enable **HSM policy 50**. Refer to the CCC CRN for details.

> **NOTE** After setting HSM policy 50, you must add the following entry to the **Chrystoki.conf/crystoki.ini** configuration file before you can re-initialize the HSM:
>
> ```
> [Misc]
> LoginAllowedOnFMEnabledHSMs=1
> ```

HSM Policy 51: Allow SMFS Auto Activation

With this policy enabled, the Secure Memory File System (SMFS) is automatically activated on startup, providing a secure, tamper-enabled location in the HSM memory where Functionality Modules can load keys and parameters. Auto-activation for SMFS, like auto-activation for multifactor quorum-authenticated partitions in general, persists through a power outage of up to 2 hours duration.If disabled, the HSM SO must manually activate the SMFS each time the HSM reboots or loses power.

Thales recommends setting HSM policy 51 to 1 (ON) to avoid having to manually re-activate the SMFS if you need to reboot the HSM. Changing this policy destroys all existing application partitions.

HSM Policy 52: Restrict FM Privilege Level

With this policy enabled, FM privilege is restricted. By default, FM privilege permits FMs to see the sensitive key attributes (including key values) of cryptographic objects on application partitions. This privilege is necessary for most FMs, so that the Crypto Officer (CO) and Crypto User (CU) roles can use partition objects with the FM. However, some FMs might not require this privilege and it can be restricted to satisfy some certification requirements (such as Common Criteria).

FM privilege permits FMs to see the sensitive key attributes (including key values) of cryptographic objects on application partitions. This privilege is necessary for most FMs, so that the Crypto Officer (CO) and Crypto User (CU) roles can use partition objects with the FM. However, some FMs might not require this privilege and it can be restricted to satisfy some certification requirements (such as Common Criteria).

Unless you require CC certification, Thales does not recommend changing this policy from its default setting (OFF). Changing this policy destroys all existing application partitions.

HSM Policy 53: Encrypt Keys Passing from FM to HSM

With this policy enabled, keys created by an FM are encrypted before crossing from the FM to the Functionality Module Crypto Engine interface (FMCE). This internal encryption may be required to satisfy some certification requirements (such as Common Criteria).

Unless you require CC certification, Thales does not recommend changing this policy from its default setting (OFF). Changing this policy (OFF-to-ON or ON-to-OFF) will destroy all existing application partitions.

Building and Signing an FM

Once you have written your FM code, you must build the binary and then sign it using a private key on the HSM. A self-signed certificate is used to confirm the authenticity of the FM. This procedure will allow you to install the FM into your HSM firmware. Luna FMs must be built on a Linux system, so you can use the native **make** command. The following example uses the **skeleton** sample FM, included with the Luna FM SDK.

The FM binary must be signed with a private key, and loaded into the HSM firmware with a self-signed certificate from the same keypair to verify its authenticity. You can use **mkfm**, included with the Luna HSM Client FM Tools, to sign your FM using a Luna application partition or your own Cryptoki signing station. The procedure below will show you how to use **mkfm**.

Prerequisites

> The FM binary must be built on a Linux client. You can use either a Windows or Linux client to perform the signing operation.

> The FM Tools option in the Luna HSM Client software must be installed on the client or signing station.

> **mkfm** requires access to a Cryptoki token (such as a Luna application partition) capable of using the CKM_SHA512_RSA_PKCS mechanism.

To build an FM binary

1. On your Linux client, navigate to the directory containing your FM code (<filename>**.c**). By default, FM samples provided with the Luna FM SDK are installed in **/usr/safenet/lunafmsdk/samples/**.

```
[user@myLunaClient ~]# cd /usr/safenet/lunafmsdk/samples/skeleton/fm/
[user@myLunaClient fm]# ls
hdr.c  makefile  skeleton.c
```

2. Use the Linux **make** command to build the FM binary.

   # make

   The **make** process creates two new sub-directories, **bin-ppc** and **obj-ppc**. Your FM binary is located in **bin-ppc**, named <filename>**.bin**.

```
[user@myLunaClient ~]# cd /usr/safenet/lunafmsdk/samples/skeleton/fm/bin-ppc/
[user@myLunaClient bin-ppc]# ls
skeleton.bin
```

To create an FM signing certificate on an application partition

1. If this is the first FM you are signing, you must first create a keypair and self-signed certificate on the application partition. If you already have a certificate for FM signing stored on the HSM, skip this procedure.

   > **NOTE** A certificate used to sign an FM must have attribute CKA_PRIVATE set as true.
   >
   > If an existing certificate has Private=F, you can use the CMU tool to export that cert, then re-import it while setting **-private=T**.
   >
   > Or, if the partition retains the FM signing keypair, you can run cmu selfsigncertificate again to re-create the certificate, this time setting **-private=T** explicitly.

   To sign an FM with **mkfm**, you must use an RSA private key at least 2048 bits long. The Crypto Officer can use the **cmu** utility to create the keypair. You will be prompted for the CO credential.

   > **NOTE** Always provide unique labels for your keys. If multiple private keys exist with the same label, **mkfm** will use the newest key (with the greatest object handle value).

   cmu generatekeypair -labelpublic=<public_key_label> **-labelprivate=**<private_key_label> **-keytype=rsa -sign=1 -verify=1**

   ```
   [user@myLunaClient bin]# ./cmu generatekeypair -labelpublic=FMpub -labelprivate=FMpriv -
   keytype=rsa -sign=1 -verify=1
   Certificate Management Utility (64-bit) v7.4.0-208. Copyright (c) 2018 SafeNet. All rights
   reserved.

   Select token
    [3] Token Label: myPartition
    [4] Token Label: myPCIeHSM
    Enter choice: 3
   Please enter password for token in slot 3 : ********


   Select RSA Mechanism Type -
   [1] PKCS [2] FIPS 186-3 Only Primes [3] FIPS 186-3 Auxiliary Primes : 2
   Enter modulus length (8 bit multiple) : 2048
   ```

2. Check the contents of the partition to find the key handles.

   cmu list

   ```
   [user@myLunaClient bin]# ./cmu list
   Certificate Management Utility (64-bit) v7.4.0-208. Copyright (c) 2018 SafeNet. All rights
   reserved.

   Select token
    [3] Token Label: myPartition
    [4] Token Label: pcie7pwd45
   Enter choice: 3
   Please enter password for token in slot 3 : ********
   ```

```
handle=48       label=FMpriv
handle=45       label=FMpub
```

3. Create a self-signed certificate on the partition by specifying a label, the public and private key handles, and any other attributes you wish to assign. You are prompted for required attributes (Common Name, serial number, start/end dates) that you do not specify.

   cmu selfsigncertificate -slot <slot_number> **-label** <cert_label> **-publichandle=**<handle> **-privatehandle=**<handle>

```
[user@myLunaClient bin]# ./cmu selfsigncertificate -slot 3 -publichandle=45 -
privatehandle=48 -label FMsign
Certificate Management Utility (64-bit) v7.4.0-208. Copyright (c) 2018 SafeNet. All rights
reserved.

Please enter password for token in slot 3 : ********

Enter certificate serial number : 1
Enter Subject 2-letter Country Code (C) : CA
Enter Subject State or Province Name (S) : ON
Enter Subject Locality Name (L) : Ottawa
Enter Subject Organization Name (O) : Thales
Enter Subject Organization Unit Name (OU) :
Enter Subject Common Name (CN) : FMsign
Enter EMAIL Address (E) :
Enter validity start date
 Year   : 2018
 Month  : 12
 Day    : 05
Enter validity end date
 Year   : 2019
 Month  : 12
 Day    : 31
Using "CKM_SHA256_RSA_PKCS" Mechanism
```

4. Export the certificate to the host file system, specifying the desired filename with **.cert** extension.

   cmu export -slot <slot_number> **-label** <cert_label> **-outputfile=**<filename.cert>

```
[user@myLunaClient bin]# ./cmu export -slot 3 -label FMsign -outputfile=FMsign.cert
Certificate Management Utility (64-bit) v7.4.0-208. Copyright (c) 2018 SafeNet. All rights
reserved.

Please enter password for token in slot 3 : ********
```

To sign an FM

1. Use the **mkfm** utility included with the Luna HSM Client FM Tools to sign the FM, specifying the unsigned FM binary, the desired FM filepath/filename (with **.fm** extension), the slot number/name of the partition/token where the keypair is stored, and the private key label.

   If you are specifying a slot number, include **-k SLOTID=**<#> instead of the partition name. If you are using a Cryptoki signing station other than a Luna 7.x application partition, include the **-c** option. You are prompted for the partition/token credential. By default, the Crypto Officer role is used; to use the Crypto User role instead, include the **-u** option.

   mkfm -f <filepath/name>**.bin -o** <filepath/name>**.fm -k** <token_or_partition_name/<private_key_label> [**-c**] [**-u**]

```
[root@k7tower bin-ppc]# ./mkfm -f /usr/safenet/lunafmsdk/samples/skeleton/fm/bin-
ppc/skeleton.bin -o /usr/safenet/lunafmsdk/samples/skeleton/fm/bin-ppc/skeleton.fm -k
myLunaPartition/FMpriv
Luna Functionality Module Signer Utility (64-bit) v7.4.0-208. Copyright (c) 2018 SafeNet.
All rights reserved.


Please Enter the PIN: (for user 'co' on slot 3) ********

mkfm: Processing ELF file /usr/safenet/lunafmsdk/samples/skeleton/fm/bin-ppc/skeleton.bin

File successfully signed
```

The signed FM is now located in the directory you specified:

```
[user@myLunaClient ~]# cd /usr/safenet/lunafmsdk/samples/skeleton/fm/bin-ppc/
[user@myLunaClient bin-ppc]# ls
skeleton.bin skeleton.fm
```

Next, see "Loading an FM Into the HSM Firmware" below.

Loading an FM Into the HSM Firmware

A signed FM must be loaded into the HSM firmware to provide new functionality. The HSM SO can load FMs using the **ctfm** tool provided with the Luna HSM Client software and the following procedure.

> **NOTE** A certificate used to sign an FM must have attribute CKA_PRIVATE set as true.
>
> If an existing certificate has Private=F, you can use the CMU tool to export that cert, then re-import it while setting **-private=T**.
>
> Or, if the partition retains the FM signing keypair, you can run cmu selfsigncertificate again to re-create the certificate, this time setting **-private=T** explicitly.

Prerequisites

- Your HSM must meet the criteria described in "Preparing the Luna PCIe HSM 7 to Use FMs" on page 301.
- **HSM policy 50: Allow Functionality Modules** must be enabled.
- **HSM policy 51: Enable SMFS Auto Activation** must be enabled, if you intend to use auto-activation (recommended). Changing this policy later will erase all partitions and installed FMs.
- Ensure that all destructive policies are set before you load FMs into the HSM firmware. Any change of a destructive policy will erase all loaded FMs.
- The FM must be signed as described in "Building and Signing an FM" on page 304, using Luna HSM Client 7.4.0 or newer. FMs built using the Luna 7.0.4 Tech Preview release are not compatible with this Luna version.
- You require the FM signing certificate. If you have previously loaded an FM signed by the same key, the correct certificate is already present in the HSM Admin partition.

> **NOTE** If you load an FM with the same FM ID as an already-loaded FM, it is considered an update, and replaces the existing FM.

To load an FM into the HSM firmware

1. Use **ctfm** on the Luna PCIe HSM 7 host workstation to load the FM, specifying filepaths for the FM and the signing certificate. If you have previously loaded an FM signed by the same private key, the certificate is already stored on the HSM Admin partition, and you only need to specify the certificate label. If you have more than one Luna PCIe HSM 7 installed, specify the Admin partition slot number for the desired HSM. You are prompted for the HSM SO credential.

   ctfm i -f <filepath/fm_filename>**.fm** {**-c** <filepath/cert_filename>**.cert** | **-l** <stored_cert_label>} [**-s** <slot_number>]

2. Reset the HSM.

   lunareset <dev_path>

   lunacm:> **hsm restart**

   > **NOTE** If you have FMs loaded, you must restart the HSM whenever you perform any of the following operations:
   > > create a new partition (even if it has the same slot number as a recently-deleted partition),
   > > make a destructive change like re-initializing or zeroizing the HSM, or changing a destructive policy.
   > You will be unable to use the loaded FMs with new partitions until you restart the HSM. Use lunacm:> **hsm restart** or the **lunareset** utility.

3. Activate the Secure Memory File System (SMFS). You are prompted for the HSM SO credential.

   ctfm a

4. [Optional] Confirm the FM status.

   ctfm q

Deleting an FM From the HSM Firmware

This procedure allows the HSM SO to delete a specified FM from the HSM firmware using the **ctfm** tool provided with the Luna HSM Client software.

> **NOTE** If you are replacing the currently-loaded FM with an updated version, you do not need to delete the old version. If the new version has the same FM ID, it will replace the original version in the HSM firmware (see "Loading an FM Into the HSM Firmware" on the previous page).

In addition to the procedure below, other actions can cause FMs to be deleted from the HSM and the SMFS to be erased. See "Effects of Administrative Actions on Functionality Modules" on page 310.

Prerequisites

> You require the FM ID of the FM you wish to delete.

To delete an FM from the HSM firmware

1. [Optional] Use **ctfm** to list the FMs currently loaded on the HSM and see the desired FM ID.

   ctfm q

2. Delete the FM by specifying its FM ID. You are prompted for the HSM SO credential.

   ctfm d -i <FM_ID>

3. [Optional] Check the FM status again. The deleted FM's status is listed as "Zombie". At this point the FM is disabled, and its data will be fully deleted the next time you restart the HSM.

   ctfm q

```
[user@myLunaClient bin]# ./ctfm
Luna Functionality Module Management Utility (64-bit) v7.4.0-208. Copyright (c) 2018
SafeNet. All rights reserved.

Getting status of the FM on all available devices


Current Functionality Module Configuration for device 0:
Serial # : 67842
Model    : Luna K7
SMFS     : Activated

FM Label      : skeleton
FM ID         : a000
Version       : 1.01
Manufacturer  : Safenet Inc.
Build Time    : Wed Dec  5 14:44:47 2018 - EST
Fingerprint   : 78 7C E3 C2 01 54 B3 99 08 59
ROM size      : 7302
Status        : Zombie (reboot HSM to cleanup)
Startup Status: OK
```

4. Launch LunaCM, change the active slot to the Admin partition, and restart the HSM.

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

   lunacm:> **hsm restart**

Recovering the HSM After FM Failure

In the event that an FM bug causes problems on the HSM, such as halting the HSM or other functionality issues, the HSM SO can take steps to recover the HSM. If you have important FM key objects stored in the Secure Memory File System (SMFS), you may be able to regain access to them. If you encounter issues with FM functionality, try the following before you proceed with recovery operations:

1. Debug your FM code. Build and sign the FM ("Building and Signing an FM" on page 304), and attempt to load it onto the HSM ("Loading an FM Into the HSM Firmware" on page 307). Loading an updated FM with the same FM ID will erase the old version and replace it.
2. If this does not fix the problem, or you are unable to load the patched FM, delete the old FM first ("Deleting an FM From the HSM Firmware" on the previous page).
3. If this does not work, continue to the recovery procedure below.

The Luna HSM Client FM Tools include **fmrecover**, which allows you to delete all FMs currently loaded on the HSM, erase the SMFS, or both. This provides a last resort for recovering HSM functionality when an FM causes a failure.

Prerequisites

> Try the methods above before continuing. If you are running multiple FMs, it may be simpler to delete and replace the one that is causing the issue.

To recover the HSM after FM failure

1. Erase all FMs currently loaded on the HSM. This will leave the SMFS intact and preserve any key material you may have stored there. You must specify the Luna PCIe HSM 7 device node:

   fmrecover --fm <K7_node>

   You may now attempt to load a patched version of your FM that addresses the cause of the issue. If this does not resolve the problem, continue to step 2.

2. Erase the SMFS.

   > **CAUTION!** This will erase any cryptographic objects you have stored in the SMFS. If this is important key material, erasing the SMFS is a last resort to restore HSM functions.

   fmrecover --smfs <K7_node>

3. Load your patched FM and restart the SMFS (see "Loading an FM Into the HSM Firmware" on page 307).

Effects of Administrative Actions on Functionality Modules

Action	Deletes FMs
Destructive HSM Policy	Yes
Zeroize on 3 bad SO attempts	No
hsm zeroize command	No
hsm factoryReset command	Yes
Decommission	Yes
hsm init when already initialized	No
Destructive CUF application	Yes
NOTE: In all the above cases, the Secure Memory File System is re-initialized, destroying all contents.	

> **NOTE** Ensure that all destructive policies are set before you load FMs into the HSM firmware. Any change of a destructive policy will erase all loaded FMs.

CHAPTER 15: Zeroizing or Resetting the HSM to Factory Conditions

During the lifetime of a Luna HSM, you might have cause to take the HSM out of service, and wish to perform actions to ensure that no trace of your sensitive material remains. Those events might include:

- Placing the unit into storage, perhaps as a spare
- Shipping to another location or business unit in your organization
- Shipping the unit back to Thales for repair/re-manufacture
- Removing the HSM permanently from operational use, for disposal at end-of-life

This chapter describes the available options in the following sections:

- "HSM Zeroization" below
- "Resetting the Luna PCIe HSM 7 to Factory Condition" on page 314
- "Decommissioning the HSM Card" on the next page
- "Comparing Zeroize, Decommission, and Factory Reset" on page 315
- "Comparison of Destruction/Denial Actions" on page 315
- "Stored Data Integrity" on page 317
- "Effects of Administrative Actions on Functionality Modules" on page 310
- "RMA and Shipping Back to Thales" on page 317
- "End of Service and Disposal" on page 318

HSM Zeroization

In the context of HSMs in general, the term "zeroize" means to erase all plaintext keys. Some HSMs keep all keys in plaintext within the HSM boundary. Luna HSMs do not.

In the context of Luna HSMs, keys at rest (keys or objects that are stored in the HSM/cryptographic module) are encrypted. Keys are decrypted into a volatile working memory space inside the HSM only while they are being used. Items in volatile memory disappear when power is removed. The action that we loosely call "zeroizing", or clearing, erases volatile memory as well as destroying the key that encrypts stored objects.

Any temporarily decrypted keys are destroyed, and all customer keys on the HSM are immediately rendered inaccessible and unrecoverable whenever you:

- perform **hsm factoryreset**
- make too many bad login attempts on the SO account
- short the pins of the decommission header
- set a "destructive" HSM policy

> perform HSM firmware rollback

The KEK (key encryption key that encrypts all user objects, partition structure, cloning vectors, masking vectors, etc.) is destroyed by a zeroization (erasure) or decommission event. At that point, any objects or identities in the HSM become effectively random blobs of bits that can never be decoded.

> **NOTE** The next HSM power-up following a KEK zeroization automatically erases the contents of user storage, which were already an indecipherable blob without the original KEK. That is, any zeroizing event instantly makes encrypted objects unusable, and as soon as power is re-applied, the HSM immediately erases even the encrypted remains before it allows further use of the HSM.

The HSM must now be re-initialized in order to use it again, and initialization overwrites the HSM with new user parameters. Everything is further encrypted with a new KEK unique to that HSM.

Keys not encrypted by the KEK are those that require exemption and are not involved in user identities or user objects:

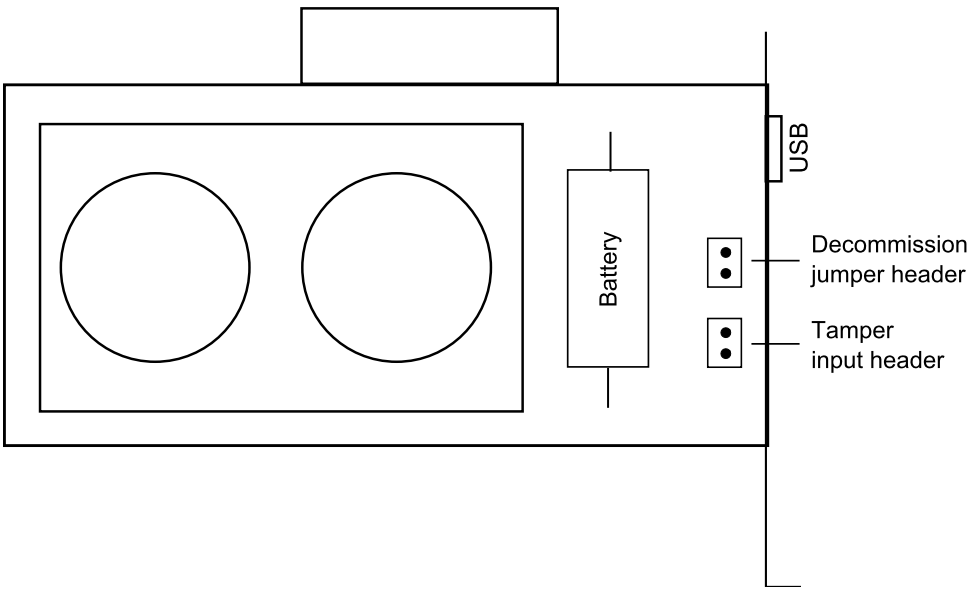
> The Master Tamper Key, which enables tamper handling

> The Remote PED Vector, to allow Remote PED-mediated recovery from tamper or from Secure Transport Mode

> The hardware origin key that certifies the HSM hardware as having been built by Thales

Decommissioning the HSM Card

The Luna PCIe HSM 7 is equipped with a two-pin decommission jumper header, as illustrated below.



By default, short-circuiting the decommission jumper header decommissions the HSM. You can use the blade of a screwdriver, or other conductive tool to short-circuit the two pins of the decommission header, or you can connect a switch to the decommission header if desired. Power is not required to decommission the HSM, that is, you can decommission the HSM after removing it from the chassis.

When you decommission a Luna PCIe HSM 7, the HSM is zeroized, all user accounts are deleted, and the HSM is returned to its factory state. Any firmware or partition upgrade packs installed on the HSM are retained.

You can also set **HSM Policy 40: Decommission on Tamper** to automatically decommission the HSM for selected tamper events. See "Tamper Events" on page 229 for details.

Disabling Decommissioning

You can disable the decommissioning feature if desired, by enabling **HSM Policy 46: Disable Decommission** (see "HSM Capabilities and Policies" on page 210). The primary reason for disabling decommissioning is to prevent the HSM from being automatically decommissioned due to loss of battery (see "Tamper Events" on page 229). If decommissioning is disabled, the Luna PCIe HSM 7 has an indefinite shelf life, as far as the battery is concerned.

To disable decommissioning

1. Launch LunaCM and log in as HSM SO.

   lunacm:> role login **-name so**

2. Enable **HSM Policy 46: Disable Decommission**:

   lunacm:> hsm changehsmpolicy **-policy 46 -value 1**

Resetting the Luna PCIe HSM 7 to Factory Condition

These instructions will allow you to restore your Luna PCIe HSM 7 to its original factory configuration. The HSM is zeroized, all partitions erased, and HSM policies are returned to their default settings. If you have performed firmware updates, those remain in place, and are not affected by this procedure.

To roll back the HSM firmware to the previous version, see "Rolling Back the Luna HSM Firmware" on page 295.

For eIDAS compliance, **hsmrecover** function is added to factoryreset commands - see "Stored Data Integrity" on page 317.

The standalone **hsmrecover** tool in the tools folder performs the same action, but can present additional messages that might be useful to Support engineers.

Prerequisites

> If you have STC enabled on the HSM, disable it by turning off **HSM policy 39** before continuing (see "Setting HSM Policies Manually" on page 223).

To reset the HSM to factory condition

1. Set the active slot to the admin partition.

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

2. Reset the HSM to factory settings.

   lunacm:> **hsm factoryreset**

Comparing Zeroize, Decommission, and Factory Reset

You can clear the contents of your Luna HSM, or the HSM may be cleared in response to an event. How this affects the contents and configuration of your HSM depends on whether the user partitions were deleted or whether the HSM was zeroized, decommissioned, or factory reset as detailed below:

Action	Command/Event	Description
Erase User Partitions	> Enable or disable a destructive HSM policy	Destroy/erase all user partitions, but do not zeroize the HSM. Policy 46 "Disable Decommission" is the exception in that it zeroizes the HSM and erases all user partitions if the policy is changed. To bring the HSM back into service, you need to: **1.** Recreate the partitions **2.** Reinitialize the partition roles
Zeroize	> Too many bad login attempts on the HSM SO account > Perform an HSM firmware rollback > lunacm:> **hsm zeroize**	Deletes all partitions and their contents, but retains the HSM configuration (audit role and configuration, policy settings). To bring the HSM back into service, you need to: **1.** Reinitialize the HSM **2.** Recreate the partitions **3.** Reinitialize the partition roles
Decommission	> Press the decommission button on the rear of the appliance. > Enable **HSM Policy 40: Decommission on Tamper**, and tamper the HSM.	Deletes all partitions and their contents, the audit role, and the audit configuration. Retains the HSM policy settings. To bring the HSM back into service, you need to: **1.** Reinitialize the HSM **2.** Reinitialize the audit role and reconfigure auditing **3.** Recreate the partitions **4.** Reinitialize the partition roles
Factory Reset	lunacm:> **hsm factoryreset**	Deletes all partitions and their contents, and resets all roles and policy configurations to their factory default values. To bring the HSM back into service, you need to completely reconfigure the HSM as though it were new from the factory.

Comparison of Destruction/Denial Actions

Various operations on the Luna PCIe HSM 7 are intended to make HSM contents unavailable to potential intruders. The effect of those actions are summarized and contrasted in the following table, along with notes on how to recognize and how to recover from each scenario.

Scenario 1: MTK is destroyed, HSM is unavailable, but use/access can be recovered after reboot (See Note 1)

Scenario 2: KEK is destroyed (Real-Time Clock and NVRAM), HSM contents cannot be recovered without restore from backup See Note 2)

Event	Scen. 1	Scen. 2	How to discover (See Note 3)	How to recover
> Three bad SO login attempts > lunacm:> **hsm zeroize** > lunacm:> **hsm factoryreset** > Any change to a destructive policy > Firmware rollback (See Note 4)	NO	YES	> Log entry > "Partition Status -> Zeroized" in HSM info (from **hsm showinfo** on admin partition)	Restore HSM objects from Backup
Hardware tamper > Undervoltage or overvoltage during operation > Under-temperature or over-temperature during operation > Chassis interference (such as cover, fans, etc.) Software (command-initiated) tamper > lunacm:> **stm transport**	YES	NO	Parse logs for text like "tamper", "TVK was corrupted", or "Generating new TVK", indicating that a tamper event was logged. Example: `RTC: external tamper latched/ MTK: security function was zeroized on previous tamper event and has not been restored yet` Also, keywords in logs like: "HSM internal error", "device error"	Reboot [See Note 1]
Decommission > Short-circuiting the tamper header pins	NO	YES	Look for log entry like: RTC: tamper 2 signal/Zeroizing HSM after decommission...LOG(INFO): POWER-UP LOG DUMP END	Restore HSM objects from Backup

Event	Scen. 1	Scen. 2	How to discover (See Note 3)	How to recover

Note 1: MTK is an independent layer of encryption on HSM contents, to manage tamper and Secure Transport Mode. A destroyed MTK is recovered on next reboot. If MTK cannot be recovered, only restoring from backup onto a new or re-manufactured HSM can retrieve your keys and HSM data.

Note 2: KEK is an HSM-wide encryption layer that encrypts all HSM objects, excluding only MTK, RPK, a wrapping key, and a couple of keys used for legacy support. A destroyed KEK cannot be recovered. If the KEK is destroyed, only restoring from backup can retrieve your keys and HSM data.

Note 3: To check the health of a remote HSM, script a frequent login to the HSM host and execution of a subset of HSM commands. If a command fails, check the logs for an indication of the cause.

Note 4: These actions all create a situation where **hsm init** is required, or strongly recommended before the HSM is used again.

In addition, another event/action that has a destructive component is HSM initialization. See "Initializing the HSM" on page 202.

Stored Data Integrity

Beginning with Luna HSM Firmware 7.7.0, a new eIDAS-supporting feature called SDI, Stored Data Integrity, has been added that checks the integrity of the stored data. The HSM firmware will halt if it detects that objects have been corrupted. An *hsmrecover* function has been introduced, as part of the **hsm factoryReset** command to clear the storage and recover the HSM from the halt state without requiring RMA of the appliance.

If the HSM firmware halts because data in the volatile memory is corrupted, restarting the HSM using lunacm:>**hsm restart** should recover the HSM without losing data in permanent storage.

If the HSM firmware halts because data in the permanent flash storage is corrupted, the HSM is recovered by using the newly enhanced **hsm factoryReset** command which deletes all the partitions, zeroizes all the objects, and resets the policies.

Since **hsm factoryReset** is destructive, it is important to keep a regular backup of HSM objects in case the HSM ever goes into a state that requires factory reset.

Running the **hsm factoryReset** command, while the HSM is in normal working state, has the same behavior as before Luna HSM Firmware 7.7.0.

Running the **hsm factoryReset** command, while the HSM is in a halt state (where the normal "factoryReset" fails), invokes the recovery process, which takes several minutes (6+ minutes) to complete. It is important to wait for the **hsm factoryReset** command to complete without interruption.

For an example of the output, see **hsm factoryreset**. Also see "Comparison of Destruction/Denial Actions" on page 315.

RMA and Shipping Back to Thales

Although rare, it could happen that you need to ship a Luna PCIe HSM 7 back to Thales. Contact your Thales representative to obtain the Return Material Authorization (RMA) and instructions for packing and shipping. You might wish (or your security policy might require you) to take maximum precaution with any contents in your HSM before it leaves your possession.

RMA Process for Thales Luna HSM Devices Containing Sensitive Customer Key Material

Thales Luna Hardware Security Modules (HSMs) are designed, manufactured and tested to the highest level of quality. On occasion, a product may fail in the field after use by the customer. Products that fail in the field, when covered by a maintenance agreement or during the warranty period, may be eligible for an RMA.

Secure RMA Without Access to Key Material

Thales recognizes that Luna HSMs may contain sensitive customer key material. In case of an RMA, Thales does not have access to key material:

> Keys stored in the HSM are encrypted using a master key based on the customer's authentication method.

> It is impossible for Thales to retrieve or use sensitive customer key material from either a functioning or a failed HSM without the password or PED keys.

> Without these authentication devices or passwords, Thales cannot access key material in the device, even by reading the flash memory, as per FIPS 140-2 Level 3 and Common Criterial EAL4+ validation processes.

HSM Decommissioning

The general practice before returning a Luna HSM under an RMA is to decommission the HSM following the instructions in the user documentation (see "Decommissioning the HSM Card" on page 313 for instructions). This deletes the master key.

> **NOTE** Ensure you have a backup of your keys.

End of Service and Disposal

Luna HSMs and appliances are deployed into a wide variety of markets and environments. Arranging for the eventual disposal of a Luna PCIe HSM 7 that is no longer needed can be a simple accounting task and a call to your local computer recycling service, or it can be a complex and rigorous set of procedures intended to protect very sensitive information.

Needs Can Differ

Some users of Luna HSMs employ cryptographic keys and material that have a very short "shelf life". A relatively short time after the HSM is taken out of service, any objects that it contains are no longer relevant. The HSM could be disposed of, with no concern about any material that might remain in it.

The majority of our customers are concerned with their keys and objects that are stored on the HSM. It is important to them that those items never be exposed. The fact is that they are never exposed, but see below for explanations and actions that address the concerns of auditors who might be more accustomed to other ways of safeguarding HSM contents.

Luna HSM Protects Your Keys and Objects

The design philosophy of our Luna HSMs ensures that contents are safe from attackers. Unlike other HSM products on the market, Luna HSMs never store sensitive objects, like cryptographic keys, unencrypted. Therefore, Luna HSMs have no real need - other than perception or "optics" - to perform active erasure of HSM

contents, in case of an attack or tamper event.

Instead, the basic state of a Luna HSM is that any stored keys and objects are strongly encrypted. They are decrypted only for current use, and only into volatile memory within the HSM.

If power is removed from the HSM, or if the current session closes, the temporarily-decrypted objects instantly evaporate. The encrypted originals remain, but they are unusable by anyone who does not have the correct HSM keys to decrypt them.

How the HSM encryption keys protect your sensitive objects

In addition to encryption with the user specific access keys or passwords, all objects on the HSM are encrypted by the HSM's global key encryption key (KEK) and the HSM's unique Master Tamper Key (MTK).

If the HSM experiences a Decommission event (pressing of the small red button on back of Luna Network HSM 7, or shorting of the pins of the decommission header on the HSM card, then the KEK is deleted.

If the HSM experiences a tamper event (physical intrusion, environmental excursion), then the MTK is destroyed.

Destruction of either of those keys instantly renders any objects in the HSM unusable by anyone. In the case of a Decommission event, when the HSM is next powered on, it requires initialization, which wipes even the encrypted remains of your former keys and objects.

We recognize that some organizations build their protocols around assumptions that apply to other suppliers' HSMs - where keys are stored unencrypted and must be actively erased in the event of an attack or removal from service. If your policies include that assumption, then you can re-initialize after Decommission - which actively erases the encrypted objects for which no decrypting key existed. For purposes of security, such an action is not required, but it can satisfy pre-existing protocols that presume a weakness not present in Luna HSMs.