



Thales Luna USB HSM 7

PARTITION ADMINISTRATION GUIDE



Document Information

Last Updated	2025-05-28 09:09:59 GMT-05:00
--------------	-------------------------------

Trademarks, Copyrights, and Third-Party Software

Copyright 2001-2025 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 Partition Administration Guide	9
Customer Release Notes	9
Audience	9
Document Conventions	10
Support Contacts	12
 Chapter 1: Multifactor Quorum Authentication	13
Multifactor Quorum Authentication Architecture	13
Comparing Password and Multifactor Quorum Authentication	14
iKeys	14
iKey Types and Roles	15
Shared iKey Secrets	16
M of N Split Secrets (Quorum)	17
iKey Management Using Luna USB HSM 7	18
Creating iKey Using Luna USB HSM 7	18
Authenticating a Role Using Luna USB HSM 7	20
Consequences of Losing iKeys	22
Identifying an iKey Secret Using Luna USB HSM 7	24
Duplicating an Existing iKey Using Luna USB HSM 7	24
Changing an iKey Credential	25
 Chapter 2: Domain Planning and Key Cloning	28
Overview and Key Concepts	28
Domain Planning	28
What is a security domain or cloning domain?	28
Only one domain per partition - no copying across domains	29
No common domains across password-authenticated and multifactor quorum-authenticated HSMs	29
Characteristics of Cloning Domains	30
Cloning Objects to Another Application Partition	31
Cloning Keys Between Luna 6, Luna 7, and Luna Cloud HSM	32
Luna/Luna Cloud HSM Cloning	32
Mismatched Partition Policies and FIPS 140 Approved Configuration	33
Mismatched Key Types/Cryptographic Mechanisms	33
Minimum Key Sizes	34
 Chapter 3: Initializing the Application Partition	35
Initializing a New Partition	35
Re-initializing an Existing Partition	36
 Chapter 4: Partition Roles	37

Logging In to the Application Partition	40
Initializing the Crypto Officer and Crypto User Roles	42
Resetting the Crypto Officer or Crypto User Credential	43
Activation on Multifactor Quorum-Authenticated Partitions	44
Enabling Activation on a Partition	44
Activating a Role	45
Security of Your Partition Challenge	46
Name, Label, and Password Requirements	47
HSM Labels	47
Cloning Domains	48
Partition Labels	48
Role Passwords or Challenge Secrets	48
Chapter 5: Partition Capabilities and Policies	49
Setting Partition Policies Manually	59
Setting Partition Policies Using a Template	60
Creating a Partition Policy Template	60
Editing a Partition Policy Template	61
Applying a Partition Policy Template	63
Chapter 6: Cloning or Export of Private Keys	64
Setting Cloning Mode on a Partition	64
Setting Key Export Mode on a Partition	65
Setting No Backup Mode on a Partition	66
Chapter 7: V0 and V1 Partitions	67
Setting V0 or V1 on an Application Partiton	67
Special Characteristics of V0 Partitions	68
Special Characteristics of V1 Partitions	68
Chapter 8: Scalable Key Storage	70
The SKS Model	70
When to use SKS	72
SKS Master Key Types	72
High Availability and SKS	73
Backup/Restore and SKS	74
Using SKS	74
Prerequisites	74
Using SKS with the PKCS#11 API	74
Workflow Example Using ckdemo	75
Using the Provided Java Sample	76
Changing the SMK	76
Chapter 9: Per-Key Authorization	78
Example Use Case	78
New Role and Handling	79
No New Administrative Commands	79

Dependencies and Interactions with Other Features	79
Chapter 10: High-Availability Groups	80
Performance	80
Load Balancing	81
The Primary Partition	82
Network Topography	82
Key Replication	82
Failover	84
Mid-operation failures	84
Recovery	85
Auto-recovery	85
Manual Recovery	85
Failure of All Group Members	86
Permanent Failures	86
Standby Members	86
Application Object Handles	86
Virtual slots and virtual objects	87
The virtual object table	87
C_FindObjects behavior and application performance	87
Planning your HA Group Deployment	88
HSM and Partition Prerequisites	88
Sample Configurations	89
Configuring a High-Availability Group	90
Verifying an HA Group	92
Setting an HA Group Member to Standby	93
Configuring HA Auto-Recovery	94
Enabling/Disabling HA Only Mode	95
HA Logging	95
Configuring HA Logging	96
HA Log Messages	97
Managing HA Groups	99
Adding/Removing an HA Group Member	99
Manually Recovering a Failed HA Group Member	101
Replacing an HA Group Member	101
Deleting an HA Group	102
Chapter 11: Migrating Keys to Your New Luna USB HSM 7	104
Migration Using Slot-to-Slot Cloning	104
Chapter 12: Partition Backup and Restore	106
Key Concepts for Backup and Restore Operations	106
Credentials Required to Perform Backup and Restore Operations	107
Client Software Required to Perform Backup and Restore Operations	108
Multifactor Quorum Authentication with Luna Backup HSM 7 v1	108
Planning Your Backup HSM Deployment	108
Backup to Another Luna USB HSM 7	109

Partition to Partition	109
Backup to Luna Cloud HSM	109
Backup HSM Connected to the Client Workstation	109
Backup HSM Installed Using Remote Backup Service	110
Backup and Restore Best Practices	111
Backup to Another Luna USB HSM 7	111
Backup to Luna Cloud HSM	112
Luna Backup HSM 7	113
Multifactor Quorum Authentication	113
Password Authentication	114
Luna Backup HSM 7 Hardware Installation	114
Luna Backup HSM 7 Required Items	114
Luna Backup HSM 7 Hardware Functions	115
Installing the Luna Backup HSM 7 Hardware	116
Managing the Luna Backup HSM 7	117
Recovering the Luna Backup HSM 7 from Secure Transport Mode	117
Configuring the Luna Backup HSM 7 for FIPS Compliance	118
Updating the Luna Backup HSM 7 Firmware	118
Rolling Back the Luna Backup HSM 7 Firmware	120
Luna Backup HSM 7 Using Direct Multifactor Quorum Authentication	121
Initializing the Luna Backup HSM 7	121
Configuring the Luna Backup HSM 7 for FIPS Compliance	123
Backing Up a Multifactor Quorum-Authenticated Partition	123
Restoring To a Multifactor Quorum-Authenticated Partition	127
Luna Backup HSM 7 Using Remote Multifactor Quorum Authentication	129
Initializing the Luna Backup HSM 7 for Multifactor Quorum Authentication	129
Configuring the Luna Backup HSM 7 for FIPS Compliance	132
Backing Up a Multifactor Quorum-Authenticated Partition	133
Restoring To a Multifactor Quorum-Authenticated Partition	136
Luna Backup HSM 7 Using Password Authentication	139
Initializing the Luna Backup HSM 7 for Password Authentication	140
Configuring the Luna Backup HSM 7 for FIPS Compliance	141
Backing Up a Password-Authenticated Partition	142
Restoring to a Password-Authenticated Partition	144
Luna Backup HSM G5	146
Considerations when Performing Cloning and Backup-Restore Operations, when SKS is Involved	147
Luna Backup HSM G5 Hardware Installation	147
Luna Backup HSM G5 Required Items	147
Optional Items	148
Physical Features	149
Installing the Luna Backup HSM G5	150
Managing the Luna Backup HSM G5	150
Storage and Maintenance	150
Initializing the Luna Backup HSM G5 Remote PED Vector	151
Updating the Luna Backup HSM G5 Firmware	152
Resetting the Luna Backup HSM G5 to Factory Conditions	153
Installing or Replacing the Luna Backup HSM G5 Battery	154

About Luna Backup HSM G5 Secure Transport and Tamper Recovery	156
Creating a Secure Recovery Key	157
Setting Secure Transport Mode	158
Recovering From a Tamper Event or Secure Transport Mode	158
Disabling Secure Recovery	159
Backup/Restore Using Luna Backup HSM G5	159
Initializing the Luna Backup HSM G5	160
Backing Up an Application Partition	161
Restoring an Application Partition from Backup	163
Configuring a Remote Backup Server	164
Installing and Configuring the Remote Backup Service	164

PREFACE: About the Partition Administration Guide

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

- > ["Multifactor Quorum Authentication" on page 13](#)
- > ["Domain Planning and Key Cloning" on page 28](#)
- > ["Initializing the Application Partition" on page 35](#)
- > ["Partition Roles" on page 37](#)
- > ["Partition Capabilities and Policies" on page 49](#)
- > ["Cloning or Export of Private Keys" on page 64](#)
- > ["V0 and V1 Partitions" on page 67](#)
- > ["Scalable Key Storage" on page 70](#)
- > ["Per-Key Authorization" on page 78](#)
- > ["High-Availability Groups" on page 80](#)
- > ["Partition Backup and Restore" on page 106](#)

The preface includes the following information about this document:

- > ["Customer Release Notes" below](#)
- > ["Audience" below](#)
- > ["Document Conventions" on the next page](#)
- > ["Support Contacts" on page 12](#)

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	<p>The bold attribute is used to indicate the following:</p> <ul style="list-style-type: none"> > 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.)
<i>italics</i>	In type, the italic attribute is used for emphasis or to indicate a related document. (See the <i>Installation Guide</i> 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> <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> <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: Multifactor Quorum Authentication

The Luna USB HSM 7 can be initialized to use multifactor quorum authentication for all roles on the HSM. The authentication secrets are stored on USB iKeys, and are presented by inserting them directly into the Luna USB HSM 7. This means that the iKeys and direct access to the Luna USB HSM 7 are the only method of accessing the HSM's administrative functions. This prevents key-logging exploits on workstations connected to the client HSM, because authentication takes place entirely within the device itself. No password is entered via computer keyboard.

This section contains the following information about Luna USB HSM 7 multifactor quorum authentication:

- > ["Multifactor Quorum Authentication Architecture" below](#)
 - ["Comparing Password and Multifactor Quorum Authentication" on the next page](#)
- > ["iKeys" on the next page](#)
 - ["iKey Types and Roles" on page 15](#)
 - ["Shared iKey Secrets" on page 16](#)
 - ["Domain iKeys" on page 17](#)
 - ["iKey PINs" on page 17](#)
 - ["M of N Split Secrets \(Quorum\)" on page 17](#)
- > ["iKey Management Using Luna USB HSM 7" on page 18](#)

Multifactor Quorum Authentication Architecture

The multifactor quorum authentication architecture consists of the following components:

- > Luna USB HSM 7: Role secrets stored on iKeys are presented directly to the Luna USB HSM 7 by connecting them to the USB-C input.
- > **Authentication secrets:** Cryptographic secrets generated by the HSM and stored on iKeys. 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.
- > **iKeys:** physical USB-connected devices that contain authentication secrets, created by the HSM (see ["iKeys" on the next page](#)). iKeys have the following custom authentication features:
 - **Shared Secrets:** iKeys of the same type can be reused or shared among HSMs or partitions, allowing domain sharing (necessary for backup configurations) and other custom configurations. See ["Shared iKey Secrets" on page 16](#).
 - **iKey PINs:** optional PINs associated with specific iKeys, set by the owner of the iKey at the time of creation. iKey PINs offer an extra layer of security for iKeys which could be lost or stolen. See ["iKey PINs" on page 17](#).

- **M of N Split Key Scheme:** optional configuration which allows a role to split its authentication secret across multiple iKeys, 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 "[M of N Split Secrets \(Quorum\)](#)" on page 17.

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	<ul style="list-style-type: none"> > Knowledge of role password is sufficient > For backup/restore, knowledge of partition domain password is sufficient 	<ul style="list-style-type: none"> > Ownership of the black Crypto Officer iKey is mandatory > For backup/restore, ownership of both black CO and red domain iKeys 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 iKey, imposing a two-factor authentication requirement on any role
Dual Control	<ul style="list-style-type: none"> > Not available 	<ul style="list-style-type: none"> > 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
Key-custodian responsibility	<ul style="list-style-type: none"> > Password knowledge only 	<ul style="list-style-type: none"> > Linked to partition password knowledge > Linked to black iKey(s) ownership and optional PIN knowledge
Two-factor authentication for remote access	<ul style="list-style-type: none"> > Not available 	<ul style="list-style-type: none"> > Remote PED and orange (Remote PED Vector) iKey deliver highly secure remote management of HSM, including remote backup

iKeys

The iKey is a USB authentication device, embedded in a molded plastic body. It 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 USB HSM 7 does not hold the authentication secrets. They reside only on the portable iKeys.




iKeys are created when an HSM, partition, or role is initialized. Each iKey can contain only one authentication secret at a time, but it can be overwritten with a new authentication secret. See ["iKey Management Using Luna USB HSM 7" on page 18](#).




CAUTION! Do not subject iKeys to extremes of temperature, humidity, dust, or vibration. Use the included key cap to protect the USB connector.

iKey Types and Roles

Once initialized for multifactor quorum authentication, the Luna USB HSM 7 uses iKeys for all credentials. You can apply the appropriate labels included with your iKeys, according to the table below, as you create them.

The iKey colors correspond with the HSM roles described in [HSM Roles](#) and ["Partition Roles" on page 37](#). The following table describes the keys associated with the various roles:

Lifecycle	iKey	Secret	Function
HSM Administration	Blue	HSM Security Officer (HSM SO) secret	Authenticates the HSM SO role. The HSM SO manages provisioning functions and security policies for the HSM. Mandatory
	Red 	HSM Domain or Key Cloning Vector	Cryptographically defines the set of HSMs that can participate in cloning for backup. See "Domain iKeys" on page 17 . Mandatory
HSM Auditing	White 	Auditor (AU) secret	Authenticates the Auditor role, responsible for audit log management. This role has no access to other HSM services. Optional
Partition Administration	Blue	Partition Security Officer (PO) secret	Authenticates the Partition SO role. The PO manages provisioning activities and security policies for the partition. NOTE: If you want the HSM SO to also perform Partition SO duties, you can use the same blue key to initialize both roles. Mandatory
	Red 	Partition Domain or Key Cloning Vector	Cryptographically defines the set of partitions that can participate in cloning for backup or high-availability. See "Domain iKeys" on page 17 . Mandatory

Lifecycle	iKey	Secret	Function
Partition Operation	Black 	Crypto Officer (CO) secret	Authenticates the Crypto Officer role. The CO can perform both cryptographic services and key management functions on keys within the partition. Mandatory
	Gray 	Limited Crypto Officer (LCO) secret	Authenticates the Limited Crypto Officer role. The LCO can perform a subset of the actions available to the Crypto Officer. Optional (used in eIDAS-compliant schemes)
	Gray 	Crypto User (CU) secret	Authenticates the Crypto User role. The CU can perform cryptographic services using keys already existing within the partition. It can create and back up public objects only. NOTE: If administrative separation is not important, you can use a single black key to initialize the Crypto Officer and Crypto User roles and still have two separate challenge secrets to distinguish read-write and read-only role privileges. Optional

NOTE 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 iKeys.

Shared iKey Secrets

The Luna USB HSM 7 identifies the type of authentication secret on an inserted iKey, 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 iKey(s) to authenticate multiple HSMs or partitions. This is useful for:

- > 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 iKeys" 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 iKey secret to authenticate multiple roles, HSMs, or partitions is less secure than giving each its own iKey. Refer to your organization's security policy for guidance.

Domain iKeys

A red domain iKey holds the key-cloning vector (the domain identifier) that allows key cloning between HSMs and partitions, and is therefore the iKey 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 the 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 or partition.

iKey PINs

The Luna USB HSM 7 allows the holder of a iKey to set a numeric PIN, 4-48 characters long, to be associated with that iKey. This PIN must then be entered on the touchscreen for all future authentication. The PIN provides two-factor authentication and ensures security in case an iKey is lost or stolen. If you forget your PIN, it is the same as losing the iKey entirely; you cannot authenticate the role.

PINs can be set only at the time of key creation, and can be changed only by changing the secret on the iKey. Duplicate keys are true copies with the same PIN, intended as backups for one person (see ["Duplicating an Existing iKey Using Luna USB HSM 7" on page 24](#)). Duplicates of the iKey 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 iKeys" on page 22](#).

M of N Split Secrets (Quorum)

The Luna USB HSM 7 can split an authentication secret among multiple 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.

This can be likened to a club 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 between 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 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 scenario, the HSM SO authentication secret is split among five blue iKeys, and at least three of those keys must be presented to the Luna USB HSM 7 to log in as HSM SO.

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 an 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 recommended only when you want multiple people to access the role, each with their own unique iKey PIN.

NOTE Using an M of N split secret can greatly increase the number of iKeys you require. Ensure that you have enough blank or rewritable iKeys on hand before you begin initializing your M of N scheme.

Activated Partitions and 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 iKey (see ["Activation on Multifactor Quorum-Authenticated Partitions" on page 44](#)). 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 iKeys) before normal operations can resume.

iKey Management Using Luna USB HSM 7

Once you have connected your Luna USB HSM 7 to a workstation and installed Luna HSM Client, you can proceed with initializing roles on the HSM using multifactor quorum authentication. The procedures in this section will guide you through the touchscreen prompts at each stage of iKey creation, authentication, and other iKey operations with the Luna USB HSM 7.

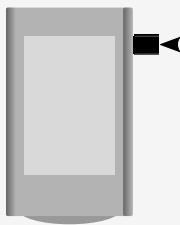
- > ["Creating iKey Using Luna USB HSM 7" below](#)
- > ["Authenticating a Role Using Luna USB HSM 7" on page 20](#)
- > ["Consequences of Losing iKeys" on page 22](#)
- > ["Identifying an iKey Secret Using Luna USB HSM 7" on page 24](#)
- > ["Duplicating an Existing iKey Using Luna USB HSM 7" on page 24](#)
- > ["Changing an iKey Credential" on page 25](#)

Creating iKey Using Luna USB HSM 7

When you initialize an HSM, partition, or role, the Luna USB HSM 7 issues a series of prompts for you to follow to create your iKeys. iKey actions have a timeout setting (default: 120 seconds); ensure that you have everything you need before issuing an initialization command. The requirements for the operation depend on the iKey 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 iKey or keyset, the owners of those keys must be present with their iKey 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. It is advisable for each iKey holder to create backup duplicates, so you must have a sufficient number of blank or rewritable iKeys ready before you begin.
- > If you plan to make backup duplicates of iKeys, you must have a sufficient number of blank or rewritable iKeys ready.
- > If you plan to use PINs, ensure that they can be privately entered on the Luna USB HSM 7 and memorized, or written down and securely stored.

NOTE Whenever the Luna USB HSM 7 prompts you to insert an iKey, use the USB-C adapter in the USB port on the right side of the Luna USB HSM 7:



To initiate iKey creation

1. Issue one of the following LunaCM commands to initialize the applicable role, domain, or vector.

- **Blue HSM SO and Red HSM Domain Keys:**

```
lunacm:> hsm init -label <label> -iped
```

- **Orange Remote iKey:**

```
lunacm:> ped vector init
```

- **Blue Partition SO and Red Partition Domain iKeys:**

```
lunacm:> partition init
```

- **Black Crypto Officer iKey:**

```
lunacm:> role init -name co
```

- **Gray Limited Crypto Officer iKey**

```
lunacm:> role init -name lco
```

- **Gray Crypto User iKey:**

```
lunacm:> role init -name cu
```

- **White Audit User iKey:**

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

2. Follow the touchscreen prompts in the following four stages.

Stage 1: Reusing Existing iKeys

If you want to use an iKey or quorum of iKeys with an existing authentication secret, have them ready to present to the HSM. Reasons for reusing iKeys may include:

- > You want to use the same iKey to authenticate multiple HSMs/partitions
- > You want to initialize a partition in an already-existing cloning domain (to allow cloning of cryptographic objects between partitions)

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 iKey Secrets" on page 16](#) and ["Domain iKeys" on page 17](#) for more information.

The first touchscreen prompt asks if you want to create a new quorum of iKeys or reuse an existing quorum. Make your selection and follow the instructions on the touchscreen. If you are creating a new quorum, go to ["Stage 2: Defining M of N" below](#).

Stage 2: Defining M of N

If you chose to create a new keyset, the Luna USB HSM 7 prompts you to define the M of N scheme (quorum and pool of splits) for the role, domain, or vector. See ["M of N Split Secrets \(Quorum\)" on page 17](#) for more information. If you do not want to use M of N (authentication by one iKey), enter a value of **1** for both M and N.

For each iKey in the quorum, proceed to ["Stage 3: Setting a PIN" below](#).

Stage 3: Setting a PIN

If you are creating a new iKey, 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 ["iKey PINs" on page 17](#) for more information.

CAUTION! If you forget your PIN, it is the same as losing the iKey entirely; you cannot authenticate the role. See ["Consequences of Losing iKeys" on page 22](#).

You now have the opportunity to create a duplicate of the new iKey in ["Stage 4: Duplicating New iKey" below](#). If you decline to create a duplicate now, repeat this stage for each new iKey in the quorum.

Stage 4: Duplicating New iKey

You now have the option to create duplicates of your newly-created iKey(s) in case of key loss or theft.

Authenticating a Role Using Luna USB HSM 7

When connected, the Luna USB HSM 7 responds to authentication commands in LunaCM. Commands that require authentication include:

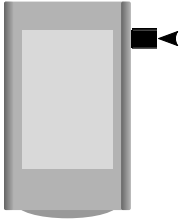
- > Role login commands (blue, black, gray, or white iKeys)
- > Backup/restore commands (red iKeys)
- > Remote PED connection commands (orange iKey)

When you issue a command that requires authentication, the interface returns a message like the following:

```
lunacm:>role login -name po
```

Please attend to the PED.

Whenever the Luna USB HSM 7 prompts you to insert a iKey, use the USB port on the right side of the Luna USB HSM 7:



CAUTION! Multiple failed authentication attempts result in zeroization of the HSM or partition, or role lockout, depending on the role. This is a security measure designed to thwart repeated, unauthorized attempts to access cryptographic material. For details, see [Logging In as HSM Security Officer](#) or "[Logging In to the Application Partition](#)" on page 40.

To perform multifactor quorum authentication

1. The touchscreen prompts for the corresponding iKey. Insert the iKey (or the first M of N split-secret key) and follow the instructions on the touchscreen.

```
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 USB HSM 7 returns control to the command interface.

Command Result : No Error

2. If a PIN is associated with the iKey, the touchscreen prompts for the PIN.

- If the key you inserted is an M of N split, continue to step 3.
- Otherwise, authentication is complete and the Luna USB HSM 7 returns control to the command interface.

Command Result : No Error

3. The touchscreen prompts for the next M of N split-secret key. Insert the next iKey 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. At this point, authentication is complete and the Luna USB HSM 7 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 iKey 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 iKeys have been presented.

Consequences of Losing iKeys

iKeys are the only means of authenticating roles, domains, and RPVs on the multifactor quorum-authenticated Luna USB HSM 7. Losing an iKey effectively locks the user out of that role. Always keep secure backups of your iKeys, including M of N split secrets. Forgetting the PIN associated with an iKey is equivalent to losing the iKey entirely. Losing a split-secret iKey is less serious, unless enough splits are lost so that M cannot be satisfied.

If an iKey is lost or stolen, log in with one of your backup keys and change the existing role secret immediately, to prevent unauthorized HSM access.

The consequences of a lost iKey with no backup vary depending on the type of secret:

- > ["Blue HSM SO iKey" below](#)
- > ["Red HSM Domain iKey" below](#)
- > ["Blue Partition SO iKey" on the next page](#)
- > ["Red Partition Domain iKey" on the next page](#)
- > ["Black Crypto Officer iKey" on the next page](#)
- > ["Gray Crypto User iKey" on page 24](#)
- > ["White Audit User iKey" on page 24](#)

Blue HSM SO iKey

If the HSM SO secret is lost, you can no longer perform administrative tasks on the HSM, including partition creation and client assignment. The contents of the HSM Admin partition are unrecoverable and you can no longer configure the HSM. Take the following steps:

1. Contact the Crypto Officer and have them immediately make a backup of their existing partition.
2. When all important cryptographic material is backed up, execute a factory reset of the HSM.
3. Initialize the HSM and create a new HSM SO secret.
4. Recreate the application partition.
5. The Partition SO must initialize the new partition using their original blue and red iKey(s), and initialize the Crypto Officer role (and Activation secret, if applicable). Supply the new black CO iKey to the Crypto Officer.
6. The Crypto Officer must change the login credentials from the new black CO iKey to their original black iKeys (and reset the Activation secret password, if applicable).
7. The Crypto Officer can now restore all partition contents from backup.
8. If you are using Remote PED, you must recreate the Remote PED Vector (RPV). You can re-use the original orange iKey.

Red HSM Domain iKey

If the HSM Key Cloning Vector is lost, you can no longer perform backup/restore operations on the HSM Admin partition. 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.

Blue Partition SO iKey

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 iKey

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 iKey

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 iKey

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 iKey

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 an iKey Secret Using Luna USB HSM 7

You can use this procedure to identify the type of secret (role, domain, or RPV) stored on an unidentified iKey. 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 USB HSM 7 in Admin Mode
- > the key you want to identify

To identify the type of secret stored on the iKey

1. Insert the iKey you want to identify.
2. Tap the **ADMIN** tab on the touchscreen to enter Admin mode.

The role secret type is identified on-screen.

Duplicating an Existing iKey Using Luna USB HSM 7

During the key creation process, you have the option to create multiple copies of iKeys. If you want to make backups of your keys later, you can use this procedure to copy iKeys. You require:

- > Luna USB HSM 7 in Admin Mode
- > Enough blank or rewritable keys to make your copies

The iKey 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 an 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 M of N keyset. See "[M of N Split Secrets \(Quorum\)](#)" on page 17.

To duplicate an existing iKey

1. Insert the iKey you want to duplicate. Have a blank or rewritable iKey ready.
2. Tap the **ADMIN** tab on the touchscreen to enter Admin mode.
3. Tap **Duplicate this iKey** and follow the instructions on the touchscreen.

Changing an iKey Credential

It may be necessary to change the iKey secret associated with a role. Reasons for changing credentials include:

- > Regular credential rotation as part of your organization's security policy
- > Compromise of a role due to loss or theft of a iKey
- > 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 iKey credential depends on the type of key. Procedures for each type are provided below.

CAUTION! If you are changing an iKey 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 iKey credentials, you must always present the old keyset first; do not overwrite your old iKeys until you have no further need for them.

If you overwrite the original iKey with a new credential and the operation fails, it is possible for the iKey credential to be overwritten while the role remains tied to the old credential. If this happens, all login attempts with the overwritten iKey will fail. Ensure that you keep at least one backup copy of the old iKey credential until the role is successfully set to a new credential.

- > ["Blue HSM SO iKey" on page 22](#)
- > ["Red HSM Domain iKey" on page 22](#)
- > ["Orange Remote PED Vector iKey" on the next page](#)
- > ["Blue Partition SO iKey" on page 23](#)
- > ["Red Partition Domain iKey" on page 23](#)
- > ["Black Crypto Officer iKey" on page 23](#)
- > ["Gray Crypto User iKey" on the previous page](#)
- > ["White Audit User iKey" on the previous page](#)

Blue HSM SO iKey

The HSM SO can use this procedure to change the HSM SO credential.

To change the blue HSM SO iKey credential

1. In LunaCM, set the active slot to the Admin partition and log in as HSM SO.

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

2. Initiate the iKey change.

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

3. You are prompted to present the original blue key(s) and then to create a new HSM SO keyset. See ["Creating iKey Using Luna USB HSM 7" on page 18](#).

Red HSM Domain iKey

It is not possible to change an HSM's cloning domain without performing a factory reset of 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 iKey

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 log in as HSM SO.
lunacm:> **role login -name so**
2. Initialize the RPV.
lunacm:> **ped vector init**
You are prompted to create a new Remote iKey.
3. Distribute a copy of the new orange key to the administrator of each Remote PED server.

Blue Partition SO iKey

The Partition SO can use this procedure to change the Partition SO credential.

To change a blue Partition SO iKey credential

1. In LunaCM, log in as Partition SO.
lunacm:> **role login -name po**
2. Initiate the iKey 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.

Red Partition Domain iKey

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.

Black Crypto Officer iKey

The Crypto Officer can use this procedure to change the Crypto Officer credential.

To change a black Crypto Officer iKey credential

1. In LunaCM, log in as Crypto Officer.
lunacm:> **role login -name co**
2. Initiate the iKey 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.

Gray Crypto User iKey

The Crypto User can use this procedure to change the Crypto User credential.

To change a gray Crypto User iKey credential

1. In LunaCM, log in as Crypto User.
lunacm:> **role login -name cu**
2. Initiate the iKey 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.

White Audit User Key

The Audit User can use this procedure to change the Audit User credential.

To change the white Audit User iKey credential

1. In LunaCM, set the active slot to the Admin partition and log in as Auditor.
lunacm:> **role login -name au**
2. Initiate the iKey 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.

CHAPTER 2: Domain Planning and Key Cloning

You can clone key material between partitions to back up the keys, or to migrate the keys from one HSM to another. The rules, prerequisites, and procedures for migrating your key material are described in the following topics:

- > ["Overview and Key Concepts" below](#)
- > ["Domain Planning" below](#)
- > ["Cloning Objects to Another Application Partition" on page 31](#)
- > ["Domain Planning and Key Cloning" above](#)
 - ["Luna/Luna Cloud HSM Cloning" on page 32](#)
 - ["Mismatched Partition Policies and FIPS 140 Approved Configuration" on page 33](#)
 - ["Mismatched Key Types/Cryptographic Mechanisms" on page 33](#)
 - ["Minimum Key Sizes" on page 34](#)

Overview and Key Concepts

A Crypto Officer can clone the cryptographic objects (keys) from one user partition to another user partition provided that:

- > The user partitions share the same domain. See ["Domain Planning" below](#).
- > The user partitions use the same authentication method (iKey or password).
- > The CO has the required credentials on both user partitions.
- > The capabilities and policies set on the source and target HSM and user partitions allow cloning. See [HSM Capabilities and Policies](#) and ["Partition Capabilities and Policies" on page 49](#).

Domain Planning

The cloning or security domain is an element of [Layered Encryption](#).

What is a security domain or cloning domain?

A security domain or cloning domain is a layer of encryption that is created, during initialization, on an HSM or HSM partition that you control. The domain determines whether a cryptographic object can leave the HSM, and where it can go if it is allowed to leave.

Cloning is a secure-copy operation by which sensitive HSM objects are copied, while strongly encrypted, from one HSM to another HSM. The security domain, or cloning domain, is a special-purpose secret that is attached to a partition on an HSM. It determines *to* which, and *from* which, other partitions (on the same HSM or on other HSMs) the current partition can clone objects. Partitions that send or receive partition objects by means of the cloning protocol must share identical cloning domain secrets. That is, the protocol verifies that the destination domain matches the source domain; otherwise an error is displayed and the attempted operation fails. This is important for cloning in backup and restore operations.

Only one domain per partition - no copying across domains

An application partition can have one cloning domain. It is not possible to clone objects from two or more different cloning domains to a single partition. By design, there is no provision to change the cloning domain of a partition without initializing it, which destroys any objects in that partition.

No common domains across password-authenticated and multifactor quorum-authenticated HSMs

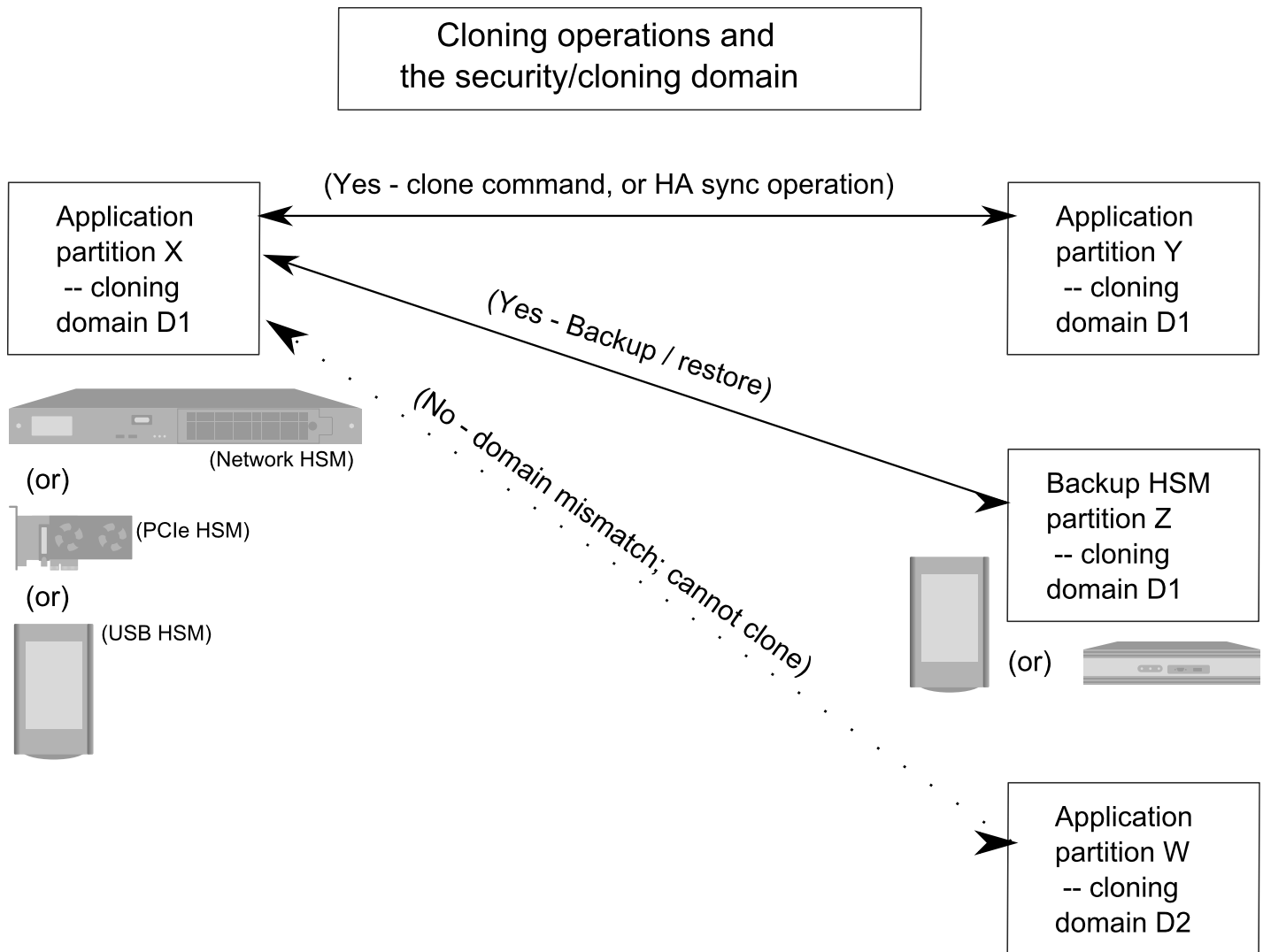
Password-authenticated application partitions with identical security domains can clone partition contents one to the other, if the partition policies support cloning.

Multifactor Quorum-authenticated application partitions with identical security domains can clone partition contents one to the other, if the partition policies support cloning.

Password-authenticated HSM partitions cannot perform cloning with multifactor quorum-authenticated HSM partitions.

The security design consideration is that, if you have a key or object stored in a multifactor quorum-authenticated partition:

- > It cannot be altered to a less-secure state and moved outside the protection of its original security/cloning domain.
- > You are assured that the key or object has never been outside its original security/cloning domain, or in any less-secure state.



Characteristics of Cloning Domains

Password-authenticated HSMs have text-string cloning domains for the HSM admin partition and for any partitions that are created on the HSM. HSM and Partition domains are typed at the command line of the client computer, when required. Password authentication cloning domains are created by you.

Multifactor Quorum-authenticated cloning domains are created by a Luna HSM, which could be the current HSM, or a previously-initialized HSM that you wish to include in a cloning group with the current HSM. Multifactor Quorum-authenticated HSMs have cloning domains in the form of encrypted secrets on red iKeys, for the admin partition and for any partitions that are created on the HSM.

The following characteristics are common to security (cloning) domains on all Luna HSMs.

- > The unique admin partition security domain can be created in the HSM at initialization time, or it can be imported, meaning that it is shared with one-or-more other HSMs.
- > The application partition security domain can be created by the current HSM when the partition is initialized, or it can be imported, meaning that it is shared with one-or-more other HSM partitions, and therefore direct cloning and backup/restore can be performed among the partitions that share a given domain.

- > The application partition security domain is usually distinct from the HSM domain, as they are controlled by different people; on multi-partition HSMs, the PSO is usually not the same person as the HSM SO, but on a single-partition HSM the two SOs might be the same person.
- > The application partition security domain can be the same as the domain of another partition on the same HSM (for HSMs that support multiple partitions).

For multifactor quorum-authenticated HSMs, the domain secret for the admin partition or for an application partition can be a single red iKey, or it can be split (by the MofN quorum feature) over several red keys, which are then distributed among trusted personnel such that no single person is able to provide the cloning domain without oversight from other trusted personnel.

In scenarios where multiple HSM partitions are in use, it can be useful to segregate those partitions according to department or business unit, or according to function groups within your organization. This ensures that personnel in a given group are able to clone or backup/restore only the contents of partitions sharing the domain for which they are responsible. The segregation is maintained by physical and procedural control of the relevant iKeys that each group is allowed to handle.

For password-authenticated HSMs, that sort of segregation is maintained entirely by procedure and by trust, as you rely on personnel not to share the domain text strings, just as you rely on them not to share other passwords.

Have your naming conventions and allotments planned out ahead of HSM initialization and partition creation, including a well-thought-out map of who should control cloning domain access for admin partitions and for application partitions. These decisions must be made before you create the partitions.

Cloning Objects to Another Application Partition

You can back up partition objects from an application partition to any other partition that shares its cloning domain. The Crypto Officer of both partitions can perform this operation using LunaCM.

Prerequisites

- > **Partition policy 0: Allow private key cloning** must be set to **1 (ON)** on both the source and target partitions.
- > The target partition must be initialized with the same cloning domain as the source partition.
- > You require the Crypto Officer credential for both the source and the target partition.
- > Both partitions must be visible as slots in LunaCM.
- > [Remote PED] This procedure is simpler when both partitions are activated (see "[Activation on Multifactor Quorum-Authenticated Partitions](#)" on page 44). If the partitions are not activated, you must connect the source partition to PEDserver before logging in, disconnect it, and then connect the target partition to PEDserver by specifying its slot.

```
lunacm:> ped connect [-ip <IP>] [-port <port>]
```

```
lunacm:> ped disconnect
```

```
lunacm:> ped connect -slot <target_slot> [-ip <IP>] [-port <port>]
```

To clone partition objects to another application partition

1. In LunaCM, set the active slot to the source partition and log in as Crypto Officer.

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

```
lunacm:> role login -name co
```

2. [Optional] View the partition objects and their object handles.

```
lunacm:> partition contents
```

3. Clone objects on the partition to the target partition by specifying the target slot. You can choose which objects to clone by specifying a comma-separated list of object handles, or specify **all** to clone all objects on the partition. Present the target partition's Crypto Officer credential when prompted.

```
lunacm:> partition clone -slot <slotnum> -objects <comma-separated_list/all>
```

The specified objects are cloned to the target partition. Any objects that already exist on the target are not cloned.

Cloning Keys Between Luna 6, Luna 7, and Luna Cloud HSM

Luna HSM Client allows you to clone keys between Luna 6 partitions, Luna 7 partitions, and Thales Data Protection on Demand (DPoD) Luna Cloud HSM services. This includes creating HA groups made up of different HSM versions. This configuration is useful for:

- > migrating your keys directly from Luna 6 to your new Luna 7 HSMs
- > migrating your keys from Luna USB HSM 7 to the cloud, or vice-versa
- > gradually upgrading your on-premises production environment from Luna 6 to Luna 7 HSMs
- > maintaining a real-time, cloud-based backup of your cryptographic objects

This page contains guidelines and general considerations for cloning keys between the different HSMs, or using mixed-version HA groups. Mixed-version HA groups have all the same requirements of standard HA groups (see ["Planning your HA Group Deployment" on page 88](#)), in addition to the considerations listed below.

Luna/Luna Cloud HSM Cloning

Cloning between Luna partitions and Luna Cloud HSM services require the following special considerations, in addition to the general considerations below.

Authentication

Luna Cloud HSM services use password authentication, and therefore they can clone objects to and from password-authenticated Luna USB HSM 7s only. It is not possible to clone keys between a Luna Cloud HSM service and a multifactor quorum-authenticated Luna HSM.

Network Latency and Luna Cloud HSM as Active HA Member

Requests performed by cloud services like Luna Cloud HSM may experience greater network latency than those sent to on-premise HSMs. Thales recommends using a Luna Cloud HSM service as a standby HA member to achieve the best performance. By default, you can add a Luna Cloud HSM service as a standby HA member only. If all other HA members fail and the Luna Cloud HSM service becomes active, it will revert to standby when another member recovers.

If you prefer to use the Luna Cloud HSM service as an active HA member, you must first edit the following toggle in the **Chrystoki.conf/crystoki.ini** configuration file (see [Configuration File Summary](#)):


```
[Toggles]
lunacm_cv_ha_ui = 0
```

Cloning Capacity Limitations

The following limitations apply to clients accessing a Luna Cloud HSM service:

- > 100 token objects (or 50 RSA-2048 key pairs) per service.
- > 100 session objects (or 50 RSA-2048 key pairs) per application.
- > 100 simultaneous sessions per application.

Clients that exceed the token object and session object limits can experience slow or failed request responses. The session limit is enforced, and the client receives the error `CKR_MAX_SESSION_COUNT` when the application reaches the limit.

If you exceed the recommended maximum number of objects cloned to/from a Luna Cloud HSM service in a single cloning operation, the operation sometimes fails with `CKR_DEVICE_ERROR`. In the case of HA groups, this could include key creation operations, since objects are then cloned to the Luna Cloud HSM service.

Mismatched Partition Policies and FIPS 140 Approved Configuration

Partitions in an HA group, and the HSMs on which they reside, must be configured with the same policy settings (see ["HSM and Partition Prerequisites" on page 88](#)). For example, Luna 6 HSMs have certain policies that have been removed from Luna 7 and Luna Cloud HSM, and new policies have been introduced.

Ensure that policies common to Luna 6/7/Luna Cloud HSM members have the same settings, according to your deployment requirements.

lunacm:> [partition showpolicies](#)

CAUTION! In particular, FIPS 140 approved configuration (formerly FIPS mode) must be consistent across all HA members (on or off).

Mismatched Key Types/Cryptographic Mechanisms

Cloning is limited to key types that are recognized by the firmware on both HSMs. If an HSM does not recognize the type of key being cloned to it, the cloning operation may fail. Ensure that the firmware on the destination HSM is capable of recognizing all cryptographic objects stored on the source HSM.

NOTE Luna HSMs comply closely with the relevant FIPS standards and their generally accepted interpretations. These are moving targets, as the crypto and security climate continues to evolve. It is possible for a validated HSM version (firmware) to be fully compliant when its NIST certificate is issued, and for same-model HSMs with newer firmware and more stringent restrictions to refuse to accept "less secure" objects.

Alternatively, the more up-to-date HSM might accept an object from an earlier-firmware HSM, but permit only limited uses of such an object.

If you are cloning between HSMs operating in FIPS 140 approved configuration (formerly FIPS mode), please consult [Supported Mechanisms](#) for the destination HSM's version to determine if all key types can be cloned.

Minimum Key Sizes

Minimum key sizes are enforced when using certain cryptographic algorithms. These minimums may differ between versions. If a Luna 6 partition creates a key that is smaller than the minimum size required by Luna 7 or Luna Cloud HSM, the key will not be replicated to the other partitions in the HA group.

NOTE Minimum key sizes for many mechanisms are larger in FIPS 140 approved configuration (formerly FIPS mode), and FIPS minimums may vary among firmware releases.

To avoid this, use LunaCM to check a mechanism's minimum key size. Check the same mechanism on each HA member slot, and always use the highest minimum reported in the HA group.

lunacm:> **partition showmechanism -m** <mechanism_ID>

CHAPTER 3: Initializing the Application Partition

Before it can be used to store cryptographic objects or perform operations, an application partition must be initialized. Initialization is performed by the Partition Security Officer and sets the authentication credential. There are two scenarios where the Partition SO would initialize the partition:

- > **Preparing a new partition:** On a new partition, initialization sets the Partition SO authentication credential, an identifying label for the partition, and the partition's cloning domain (see ["Initializing a New Partition" below](#)).
- > **Erasing an existing partition:** The Partition SO can re-initialize a partition to erase all cryptographic objects and the Crypto Officer/Crypto User roles, and select a new partition label. The Partition SO credential and the cloning domain remain the same (see ["Re-initializing an Existing Partition" on the next page](#)).

Initializing a New Partition

Initializing an application partition for the first time establishes you as the Partition SO and sets a cloning domain for the partition. This procedure must be performed from the Luna USB HSM 7 client using LunaCM commands.

Prerequisites

- > The new partition must be created on the HSM and visible in LunaCM (see [Creating or Deleting the Application Partition](#)).
- > If you want to configure the partition's policies with a policy template using LunaCM, the template file must be available on the client (see ["Setting Partition Policies Using a Template" on page 60](#)).
- > Multifactor Quorum authentication: Ensure that you have enough blue (Partition SO) and red (Domain) iKeys for your planned authentication scheme (see ["Creating iKey Using Luna USB HSM 7" on page 18](#)).

To initialize a new application partition

1. Launch LunaCM on the client workstation.
2. Set the active slot to the partition you want to initialize.
`lunacm:> slot set -slot <slot_number>`
3. Initialize the partition by specifying an identifying label. To initialize the partition using a policy template, specify the path to the template file.

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.

- **Password authentication:** You can specify a Partition SO password and/or a domain string with the initialization command, or enter them when prompted.

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.

Passwords and activation challenge secrets must be 8-255 characters in length. Spaces are allowed; to specify a password with spaces using command-line options, enclose the password in double quotation marks. The space character may not be used as the first character in a password.

The following characters are allowed:

!#\$%&'()*+,-./0123456789:=? @ABCDEFGHIJKLMNOPQRSTUVWXYZ[]^_abcdefghijklmnopqrstuvwxyz{|}~

This character set is enforced when using [Luna HSM Client 10.8.0](#) or newer, and recommended for all previous versions. Previously-set passwords and challenge secrets are unaffected, but the new character set is enforced when these passwords are changed.

```
lunacm:> partition init -label <label> [-applytemplate <template_file>] [-password <password>] [-domain <domain_string>]
```

- **Multifactor Quorum authentication:**

```
lunacm:> partition init -label <label> [-applytemplate <template_file>]
```

Respond to the touchscreen (see "[Creating iKey Using Luna USB HSM 7" on page 18](#)) prompts to create the blue Partition SO key and the red domain key.

Re-initializing an Existing Partition

The Partition SO can re-initialize an existing partition at any time. Re-initialization erases all cryptographic objects on the partition, and the login credentials for the Crypto Officer and Crypto User roles. The Partition SO login credential and cloning domain are retained.

Prerequisites

- > The partition must be already initialized.
- > Back up any important cryptographic objects stored on the partition.

To re-initialize an existing application partition

1. Launch LunaCM on the client workstation.
2. Set the active slot to the partition you want to re-initialize.
lunacm:> **slot set -slot** <slot_number>
3. Initialize the partition by specifying an identifying label. You must specify a label for the partition (the same label or a new one). You are prompted for the current Partition SO credential.

```
lunacm:> partition init -label <label>
```

CHAPTER 4: Partition Roles

The security of an HSM and its cryptographic contents depends on well-controlled access to that HSM. A controlled access policy is defined by:

- > the set of users with valid login credentials for the client 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.

All cryptographic operations take place on an application partition. This partition is created on the Luna USB HSM 7 by the HSM SO and is designed to function independently of the Admin partition, with its own Security Officer and users. This provides more flexibility in meeting the security needs of your organization. Personnel holding the roles described below must have administrative access to the Luna USB HSM 7 host workstation.

The partition-level roles are as follows:

Partition Security Officer (PO)

The Partition SO handles all administrative and configuration tasks on the application partition, including:

- > Initializing the partition, setting the PO credential, and setting a cloning domain for the partition (see ["Initializing the Application Partition" on page 35](#))
- > Configuring partition policies (see ["Partition Capabilities and Policies" on page 49](#))
- > Initializing the Crypto Officer role (see ["Initializing the Crypto Officer Role" on page 42](#))
- > Activating the partition (see ["Activation on Multifactor Quorum-Authenticated Partitions" on page 44](#))

Managing the Partition SO Role

Refer also to the following procedures to manage the PO role:

- > ["Logging In to the Application Partition" on page 40](#)
- > [Changing a Role Credential](#)

Crypto Officer (CO)

The Crypto Officer is the primary user of the application partition and the cryptographic objects stored on it. The Crypto Officer has the following responsibilities:

- > Creating, deleting, and modifying cryptographic objects via user applications
- > Performing cryptographic operations via user applications
- > Managing backup and restore operations for partition objects (see ["Partition Backup and Restore" on page 106](#)):
- > Initializing the Crypto User role (see ["Initializing the Crypto User Role" on page 42](#))

- > The CO can modify keys - must provide per-key authorization (PKA) data for unassigned keys
- > The CO can unblock blocked (due to per-key auth failures) PKA keys
- > The CO can increment usage counters and change/set the limit
- > The CO can perform SMK rollover

Managing the Crypto Officer Role

Refer also to the following procedures to manage the CO role:

- > ["Logging In to the Application Partition" on page 40](#)
- > [Changing a Role Credential](#)

Limited Crypto Officer (LCO)

The Limited Crypto Officer is a role needed for eIDAS compliance and the performance of Per-Key Authorization functions, with a subset of the abilities and responsibilities of the Crypto Officer, but wider authority and ability than the Crypto User. The LCO is created by the partition CO. The Limited Crypto Officer has the following abilities and responsibilities:

- > Creating, deleting, and modifying cryptographic objects via user applications
- > Performing cryptographic operations via user applications
 - The LCO can copy and modify keys and private objects- must provide per-key authorisation (PKA) data for unassigned keys
 - The LCO can increment usage counters, but cannot change/set the limit
 - The LCO cannot unblock blocked (due to per-key auth failures) PKA keys
 - The LCO can wrap/unwrap keys - must specify the per-key auth data for both the wrapping/unwrapping keys and the wrapped/unwrapped keys
 - The LCO can derive keys - must provide the per-key auth data for the key used for derivation and specify the per-key auth data for the key being derived in the template
 - The LCO can derive-and-wrap - must provide per-key auth data as above
 - The LCO can perform SKS operations (SIMExtract / SIMInsert)
 - The LCO cannot perform SMK rollover
- > Managing backup and restore operations for partition objects:
 - ["Partition Backup and Restore" on page 106](#)
- > Creating and configuring HA groups (see ["Configuring a High-Availability Group" on page 90](#))
- > Initializing the Crypto User role (see ["Initializing the Crypto User Role" on page 42](#))

Managing the Limited Crypto Officer Role

Refer also to the following procedures to manage the LCO role:

- > ["Logging In to the Application Partition" on page 40](#)
- > [Changing a Role Credential](#)
- > The LCO role does not support cloning

- > The LCO role is not visible for V0 partitions.
- > The LCO role is subject to role-affecting partition policies like
 - Minimum PIN length [25]
 - Maximum PIN length [26]
 - Maximum failed challenge responses [15]
 - Maximum failed user logins allowed [20]
 - Upon reaching the limit, the LCO is locked out; CO and CU remain operational
 - Partition CO can unlock a locked LCO by resetting its credentials
- > The LCO can create and destroy private objects
- > The LCO can generate keys assigned or unassigned, but cannot assign a key after it is generated.
- > The LCO can delete keys
 - Unlike CO role, LCO must provide per-key authorization (PKA) data
 - LCO supports the “single-use signing keys” scenario where a user generates a key, signs with that key, and deletes the key
- > The LCO can modify keys - must provide per-key authorisation (PKA) data for unassigned keys
- > The LCO can increment usage counters but, unlike CO, cannot change/set the limit
- > The LCO can wrap/unwrap
 - PKA behaviour for wrap: must provide the per-key auth data for both the wrapping and the wrapped keys
 - PKA behaviour for unwrap: must provide the per-key auth data for unwrapping key and specify the per-key auth data for the unwrapped key in the template
- > For PKA operation
 - The LCO can derive keys
 - The LCO can derive-and-wrap
- > The LCO can extract/insert in all scenarios
 - Including SKS key migration (old SKS: Insert; no Extract)
 - Including new SKS (Extract and Insert)
- > The LCO cannot clone/replicate in any scenario - this means that LCO is not self-sufficient for HA; the CO is needed to clone SMK(s)
- > Unlike the CO, the LCO cannot perform SMK rollover

Crypto User (CU)

The Crypto User is an optional role that can perform cryptographic operations using partition objects in a read-only capacity, but can create only public objects. This role is useful in that it provides limited access; the Crypto Officer is the only role that can make significant changes to the contents of the partition. The Crypto User has the following capabilities:

- > Performing operations like encrypt/decrypt and sign/verify using objects on the partition
- > Creating and backing up public objects (see ["Partition Backup and Restore" on page 106](#)):

- > The CU can increment usage counters but, unlike CO, cannot change/set the limit

Managing the Crypto User Role

Refer also to the following procedures to manage the CU role:

- > ["Logging In to the Application Partition" below](#)
- > [Changing a Role Credential](#)

Logging In to the Application Partition

Before you can perform administrative tasks on the partition or its stored cryptographic objects, you must log in with the appropriate role:

- > Partition Security Officer (specify **po** for <role>)
- > Crypto Officer (specify **co** for <role>)
- > Crypto User (specify **cu** for <role>)

To log in to the application partition

1. Launch LunaCM on the Luna USB HSM 7 client workstation.
2. Set the active slot to the desired partition.
lunacm:> **slot set -slot** <slotnum>
3. Log in by specifying your role on the partition.

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

You are prompted for the role's credential.

Failed Partition Login Attempts

The consequences of multiple failed login attempts vary by role, depending on the severity of the security risk posed by that role being compromised. This is a security feature meant to thwart repeated, unauthorized attempts to access your cryptographic material.

NOTE The system must actually receive some erroneous/false information before it logs a failed attempt; if you merely forget to insert the iKey, or insert the wrong color key, that is not counted as a failed attempt. You must insert an incorrect iKey of the correct type, or enter an incorrect PIN or challenge secret, to fail a login attempt.

Partition Security Officer

If you fail ten consecutive Partition SO login attempts, the partition is zeroized and all cryptographic objects are destroyed. The Partition SO must re-initialize the partition and Crypto Officer role, who can restore key material from a backup device.

Crypto Officer

If you fail ten consecutive Crypto Officer login attempts, the CO and CU roles are locked out. But see below for the exception. The default lockout threshold of 10 is governed by partition policy 20: Max failed user logins allowed, and the Partition SO can set this threshold lower if desired (see ["Partition Capabilities and Policies" on page 49](#)).

Is recovery possible from lockout or loss of the partition role credential?

Yes, and no, depending on configuration options you might choose.

Separation of roles ensures that,

- > while the Partition Crypto Officer (and subsidiary roles) can see and manage the *content* of an application partition,
- > the partition SO cannot access or manage the content of a partition; SO manages at the provisioning and security level for the partition.

If you lose the use of your CO credential, the contents of the partition are no longer accessible. The Partition SO might not be able to help in that situation, for the following reason.

The partition SO cannot just reset the password of the partition CO if you have disallowed it

Recovery from partition role lockout depends on the setting of **HSM policy 15: Enable SO reset of partition PIN**:

- > If HSM policy 15 is set to **1** (enabled), the CO and CU and LCO roles are temporarily locked out by too many bad authentication attempts. The Partition SO must unlock the CO role and reset the credential (see ["Resetting the Crypto Officer or Crypto User Credential" on page 43](#)).
- > If HSM policy 15 is set to **0** (disabled), the CO and CU and LCO roles are permanently locked out and the partition contents are no longer accessible. The Partition SO must re-initialize the partition (destroying all contents) and the Crypto Officer role, who can restore key material from a backup. This is the default setting.

NOTE If you have a backup and know its password, you can recover material. If you do not have a backup, or the backup that you have is not secured by a known password, then the material is lost.

CAUTION! If loss of partition contents is not the desired outcome, ensure that the HSM SO enables this destructive policy *before* creating partitions and assigning to clients.

Crypto User

If you fail ten consecutive Crypto User login attempts, the CU role is locked out. The default lockout threshold of 10 is governed by partition policy **20: Max failed user logins allowed**, and the Partition SO can set this threshold lower if desired (see ["Partition Capabilities and Policies" on page 49](#)). The CO must unlock the CU role and reset the credential (see ["Resetting the Crypto Officer or Crypto User Credential" on page 43](#)).

Initializing the Crypto Officer and Crypto User Roles

The following procedures will allow you to initialize the Crypto Officer (CO) and Crypto User (CU) roles and set an initial credential.

Initializing the Crypto Officer Role

The Crypto Officer (CO) is the primary user of the application partition and the cryptographic objects stored on it. The Partition Security Officer (PO) must initialize the CO role and assign an initial credential.

To initialize the Crypto Officer role from the Client via lunacm

1. In LunaCM, log in to the partition as Partition SO (see ["Logging In to the Application Partition" on page 40](#)).
lunacm:> **role login -name po**
2. Initialize the Crypto Officer role. If you are using a password-authenticated partition, specify a CO password. If you are using a multifactor quorum-authenticated partition, ensure that you have a blank or rewritable black iKey available. Refer to ["Creating iKey Using Luna USB HSM 7" on page 18](#) for details.

Passwords and activation challenge secrets must be 8-255 characters in length. Spaces are allowed; to specify a password with spaces using command-line options, enclose the password in double quotation marks. The space character may not be used as the first character in a password. The following characters are allowed:
!#\$% ' () * + , - . / 0 1 2 3 4 5 6 7 8 9 : = ? @ A B C D E F G H I J K L M N O P Q R S T U V W X Y Z [] ^ _ ` a b c d e f g h i j k l m n o p q r s t u v w x y z { } ~
This character set is enforced when using [Luna HSM Client 10.8.0](#) or newer, and recommended for all previous versions. Previously-set passwords and challenge secrets are unaffected, but the new character set is enforced when these passwords are changed.

lunacm:> **role init -name co**
3. Provide the CO credential to your designated Crypto Officer.

NOTE If **HSM policy 21: Force user PIN change after set/reset** is enabled (this is the default setting), the CO must change the credential before any other actions are permitted. See [Changing a Role Credential](#).

Initializing the Crypto User Role

The Crypto User (CU) is an optional role that can perform cryptographic operations using partition objects in a read-only capacity, but can only create public objects. The Crypto Officer must initialize the CU role and assign an initial credential.

To initialize the Crypto User role

1. In LunaCM, log in to the partition as Crypto Officer (see ["Logging In to the Application Partition" on page 40](#)).
lunacm:> **role login -name co**
2. Initialize the Crypto User role. If you are using a password-authenticated partition, specify a CU password. If you are using a multifactor quorum-authenticated partition, ensure that you have a blank or rewritable gray iKey available. Refer to ["Creating iKey Using Luna USB HSM 7" on page 18](#) for details.

Passwords and activation challenge secrets must be 8-255 characters in length. Spaces are allowed; to specify a password with spaces using command-line options, enclose the password in double quotation marks. The space character may not be used as the first character in a password. The following characters are allowed:

```
!#$%&'()*+,-./0123456789:=? @ABCDEFGHIJKLMNOPQRSTUVWXYZ[]^_abcdefghijklmnopqrstuvwxyz{}~
```

This character set is enforced when using [Luna HSM Client 10.8.0](#) or newer, and recommended for all previous versions. Previously-set passwords and challenge secrets are unaffected, but the new character set is enforced when these passwords are changed.

```
lunacm:> role init -name cu
```

3. Provide the CU credential to your designated Crypto User.

NOTE If **HSM policy 21: Force user PIN change after set/reset** is enabled (this is the default setting), the CU must change the credential before any other actions are permitted. See [Changing a Role Credential](#).

Resetting the Crypto Officer or Crypto User Credential

If necessary, the Crypto Officer can reset the Crypto User credential at any time, without providing the current credential. This is useful in cases where the Crypto User credential has been lost or otherwise compromised.

Prerequisites for Crypto Officer Reset

The Partition SO can also reset the Crypto Officer's credential, if **HSM policy 15: Enable SO reset of partition PIN** is enabled. By default, this policy is not enabled, and changing it is destructive. If you want the Partition SO to be able to reset the CO's credential, the HSM SO must enable this policy before creating the application partition (see ["Partition Capabilities and Policies" on page 49](#)).

CAUTION! HSM policy 15 is destructive when turned on. All partitions on the HSM and their contents will be erased.

To reset the Crypto Officer or Crypto User credential

1. Log in with the appropriate role (see ["Logging In to the Application Partition" on page 40](#)).
2. Reset the desired role's credential.

Passwords and activation challenge secrets must be 8-255 characters in length. Spaces are allowed; to specify a password with spaces using command-line options, enclose the password in double quotation marks. The space character may not be used as the first character in a password. The following characters are allowed:

```
!#$%&'()*+,-./0123456789:=? @ABCDEFGHIJKLMNOPQRSTUVWXYZ[]^_abcdefghijklmnopqrstuvwxyz{}~
```

This character set is enforced when using [Luna HSM Client 10.8.0](#) or newer, and recommended for all previous versions. Previously-set passwords and challenge secrets are unaffected, but the new character set is enforced when these passwords are changed.

```
lunacm:> role resetpw -name <role>
```

You are prompted to set a new credential for the role.

3. Provide the new credential to the Crypto Officer or Crypto User.

NOTE If **HSM policy 21: Force user PIN change after set/reset** is enabled, the user must change the credential before any other actions are permitted. See [Changing a Role Credential](#). The CO can reset the LCO's primary credentials (lunacm:> **role resetpw**) regardless of the status of "Enable SO reset of a partition PIN" policy 15.

Activation on Multifactor Quorum-Authenticated Partitions

A multifactor quorum-authenticated partition requires an iKey each time a role (Partition SO, Crypto Officer, Crypto User) logs in. For some use cases, such as key vaulting, this physical key requirement is desirable. For many applications, however, it is impractical to require the full authentication procedure every time.

For these use cases, the Partition SO can activate the partition and set a secondary password referred to as a challenge secret. When a partition is activated, the HSM caches the Crypto Officer and Crypto User secrets upon first login, and subsequent logins require the challenge secret only. The iKey secret remains cached until the role is explicitly deactivated or the HSM loses power due to a reboot or power outage.

Activation does not provide much advantage for clients that log in to the partition and remain logged in. It is an indispensable advantage in cases where the client application repeatedly logs in to perform a task, and then logs out or closes the cryptographic session after the task is completed.

Tamper events and activation

When a tamper event occurs, or if an uncleared tamper event is detected on reboot, the cached iKey data is zeroized, and activation is disabled. See [Tamper Events](#) and ["Partition Capabilities and Policies" on page 49](#) for more information.

This section contains instructions for the following procedures:

- > ["Enabling Activation on a Partition" below](#)
- > ["Activating a Role" on the next page](#)
- > ["Deactivating a Role" on page 46](#)

Enabling Activation on a Partition

The Partition SO can enable activation on a partition by setting **partition policy 22: Allow activation** to **1** (on). This setting enables activation for the Crypto Officer and Crypto User roles. When partition policy 22 is enabled, the Partition SO can set an initial challenge secret for the Crypto Officer.

Prerequisites

- > The partition must be initialized (see ["Initializing the Application Partition" on page 35](#)).

To enable activation on a partition

1. Log in to the partition as Partition SO (see ["Logging In to the Application Partition" on page 40](#)).
lunacm:> **role login -name po**
2. Enable partition policy 22.
lunacm:> **partition changepolicy -policy 22 -value 1**

Activating a Role

After enabling partition policy 22, activate the CO or CU roles on the partition. You must set a challenge password for each role you want to activate. The Partition SO must set the initial challenge secret for the Crypto Officer, who must set it for the Crypto User. The role becomes activated the first time the user logs in to the partition.

Prerequisites

- > **Partition policy 22: Allow activation** must be enabled on the partition (see ["Enabling Activation on a Partition" on the previous page](#)).
- > The role you wish to activate must be initialized on the partition (see ["Initializing the Crypto Officer and Crypto User Roles" on page 42](#)).

To activate a role

1. Log in to the partition using the appropriate role (see ["Logging In to the Application Partition" on page 40](#)):
 - If you are activating the Crypto Officer role, log in as Partition SO.
 - If you are activating the Crypto User role, log in as Crypto Officer.

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

2. Set an initial challenge secret for the role you wish to activate. The length of the challenge secret is configurable by the Partition SO (see ["Partition Capabilities and Policies" on page 49](#)).

Passwords and activation challenge secrets must be 8-255 characters in length. Spaces are allowed; to specify a password with spaces using command-line options, enclose the password in double quotation marks. The space character may not be used as the first character in a password. The following characters are allowed:

!#\$%&'()*+,-./0123456789:=? @ABCDEFGHIJKLMNPQRSTUVWXYZ[]^_abcdefghijklmnopqrstuvwxyz{}~

This character set is enforced when using [Luna HSM Client 10.8.0](#) or newer, and recommended for all previous versions. Previously-set passwords and challenge secrets are unaffected, but the new character set is enforced when these passwords are changed.

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

NOTE Activation requires that a challenge secret is set for the specified role. If the role does not have a challenge secret, you are prompted for the iKey, regardless of the policy setting.

3. Log out of the partition.

lunacm:> **role logout**

4. Provide the initial challenge secret to the designated CO or CU by secure means. The iKey secret is cached when they log in for the first time. The CO or CU can store the black or gray iKey in a safe place. The cached iKey secret allows their application(s) to open and close sessions and perform operations within those sessions.

NOTE If HSM policy 21: Force user PIN change after set/reset is enabled (this is the default setting), the CO or CU must change the challenge secret before any other actions are permitted. See [Changing a Role Credential](#).

Deactivating a Role

An activated role on a partition remains activated until it is explicitly deactivated, or the HSM loses power due to a reboot or power outage. This deletes the cached authentication secret for the role.

Prerequisites

- > You must be authorized to deactivate the role. The CO and CU can manually deactivate their own or each other's roles. The Partition SO can deactivate both the CO and CU roles.

To deactivate a role on a partition

1. Log in to the partition with the appropriate role (see ["Logging In to the Application Partition" on page 40](#)).

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

2. Specify the role you wish to deactivate.

```
lunacm:> role deactivate -name <role>
```

This deletes the cached authentication credential for the role. The next time the role logs in, the credential is re-cached.

3. If you wish to disable activation entirely, so that credentials are not re-cached at the next login, the Partition SO can disable **partition policy 22: Allow activation**.

```
lunacm:> partition changepolicy -policy 22 -value 0
```

Security of Your Partition Challenge

For Luna USB HSM 7s with Password Authentication, the partition password used for administrative access by the Crypto Officer is also the partition challenge secret or password used by client applications.

For Luna USB HSM 7s with multifactor quorum authentication, the partition authentication used for administrative access by the Crypto Officer is the secret on the black iKey(s) for that partition. The partition challenge secret or password used by client applications is a separate character string, set by the Partition SO and then changed by the Crypto Officer (mandatory) for the CO's use. This is one way in which we implement separation of roles in the Luna HSM security paradigm.

How Secure Is the Challenge Secret or Password?

The underlying concern is that a password-harvesting attack might eventually crack the secret that protects the partition. Layers of protection are in place, to minimize or eliminate such a risk.

First, such an attack must be run from a Luna HSM Client computer. For interaction with HSM partitions on a Luna Network HSM 7, a Luna HSM Client computer is one with Luna software installed, on which you have performed the exchange of certificates to create a Network Trust Link (NTL). That exchange requires the

knowledge and participation of the appliance administrator and the Partition SO (who might, or might not, be the same person). It is not possible to secretly turn a computer into a Client of a Luna HSM partition - an authorized person within your organization must participate.

Second, for Luna HSMs with password authentication, you set the partition password directly when you create the partition, so you can make it as secure as you wish (for an example of guidance on password strength, see <http://howsecureismypassword.net/> or <http://xkcd.com/936/>)

For Luna HSMs with multifactor quorum authentication, an optional partition password (also called a challenge secret) may be added for the initialized Crypto Officer (CO) and/or Crypto User (CU) roles. See [role createchallenge](#) for the proper command syntax.

Using LunaCM or LunaSH, you can change the partition password (or challenge secret) if you suspect it has been compromised, or if you are complying with a security policy that dictates regular password changes.

As long as you replace any password/challenge secret with one that is equally secure, the possible vulnerability is extremely small.

Conversely, you can choose to replace a secure, random password/challenge-secret with one that is shorter or more memorable, but less secure - you assume the risks inherent in such a tradeoff.

Third, Luna HSM **partition policy 15: Ignore failed challenge responses** can be set to **0** (off). When that policy is off, the HSM stops ignoring bad challenge responses (that is, attempts to submit the partition secret) and begins treating them as failed login attempts. Each bad login attempt is counted. **Partition policy 20: Max failed user logins allowed** determines how high that count can go before the partition is locked out.

Once a partition is locked by bad login attempts, it cannot be accessed until the HSM Security Officer (SO) unlocks it. This defeats an automated harvesting attack that relies on millions of attempts occurring at computer-generated speeds. As well, after one or two lockout cycles, the HSM SO would realize that an attack was under way and would rescind the NTL registration of the attacking computer. That computer would no longer exist as far as the HSM partition was concerned. The SO or your security organization would then investigate how the client computer had been compromised, and would correct the problem before allowing any new NTL registration from that source. See ["Logging In to the Application Partition" on page 40](#) for more information.

As the owner/administrator of the HSM, you determine any tradeoffs with respect to security, convenience, and other operational parameters.

Name, Label, and Password Requirements

This page describes length and character requirements for setting labels, domains, passwords, and challenge secrets on the Luna USB 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" on the next page](#)
- > ["Partition Labels" on the next page](#)
- > ["Role Passwords or Challenge Secrets" on the next page](#)

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

Passwords and activation challenge secrets must be 8-255 characters in length. Spaces are allowed; to specify a password with spaces using command-line options, enclose the password in double quotation marks. The space character may not be used as the first character in a password.

The following characters are allowed:

!#\$%&'()*+,-./0123456789:;=?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[]^_abcdefghijklmnopqrstuvwxyz{|}~

This character set is enforced when using [Luna HSM Client 10.8.0](#) or newer, and recommended for all previous versions. Previously-set passwords and challenge secrets are unaffected, but the new character set is enforced when these passwords are changed.

CHAPTER 5: Partition Capabilities and Policies

An application partition can be configured to provide a range of different functions. The Partition Security Officer can customize this functionality using partition policies. This configuration is governed by the following settings:

- > **Partition Capabilities** are features of partition functionality that are inherited from the parent HSM policies (see [HSM Capabilities and Policies](#)). The HSM SO can configure HSM policies to allow or disallow partition capabilities. Some capabilities have corresponding modifiable partition policies.
- > **Partition Policies** are configurable settings that allow the Partition Security Officer to modify the function of their corresponding capabilities.

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

- > ["Setting Partition Policies Manually" on page 59](#)
- > ["Setting Partition Policies Using a Template" on page 60](#)

Destructive Policies

As a security measure, changing some partition policies forces deletion of all cryptographic objects on the partition. These policies are listed as **destructive** in the table below. Some policy changes are destructive in either direction (**OFF-to-ON** and **ON-to-OFF**), while others are destructive only in the direction resulting in lowered partition security.

Use `lunacm:> partition showpolicies -verbose` to check whether the policy you want to enable/disable is destructive.

#	Partition Capability	Partition Policy
0	<p>Enable private key cloning</p> <p>Always 1. This capability allows private keys to be cloned to another Luna HSM partition (required for backup).</p> <div> <p>NOTE The HSM SO can disable cloning for all partitions on the HSM by turning off HSM policy 7 (see HSM Capabilities and Policies). In this case, cloning is not possible on the partition, regardless of this capability/policy's setting.</p> </div>	<p>Allow private key cloning</p> <p><i>Destructive OFF-to-ON</i></p> <ul style="list-style-type: none"> > 1 (default): The partition is capable of cloning private keys to another partition. This policy must be enabled to back up partitions or create HA groups. Public keys and objects can always be cloned, regardless of this policy's setting. > 0: Private keys can never be cloned to another application partition. <p>Partition policies 0 and 1 may not be set to 1 (ON) at the same time.</p> <div> <p>NOTE Key attributes can be set modifiable, and a key can then be set with (for example) attribute - extractable (see cmu generatekeypair), but Partition Policies overrule object attributes; Cloning ON and Private Key Wrapping OFF would prevent export despite the attribute settings.</p> </div>
1	<p>Enable private key wrapping</p> <p>Always 1. This capability allows private keys to be encrypted (wrapped) and exported off the partition.</p>	<p>Allow private key wrapping</p> <p><i>Destructive OFF-to-ON</i></p> <ul style="list-style-type: none"> > 1: Private keys may be wrapped and saved to an encrypted file off the partition. Public keys and objects can always be wrapped and exported, regardless of this policy's setting. > 0 (default): Private keys can never be wrapped and exported off the partition. <p>Partition policies 0 and 1 may not be set to 1 (ON) at the same time.</p> <div> <p>NOTE Key attributes can be set modifiable, and a key can then be set with (for example) attribute - extractable (see cmu generatekeypair), but Partition Policies overrule object attributes; Cloning ON and Private Key Wrapping OFF would prevent export despite the attribute settings.</p> </div>

#	Partition Capability	Partition Policy
2	Enable private key unwrapping Always 1. This capability allows wrapped private keys to be imported to the partition.	Allow private key unwrapping <ul style="list-style-type: none"> > 1 (default): Private keys can be unwrapped and stored on the partition. > 0: Private keys cannot be unwrapped onto the partition.
3	Enable private key masking Private keys can be masked off the partition.	Allow private key masking <ul style="list-style-type: none"> > 1 (default for V1 partitions): Private keys can be masked off the partition. > 0 (default for V0 partitions): Private keys cannot be masked off the partition.
4	Enable secret key cloning Always 1. This capability allows secret keys to be cloned to another Luna HSM partition (required for backup). <div> NOTE The HSM SO can disable cloning for all partitions on the HSM by turning off HSM policy 7 (see HSM Capabilities and Policies). In this case, cloning is not possible on the partition, regardless of this capability/policy's setting. </div>	Allow secret key cloning <i>Destructive OFF-to-ON</i> <ul style="list-style-type: none"> > 1 (default): Secret keys on the partition can be cloned to another partition. This is required for partition backup and HA groups. > 0: Secret keys cannot be backed up, and will not be cloned to other HA group members.
5	Enable secret key wrapping Always 1. This capability allows secret keys to be encrypted (wrapped) and exported off the partition.	Allow secret key wrapping <i>Destructive OFF-to-ON</i> <ul style="list-style-type: none"> > 1 (default): Secret keys can be wrapped and saved to an encrypted file off the partition. > 0: Secret keys can never be wrapped and exported off the partition.
6	Enable secret key unwrapping Always 1. This capability allows wrapped secret keys to be imported to the partition.	Allow secret key unwrapping <ul style="list-style-type: none"> > 1 (default): Secret keys can be unwrapped and stored on the partition. > 0: Secret keys cannot be unwrapped onto the partition.
7	Enable secret key masking Enable masking secret keys off the partition.	Allow secret key masking <ul style="list-style-type: none"> > 1 (default for V1 partitions): Secret keys can be masked and stored off the partition. > 0 (default for V0 partitions): Secret keys cannot be masked off the partition.

#	Partition Capability	Partition Policy
9	<p>Enable DigestKey</p> <p>Always 1.</p> <p>Enable the C_DigestKey function to hash a symmetric key and return the hash to the calling application. The hashing is a subset of steps performed in many HASH/HMAC-based KDFs. The HSM firmware checks the policy every time LUNA_DIGEST_KEY is called, and returns LUNA_RET_OPERATION_RESTRICTED if the policy is off.</p> <p>Only FIPS-compliant hashes are allowed, so the state of the policy does not affect overall FIPS compliance.</p> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p>NOTE DigestKey can allow replication of Key Derive Functions externally, permitting some keys to be derived outside the HSM.</p> </div> <p>Requires Luna USB HSM 7 Firmware 7.7.3 or newer.</p>	<p>Allow DigestKey</p> <p><i>Destructive OFF-to-ON</i></p> <ul style="list-style-type: none"> > 1 : Key Derive Functions can be performed using DigestKey > 0 (default): Key Derive Functions cannot use DigestKey. <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p>NOTE Partition policy 9: "Allow DigestKey" above is set to 0 by default when you update to "Partition Capabilities and Policies" on page 49 or newer, and it is destructive when changed from 0 to 1. If you were using C_DigestKey with Luna USB HSM 7 Firmware 7.7.2, and you need to continue using it, you must back up the contents of your application partition and restore them after changing the policy. Refer to "Partition Backup and Restore" on page 106.</p> </div>
10	<p>Enable multipurpose keys</p> <p>Always 1. This capability allows keys that are created or unwrapped on the partition to have more than one of the following attributes enabled (set to 1), and can therefore be used for multiple types of operation:</p> <ul style="list-style-type: none"> • Encrypt/Decrypt • Sign/Verify • Wrap/Unwrap • Derive 	<p>Allow multipurpose keys</p> <p><i>Destructive OFF-to-ON</i></p> <ul style="list-style-type: none"> > 1 (default): Keys that are created or unwrapped on the partition may be used for multiple operations. > 0: Keys that are created or unwrapped on the partition may have only one of the affected attributes enabled. Thales recommends that you create keys with only the attributes required for their intended purpose. Disabling this policy enforces this rule on the partition. <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p>NOTE This policy does not affect Diffie-Hellman keys, which are always created with only Derive set to 1.</p> </div>

#	Partition Capability	Partition Policy
11	Enable changing key attributes Always 1. This capability allows the Crypto Officer to modify the following non-sensitive attributes of keys on the partition, changing key functions: <ul style="list-style-type: none"> > CKA_ENCRYPT > CKA_DECRYPT > CKA_WRAP > CKA_UNWRAP > CKA_SIGN > CKA_SIGN_RECOVER > CKA_VERIFY > CKA_VERIFY_RECOVER > CKA_DERIVE > CKA_EXTRACTABLE 	Allow changing key attributes <i>Destructive OFF-to-ON</i> <ul style="list-style-type: none"> > 1 (default): The Crypto Officer can modify the non-sensitive attributes of keys on the partition. > 0: Keys created on the partition cannot be modified.
15	Allow failed challenge responses Always 1. This capability/policy applies to multifactor quorum-authenticated Luna USB HSM 7 only. It determines whether failed login attempts using a challenge secret count towards a partition lockout.	Ignore failed challenge responses <i>Destructive OFF-to-ON</i> <ul style="list-style-type: none"> > 1 (default): Failed challenge secret login attempts are not counted towards a partition lockout. Only failed iKey authentication attempts increment the counter. > 0: Failed login attempts using either an iKey or a challenge secret will count towards a partition lockout. <p>See "Activation on Multifactor Quorum-Authenticated Partitions" on page 44 and "Logging In to the Application Partition" on page 40 for more information.</p>
16	Enable operation without RSA blinding Always 1. This is always disabled on Luna USB HSM 7.	Operate without RSA blinding <ul style="list-style-type: none"> > Always 1: The partition does not use RSA blinding. This policy cannot be changed.
17	Enable signing with non-local keys Always 1. Keys generated on the HSM have the attribute CKA_LOCAL=1. Keys that are imported (unwrapped) to the HSM have CKA_LOCAL=0. These attributes are maintained if keys are backed up or cloned to another HSM partition.	Allow signing with non-local keys <ul style="list-style-type: none"> > 1 (default): Only keys with attribute CKA_LOCAL=1 can be used to sign data on the partition. > 0: Keys with attribute CKA_LOCAL=0 can be used for signing, and their trust history is not assured.

#	Partition Capability	Partition Policy
18	Enable raw RSA operations Always 1 . This capability enables the RSA mechanism CKM_RSA_X_509 on the partition, which allows weak signatures and weak encryption.	Allow raw RSA operations <i>Destructive OFF-to-ON</i> > 1 (default): The partition allows operations using the RSA mechanism CKM_RSA_X_509 . > 0 : Operations using CKM_RSA_X_509 are blocked on the partition.
20	Max failed user logins allowed Displays the maximum number of failed partition login attempts (10) before the partition is locked out (see " Logging In to the Application Partition " on page 40).	Max failed user logins allowed The Partition SO can lower the effective number of failed logins below the maximum if desired. Default: 10
21	Enable high availability recovery Always 1 . This capability enables the RecoveryLogin feature on the partition. This feature allows other HA group members to restore the login state of the partition in the event of a power outage or other such deactivation.	Allow high availability recovery > 1 (default): RecoveryLogin is enabled on the partition. This feature must be configured in advance (see role recoveryinit and role recoverylogin). > 0 : RecoveryLogin is disabled on the partition.
22	Enable activation This capability allows the partition to be activated. See " Activation on Multifactor Quorum-Authenticated Partitions " on page 44. > 1 : Always enabled on multifactor quorum-authenticated HSMs. > 0 : Always disabled on password-authenticated HSMs.	Allow activation > 1 : The black and/or gray iKey secrets can be encrypted and cached, so that only a keyboard-entered challenge secret is required to log in. > 0 (default): iKeys must be presented at each login, whether via LunaCM or a client application. This policy is overridden and activation is disabled if a tamper event occurs, or if an uncleared tamper event is detected on reboot. See Tamper Events for more information.
23	Enable auto-activation Always 0 . Not supported on Luna USB HSM 7.	N/A

#	Partition Capability	Partition Policy
25	<p>Minimum PIN length</p> <p>Always 247 (8 characters).</p> <p>The absolute minimum length for a role password/challenge secret is 8 characters. This is displayed as a value subtracted from 255.</p> <p>The reason for this inversion is that a policy can only be set to a value equal to or lower than the value set by its capability. If the absolute minimum length was set to 8, the Partition SO would be able to set the preferred minimum to 2, a less-secure policy. The Partition SO may only change the minimum length to increase security by forcing stronger passwords.</p>	<p>Minimum PIN length</p> <p>The Partition SO can choose to increase the effective minimum length of a role password/challenge secret by setting this policy. The policy value is determined as follows:</p> <p>Subtract the desired minimum length from 255 (the absolute maximum length), and set policy 25 to that value.</p> <p>255 - (desired length) = (policy value)</p> <p>For example, to set the minimum length to 10 characters, set the value of this policy to 245:</p> <p>255 - 10 = 245</p> <p>Default: 247 (8 characters)</p>
26	<p>Maximum PIN length</p> <p>Always 255. The absolute maximum length for a role password/challenge secret is 255 characters.</p>	<p>Maximum PIN length</p> <p>The effective maximum role password/challenge secret length may be changed by the Partition SO. It must always be greater than or equal to the effective minimum length, determined by the formula described in policy 25 (above).</p> <p>Default: 255</p>
28	<p>Enable Key Management Functions</p> <p>Always 1. This capability allows cryptographic objects to be created or deleted on the partition.</p>	<p>Allow Key Management Functions</p> <p><i>Destructive OFF-to-ON</i></p> <ul style="list-style-type: none"> > 1 (default): The Crypto Officer can manage (create/delete) objects on the partition. The Crypto User is restricted to read-only operations. > 0: Partition objects are read-only for both the CO and CU roles.
29	<p>Enable RSA signing without confirmation</p> <p>Always 1. This capability governs the HSM's internal signing verification.</p>	<p>Perform RSA signing without confirmation</p> <p><i>Destructive OFF-to-ON</i></p> <ul style="list-style-type: none"> > 1 (default): No internal signing verification is performed. > 0: The HSM performs an internal verification of signing operations to validate the signature. This has a performance impact on signature operations.

#	Partition Capability	Partition Policy
31	Enable private key unmasking Always 1 . Private keys can be unmasked onto the partition.	Allow private key unmasking <ul style="list-style-type: none"> > 1 (default for V1 partitions): Private keys can be unmasked onto the partition (meaning they also can be migrated from legacy Luna HSMs that used SIM). > 0 (default for V0 partitions): Private keys cannot be unmasked onto the partition (meaning that migration of private keys from legacy HSMs using SIM is also not possible).
32	Enable secret key unmasking Enable unmasking of a secret key onto the partition.	Allow secret key unmasking <ul style="list-style-type: none"> > 1 (default for V1 partitions): Secret keys can be masked and stored onto the partition. > 0 (default for V0 partitions): Secret keys cannot be masked onto the partition.
33	Enable RSA PKCS mechanism Always 1 . The mechanism CKM_RSA_PKCS has known weaknesses, which you can address in your applications. If you are not prepared to address these issues, you can choose to disable the mechanism entirely.	Allow RSA PKCS mechanism <i>Destructive OFF-to-ON</i> <ul style="list-style-type: none"> > 1 (default): CKM_RSA_PKCS is enabled on the partition. Using Luna USB HSM 7 Firmware 7.7.3 or newer, when the partition is in FIPS approved configuration (HSM policy 12: Allow non-FIPS algorithms or partition policy 43: "Allow Non-FIPS algorithms" on page 58 set to 0), the mechanism is disabled even if this policy is set to 1. > 0: CKM_RSA_PKCS is disabled on the partition.

#	Partition Capability	Partition Policy
34	<p>Enable CBC-PAD (un)wrap keys of any size</p> <p>Always 1. There are known vulnerabilities using small keys wrapped/unwrapped with CBC_PAD mechanisms (and with small keys in general). You can choose to enforce a size restriction so that small weak keys cannot be unwrapped onto the partition. The following mechanisms are affected:</p> <ul style="list-style-type: none"> > CKM_AES_CBC_PAD > CKM_AES_CBC_PAD_IPSEC > CKM_ARIA_CBC_PAD > CKM_ARIA_L_CBC_PAD > CKM_CAST3_CBC_PAD > CKM_CAST5_CBC_PAD > CKM_DES_CBC_PAD > CKM_DES3_CBC_PAD > CKM_DES3_CBC_PAD_IPSEC > CKM_RC2_CBC_PAD > CKM_RC5_CBC_PAD > CKM_SEED_CBC_PAD > CKM_SM4_CBC_PAD 	<p>Allow CBC-PAD (un)wrap keys of any size</p> <ul style="list-style-type: none"> > 1 (default): All keys can be wrapped or unwrapped using CBC_PAD mechanisms. > 0: Small keys cannot be wrapped or unwrapped using CBC_PAD mechanisms.
37	<p>Enable enforcing Secure Trusted Channel</p> <p>Always 0. Not applicable to Luna USB HSM 7.</p>	N/A
39	<p>Enable Start/End Date Attributes</p> <p>Always 1. This capability allows you to enforce the CKA_START_DATE and CKA_END_DATE attributes of partition objects.</p>	<p>Allow Start/End Date Attributes</p> <p><i>Destructive ON-to-OFF</i></p> <ul style="list-style-type: none"> > 1: CKA_START_DATE and CKA_END_DATE attributes are enforced for all partition objects. > 0 (default): These attributes can be set for partition objects, but their values are ignored.
40	<p>Enable Per-Key Authorization Data</p> <p>Both assigned and unassigned secret keys (symmetric or private) are given per-key authorization attributes in the form of CKA_AUTH_DATA, in any partition. For V0 partitions, PKA is ignored and applications can use the pre-existing APIs as before. For V1 partitions it is actively used, for eIDAS compliance with newer API.</p>	<p>Require Per-Key Authorization Data</p> <ul style="list-style-type: none"> > 1 (default for V1 partitions): Per-Key Authorization is on by default, but can be turned off for performance. > 0 (only setting for V0 partitions): Per-Key Authorization is off by default, and cannot be turned on - V0 partitions do not allow policy changes that would require new clients.

#	Partition Capability	Partition Policy
41	Enable Partition Version Always 1. This capability allows you to switch a partition between version V0 and V1.	Partition Version <i>Destructive ON-to-OFF</i> <ul style="list-style-type: none"> > 1 : Version 1 (V1) partition supports all features of Luna HSM Firmware 7.7.0 (or newer). <ul style="list-style-type: none"> • cloning is used/permitted only for SMKs • key objects are transferred using SKS > 0 (default): Version 0 (V0) supports older API and your pre-existing applications, enhanced by fixes and security updates, but Per-Key Authorization, SKS, and other V1-dependent features are not available.
42	Enable CPv1 This capability allows the partition to use the cloning protocol required for cloning/HA with Luna partitions with firmware older than 7.7.2, or Luna Cloud HSM services.	Allow CPv1 <i>Destructive OFF-to-ON</i> <ul style="list-style-type: none"> > 1 (default for V0 partitions created when the HSM is in non-FIPS 140 approved configuration): Cryptographic objects on the partition can be cloned to Luna partitions with firmware older than 7.7.2, or to Luna Cloud HSM services. > 0: (only setting for V1 partitions or V0 partitions in FIPS 140 approved configuration): Cryptographic objects on the partition can be cloned only to other partitions with firmware 7.7.2 or newer, with the same FIPS settings. <p>Luna Backup HSM 7 Firmware 7.7.1 cannot restore keys to a partition with CPv1 enabled if they were backed up from a partition with CPv1 disabled. This limitation is restricted to the backup HSM -- you can still use direct slot-to-slot cloning (see "Cloning Objects to Another Application Partition" on page 31).</p>
43	Enable Non-FIPS Algorithms This capability allows the use of algorithms that are not compliant with FIPS, within the current partition. Requires that HSM policy 12 be set to ON (the HSM is in non-FIPS 140 approved configuration (formerly FIPS mode)).	Allow Non-FIPS algorithms <i>Destructive OFF-to-ON</i> <ul style="list-style-type: none"> > 1: (default for new partitions where HSM policy 12 is set to 1): Non-FIPS-compliant algorithms can be used by the partition. > 0: (only setting for partitions where HSM policy 12 is set to 0): Non-FIPS-compliant algorithms cannot be used by the partition.

A number of partition capabilities are linked to the corresponding HSM capabilities and policies including:

- > Partition Policy (0) Enable private key cloning is dependent on HSM Policy (7) Allow cloning;
- > Partition Policy (3) Enable private key masking is dependent on HSM Policy (6) Allow Masking;

- > Partition Policy (4) Enable secret key cloning is dependent on HSM Policy (7) Allow cloning;
- > Partition Policy (7) Enable secret key masking is dependent on HSM Policy (6) Allow Masking;
- > Partition Policy (22) Enable Activation is dependent on HSM Policy (1) Allow PED-based authentication;
- > Partition Policy (31) Enable private key unmasking is dependent on HSM Policy (6) Allow Masking; and
- > Partition Policy (32) Enable secret key unmasking is dependent on HSM Policy (6) Allow Masking.

In addition – the following dependencies within the partition level policies are observed:

- > Partition Policy (7) Allow cloning cannot be enabled at the same time as Partition Policy (1) Allow private key wrapping;
- > Partition Policy (1) Allow private key wrapping cannot be enabled at the same time as either one of the policies, Partition Policy (0) Enable private key cloning, Partition Policy (3) Allow private key masking, Partition Policy (31) Enable private key unmasking;
- > Partition Policy (23) Allow Activation is dependent on Partition Policy (22) Allow Activation being enabled;
- > Partition Policies related to 'Masking' (3, 7, 31 and 32) can only be enabled when Partition Policy (41) Partition Version is '0'; and
- > Partition Policy (41) Partition Version cannot be set to '1' at the same time as either Partition Policy (40) Enable Per-Key Authorisation Data or any of the Partition Policies covering key masking (3, 7, 31 and 32).
- > Partition Policy (40) Enable Per-Key Authorisation Data is enabled by default but is disabled if Partition Policy (41) Partition Version is set to '0'.

Setting Partition Policies Manually

The Partition Security Officer can change available policies to customize partition functionality. Policy settings apply to all roles/objects on the partition. Refer to ["Partition Capabilities and Policies" on page 49](#) for a complete list of partition policies and their effects.

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

To change multiple policy settings during partition initialization, see ["Setting Partition Policies Using a Template" on the next page](#).

See also ["Cloning or Export of Private Keys" on page 64](#).

Prerequisites

- > The partition must be initialized (see ["Initializing the Application Partition" on page 35](#)).
- > If you are changing a destructive policy, back up any important cryptographic objects (see ["Partition Backup and Restore" on page 106](#)).

NOTE If you are running more than one LunaCM session against the same partition, and change a partition policy in one LunaCM session, the new policy setting is visible in that session only (although it is in effect). You must exit and restart the other LunaCM sessions to display the new policy setting.

To manually set or change a partition policy

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

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

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

lunacm:> **partition showpolicies**

3. Log in as Partition SO (see ["Logging In to the Application Partition" on page 40](#)).

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

4. Change the policy setting by specifying the policy number and the desired value (**0**, **1**, or a number in the accepted range for that policy). You can specify multiple policy changes in the same command by using comma-separated lists (for example, **-policy 33,37,40 -value 0,1,1**).

lunacm:> **partition changepolicy -policy** <policy_ID> **-value** <value>

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

Setting Partition Policies Using a Template

A partition policy template is a file containing a set of preferred partition policy settings, used to initialize partitions with those settings. You can use the same file to initialize multiple partitions, rather than changing policies manually after initialization. This can save time and effort when initializing partitions that are to function as an HA group, or must comply with your company's overall security strategy. Templates enable scalable policy management and simplify future audit and compliance requirements.

See also [Setting HSM Policies Using a Template](#).

You can create a partition policy template file from an initialized or uninitialized partition, and edit it using a standard text editor. Partition policy templates have additional customization options.

Policy templates cannot be used to alter settings for an initialized partition. Once a partition has been initialized, the Partition SO must change individual policies manually (see ["Setting Partition Policies Manually" on the previous page](#)).

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

- > ["Creating a Partition Policy Template" below](#)
- > ["Editing a Partition Policy Template" on the next page](#)
- > ["Applying a Partition Policy Template" on page 63](#)

Creating a Partition Policy Template

The following procedure describes how to create a policy template for a partition. This can be done optionally at two points in the partition setup process:

- > before the partition is initialized: this produces a template file containing the default policy settings, which can then be edited
- > after initializing and setting the partition policies manually: this produces a template file with the current policy settings, which can then be used to initialize other partitions with the same settings. The Partition SO must complete the procedure.

To create a partition policy template

1. Launch LunaCM and set the active slot to the partition. If you are creating a template from an initialized partition, you must log in as Partition SO.
 lunacm:> **slot set -slot** <slotnum>
 lunacm:> **role login -name** po
2. Create the partition policy template file. Specify an existing save directory and original filename. No file extension is required. If a template file with the same name exists in the specified directory, it is overwritten.

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

```
lunacm:> partition showpolicies -exporttemplate /usr/safenet/lunaclient/templates/ParPT
```

```
Partition policies for Partition: myPartition1 written to
/usr/safenet/lunaclient/templates/ParPT
```

Command Result : No Error

Editing a Partition Policy Template

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

Partition Policy Template Example

This example shows the contents of a partition policy template created using the factory default policy settings. Use a standard text editor to change the policy and/or destructiveness values (0=OFF, 1=ON, or the desired value 0-255).

Partition policy template entries have two additional fields: **Off to on destructive** and **On to off destructive** (see example below). Change these values to **0** or **1** to determine whether cryptographic objects on the partition should be deleted when this policy is changed in the future. Policies that lower the security level of the objects stored on the partition are normally destructive, but it may be useful to customize this behavior for your own security strategy. See ["Partition Capabilities and Policies" on page 49](#) for more information.

CAUTION! Setting policy destructiveness to **0** (OFF) makes partitions less secure. Use this feature only if your security strategy demands it.

If you export a policy template from an uninitialized partition, the **Sourced from partition** header field remains blank. This field is informational and you can still apply the template.

The **Policy Description** field is included in the template for user readability only. Policies are verified by the number in the **Policy ID** field.

```
# Policy template FW Version 7.1.0
# Field format - Policy ID:Policy Description:Policy Value:Off to on destructive:On to off
destructive
# Sourced from partition: myPartition1, SN: 154438865290
```

```
0:"Allow private key cloning":1:1:0
1:"Allow private key wrapping":0:1:0
2:"Allow private key unwrapping":1:0:0
3:"Allow private key masking":0:1:0
```

```

4:"Allow secret key cloning":1:1:0
5:"Allow secret key wrapping":1:1:0
6:"Allow secret key unwrapping":1:0:0
7:"Allow secret key masking":0:1:0
10:"Allow multipurpose keys":1:1:0
11:"Allow changing key attributes":1:1:0
15:"Ignore failed challenge responses":1:1:0
16:"Operate without RSA blinding":1:1:0
17:"Allow signing with non-local keys":1:0:0
18:"Allow raw RSA operations":1:1:0
20:"Max failed user logins allowed":10:0:0
21:"Allow high availability recovery":1:0:0
22:"Allow activation":0:0:0
23:"Allow auto-activation":0:0:0
25:"Minimum pin length (inverted 255 - min)":248:0:0
26:"Maximum pin length":255:0:0
28:"Allow Key Management Functions":1:1:0
29:"Perform RSA signing without confirmation":1:1:0
31:"Allow private key unmasking":1:0:0
32:"Allow secret key unmasking":1:0:0
33:"Allow RSA PKCS mechanism":1:1:0
34:"Allow CBC-PAD (un)wrap keys of any size":1:1:0
39:"Allow Start/End Date Attributes":0:1:0
40:"Require Per-Key Authorization Data":0:1:0
41:"Partition Version":0:0:1

```

Editing Guidelines and Restrictions

When creating or editing partition policy templates:

- > You can remove a policy from the template by adding **#** at the beginning of the line or deleting the line entirely. When you apply the template, the partition will use the default values for that policy.
- > Partition policy templates from older Luna versions (6.x or earlier) cannot be applied to Luna 7.x partitions.
- > This version of the partition policy template feature is available on Luna 7.x application partitions only. When the active slot is set to the Admin partition, the **-exporttemplate** option is not available. To create an HSM policy template from the Admin partition, see [Setting HSM Policies Using a Template](#).
- > The following restrictions apply when configuring partitions for Cloning or Key Export (see ["Cloning or Export of Private Keys" on page 64](#) for more information):
 - **Partition policy 0: Allow private key cloning** and **partition policy 1: Allow private key wrapping** can never be set to **1** (ON) at the same time. Initialization fails if the template contains a value of **1** for both policies.
 - **Partition policy 1: Allow private key wrapping** must always have **Off-to-on** destructiveness set to **1** (ON). Initialization fails if the template contains a value of **0** in this field.
- > You may not use invalid policy values (outside the acceptable range), or values that conflict with your HSM or partition's capabilities. For example, **Partition capability 3: Enable private key masking** is always **0**, so you cannot set the corresponding partition policy to **1**. If you attempt to initialize a partition with a template containing invalid policy values, an error is returned and initialization fails.

If there is a mismatch between template policies and the default values of newer or dependent policies, then the attempt to apply the old policy would fail with **CKR_FAILED_DEPENDENCIES**.

You have the option to edit a policy file before applying it, to add newer policies.

Applying a Partition Policy Template

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

To apply a policy template to a new partition

1. Ensure that the template file is saved on the client workstation.
2. Launch LunaCM and set the active slot to the new partition.
`lunacm:> slot set -slot <slotnum>`
3. Initialize the partition, specifying a label and the policy template file. If the template file is not in the same directory as LunaCM, include the correct filepath.
`lunacm:> partition init -label <label> -applytemplate <filepath/filename>`
4. [Optional] Verify that the template has been applied correctly by checking the partition's policy settings. Include the **-verbose** option to view the destructiveness settings.
`lunacm:> partition showpolicies [-verbose]`

CHAPTER 6: Cloning or Export of Private Keys

By default, the Luna USB HSM 7 stores all keys in hardware, allowing private keys to be copied only to another Luna HSM (cloning). Cloning allows you to move or copy key material from a partition to a backup HSM or to another partition in the same HA group. You might, however, want to export private keys to an encrypted file for off-board storage or use. Individual partitions can be configured in one of three modes for handling private keys.

The Partition SO can set the mode by changing the following policies (see ["Partition Capabilities and Policies" on page 49](#) for more information):

- > **Partition policy 0: Allow private key cloning** (default: 1)
- > **Partition policy 1: Allow private key wrapping** (default: 0)

NOTE These partition policies can never be set to 1 (ON) at the same time. An error will result (CKR_CONFIG_FAILS_DEPENDENCIES).

The policies can be set at the time of initialization, using a policy template (see ["Setting Partition Policies Using a Template" on page 60](#)) or by following the procedures described below:

- > ["Setting Cloning Mode on a Partition" below](#)
- > ["Setting Key Export Mode on a Partition" on the next page](#)
- > ["Setting No Backup Mode on a Partition" on page 66](#)

NOTE Partition configurations are listed in LunaCM as "Key Export With Cloning Mode". This indicates that the partition is capable of being configured for either Key Export or Cloning, with the mode of operation defined by the policies listed above. You can never configure a partition to allow both export and cloning of private keys at once.

Setting Cloning Mode on a Partition

A partition in Cloning mode has the following capabilities and restrictions:

- > All keys/objects can be cloned to another partition or Luna Backup HSM in the same cloning domain.
- > All keys/objects are replicated within the partition's HA group.
- > Private keys cannot be wrapped off the HSM (cannot be exported to a file encrypted with a wrapping key).

In this mode, private keys are never allowed to exist outside of a trusted Luna HSM in the designated cloning domain.

Cloning mode is the default setting for new partitions. If another mode was set previously, the Partition SO can use the following procedure to set Cloning mode. Use `lunacm:> partition showpolicies` to see the current policy settings.

CAUTION! Partition policy 0: Allow private key cloning is Off-to-On destructive by default. Back up any important cryptographic material on the partition before continuing. This destructiveness setting can be customized by initializing the partition with a policy template (see ["Editing a Partition Policy Template" on page 61](#)).

To manually set Cloning mode on a partition

1. Log in to the partition as Partition SO.
lunacm:> **slot set -slot** <slotnum>
lunacm:> **role login -name po**
2. Set partition policy 1: Allow private key wrapping to 0 (OFF).
lunacm:> **partition changepolicy -policy 1 -value 0**
3. Set partition policy 0: Allow private key cloning to 1 (ON).
lunacm:> **partition changepolicy -policy 0 -value 1**

To initialize a partition in Cloning mode using a policy template

Use a standard text editor to include the following lines in the policy template file (see ["Editing a Partition Policy Template" on page 61](#)):

```
0:"Allow private key cloning":1:1:0
1:"Allow private key wrapping":0:1:0
```

Setting Key Export Mode on a Partition

A partition in Key Export mode has the following capabilities and restrictions:

- > Private keys cannot be cloned to other partitions nor to a Luna Backup HSM.
- > The partition cannot be part of an HA group (private keys will not be replicated).
- > All keys/objects, including private keys, can be wrapped off the HSM (can be exported to a file encrypted with a wrapping key).

This mode is useful when generating key pairs for identity issuance, where transient key-pairs are generated, wrapped off, and embedded on a device. They are not used on the HSM, but generated and issued securely, and then deleted from the HSM.

The Partition SO can use the following procedure to set Key Export mode. Use lunacm:> **partition showpolicies** to see the current policy settings.

CAUTION! Partition policy 1: Allow private key wrapping is always Off-to-On destructive. Back up any important cryptographic material on the partition before continuing. This destructiveness setting cannot be changed with a policy template (see ["Editing Guidelines and Restrictions" on page 62](#)).

To manually set Key Export mode on a partition

1. Launch LunaCM and log in to the partition as Partition SO.

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

```
lunacm:> role login-name po
```

2. Set **partition policy 0: Allow private key cloning** to **0** (OFF).

```
lunacm:> partition changepolicy -policy 0 -value 0
```

3. Set **partition policy 1: Allow private key wrapping** to **1** (ON).

```
lunacm:> partition changepolicy -policy 1 -value 1
```

To initialize a partition in Key Export mode using a policy template

Use a standard text editor to include the following lines in the policy template file (see ["Editing a Partition Policy Template" on page 61](#)):

```
0:"Allow private key cloning":0:1:0
1:"Allow private key wrapping":1:1:0
```

Setting No Backup Mode on a Partition

A partition in No Backup mode has the following restrictions:

- > Private keys cannot be cloned to other partitions or to a Luna Backup HSM. All other objects can still be cloned.
- > Private keys cannot be wrapped off the HSM (exported to a file encrypted with a wrapping key). All other objects can still be wrapped off.

Without backup capability, private keys can never leave the HSM. This mode is useful when keys are intended to have short lifespans, and are easily replaced.

The Partition SO can use the following procedure to set No Backup mode. Use `lunacm:> partition showpolicies` to see the current policy settings.

To manually set No Backup mode on a partition

1. Launch LunaCM and log in to the partition as Partition SO.

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

```
lunacm:> role login -name po
```

2. If **partition policy 0: Allow private key cloning** is set to **1** (ON), set it to **0** (OFF).

```
lunacm:> partition changepolicy -policy 0 -value 0
```

3. If **partition policy 1: Allow private key wrapping** is set to **1** (ON), set it to **0** (OFF).

```
lunacm:> partition changepolicy -policy 1 -value 0
```

To initialize a partition in No Backup mode using a policy template

Use a standard text editor to include the following lines in the policy template file (see ["Editing a Partition Policy Template" on page 61](#)):

```
0:"Allow private key cloning":0:1:0
1:"Allow private key wrapping":0:1:0
```

CHAPTER 7: V0 and V1 Partitions

Luna HSMs traditionally store sensitive cryptographic objects within the HSM only, except for cloning to another Luna HSM, or secure wrapping for export. The latest Luna USB HSM 7 firmware provides a third option, ["Scalable Key Storage" on page 70](#) (SKS), which allows you to securely store more keys than will fit in HSM storage. You can enable these different options using partition policies, to select one of the following partition types:

Version 0 (V0) Partitions	Version 1 (V1) Partitions
<p>Cryptographic objects are securely stored in the HSM hardware. A partition can be configured to allow private and secret keys to leave the partition by only one of the following methods:</p> <ul style="list-style-type: none">> Cloning to another Luna HSM partition<ul style="list-style-type: none">• "Partition Backup and Restore" on page 106• "High-Availability Groups" on page 80• "Domain Planning and Key Cloning" on page 28> Secure wrapping for export off the partition<ul style="list-style-type: none">• "Setting Key Export Mode on a Partition" on page 65 <p>Refer to "Cloning or Export of Private Keys" on page 64 to enable one of these modes on a V0 partition.</p>	<p>On a V1 partition, one of the V0 modes (Cloning or Key Export) can be configured. V1 partitions also enable the following features:</p> <ul style="list-style-type: none">> "Scalable Key Storage" on page 70 allows secure storage of cryptographic objects in a database outside the HSM. This lets you manage objects without the storage restrictions of an application partition.> "Per-Key Authorization" on page 78 allows you to assign an authentication credential to individual keys, so they can only be accessed by authorized users. <p>These features are introduced to conform to FIPS SP 800-131A (revised), and to comply with current and anticipated Common Criteria and eIDAS requirements.</p>

You can migrate existing keys and objects from a Luna application partition that was created before the introduction of V0 and V1 partitions. The following sections provide additional detail about differences between the partition types and implications of switching between them.

Setting V0 or V1 on an Application Partiton

The partition version is determined by **partition policy 41** (see ["Partition Capabilities and Policies" on page 49](#)), set to V0 by default on a newly-created partition. There are three ways to set V1 on an application partition:

- > The HSM SO can set V1 at the time of partition creation (see [Creating the Application Partition](#))
- > The Partition SO can set policy 41 to 1 during initialization, using a policy template (see ["Setting Partition Policies Using a Template" on page 60](#))
- > The Partition SO can convert the default V0 partition to V1 at any time after initializing the partition (see ["Setting Partition Policies Manually" on page 59](#)). Converting a partition from V0 to V1 is non-destructive; converting from V1 to V0 is destructive, resulting in the loss of all objects on the partition.

Special Characteristics of V0 Partitions

V0 partitions maintain all the same functionality and behavior as Luna HSM partitions using firmware prior to Luna HSM Firmware 7.7.0. Keys reside only within the secure HSM hardware and can be transferred to another Luna partition that is part of the same cloning domain (backup/restore, HA groups, slot-to-slot), or wrapped and exported off the partition, depending on the partition policy settings (see ["Cloning or Export of Private Keys" on page 64](#)).

V0 partitions use a new cloning protocol with enhanced security. Since Luna HSMs do not allow cloning objects from more-secure to less-secure environments, objects can be cloned from older firmware partitions to V0 partitions, but not from V0 partitions to partitions using older Luna HSM firmware. This has implications for various HSM functions:

- > **Partition storage overhead:** V0 partitions require more space than older firmware partitions, mainly for new attributes (related to V1 functionality) that are added to cryptographic keys. These attributes become relevant only when a V0 partition is converted to a V1 partition.

The Luna USB HSM 7 provides more than enough HSM storage space to import all objects stored on an older Luna HSM.

- > **High-Availability Groups:** V0 partitions can work in an HA group with other V0 partitions. You can create an HA group with a mix of V0 and older-firmware partitions only for the purposes of migrating keys to the V0 partition. Do not attempt to use V0 partitions in a production HA group including partitions with firmware older than Luna HSM Firmware 7.7.0.

- > **Backup/Restore:** V0 partitions have the same Backup HSM firmware requirements as V1 partitions:

- [Luna Backup HSM 7 Firmware 7.7.1](#)
- [Luna Backup HSM G5 Firmware 6.28.0](#)

A Backup HSM with older firmware may be used to migrate objects from older Luna HSM firmware to a V0 partition, but this is a one-way operation only.

Special Characteristics of V1 Partitions

As described above, V1 partitions add new features that change the way objects are securely stored on the HSM (["Scalable Key Storage" on page 70](#)) and the function of individual keys (["Per-Key Authorization" on page 78](#)). In addition to these new features, the new cloning protocol has implications for various HSM functions:

- > **SKS Master Keys:** Keys that are exported to a database using ["Scalable Key Storage" on page 70](#) are encrypted with an SKS Master Key (SMK) on the partition. The SMK is generated when the Crypto Officer logs in for the first time, but can be replaced with another primary SMK cloned from another partition. During cloning operations (HA, backup/restore, slot-to-slot cloning), only the SMK is cloned, and only to another V1 partition. Key objects remain in the secure database and they cannot be cloned to another partition. Replication or archiving of objects is done using SKS only.

Each V1 partition also has additional SMK slots or holding areas for:

- Rollover SMK
- SMKs from earlier-model HSMs
- FM SMK for partitions with Functionality Modules enabled

The Primary SMK secret is used to extract and to insert keys/objects; all other SMK secrets can be used only to insert keys/objects.

NOTE For older Luna versions, or situations where only cloning protocol version one (CPv1) is available, the library attempts to perform the individual actions of a cloning operation in sequence on the respective partitions, opening and closing a separate session for each object to be copied. If the policies and partition types on the source and target partitions are incompatible, the **partition clone** command (or an attempted HA synchronization) can fail with a message like CKR_DATA_LEN_RANGE while trying to clone. This can occur if a key object from the source partition is a different size than an equivalent object expected by the target.

UPDATE: Using Luna HSM Firmware 7.8.0 and newer, when a cloning negotiation agrees on the use of CPv4, a call to clone multiple keys/objects launches a *single session for all the requested objects*, rather than opening and closing individual sessions for each object. The above portion of this note about mismatched sizes remains valid.

- > **Partition storage overhead:** V1 partitions require more space than older firmware partitions, for SMKs and new key attributes used for V1 features. This additional overhead also applies to V0 partitions, but the features only become active when the partition is converted to V1.
- > **Partition Policy 40: Enable Per-Key Authorization:** This policy is enabled by default on V1 partitions. If you do not plan to use ["Per-Key Authorization" on page 78](#), you can disable this policy to improve performance.
- > **High-Availability Groups:** V1 partitions use SKS as the method of object replication among group members, rather than cloning.
- > **Limited Crypto Officer:** V1 partitions have an additional Limited Crypto Officer role, which enables ["Per-Key Authorization" on page 78](#) (see ["Partition Roles" on page 37](#)).
- > **Backup/Restore:** V1 partitions have the same Backup HSM firmware requirements as V0 partitions:
 - [Luna Backup HSM 7 Firmware 7.7.1](#)
 - [Luna Backup HSM G5 Firmware 6.28.0](#)

A Backup HSM with older firmware may be used to migrate objects from older Luna HSM firmware to a V0 partition, but this is a one-way operation only.

CHAPTER 8: Scalable Key Storage

Scalable Key Storage (SKS) is virtually unlimited secure storage and handling of your sensitive keys.

By default, Luna HSMs have always stored keys in the HSM hardware. SKS expands the HSM's assurance boundary to a securely encrypted database, allowing you to store many more keys than would be possible in HSM hardware. All V1 partitions use SKS (see ["V0 and V1 Partitions" on page 67](#)). With SKS, keys generated on the application partition are encrypted with an SKS Master Key (SMK), securely extracted to a database for storage, and inserted back onto the partition to perform cryptographic operations. When a unique key encrypts data, the key and data can be stored as an encrypted **binary large object (blob)**, up to 64 KB in size, that can be decrypted only on the partition. Luna HSM Client provides an SKS API so that your applications can use V1 partitions to work with SKS objects.

This section contains the following information about SKS:

- > ["The SKS Model" below](#)
- > ["When to use SKS" on page 72](#)
- > ["SKS Master Key Types" on page 72](#)
- > ["High Availability and SKS" on page 73](#)
- > ["Backup/Restore and SKS" on page 74](#)
- > ["Using SKS" on page 74](#)
- > ["Changing the SMK" on page 76](#)

The SKS Model

Various models exist for handling large numbers of sensitive keys and objects:

- > **Wrap-off/wrap-on:** Keys and objects can have unknown, uncontrolled origin, outside the assurance boundary. They can be accessed outside the HSM, made available and used externally, in potentially unsafe environments. It is possible to strip security attributes from keys.
- > **SKS extract/insert:** The history of keys and objects is known, controlled, and auditable. They remain within the security and access envelope of the HSM. The master key never exists outside a Luna HSM, and all extracted keys and objects must be inserted back into the HSM at time of decryption and use. Keys always retain their attributes.

In an SKS model, in compliance with relevant standards, an application maintains thousands or millions of encrypted objects as records in a repository (such as a database, file system, cloud storage, etc). The repository might have each record/object encrypted with a unique key. Examples of applications include Remote Signing identities (Common Criteria PP 419221-5 use case). The SKS model provides the following assurances:

- > encryption and decryption of objects take place within the HSM
- > more individual object-encryption keys are needed by the application than can be accommodated by the internal capacity of any HSM

- > records or objects, and the keys that encrypt them, do not exist in-the-clear - both the record (data object, ID, etc.) and its encrypting key are stored in encrypted form
- > keys that encrypt objects or signatures originate within the assurance boundary and are only ever decrypted within the assurance boundary
- > objects extracted from a current-version HSM cannot be inserted into older version HSMs with known vulnerabilities

A key is created in an HSM partition at the direction of an application. It might be intended as an ID for signing documents and verifying by private persons, or for sealing of documents and records by organizations. It might be intended to encrypt records stored in an external database (customer-identifying records, medical records, supply-chain information, or other information that requires privacy and controlled access). Each key is encrypted for extraction by an extraction/insertion key, derived (in compliance with NIST SP800-108) from the SKS Master Key (SMK), a master encryption key that never leaves the HSM partition. From the application's perspective, the data record or key is extracted from the partition uniquely encrypted as a secure SKS blob, which can be securely stored anywhere.

Because the SKS objects are stored outside the HSM and individually inserted back into the HSM partition for use, there is *no capacity limitation*. The only limitation on scalability is the number of SKS operations that can be performed simultaneously (SKS object creation/extract/insert, and resulting cryptographic operations).

An application might use the SKS API to perform any of the following actions:

- > create an identity or a record or data object
- > acquire a suitable key for encrypting that record or data object, by:
 - request a new object-encryption key be generated by the HSM
 - provide an already existing object-encryption-key for the HSM to use
- > encrypt that ID or object with the new SKS key, or the pre-existing SKS key, which must first be inserted and decrypted for use by the HSM
- > store the encrypted record or key within the repository
- > retrieve the encrypted record or key
- > insert/decrypt the SKS blob into the HSM, using the SKS Master Key (SMK)
- > use the decrypted key to:
 - sign or seal documents or transactions in the case of RSS
 - further decrypt a database record for reading or editing, re-encrypt the record if it changed, and send the re-encrypted record back to storage
- > delete / destroy the material from the HSM, once it is not needed (the encrypted SKS blob still exists in the external repository, for the next time it is required)

The application is responsible for storing the SKS object in the repository of choice (database, file system, directory, NAS, cloud, etc.) and retrieving it.

It is possible to create data objects to store any kind of data in an HSM partition, SKS blobs included (which is essentially what is done if you choose to archive SKS objects in a Backup HSM), but that is not the ideal workflow. Instead, a practical workflow is assumed to include backing up SMKs, but not SKS blobs, since the latter are already securely encrypted and can be stored anywhere that is reasonably secure, and in quantity far greater than the capacity of any HSM. However, we cannot anticipate all use-cases, so the onboard storage option exists.

Optionally, such as in the case of Trust Service Providers, during the SKS object creation process, authentication can be added such that a password must be provided before the keys in an SKS object can be used. SKS objects use 256-bit AES-GCM encryption for confidentiality and integrity protection. SHA-512 is also used for further integrity protection. The cryptographic mechanisms employed by SKS comply with the FIPS 140-2 and PP 419221-5 standards ([Secure External Scalable Key Storage Extensions](#)). The SKS mechanism complies with the per-key authorization requirements of Common Criteria PP 419221-5 ("[Per-Key Authorization](#)" on page 78).

When to use SKS

Use SKS when you need to handle greater numbers of keys and objects than can be stored within the HSM, and you want to employ methods more secure than wrap-off / wrap-on. SKS is required to comply with a regulatory regime like eIDAS.

Any application where large numbers of very sensitive keys or records must be protected with the highest possible security, while remaining available and accessible to authorized users and applications, is a candidate for the Luna HSM with Scalable Key Storage.

A general use case for SKS is storing encrypted keys in external databases.

1. Generate keys inside the HSM
2. Using the SIMExtract API, extract the encrypted keys and store them in external databases and delete the original keys inside the HSM.
3. Insert individual encrypted keys back into the HSM when you need to use them for cryptographic operations inside the HSM.

One example might be the creation and use of electronic signatures (for natural persons) or electronic seals (for organizations) for remote signing (RSS). The signatures or seal key materials are created within the HSM, extracted (not wrapped) in strongly encrypted form that preserves attributes, and stored in a repository. When they are needed, they are found in the repository by the managing system, inserted into the HSM for decryption by a master key that never resides outside an HSM, then used for signing or sealing respectively, and discarded from the HSM (the encrypted versions remain stored in the repository for the next time they are needed).

Another example might be a database of customers, with their contact and shipping information, credit-card information, history of purchases, current/recent browse interests on your commerce site, etc. All of that is likely to be sensitive information protected by regulations and by your own published privacy policies. In this case, the primary concern is privacy of data.

A third example might be a government database of land ownership, including detailed and official property descriptions, current ownership with identifying details, history of title transfers, subdivisions, legal rulings and encumbrances (such as rights of way and covenants), liens, and so on. In this case, the data is meant to be publicly viewable, but its integrity against unauthorized change is paramount.

SKS Master Key Types

Each SKS-capable partition supports four unique SMKs, each with its own location and limitations within the partition:

- > **Primary SMK:** Generated on a V0 or V1 partition at creation, or replaced with a primary SMK from another partition using `lunacm:> partition smkclone`. The Primary SMK is used for object extraction and insertion operations.
- > **Rollover SMK:** When you generate a new primary SMK, the old primary is referred to as the rollover SMK, and temporarily stored in its own location on the partition. This allows all SKS blobs that were encrypted/extracted with the old SMK to be brought back into the partition, decrypted, and re-encrypted with the new primary SMK. When the rollover operation is complete, the rollover SMK is deleted and only the new primary SMK remains. See "[Changing the SMK](#)" on page 76.
- > **FM SMK:** This SMK is not used on the Luna USB HSM 7.
- > **Firmware 6 SMK:** This SMK is imported from a Luna HSM using firmware version 6.x, for the purposes of migrating SKS blobs from a legacy Luna 6 HSM. After the migrated blobs are inserted and decrypted, the cryptographic objects are encrypted with the primary SMK and extracted as Luna 7 SKS blobs. Whenever a partition is on the receiving end of a `partition smkclone` operation, any contents of the primary and non-primary locations from the source partition overwrite their equivalent locations in the target partition.

High Availability and SKS

SKS supports high availability configurations similar to the Luna HSM cloning model (see "[High-Availability Groups](#)" on page 80), with some minor differences. High availability and load balancing is implemented in the Luna HSM Client software and is completely transparent to the application, in that the application is configured to use a virtual slot and not a physical slot on the HSM.

In HA groups using SKS, `lunacm:> hagroup addmember` clones the SMK from the primary HA member to each additional member as it is added.

One difference, from cloning HSMs in HA configuration is that, for SKS HA, the `hagroup addmember` command clones the SMK from the initial SKS application partition to all other group member partitions as they are added. Thereafter, your application deals with the HA virtual slot, and HA operation is automatic.

NOTE V1 partitions: If you add an application partition with an existing SMK to an HA group, the primary member's SMK overwrites the existing SMK of the joining partition.

If a partition's SMK has ever been used to encrypt important SKS objects, save a backup of the SMK before adding that partition to any HA group.

To perform a cryptographic operation, the application calls the SKS API to import an SKS blob into the HSM. In an HA group with HA synchronization enabled, Luna HSM Client replicates the imported blob to all member partitions by performing `sksExtract` on the source partition and `sksInsert` on the target partition, combined into a single operation and repeated for each HA group member. When the application requests a cryptographic operation from the HA virtual slot, Luna HSM Client load-balances requests among the group member partitions. The SKS operation succeeds because all partitions in the HA group have a copy of the imported SKS object.

NOTE HA failover is not supported in the case of member failure during a `SIMInsert`, `SIMExtract`, or `SIMMultisign` operation.

TIP If your primary use-case is to insert a key and use it for one signing operation, consider using the multisign API for better HA group performance. This allows you to use the key on only one HA member partition, and prevent unnecessary replication to the other members. If this will be the main function of your HA group, disable HA synchronization to prevent keys from being unnecessarily cloned to other members.

If inserted keys are likely to be used for multiple load-balanced operations, then the overhead of replicating to all members is unavoidable and would be minimal in that context.

Backup/Restore and SKS

For most SKS implementations, only the SKS Master Key (SMK) is stored on the application partition. Cryptographic objects are stored as blobs in a database, encrypted by the SMK. Therefore, backup operations performed on a V1 partition include the SMK only by default. If you decide to store persistent SKS blobs on the partition, these are included in backup/restore operations, unless you include the **-smkonly** option. SKS backup and restore procedures are the same as standard backup/restore (see ["Partition Backup and Restore" on page 106](#)).

Using SKS

This section describes prerequisites for using SKS on a Luna USB HSM 7 application partition, and provides example workflows using the API and the [ckdemo](#) utility.

Prerequisites

- > You require at least one initialized V1 application partition available as a slot in LunaCM (see ["Initializing the Application Partition" on page 35](#)).
- > The Crypto Officer role must be initialized (see ["Initializing the Crypto Officer Role" on page 42](#)). The SMK is created when the CO logs in to the partition for the first time.

NOTE For security reasons, the SMK is not visible in the output of commands that show objects on an HSM partition (`lunacm:> partition contents`).

- > Back up the SMK to a Luna Backup HSM (see ["Partition Backup and Restore" on page 106](#)). This is highly recommended, since the loss of the SMK for any reason results in all cryptographic objects encrypted by the SMK becoming unrecoverable.

Using SKS with the PKCS#11 API

Authorization forms currently supported are none, and password.

Export a key from a partition as an SMK-encrypted blob, using `SIMExtract` function. You can extract all key objects within a given partition by specifying an empty list on the input. Otherwise, specify only individual objects that you wish to extract at one time.

```
SIM_AUTH_FORMS = (CKA_SIM_NO_AUTHORIZATION,
                  CKA_SIM_PASSWORD)
CK_RV CA_SIMExtract(CK_ULONG handleCount, CK_ULONG *handleList,
```

```

CK_ULONG authForm, CK_ULONG authDataCount, CK_ULONG subsetRequired,

CK_BYTE **authDataList,

CK_BOOL deleteAfterExtract,

CK_ULONG *pBlobSize, CK_BYTE *pBlob );

```

Import a previously extracted blob, using the `SIMInsert` function

```

CK_RV SIMInsert( CK_ULONG blobSize, CK_BYTE *pBlob,

CK_ULONG authForm, CK_ULONG authDataCount, CK_BYTE **authDataList,

CK_ULONG *pHandleListSize, CK_ULONG *pHandleList );

```

Refer to the SDK Guide ([Secure External Scalable Key Storage Extensions](#)).

Workflow Example Using `ckdemo`

This example uses the `ckdemo` utility to guide you through a possible SKS workflow.

1. Start by running `ckdemo` and executing:
 - a. **Open Session (1)** to the slot
 - b. **Login (3)** as **Crypto Officer**. Enter the partition password.
2. Generate an AES key using **Simple Generate Key (45)** and note the object handle for the generated key.
3. Execute **SIMExtract (105)**.
 - a. Enter the object handle for **Enter handle of object to add to blob**.
 - b. Enter **0** to **end the list**.
 - c. Enter **1** for **Enter authentication form**.
 - d. Enter **1** for **number of authorization secrets (N value)**.
 - e. Enter **1** for **Enter subset size required for key use (M value)**.
 - f. Enter a password.
 - g. Enter **1** for **Delete after extract**.
 - h. The masked key is saved to **blobfile.sim**.
4. List all of the objects in the partition by running **Find object (26)** with option **All Standard Objects (6)**.
5. Execute **SIMInsert (106)**.
 - a. Enter **blobfile.sim** for **Enter filename with object to insert**.
 - b. Enter **1** for **Enter authentication form**.
 - c. Enter **1** for **Enter number of authorization secrets to be provided**.
 - d. Enter the password that was entered in the previous step.
6. List all of the objects in the partition by running **Find object (26)** with option **All Standard Objects (6)**. The key that was extracted should now be present in the partition.

NOTE The example above uses the password authentication form. Other authentication forms can be used.

Using the Provided Java Sample

As a prerequisite, ensure that the **LunaProvider.jar** and **libLunaAPI.so** has been installed to your JDK.

1. Navigate to the directory that contains the java sample:
cd JavaSample
2. In the **SIMExtractInsert.java**, modify the **slot** and **hsmPass** variables appropriately.
3. Compile the sample using **javac**:
javac SIMExtractInsert.java
4. Run the sample using java.

Changing the SMK

Your organization may have mandatory rollover schedules that govern the lifetime of important keys. Therefore, it may be necessary to change the SMK after a set time. Follow the procedure below to change the SMK on a partition or HA group.

CAUTION! SMK rollover is a disruptive process and can result in significant down-time. Plan it appropriately before you continue.

If you create a new SMK on the partition, every blob in the database encrypted by the old SMK must be inserted to the partition, decrypted, the objects re-encrypted with the new SMK and extracted as a new blob, and stored back in the database. Luna USB HSM 7 stores both the new SMK and the old SMK simultaneously until this process is complete, and the old SMK is then deleted.

Prerequisites

- > Stop all applications using the partition or HA group.
- > Dismantle the HA group by removing all but the primary member, or delete the HA group.

NOTE For HA environments, if you perform SMK rollover on a member, then the new SMK must be cloned to all members. However, database / repository update for rollover should be done by directly addressing the primary physical member, and *not* using the virtual slot (to avoid the performance penalty when keys inserted to the virtual slot during rollover would be propagated to all members before the re-extraction).

- > Ensure that you have access to all databases, repositories, and backups where blobs encrypted by the old SMK are stored.

To roll over the SMK

1. In LunaCM, log in to the partition as Crypto Officer.

lunacm:> `role login -role co`

2. Initiate the rollover procedure by creating a new SMK on the partition.

lunacm:> `partition smkrollover -start`

The new SMK is stored in the **Primary SMK** location on the partition (see "SKS Master Key Types" on page 72). The old SMK is moved to the **Rollover SMK** location.

3. [Optional] Display the primary and rollover SMK OUIDs.

lunacm:> `partition showinfo`

```
Partition SMK OUIDs:
    SMK-FW4: Not Initialized
    SMK-FW6: Not Initialized
    SMK-FW7-FM: Not Initialized
    SMK-FW7-Rollover: 1e0000000e000001111e0800
    SMK-FW7-Primary: 1f0000000e000009999e0800
```

4. Perform SMK rollover by retrieving each of your encrypted blobs from the database, inserting it into the partition, and re-extracting it once it has been encrypted with the new SMK.
 - a. Insert blobs using the `SIMInsert` API and the rollover SMK (the former primary SMK).
 - b. After each blob is inserted, extract it again to external storage. The extract action is performed with `SIMExtract`, using the new primary SMK.
5. When all blobs have been retrieved, inserted, and re-extracted, end the rollover procedure by deleting the rollover SMK.

CAUTION! Ensure that you have re-encrypted all your key material before ending the rollover procedure. When the rollover SMK is deleted, any blobs it encrypted are unrecoverable.

lunacm:> `partition smkrollover -end`

6. If you are using the partition as the primary in an HA group, you can recreate it now. The new SMK is cloned to each member as you add it to the group.

lunacm:> `hagroup addmember {-serialnumber <serialnum> | -slot <slotnumber>} -group <label> -password <password>`

CHAPTER 9: Per-Key Authorization

Per-key authorization or authentication (PKA) is a feature introduced with Luna HSM Firmware 7.7.0 to support the eIDAS use case of Remote Signing and Sealing (RSS) and the relevant Protection Profile (PP 419-221.5). PKA introduces data structures to keys that are created and manipulated in the HSM such that keys can be handled in the ways that applications normally handle key material, but under the sole ownership and control of an end-user natural person or legal entity. The attributes are applied to keys that are created with Luna HSM Firmware 7.7.0 (or newer). In V0 partitions those attributes are simply ignored. In V1 partitions the attributes are actively used. See more at ["V0 and V1 Partitions" on page 67](#).

Keys for use in eIDAS schemas:

- > have authentication data structures that allow the possibility for an entity to have sole ownership and control
- > can be unassigned, waiting for distribution to eventual owners/Users, or
- > can be assigned to the control of a specific owner/User.

When a key has auth code data attached, then by definition anyone who holds the auth code is a/the key owner. But before it is assigned, the key does not have an owner/User, and might be part of a pool of unassigned keys, waiting for distribution to users. Keys do take some time to generate, so in times of high demand, it could be practical and convenient to have some ready-to-go.

Keys are intended to be used, but they must also be administered. That is, an individual natural person (or a non-natural legal entity) authorizes cryptographic usage of a key, perhaps to sign forms or documents. At the same time, the HSM has roles that perform actions within the HSM, either:

- > *generally* - the eIDAS Administrator role represented by the Crypto Officer (CO) role in the HSM) or
- > *specifically/individually* - the eIDAS User role represented by the Limited Crypto Officer (LCO) role in the HSM.

So, a citizen might log into a service and perform an action that directs the application to retrieve their existing personal key from a database/repository and insert/decrypt the key into an HSM partition, where the citizen authorizes a signing or other operation, and then the copy of the key is deleted from the HSM partition, but the archived, encrypted copy resides safely in the database for future retrieval and use. Alternatively, a citizen might request a single-use key, that is generated 'on-the-spot' in the HSM partition (by the LCO role) and is authorized by that citizen to perform one action (like signing) and then the key is permanently deleted, with no copy existing anywhere.

Generally, these and other operations related to keys are not performed by administrative commands (tools like LunaCM); rather, they are performed via the PKA API or REST, while the performing application is logged in as one or the other of the partition roles.

Example Use Case

For example, an application might be instructed to retrieve a certain key and use it to sign a document on behalf of a citizen. The application acquires the key from a database (in the form of an encrypted blob) and inserts it into an HSM where it is decrypted to reveal the key that is to be used. But the application is able to actually use that

key only when the owner/citizen presents her/his unique authentication data, which is part of the key attributes.

New Role and Handling

In order to manage this service, the individual application partition's Crypto Officer role and a new role called Limited Crypto User handle the actions of creating, modifying, and using keys containing auth data.

A key can be created in an assigned state, where it is immediately associated with an entity, or a key can be created in unassigned state and only later assigned to an owner, when convenient.

No New Administrative Commands

Because the operations around PKA and RSS are handled programmatically, no particular administrative commands are introduced - only a new **-version** option for partition creation and a new partition policy 40, which is off for V0 partitions, and which defaults to on for V1 partitions, but can be turned off if desired. Everything else about PKA is handled by the [PKA API](#).

Dependencies and Interactions with Other Features

The feature is ignored by older clients and applications that do not know how to make use of it. Active use of PKA requires a V1 partition, which means that cloning is used for:

- > incoming keys and objects from older firmware, but not outgoing (that is, on V1 partitions, cloning of keys is inbound migration, only)
- > copying (such as for HA), or backing-up/restoring, of the SMK

All other objects are stored, encrypted by the SMK, in external storage using the Scalable Key Storage (SKS) feature.

Stored Data Integrity (SDI) is also mandated by eIDAS and is therefore applied by Luna HSM Firmware 7.7.0 and newer.

HA Indirect Login support is constrained differently for V0 and V1 partitions - see section "V0 Partitions" in [PKA API](#).

CHAPTER 10: High-Availability Groups

Luna HSMs can provide scalability and redundancy for cryptographic applications that are critical to your organization. For applications that require continuous, uninterruptible uptime, the Luna HSM Client allows you to combine application partitions on multiple HSMs into a single logical group, known as a High-Availability (HA) group.

This feature is best suited to provide redundancy to the Luna Network HSM 7 product. It has been tested with for limited application with small groups of Luna USB HSM 7s (see ["Planning your HA Group Deployment" on page 88](#)).

An HA group allows your client application to access cryptographic services as long as one member HSM is functional and network-connected. This allows you to perform maintenance on any individual member without ever pausing your application, and provides redundancy in the case of individual failures. Cryptographic requests are distributed across all active group members, enabling a performance gain for each member added. Cryptographic objects are replicated across the entire group, so HA can also be used to keep a current, automatic, remote backup of the group contents.

HA functionality is handled by the Luna HSM Client software. The individual partitions have no way to know they are configured in an HA group, so you can configure HA on a per-application basis. The way you group your HSMs depends on your circumstances and desired performance.

- > ["Planning your HA Group Deployment" on page 88](#)
- > ["Configuring a High-Availability Group" on page 90](#)
- > ["Managing HA Groups" on page 99](#)

Performance

For repetitive operations (for example, many signings using the same key), an HA group provides linear performance gains as group members are added. The best approach is to maintain an HA group at a size that best balances application server capability and the expected loads, with an additional unit providing capacity for bursts of traffic.

For best overall performance, keep all group members running near their individual performance ideal, about 30 simultaneous threads per HSM. If you assemble an HA group that is significantly larger than your server(s) can manage, you might not achieve full performance from all members. Gigabit Ethernet connections are recommended to maximize performance.

Performance is also affected by the kind of cryptographic operations being requested. For some operations, an HA group can actually hinder performance by requiring extra operations to replicate new key objects. For example, if the operation involves importing and unwrapping keys:

Using an HA group	Using an individual partition
<ol style="list-style-type: none"> 1. Encryption (to wrap the key) 2. Decryption on the primary member partition (to unwrap the key) 3. Object creation on the primary member partition (the unwrapped key is created and stored as a key object) 4. Key replication across the HA group: <ol style="list-style-type: none"> a. RSA 4096-bit operation is used to derive a shared secret between HSMs b. Encryption of the key on the primary HA member using the shared secret c. Decryption of the key on each HA member using the shared secret d. Object creation on each HA member 5. Encryption (using the unwrapped key object to encrypt the data) 	<ol style="list-style-type: none"> 1. Encryption (to wrap the key) 2. Decryption (to unwrap the key) 3. Object creation (the unwrapped key is created and stored as a key object) 4. Encryption (using the unwrapped key object to encrypt the data)

In this case, the HA group must perform many more operations than an individual partition, most significantly the RSA-4096-bit operation and creating the additional objects. Those two operations are by far the most time-consuming on the list, and so this task would have much better performance on an individual partition.

The crucial HA performance consideration is whether the objects on the partitions are constant, or always being created and replaced. If tasks make use of already-existing objects, those objects exist on all HA group members; operations can be performed by different group members, boosting performance. If new objects are created, they must be replicated across the entire group, causing a performance loss.

NOTE The way your application uses the **C_FindObjects** function to search for objects in a virtual HA slot can have a significant impact on your application performance (see ["Application Object Handles" on page 86](#)).

Load Balancing

Cryptographic requests sent to the HA group's virtual slot are load-balanced across all active members of the HA group. The load-balancing algorithm sends requests for cryptographic operations to the least busy partition in the HA group. This scheme accounts for operations of variable length, ensuring that queues are balanced even when some partitions are assigned very long operations. When an application requests a repeated set of operations, this method works. When the pattern is interrupted, however, the request type becomes relevant, as follows:

- > Single-part (stateless) cryptographic operations are load-balanced.
- > Multi-part (stateful) cryptographic operations are load-balanced.

- > Multi-part (stateful) information retrieval requests are not load-balanced. In this case, the cost of distributing the requests to different HA group members is generally greater than the benefit. For this reason, multi-part information retrieval requests are all targeted at one member.
- > Key management requests are not load-balanced. Operations affecting the state of stored keys (creation, deletion) are performed on a single HA member, and the result is then replicated to the rest of the HA group.

For example, when a member partition is signing and an asymmetric key generation request is issued, additional operations on that member are queued while the partition generates the key. In this case, the algorithm schedules more operations on other partitions in the HA group.

The load-balancing algorithm operates independently in each application process. Multiple processes on the same client or on different clients do not share information when scheduling operations. Some mixed-use cases might cause applications to use some partitions more than others (see ["Planning your HA Group Deployment" on page 88](#)). If you increase key sizes, interleave other cryptographic operations, or if network latency increases, performance may drop for individual active members as they become busier.

NOTE Partitions designated as standby members are not used to perform cryptographic operations, and are therefore not part of the load-balancing scheme (see ["Standby Members" on page 86](#)).

The Primary Partition

The primary partition is the first partition you specify as a member of the HA group. While cryptographic operations are load-balanced across all the partitions in the group, new keys are always created on the primary partition, and then replicated on the other partitions (see ["Key Replication" below](#)). Depending on how many new keys you are creating on your HA group, this can mean that the primary partition has a heavier workload than the other partitions in the group. If your HSMs are in different remote locations, you could select one with the least latency as the primary partition.

Despite its name, the primary partition is not more critical than any other partition in the HA group. If the primary partition fails, its operations fail over to other partitions in the group, and the next member added to the group becomes the new primary partition.

Network Topography

The network topography of the HA group is generally not important to the functioning of the group. As long as the client has a network path to each member, the HA logic will function. Different latencies between the client and each HA member cause a command scheduling bias towards the low-latency members. Commands scheduled on the long-latency devices have a longer overall latency associated with each command.

In this case, the command latency is a characteristic of the network. To achieve uniform load distribution, ensure that partitions in the group have similar network latency.

Key Replication

Objects (session or token) are replicated immediately to all members in an HA group when they are generated in the virtual HA slot. Similarly, deletion of objects (session or token) from the virtual HA slot is immediately replicated across all group members. Therefore, when an application creates a key on the virtual HA slot, the HA library automatically replicates the key across all group members before reporting back to the application. Keys

are created on one member partition and replicated to the other members. If a member fails during this process, the HA group reattempts key replication to that member until it recovers, or failover attempts time out. Once the key exists on all active members of the HA group, a success code is returned to the application.

NOTE If you are using [Luna HSM Client 10.4.0](#) or newer and are setting up an HA group with a mix of FIPS and non-FIPS partitions as members, objects will not replicate across all HSMs in the group in the following cases:

- > If you have set a non-FIPS primary, a FIPS secondary, and created a non-FIPS approved key on the group, the key will not replicate to the FIPS secondary. No error is returned when this occurs.
- > If you synchronize group members with the [hagroup synchronize](#) LunaCM command, any non-FIPS keys will fail to replicate to the FIPS member(s). An error is returned when this occurs, but lunaCM synchronizes everything else.

NOTE If your application bypasses the virtual slot and creates or deletes directly in a physical member slot, the action occurs only in that single physical slot, and can be overturned by the next synchronization operation. For this reason we generally advise to enable HA-only, unless you have specific reason to access individual physical slots, and are prepared (in your application) to perform the necessary housekeeping.

Key replication, for pre-firmware-7.7.0 HSM partitions and for V0 partitions, uses the Luna cloning protocol, which provides mutual authentication, confidentiality, and integrity for each object that is copied from one partition to another. Therefore, *prior to Luna HSM Firmware 7.8.0*, all HA group member partitions must be initialized with the same cloning domain.

Key replication, for Luna HSM Firmware 7.8.0 (and newer) HSM partitions and for V0 partitions, and [Luna HSM Client 10.5.0](#) (and newer), becomes more versatile with Extended Domain Management, as each member partition can have as many as three cloning/security domains. It becomes possible to easily mix password-authenticated and multi-factor (PED) authenticated partitions in HA groups. Any member must have at least one of its domains in common with the current primary member. [For reasons of redundancy and overlap, we recommend that you *not* create (say) a 4-member group where the primary has domains A, B, C, and the three secondary members include one member with domain A, one member with domain B, and one member with domain C, where no other domains belong to the group -- such a group could function only until the primary failed/went-offline, at which point the next primary would have no domain peers with which to synchronize. Therefore, consider redundancy overlap when using Extended Domain Management with HA group members.

Key replication for V1 partitions uses the Luna cloning protocol to ensure that all HA group members have the same SMK, and uses SKS to export a key originating at one member and to import and decrypt that key (using the common SMK) on each other member in the group. Again, all HA group member partitions must be initialized with the same cloning domain in order that the common SMK can be available on every member.

The cloning or SKS protocol is invoked separately for each object to be replicated and the sequence of required calls must be issued by an authorized client library residing on a client platform that has been authenticated to each of the partitions in the HA group).

Failover

When any active HA group member fails, a failover event occurs – the affected partition is dropped from the list of available HA group members, and all operations that were pending on the failed partition are transparently rescheduled on the remaining member partitions. The Luna HSM Client continuously monitors the health of member partitions at two levels:

- > network connectivity – disruption of the network connection causes a failover event after a 20-second timeout.
- > command completion – any command that is not executed within 20 seconds causes a failover event.

NOTE Most commands are completed within milliseconds. Some can take longer, either because the command itself is time-consuming (for example, key generation), or because the HSM is under extreme load. The HSM automatically sends a "heartbeat" signal every two seconds for commands that are pending or in progress. The client extends the 20-second timeout whenever it receives a heartbeat, preventing false failover events.

When an HA group member fails, the HA group status (see [hagroup listgroups](#)) reports a device error for the failed member. The client tries to reconnect the failed member at a minimum retry rate of once every 60 seconds, for the specified number of times (see ["Recovery" on the next page](#)).

When a failover occurs, the application experiences a latency stall on the commands in process on the failing unit, but otherwise there is no impact on the transaction flow. The scheduling algorithm described in ["Load Balancing" on page 81](#) automatically minimizes the number of commands that stall on a failing unit during the 20-second timeout.

As long as one HA group member remains functional, cryptographic service is maintained no matter how many other group members fail. As described in ["Recovery" on the next page](#), members can be returned to service without restarting the application.

Mid-operation failures

Any operation that fails mid-point needs to be re-sent from the calling application. The entire operation returns a failure (CKR_DEVICE_ERROR). This is more likely to happen in a multi-part operation, but a failure could conceivably happen during a single atomic operation as well.

For example, multi-part operations could be block encryption/decryption or any other command where the previous state of the HSM is critical to the processing of the next command. These operations must be re-sent, since the HA group does not synchronize partitions' internal memory state, only the stored key material.

NOTE You must ensure that your applications can deal with the rare possibility of a mid-operation failure, by re-issuing the affected commands.

Possible Causes of Failure

In most cases, a failure is a brief service interruption, like a system reboot. These temporary interruptions are easily dealt with by the failover and auto-recovery functions. In some cases, additional actions may be required

before auto-recovery can take place. For example, if a partition becomes deactivated, it must be reactivated by the Crypto Officer (see ["Activation on Multifactor Quorum-Authenticated Partitions" on page 44](#)). Some permanent failures may require manual recovery (see ["Recovery" below](#)).

Recovery

Recovery of a failed HA group member is designed to be automatic in as many cases as possible. You can configure your auto-recovery settings to require as much manual intervention as is convenient for you and your organization. In either an automated or manual recovery process, there is no need to restart your application. As part of the recovery process:

- > Any cryptographic objects created while the member was offline are automatically replicated to the recovered partition.
- > The recovered partition becomes available for its share of load-balanced cryptographic operations.

Auto-recovery

When auto-recovery is enabled, Luna HSM Client performs periodic recovery attempts when it detects a member failure. You can adjust the frequency (maximum once per minute) and the total number of retries (no limit). If the failed partition is not recovered within the scheduled number of retries, it remains a member of the HA group, but the client will no longer attempt to recover it. You must then address whatever equipment or network issue caused the failure, and execute a manual recovery of the member partition.

With each recovery attempt, a single application thread experiences a slight latency delay of a few hundred milliseconds while the client uses the thread to recover the failed member partition.

There are two HA auto-recovery modes:

- > **activeBasic** – uses a separate, non-session-based Active Recovery Thread to perform background checks of HA member availability, recover failed members, and synchronize the contents of recovered members with the rest of the group. It does not restore existing sessions if all members fail simultaneously and are recovered.
- > **activeEnhanced** – works the same as activeBasic, but restores existing sessions and login states if all members fail and are recovered.

HA auto-recovery is disabled by default. It is automatically enabled when you set the recovery retry count (see ["Configuring HA Auto-Recovery" on page 94](#)). Thales recommends enabling auto-recovery in all configurations.

NOTE If a member partition loses Activation, you must present the black Crypto Officer iKey to re-cache the authentication secret before the member can be recovered.

Manual Recovery

When auto-recovery is disabled, or fails to recover the partition within the scheduled number of retries, you must execute a manual recovery in LunaCM. Even if you use manual recovery, you do not need to restart your application. When you execute the recovery command, the client makes a recovery attempt the next time the application uses the group member (see ["Manually Recovering a Failed HA Group Member" on page 101](#)).

Even with auto-recovery enabled and configured for a large number of retries, there are some rare occasions where a manual recovery may be necessary (for example, when a member partition and the client application fail at the same time).

CAUTION! Never attempt a manual recovery while the application is running and auto-recovery is enabled. This can cause multiple concurrent recovery processes, resulting in errors and possible key corruption.

Failure of All Group Members

If all members of an HA group fail (and no standby members are configured), all logged-in sessions are lost, and operations that were active when the last member failed are terminated. If you have set the HA auto-recovery mode to `activeEnhanced`, all sessions will be restarted when one or more members are recovered, and normal operations will resume. Otherwise, you must restart the client application once the group members have been recovered.

Permanent Failures

Sometimes an HSM failure is permanent (from the perspective of the HA group). For example, if the HSM is re-initialized, the member partition is erased and must be recreated. In this case, you can decide to recreate the original member or deploy a new member to the group. The client automatically replicates cryptographic objects to the new member and begins assigning operations to it (see ["Replacing an HA Group Member" on page 101](#)).

Standby Members

After you add member partitions to an HA group, you can designate some as standby members. Cryptographic objects are replicated on all members of the HA group, including standby members, but standby members do not perform any cryptographic operations unless all the active members go offline. In this event, all standby members are immediately promoted to active service, and operations are load-balanced across them. This provides an extra layer of assurance against a service blackout for your application.

Since standby members replicate keys but do not perform operations, they can also serve as an automatic backup partition for the cryptographic objects on the HA group. The contents of standby partitions are always kept up-to-date, so it is not possible to keep multiple backups (different generations of preserved material) using an HA group (see ["Planning your HA Group Deployment" on page 88](#)). You can consider HA standby members to be your backup only in the case where the most recent sync always replicates all objects you are interested in preserving and recovering.

If you have audit-compliance rules or other mandate to preserve earlier partition contents (keys and objects), then you should perform intentional backups with dedicated backup devices (see ["Partition Backup and Restore" on page 106](#)).

Application Object Handles

Application developers should be aware that the PKCS #11 object handle model is fully virtualized when using an HA slot. The application must not assume fixed handle numbers across instances of an application. A handle's value remains consistent for the life of a process; but it might be a different value the next time the application is executed.

When you use an HA slot with your applications, the client behaves as follows when interacting with the application:

1. Intercept the call from the application.
2. Translate virtual object handles to physical object handles using the mappings specified by the virtual object table. The virtual object table is created and updated for the current session only, and only contains a list of the objects accessed in the current session.
3. Launch any required actions on the appropriate HSM or partition.
4. Receive the result from the HSM or partition and forward the result to your application,
5. Propagate any changes in objects on the physical HSM that performed the action to all of the other members of the HA group.

Virtual slots and virtual objects

When an application uses a non-HA physical slot, it addresses all objects in the slot by their physical object handles. When an application uses an HA slot, however, a virtual layer of abstraction overlays the underlying physical slots that make up the HA group, and the HA group is presented to the application as a virtual slot. This virtual slot contains virtual objects that have virtual object handles. The object handles in an HA slot are virtualized since the object handles on each of the underlying physical slots might be different from slot to slot. Furthermore, the physical object handles could change if a member of the HA group drops out (fails or loses communication) and is replaced.

The virtual object table

HA slots use a virtual object table to map the virtual objects in the virtual HA slot to the real objects in the physical slots that make up the HA group. The HA client builds a virtual object table for each application that loads the library. The table is ephemeral, and only exists for the current session. It is created and updated, if necessary, each time an application makes a request to access an object. To maximize performance and efficiency, the table only contains a list of the objects accessed in the current session. For example, the first time an application accesses an object after application start up, the table is created, a look up is performed to map the virtual object to its underlying physical objects, and an entry for the object is added to the table. For each subsequent request for that object, the data in the table is used and no look up is required. If the application then accesses a different object that is not listed in the table, a new look up is performed and the table is updated to add an entry for the new object.

C_FindObjects behavior and application performance

Since the client must perform a lookup to create the virtual object table, the way you use the C_FindObjects function can have a significant impact on the performance of your applications. For example, if you use the C_FindObjects function to ask for specific attributes, the client only needs to update the table to include the requested objects. If, however, you use the C_FindObjects function to find all objects, the client queries each HSM/partition in the group, for each object, to create the table. This can take a significant amount of time if the slot contains a large number of objects, or if the HA group includes many members.

To mitigate performance degradation when using the C_FindObjects function to list the objects on an HA slot, we recommend that you structure your applications to search by description, handles, or other attributes, rather than searching for all objects. Doing so minimizes the number of objects returned and the time required to create or

update the table. If your application must find all objects, we recommend that you add the `C_FindObjects` all function call to the beginning of your application so that the table is built on application start up, so that the table is available to the application for all subsequent `C_FindObjects` function calls.

Planning your HA Group Deployment

This section describes important considerations and constraints to keep in mind as you plan your High-Availability (HA) group deployment. The benefits of HA are described in detail in ["High-Availability Groups" on page 80](#).

- > ["HSM and Partition Prerequisites" below](#)
- > ["Sample Configurations" on the next page](#)

HSM and Partition Prerequisites

The HSM partitions you plan to use in an HA group must meet the following prerequisites before you can use them in an HA group.

Compatible HSM Firmware Versions

All HSMs in an HA group must have the same firmware version installed.

Common Partition Versions

All partitions in an HA group must be the same version, either V0 or V1 (see ["V0 and V1 Partitions" on page 67](#)).

Common Cloning Domain

All key replication in an HA group uses the Luna cloning protocol, which provides mutual authentication, confidentiality, and integrity for each object that is copied from one partition to another. Therefore, all HA group member partitions must be initialized with the same cloning domain. If you are planning to combine already-existing partitions into an HA group, you must first re-initialize them using the same domain string or red iKey.

Common Crypto Officer Credentials

An HA group essentially allows you to log in to all its member partitions simultaneously, using a single credential. Password-authenticated partitions must all be initialized with the same Crypto Officer password. Multifactor Quorum-authenticated partitions must all be initialized with the same black Crypto Officer iKey and activated with the same CO challenge password.

It is not possible to create an HA group made up of both password- and multifactor quorum-authenticated partitions.

Common HSM/Partition Policies (FIPS 140 approved configuration [formerly FIPS mode])

Generally, all HSMs/partitions used in an HA group must have the same policy configuration, especially FIPS 140 approved configuration (formerly FIPS mode). Do not attempt to use an HA group combining HSMs with FIPS 140 approved configuration (formerly FIPS mode) on and others with FIPS 140 approved configuration off.

Sample Configurations

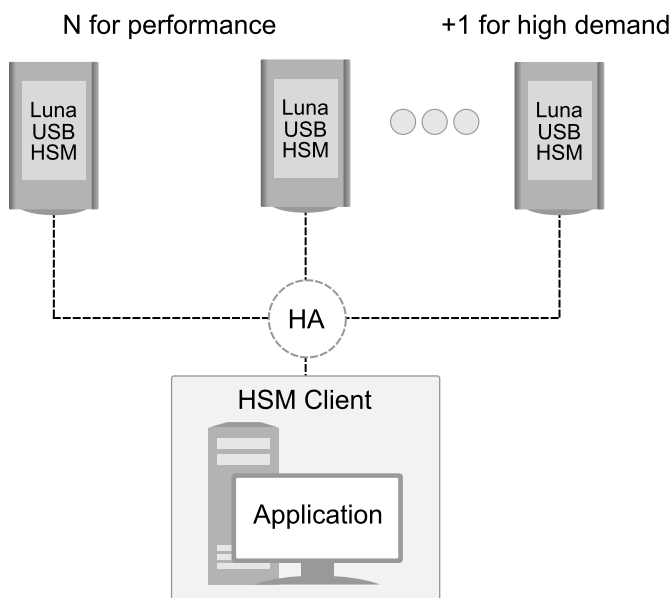
Your ideal HA group configuration depends on the number of HSMs you have available and the purpose of your application(s). Sample configurations for different types of deployment are described below.

Performance and Load Balancing

If your application is designed to perform many cryptographic operations as quickly as possible, using keys or other objects that do not change often, you can create a large HA group using partitions on multiple HSMs. This deployment uses load balancing to provide linear performance gains for each HSM added to the group.

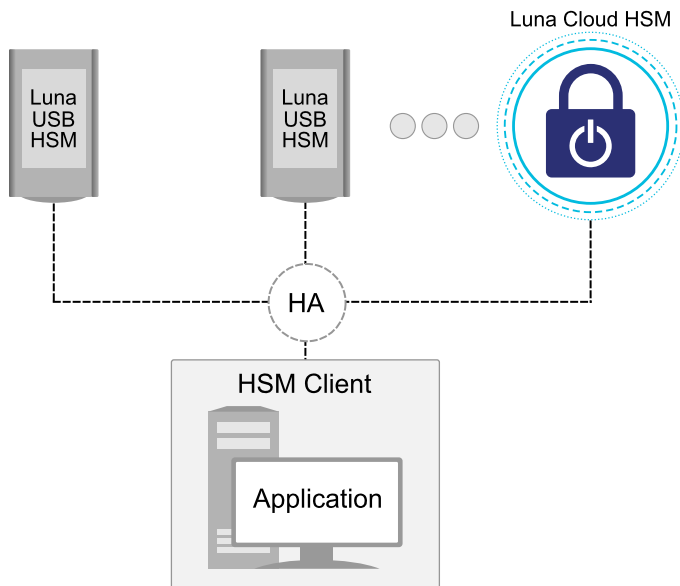
For example: your application uses keys stored on the HSM to perform many encrypt/decrypt or sign/verify operations. You want to minimize transaction latency by providing enough HSMs to handle capacity.

The best approach in this example is to add enough group members to handle the usual number of operations, plus enough extra members to handle periods of high demand.



Automatic Remote Backup

Since the contents of member partitions are always kept up-to-date, you can use an HA group to keep an automatic backup of your cryptographic objects. Set the backup member to standby mode so that it does not perform operations. If the regular member(s) fail, the standby member takes over operations. The backup member can be a local HSM or a Luna Cloud HSM service. Refer to ["Cloning Keys Between Luna 6, Luna 7, and Luna Cloud HSM" on page 32](#).



Configuring a High-Availability Group

Use LunaCM to create an HA group from partitions assigned to your client. This procedure is completed by the Crypto Officer. Ensure that you have met all necessary prerequisites before proceeding with group creation. For a detailed description of HA functionality, see ["High-Availability Groups" on page 80](#).

NOTE Your LunaCM instance needs to update the **Chrystoki.conf** (Linux/UNIX) or **crystoki.ini** file (Windows) when setting up or reconfiguring HA. Ensure that you have Administrator privileges on the client workstation.

V1 partitions: If you add an application partition with an existing SMK to an HA group, the primary member's SMK overwrites the existing SMK of the joining partition. If a partition's SMK has ever been used to encrypt important SKS objects, save a backup of the SMK before adding that partition to any HA group.

The following procedures are included in the configuration process:

- > ["Verifying an HA Group" on page 92](#)
- > ["Setting an HA Group Member to Standby" on page 93](#)
- > ["Configuring HA Auto-Recovery" on page 94](#)
- > ["Enabling/Disabling HA Only Mode" on page 95](#)
- > ["HA Logging" on page 95](#)

Prerequisites

HA groups are set up in LunaCM by the Crypto Officer. Before the CO can perform this setup, however, all HSMs and member partitions must meet the following prerequisites, completed by the HSM and Partition Security Officers.

HSMs

The HSM SO must ensure that all HSMs containing HA group member partitions meet the following prerequisites:

- > All HSMs must use the same authentication method (password/multifactor quorum). Luna Cloud HSM services support password authentication only.
- > All must be running one of the supported software/firmware versions. Generally, Thales recommends using HSMs with the same software/firmware for HA. However, mixed-version HA groups containing Luna USB HSM 7 member partitions and Luna Cloud HSM services are supported. See ["Cloning Keys Between Luna 6, Luna 7, and Luna Cloud HSM" on page 32](#) for more information.
- > HSM policies **7: Allow Cloning** and **16: Allow Network Replication** must be set to **1** (see [Setting HSM Policies Manually](#)).
- > HSM policies must be consistent across all HSMs, particularly **12: Allow non-FIPS algorithms**. Do not attempt to use an HA group combining HSMs with FIPS 140 approved configuration (formerly FIPS mode) on and others with FIPS 140 approved configuration off.

Partitions

The Partition SO must ensure that all partitions in an HA group meet the following prerequisites:

- > All partitions must be visible in LunaCM on the client workstation.
- > All partitions must be initialized with the same cloning domain:
 - password-authenticated partitions must share the same domain string.
 - multifactor quorum-authenticated partitions must share the same red domain iKey.
- > Partition policies **0: Allow private key cloning** and **4: Allow secret key cloning** must be set to **1** on all partitions.
- > Partition policies must be consistent across all member partitions.
- > The Crypto Officer role on each partition must be initialized with the same CO credential (password or black iKey).
- > Multifactor Quorum-authenticated partitions must have partition policy **22: Allow activation** set to **1**. Each partition must have the same activation challenge secret set (see ["Activation on Multifactor Quorum-Authenticated Partitions" on page 44](#))

NOTE If HSM policy **21: Force user PIN change after set/reset** is set to **1** (the default setting), the Crypto Officer must change the initial CO credential before using the partition for cryptographic operations. This applies to the activation challenge secret as well (see [role changepw](#)).

To set up an HA group

1. Decide which partition will serve as the primary member (see ["The Primary Partition" on page 82](#)). Create a new HA group, specifying the following information:
 - the group label (do not call the group "HA")
 - the Serial number OR the slot number of the primary member partition
 - the Crypto Officer password or challenge secret for the partition

```
lunacm:>hagroup creategroup -label <label> {-slot <slotnum> | -serialnumber <serialnum>}
```

LunaCM generates a serial number for the HA group (by adding a "1" before the primary partition serial number), assigns it a virtual slot number, and automatically restarts.

2. Add another partition to the HA group, specifying either the slot or the serial number. If the new member contains cryptographic objects, you are prompted to decide whether to replicate the objects within the HA group, or delete them. See also ["Adding/Removing an HA Group Member" on page 99](#).

```
lunacm:>hagroup addmember -group <grouplabel> {-slot <slotnum> | -serialnumber <serialnum>}
```

Repeat this step for each additional HA group member.

NOTE By default, lunacm:>**hagroup addmember** automatically adds a Luna Cloud HSM service as a standby HA member. If you prefer to use the Luna Cloud HSM service as an active HA member, you must first edit the following toggle in the **Chrystoki.conf/crystoki.ini** configuration file (see [Configuration File Summary](#)):

```
[Toggles]
lunacm_cv_ha_ui = 0
```

3. If you are adding member partitions that already have cryptographic objects stored on them, initiate a manual synchronization. You can tell whether this step is required by checking the line **Needs Sync : yes/no** in the HA group output. This will also confirm that the HA group is functioning correctly.

```
lunacm:> hagroup synchronize -group <grouplabel>
```

4. [Optional] If you created an HA group out of empty partitions, and you want to verify that the group is functioning correctly, see ["Verifying an HA Group" below](#).
5. Specify which member partitions, if any, will serve as standby members.
See ["Setting an HA Group Member to Standby" on the next page](#).
6. Set up and configure auto-recovery (recommended). If you choose to use manual recovery, you will have to execute a recovery command whenever a group member fails.
See ["Configuring HA Auto-Recovery" on page 94](#).
7. [Optional] Enable HA Only mode (recommended).
See ["Enabling/Disabling HA Only Mode" on page 95](#).
8. [Optional] Configure HA logging.
See ["HA Logging" on page 95](#) for procedures and information on reading HA logs.

The HA group is now ready for your application.

Verifying an HA Group

After creating an HA group in LunaCM, you can see the group represented as a virtual slot alongside the physical slots. The following procedure is one way to verify that your HA group is working as intended.

To verify an HA group

1. Exit LunaCM and run **multitoken** against the HA group slot number (slot 5 in the example) to create some objects on the HA group partitions.

```
./multitoken -mode <keygen_mode> -key <key_size> -no destroy -slots <HA_virtual_slot>
```

You can hit **Enter** at any time to stop the process before the partitions fill up completely. Any number of created objects will be sufficient to show that the HA group is functioning.

2. Run LunaCM and check the partition information on the two physical slots. Check the object count under "Partition Storage":

```
lunacm:> partition showinfo
```

```
Current Slot Id: 0
```

```
lunacm:> partition showinfo
```

```
Partition Storage:
  Total Storage Space: 325896
  Used Storage Space:  22120
  Free Storage Space:  303776
  Object Count:      14
  Overhead:          9648
```

```
Command Result : No Error
```

```
lunacm:> slot set slot 1
```

```
Current Slot Id: 1 (Luna User Slot 7.7.2 (PW) Signing With Cloning Mode)
```

```
Command Result : No Error
```

```
lunacm:> partition showinfo
```

```
Partition Storage:
  Total Storage Space: 325896
  Used Storage Space:  22120
  Free Storage Space:  303776
  Object Count:      14
  Overhead:          9648
```

```
Command Result : No Error
```

3. To remove the test objects, login to the HA virtual slot and clear the virtual partition.

```
lunacm:> slot set -slot <HA_virtual_slot>
```

```
lunacm:> partition login
```

```
lunacm:> partition clear
```

If you are satisfied that your HA group is working, you can begin using your application against the HA virtual slot. The virtual slot assignment will change depending on how many more application partitions are added to your client configuration. If your application invokes the HA group label, this will not matter. If you have applications that invoke the slot number, see ["Enabling/Disabling HA Only Mode" on page 95](#).

Setting an HA Group Member to Standby

Some HA group members can be designated as standby members. Standby members do not perform any cryptographic operations unless all active members have failed (see ["Standby Members" on page 86](#) for details). They are useful as a last resort against loss of application service.

Prerequisites

- > The partition you want to designate as a standby member must already be a member of the HA group (see ["Adding/Removing an HA Group Member" on page 99](#)).
- > The group member must be online.
- > The Crypto Officer must perform this procedure.

To set an HA group member to standby

1. [Optional] Check the serial number of the member you wish to set to standby mode.

```
lunacm:> hagroup listgroups
```

2. Set the desired member to standby mode by specifying the serial number.

```
lunacm:> hagroup addstandby -group <label> -serialnumber <member_serialnum>
```

To make a standby HA member active

NOTE By default, a Luna Cloud HSM service is always added to an HA group as a standby member. If you prefer to use the Luna Cloud HSM service as an active HA member, you must first edit the following toggle in the **Chrystoki.conf/crystoki.ini** configuration file (see [Configuration File Summary](#)):

```
[Toggles]
lunacm_cv_ha_ui = 0
```

1. [Optional] Check the serial number of the standby member.

```
lunacm:> hagroup listgroups
```

2. Remove the member from standby and return it to active HA use.

```
lunacm:> hagroup removestandby -group <label> -serialnumber <member_serialnum>
```

Configuring HA Auto-Recovery

When auto-recovery is enabled, Luna HSM Client performs periodic recovery attempts when it detects a member failure. HA auto-recovery is disabled by default for new HA groups. To enable it, you must set a maximum number of recovery attempts. You can also set the frequency of recovery attempts, and the auto-recovery mode (**activeBasic** or **activeEnhanced**). These settings will apply to all HA groups configured on the client.

To configure HA auto-recovery

1. Set the desired number of recovery attempts by specifying the retry count as follows:

- Set a value of **0** to disable HA auto-recovery
- Set a value of **-1** for unlimited retries
- Set any specific number of retries from **1** to **500**

```
lunacm:> hagroup retry -count <retries>
```

2. [Optional] Set the desired frequency of recovery attempts by specifying the time in seconds. The acceptable range is **60-1200** seconds (default: **60**).

lunacm:> **hagroup interval -interval** <seconds>

3. [Optional] Set the auto-recovery mode. The default is **activeBasic**.

lunacm:> **hagroup recoverymode -mode** {**activeBasic** | **activeEnhanced**}

4. [Optional] Check that auto-recovery has been enabled. You are prompted for the Crypto Officer password/challenge secret.

lunacm:> **hagroup listgroups**

Enabling/Disabling HA Only Mode

By default, client applications can see both physical slots and virtual HA slots. Directing applications at the physical slots bypasses the high availability and load balancing functionality. An application must be directed at the virtual HA slot to use HA load balancing and redundancy. HA Only mode hides the physical slots and leaves only the HA group slots visible to applications, simplifying the PKCS#11 slot numbering.

If an HA group member partition fails and is recovered, all visible slot numbers can change, including the HA group virtual slots. This can cause applications to direct operations to the wrong slot. If a physical slot in the HA group receives a direct request, the results will not be replicated on the other partitions in the group. When HA Only mode is enabled, the HA virtual slots are not affected by partition slot changes. Thales recommends enabling HA Only mode on all clients running HA groups.

NOTE Individual partition slots are still visible in LunaCM when HA Only mode is enabled. They are hidden only from client applications. Use **CKdemo** (Option **11**) to see the slot numbers to use with client applications.

To enable HA Only mode

1. Enable HA Only mode in LunaCM.

lunacm:> **hagroup haonly -enable**

2. [Optional] Since LunaCM still displays the partitions, you can check the status of HA Only mode at any time.

lunacm:> **hagroup haonly -show**

To disable HA Only mode

1. Disable HA Only mode in LunaCM.

lunacm:> **hagroup haonly -disable**

HA Logging

Logging of HA-related events takes place on the Luna HSM Client workstation. The log file **haErrorLog.txt** shows HA errors, as well as add-member and delete-member events. It does not record status changes of the group as a whole (like adding or removing the group).

The HA log rotates after the configured maximum length is reached. When it finishes writing the current record (even if that record slightly exceeds the configured maximum), the file is renamed to include the timestamp and the next log entry begins a new `haErrorLog.txt`.

Configuring HA Logging

Logging is automatically enabled when you configure an HA group, but you must configure a valid destination path before logging can begin. HA groups are configured on the client using LunaCM. The HA configuration settings are saved to the **Chrystoki.conf** (Linux/Unix) or **cryptoki.ini** (Windows) file, as illustrated in the following example:

```
VirtualToken = {
VirtualToken00Label = haGroup1; // The label of the HA group.
VirtualToken00SN = 11234840370164; // The pseudo serial number of the HA group.
VirtualToken00Members = 1234840370164, 1234924189183; // The serial number of the members.
VirtualTokenActiveRecovery = activeEnhanced; // The recovery mode.
}
HASynchronize = {
haGroup1 = 1; // Enable automatic synchronization of objects.
}
HAConfiguration = {
HAOnly = 1; // Enable listing HA groups only via PKCS#11 library.
haLogPath = /tmp/halog; // Base path of the HA log file; i.e., "/tmp/halog/haErrorLog.txt".
haLogStatus = enabled; // Enable HA log.
logLen = 100000000; // Maximum size of HA log file in bytes.
failover_on_deactivation = 1; // if a partition becomes deactivated then the client will
immediately
                                // failover and resume its operation on the other HA partitions.
This
                                // is currently an alpha feature
reconnAtt = 120; // Number of recovery attempts.
}
HARecover = {
haGroup1 = 1; // Deprecated in this release as auto recovery will cover the use case. When
cryptoki
                                // loads into memory it reads the number and if the number changes (gets
incremented)
                                // then cryptoki interprets this as a manual recovery attempt.
}
```

To configure HA logging

Use the LunaCM command **hagroup halog**.

1. Set a valid path for the log directory. You must specify an existing directory.

```
lunacm:> hagroup halog -path <filepath>
```

2. [Optional] Set the maximum length for individual log files (in bytes).

```
lunacm:> hagroup halog -maxlength <max_file_length>
```

3. [Optional] Enable or disable HA logging at any time.

```
lunacm:> hagroup halog -disable
```

```
lunacm:> hagroup halog -enable
```

4. [Optional] View the current status of the HA logging configuration.


```
lunacm:> hagroup halog -show
```

HA Log Messages

The following table provides descriptions of the messages generated by the HA sub-system and saved to the HA log. The HA log is saved to the location specified by **haLogPath** in the **Chrystoki.conf** (Linux/Unix) or **crystoki.ini** (Windows) file.

Message Format

Every HA log message has a consistent prefix consisting of the date, time, process id, and serial number (of the affected HA group). For example:

```
Wed Oct 4 16:29:21 2017 : [17469] HA group: 11234840370164 ...
```

Message Descriptions

Message ID	Message/Description
HALOG_CONFIGURED_AS_PASSWORD	<p><MessagePrefix> configured as a "PASSWORD Based" virtual device</p> <p>Description: Message advising that the virtual partition is password-authenticated. This means that you cannot add a PED-authenticated member to the group.</p>
HALOG_CONFIGURED_AS_PED	<p><MessagePrefix> configured as a "PED Based" virtual device</p> <p>Description: Message advising that the virtual partition is PED-authenticated. This means that you cannot add a password-authenticated member to the group.</p>
HALOG_DROPMEMBER	<p><MessagePrefix> has dropped member: <SerialNumber></p> <p>Description: The connection changed from valid to invalid, determined after an HSM command (such as C_Sign) fails.</p>
HALOG_DROPUNRECOVERABLE	<p><MessagePrefix> unable to reach member: <SerialNumber>. Manual Recover or Auto Recovery will be able to recover this member</p> <p>Description: The connection is invalid, as determined during a call to C_Initialize.</p>
HALOG_LOGINFAILED	<p><MessagePrefix> can not login to member: <SerialNumber>, autorecovery will be disabled. Code: <ErrorCodeHex> : <ErrorCodeString></p> <p>Description: The connection changed from valid to invalid, as determined during a call to C_Login.</p>
HALOG_MEMBER_DEACTIVATED	<p><MessagePrefix> member: <SerialNumber> deactivated</p> <p>Description: The user manually deactivated the partition, as determined after an HSM command (such as C_Sign) fails.</p>

Message ID	Message/Description
HALOG_MEMBER_NOW_ACTIVATED	<p><MessagePrefix> recovery attempt <AttemptNumber> member <SerialNumber> is now activated and will be reintroduce back into the HA group.</p> <p>Description: Additional info about the recovered partition, which was deactivated and is now becoming activated.</p>
HALOG_MEMBER_REVOKED	<p><MessagePrefix> member: <SerialNumber> revoked</p> <p>Description: The user manually revoked the partition, as determined during a periodic recovery attempt.</p>
HALOG_MEMBERS_OFFLINE	<p><MessagePrefix> all members gone offline.</p> <p>Description: A situation where all members go offline. Recovery is not possible at this point.</p>
HALOG_MGMT_THREAD_START	<p><MessagePrefix> management thread started</p> <p>Description: This thread is responsible for managing all members and HA in general while the HA group is active. The thread starts up when the application first launches.</p>
HALOG_MGMT_THREAD_TERMINATE	<p><MessagePrefix> management thread terminated</p> <p>Description: This thread is responsible for managing all members and HA in general while the HA group is active. If the client application shuts down, this thread will simply terminate. The thread will start up again once the application re-launches.</p>
HALOG_NEWMEMBER	<p><MessagePrefix> detected new member member: <SerialNumber></p> <p>Description: The user manually added a member to the HA group without restarting the application, as determined during a periodic recovery attempt.</p>
HALOG_RECOVERED	<p><MessagePrefix> recovery attempt <Integer> succeeded for member: <SerialNumber></p> <p>Description: The connection changed from invalid to valid, as determined during a periodic recovery attempt.</p>
HALOG_RECOVERY_ATTEMPT_#_REINTRODUCING	<p><MessagePrefix> recovery attempt <AttemptNumber> reintroducing <Number> token objects to recovered token <TokenNumber></p> <p>Description: Additional info about the recovered partition at which some objects were cloned.</p>
HALOG_RECOVERYFAILED	<p><MessagePrefix> recovery attempt <Integer> failed for member: <SerialNumber>. Code: <ErrorCodeHex> : <ErrorCodeString>.</p> <p>If autorecovery fails, then a second message is logged, as follows:</p> <p><MessagePrefix> exceeded maximum number of autorecovery attempts for member: <SerialNumber>. Autorecovery will be disabled</p> <p>Description: The connection remains invalid, as determined during a periodic recovery attempt.</p>

Message ID	Message/Description
HALOG_REENABLEMEMBER (deprecated)	<pre><MessagePrefix> Re-enable auto recovery process for member: <SerialNumber></pre> <p>Description: The user manually requested partition recovery, as determined during a periodic recovery attempt before an HSM command.</p>
HALOG_UNRECOVERABLE (deprecated)	<pre><MessagePrefix> recovery attempt <Integer> failed for member: <SerialNumber>. Manual Recover or Auto Recovery will not be able to recover this member. Code: <ErrorcodeHex> : <ErrorcodeString></pre> <p>Description: The connection is invalid and is not eligible for recovery.</p>
No ID*	<pre><MessagePrefix> member <SerialNumber> is not activated and is excluded from the HA group</pre> <p>Description: The HA member was not activated at the time when a C_Initialize call was made, and is therefore excluded from the HA group. Once the partition is activated, the HA group will attempt an automatic recovery, resulting in one of the two messages below</p>
No ID*	<pre><MessagePrefix> recovery attempt <SerialNumber> is not activated and cannot be reintroduced back into the HA group\n</pre> <p>Description: Recovery failed</p>
No ID*	<pre><MessagePrefix> recovery attempt <SerialNumber> is now activated and will be reintroduce back into the HA group.\n</pre> <p>Description: Recovery succeeded</p>

* You might encounter these extra messages in the HA logs. They were added for HA development testing and therefore have no Message IDs assigned to them. They could duplicate information covered by other log messages as defined above.

Managing HA Groups

If you set up your HA groups as recommended, using auto-recovery, they require very little direct maintenance. You can perform the following tasks without pausing your applications:

- > ["Adding/Removing an HA Group Member" below](#)
- > ["Manually Recovering a Failed HA Group Member" on page 101](#) --If you declined to use auto-recovery, you must manually recover group members whenever they fail
- > ["Replacing an HA Group Member" on page 101](#) -- If an HSM fails permanently, or is re-initialized, the member partition cannot be recovered
- > ["Deleting an HA Group" on page 102](#)

Adding/Removing an HA Group Member

You can add a new member to an HA group at any time using LunaCM, even if your application is running. Cryptographic objects will be replicated on the new partition and operations will be scheduled according to the load-balancing algorithm (see ["Load Balancing" on page 81](#)).

Likewise, you can remove a member at any time, and currently-scheduled operations will fail over to the rest of the group members (see ["Failover" on page 84](#)).

NOTE If you remove the partition that was used to create the group, the HA group serial number changes to reflect this. This is to prevent another HA group from being assigned the same serial number as the original. If your application queries the HA group serial number, it must redirect operations to the new serial.

Prerequisites

The new member partition must:

- > be assigned to the client and visible in LunaCM
- > be initialized with the same domain string/red domain iKey as the other partitions in the group
- > have the Crypto Officer role initialized with the same credentials as the other partitions in the group
- > be activated(multifactor quorum-authenticated)

NOTE V1 partitions: If you add an application partition with an existing SMK to an HA group, the primary member's SMK overwrites the existing SMK of the joining partition.
If a partition's SMK has ever been used to encrypt important SKS objects, save a backup of the SMK before adding that partition to any HA group.

To add an HA group member

1. Open LunaCM on the client workstation and ensure that the new partition is visible.
2. Add the new partition to the HA group by specifying either the slot or the serial number. You are prompted for the Crypto Officer password/challenge secret.

```
lunacm:> hagroup addmember -group <label> {-slot <slotnum> | -serial <serialnum>}
```

To remove an HA group member

1. Remove the partition from the group by specifying either the slot or the serial number.

```
lunacm:> hagroup removemember -group <label> {-slot <slotnum> | -serial <serialnum>}
```

NOTE If you remove the partition that was used to create the group, the HA group serial number changes to reflect this. This is to prevent another HA group from being assigned the same serial number as the original. If your application queries the HA group serial number, it must redirect operations to the new serial.

LunaCM restarts.

2. [Optional] Check that the partition was removed from the group.

```
lunacm:> hagroup listgroups
```

Manually Recovering a Failed HA Group Member

Thales recommends using auto-recovery for all HA group configurations (see ["Configuring HA Auto-Recovery" on page 94](#)). If you do not enable auto-recovery and a member partition fails, or if the recovery retry count expires before the partition comes back online, you must recover the partition manually using LunaCM. You do not need to pause your application(s) to perform a manual recovery; the HA group handles load-balancing and automatically replicates any new or changed keys to the recovered member.

To perform a manual recovery of a failed HA group member

1. [Optional] Ensure that the failed member is available and visible in LunaCM by addressing the problem that caused the failure. Display the HA group to see the failed members. You are prompted for the Crypto Officer password/challenge secret.

```
lunacm:> hagroup listgroups
```

2. If you are using a multifactor quorum-authenticated partition, log in to the partition as Crypto Officer and present the black CO iKey.

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

```
lunacm:> role login -name co
```

3. Execute the manual recovery command, specifying the HA group label.

```
lunacm:> hagroup recover
```

If you have an application running on the HA group, the failed members will be recovered the next time an operation is scheduled. Load-balancing and key replication is automatic.

4. If you do not currently have an application running, you can manually synchronize the contents of the HA group.

CAUTION! Never use manual synchronization if you have an application running. The HA group performs this automatically. Using this command on an HA group that is running an application could create conflicting key versions.

```
lunacm:> hagroup synchronize -group <label>
```

Replacing an HA Group Member

Sometimes an HSM failure is permanent (from the perspective of the HA group). For example, if the HSM is re-initialized, the member partition is erased and must be recreated. In this case, you can recreate a partition on the same HSM or another HSM, and deploy the new member to the group. You do not need to pause your application to replace an HA group member.

Prerequisites

The Crypto Officer must complete this procedure, but any new member partition must first be created by the HSM SO, and initialized by the Partition SO. All the prerequisites listed in ["Configuring a High-Availability Group" on page 90](#) must be met.

NOTE V1 partitions: If you add an application partition with an existing SMK to an HA group, the primary member's SMK overwrites the existing SMK of the joining partition.

If a partition's SMK has ever been used to encrypt important SKS objects, save a backup of the SMK before adding that partition to any HA group.

To replace an HA group member

1. [Optional] Display the HA group to see the failed member. You are prompted for the Crypto Officer password/challenge secret.
lunacm:> **hagroup listgroups**
2. Prepare the new HA group member, whether that means creating a new partition on the original HSM or configuring a new Luna USB HSM 7, and assign the new partition to the HA client. Ensure that the new member partition and the HSM on which it resides meet the prerequisites outlined in ["Configuring a High-Availability Group" on page 90](#) and is visible in LunaCM.
3. Add the new partition to the HA group by specifying either the slot or the serial number. You are prompted for the Crypto Officer password/challenge secret.
lunacm:> **hagroup addmember -group <label> {-slot <slotnum> | -serial <serialnum>}**
The new partition is now an active member of the HA group. If you have an application currently running, cryptographic objects are automatically replicated to the new member and it is assigned operations according to the load-balancing algorithm.
4. Remove the old partition from the group by specifying the serial number.
lunacm:> **hagroup removemember -group <label> -serial <serialnum>**
LunaCM restarts.
5. [Optional] If you do not currently have an application running, you can manually synchronize the contents of the HA group.

CAUTION! Never use manual synchronization if you have an application running. The HA group performs this automatically. Using this command on an HA group that is running an application could create conflicting key versions.

lunacm:> **hagroup synchronize -group <label>**

6. [Optional] If you intend to have the new partition serve as a standby member, see ["Setting an HA Group Member to Standby" on page 93](#).

Deleting an HA Group

Use LunaCM to delete an HA group from your configuration.

NOTE This procedure only removes the HA group virtual slot; the member partitions and all their contents remain intact. Only the HSM SO can delete individual partitions.

To delete an HA group

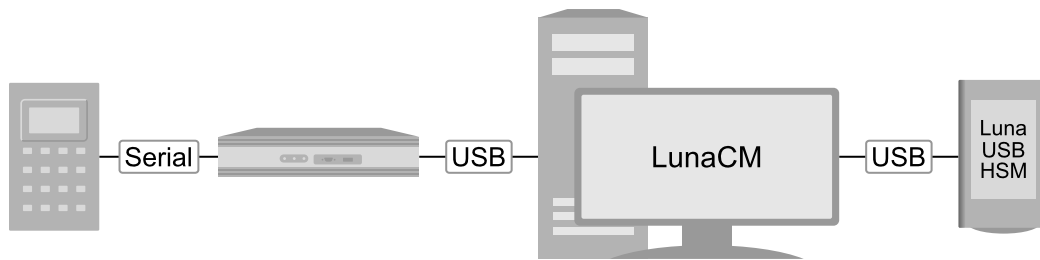
1. Stop any applications currently using the HA group.

2. Delete the group by specifying its label (see [hagroup listgroups](#)).

```
lunacm:> hagroup deletigroup -label <label>
```

CHAPTER 11: Migrating Keys to Your New Luna USB HSM 7

If your Luna USB HSM 7 is replacing an older Luna USB HSM, this page provides information on migrating your keys securely to the new HSM. The partitions on both HSMs must be initialized with the same authentication method (password or iKey) and cloning domain, and they must be connected to the same Luna HSM Client computer. You can migrate objects using direct slot-to-slot cloning, or set up an HA group to synchronize your partition contents between the two HSMs.



Refer to the following sections for preparation and procedures:

- > ["Domain Planning and Key Cloning" on page 28](#) -- the Luna USB HSM 7 partition must be initialized with the same domain as the Luna USB HSM G5 partition.
- > ["Cloning Keys Between Luna 6, Luna 7, and Luna Cloud HSM" on page 32](#) -- consider some possible restrictions on cloning from older to newer Luna firmware.
- > ["Configuring a High-Availability Group" on page 90](#) -- instructions on setting up the Luna USB HSM G5 and Luna USB HSM 7 in an HA group to be synchronized automatically.
- > ["Migration Using Slot-to-Slot Cloning" below](#) -- instructions on direct slot-to-slot cloning from Luna USB HSM G5 to Luna USB HSM 7.

Migration Using Slot-to-Slot Cloning

You can back up partition objects from an application partition to any other partition that shares its cloning domain. The Crypto Officer of both partitions can perform this operation using LunaCM.

Prerequisites

- > You require [Luna HSM Client 10.4.0](#) or newer.
- > **Partition policy 0: Allow private key cloning** must be set to **1 (ON)** on both the source and target partitions.
- > The target partition must be initialized with the same cloning domain as the source partition.
- > You require the Crypto Officer credential for both the source and the target partition.
- > Both partitions must be visible as slots in LunaCM.

- > [Remote PED] This procedure is simpler when both partitions are activated (see ["Activation on Multifactor Quorum-Authenticated Partitions" on page 44](#)). If the partitions are not activated, you must connect the source partition to PEDserver before logging in.

```
lunacm:> ped connect [-ip <IP>] [-port <port>]
```

To clone partition objects to another application partition

1. In LunaCM, set the active slot to the Luna USB HSM G5 partition and log in as Crypto Officer.

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

- If your Luna USB HSM G5 firmware is 6.21.2 or older:

```
lunacm:> partition login
```

- If your Luna USB HSM G5 firmware is 6.22.0 or newer:

```
lunacm:> role login -name Crypto Officer
```

2. [Optional] View the partition objects and their object handles.

```
lunacm:> partition contents
```

3. Clone objects on the partition to the Luna USB HSM 7 partition by specifying the target slot. You can choose which objects to clone by specifying a comma-separated list of object handles, or specify **all** to clone all objects on the partition. Present the target partition's Crypto Officer credential when prompted.

```
lunacm:> partition clone -slot <slotnum> -objects <comma-separated_list/all>
```

The specified objects are cloned to the target partition.

CHAPTER 12: Partition Backup and Restore

Luna USB HSM 7 allows secure creation, storage, and use of cryptographic data (keys and other objects). It is critically important to safeguard your important cryptographic objects against unforeseen damage or data loss. No device can offer total assurance against equipment failure, physical damage, or human error. Therefore, a comprehensive strategy for making regular backups is essential. There are multiple ways to perform these operations, depending on your implementation.

This section contains the following information:

- > ["Key Concepts for Backup and Restore Operations" below](#)
 - ["Credentials Required to Perform Backup and Restore Operations" on the next page](#)
 - ["Client Software Required to Perform Backup and Restore Operations" on page 108](#)
 - ["Multifactor Quorum Authentication with Luna Backup HSM 7 v1" on page 108](#)
- > ["Planning Your Backup HSM Deployment" on page 108](#)
- > ["Backup and Restore Best Practices" on page 111](#)

Thales recommends backing up your Luna USB HSM 7 to a second Luna USB HSM 7 unit, as this is the most efficient way to ensure redundancy and provides the easiest disaster recovery -- in the event of failure, the second unit is ready to assume production duties immediately.

Luna USB HSM 7 can perform backup and restore operations using the legacy ["Luna Backup HSM G5" on page 146](#), the updated ["Luna Backup HSM 7" on page 113](#), or a ["Backup to Luna Cloud HSM" on page 112](#) service. Refer to the section describing the variant you wish to use:

- > ["Backup to Another Luna USB HSM 7" on page 111](#)
- > ["Backup to Luna Cloud HSM" on page 112](#)
- > ["Luna Backup HSM 7" on page 113](#)
- > ["Luna Backup HSM G5" on page 146](#)

Key Concepts for Backup and Restore Operations

A Crypto Officer (CO) can use the backup HSM to back up and restore the objects in any partition they can log in to, provided that:

- > The application partition and the backup HSM partition share the same domain.
- > The application partition and the backup HSM use the same authentication method (multifactor quorum or password).
- > The CO has the required credentials on the backup HSM.

You can perform backup/restore operations on your application partitions by connecting the backup HSM to the Luna HSM Client workstation. When you connect the backup HSM to a Luna HSM Client workstation, the backup HSM Admin partition is added to the slots listed in LunaCM, allowing you to clone objects between the source application partition and the target backup partition.

NOTE To perform backup operations on Luna HSM Firmware 7.7.0 or newer (V0 or V1 partitions) you require at minimum:

- > [Luna Backup HSM 7 Firmware 7.7.1](#)
- > [Luna Backup HSM G5 Firmware 6.28.0](#)

You can use a Luna Backup HSM with older firmware to restore objects to a V0 or V1 partition, but this is supported for purposes of getting your objects from the older partitions onto the newer V0 or V1 partitions only. V0 and V1 partitions are considered more secure than partitions at earlier firmware versions - any attempt to restore from a higher-security status to lower-security status fails gracefully.

Backups are created and stored as partitions within the Admin partition on the backup HSM.

Credentials Required to Perform Backup and Restore Operations

You require the following credentials to perform backup/restore operations:

Luna USB HSM 7	Remote PED (orange) iKey. Required for multifactor quorum-authenticated backups only, using a local or remote Luna Backup HSM 7 v1, or a remote Luna Backup HSM G5 or Luna Backup HSM 7 v2.
Source Luna USB HSM 7 partition	Crypto Officer (CO). Required to access the objects in the source application partition that will be backed up. Domain. Required to allow objects to be cloned between the source application partition and target backup partition. The domains for the source application partition and target backup partition must match, otherwise the backup will fail.
Target Backup HSM	HSM Security Officer (SO). Required to create or access the target backup partition in the Admin slot, where all backups are archived. Remote PED (orange) iKey. Required for multifactor quorum-authenticated backups only, using a local or remote Luna Backup HSM 7 v1, or a remote Luna Backup HSM 7 v2 or Luna Backup HSM G5, to establish a remote PED connection to the HSM that hosts the target backup partition. Note: You create new credentials for both roles on HSM initialization, and use them for subsequent backups to the target backup HSM.
Target Backup Partition	Partition Security Officer (PO). Required to access the target backup partition on a Luna Backup HSM 7. Crypto Officer (CO). Required to access the objects in the target backup partition. Note: You create new credentials on the initial backup, and use them for subsequent backups to the target backup partition.

Client Software Required to Perform Backup and Restore Operations

You must install the Luna HSM Client software and USB driver for the backup HSM on the workstation you intend to use to perform backup and restore operations. The Luna Backup HSM 7 v1 requires minimum [Luna HSM Client 10.1.0](#). The Luna Backup HSM 7 v2 requires minimum [Luna HSM Client 10.4.0](#). Refer to [Luna HSM Client Software Installation](#).

NOTE Ensure that the backup HSM is not connected to the Luna HSM Client workstation when you install or uninstall the client software. Failure to do so may result in the backup HSM becoming unresponsive.

When you install the client software, you must select the following options:

- > The **Backup** option. This installs the driver for the backup HSM and components required for the Remote Backup Service (RBS).
- > The **USB** option. This installs the driver for the backup HSM.
- > The **Network** and/or **PCIe** options, depending on which type of HSM you intend to back up.
- > The **Remote PED** option, if you want to back up multifactor quorum-authenticated partitions. Note that you can install and use a remote PED on the same workstation used to host the backup HSM, or on a different workstation. This option is mandatory for the Luna Backup HSM 7 v1, but a local PED connection can be used for the Luna Backup HSM 7 v2 or Luna Backup HSM G5.

Multifactor Quorum Authentication with Luna Backup HSM 7 v1

The Luna Backup HSM 7 v1 is equipped with a single USB port that is used to connect the backup HSM to a Luna HSM Client workstation. As such, any PED connections to the backup HSM must use a remote PED and the **pedserver** service.

Planning Your Backup HSM Deployment

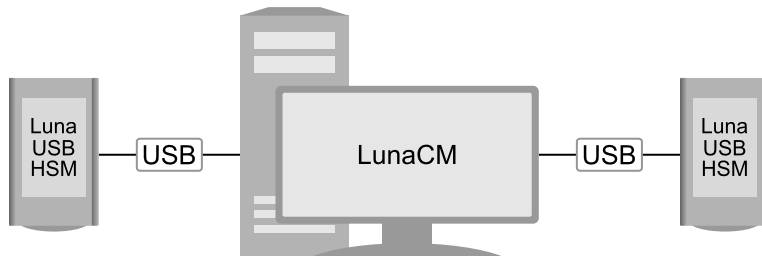
When setting up your backup deployment, you have multiple configuration options. This section will help you choose the right configuration, depending on where you prefer to keep your backups. You can use a Luna Backup HSM, Luna Cloud HSM service, or an application partition on another Luna HSM for backup/restore operations.

Backup and restore operations require that cloning be enabled.

- > ["Backup to Another Luna USB HSM 7" on the next page](#)
- > ["Partition to Partition" on the next page](#)
- > ["Backup to Luna Cloud HSM" on the next page](#)
- > ["Backup HSM Connected to the Client Workstation" on the next page](#)
- > ["Backup HSM Installed Using Remote Backup Service" on page 110](#)

Backup to Another Luna USB HSM 7

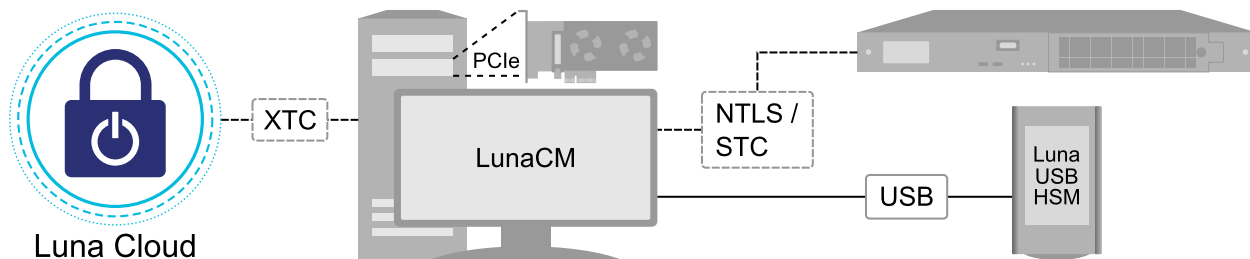
Thales recommends backing up your Luna USB HSM 7 to a second Luna USB HSM 7 unit using direct slot-to-slot cloning. The partitions on both HSMs must be initialized with the same authentication method (password or iKey) and cloning domain, and they must be connected to the same Luna HSM Client computer.



See ["Backup to Another Luna USB HSM 7" on page 111](#).

Partition to Partition

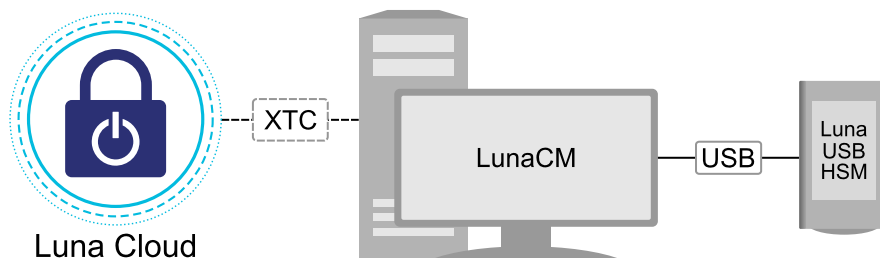
You can clone objects from any Luna 7 application partition to any other Luna 7 partition that shares its cloning domain. You must have the Crypto Officer credential for both partitions. Both partitions must use the same authentication method (either password or iKey).



See ["Cloning Objects to Another Application Partition" on page 31](#).

Backup to Luna Cloud HSM

You can securely back up the contents of any password- or multifactor quorum-authenticated Luna 7 partition to a Luna Cloud HSM service.

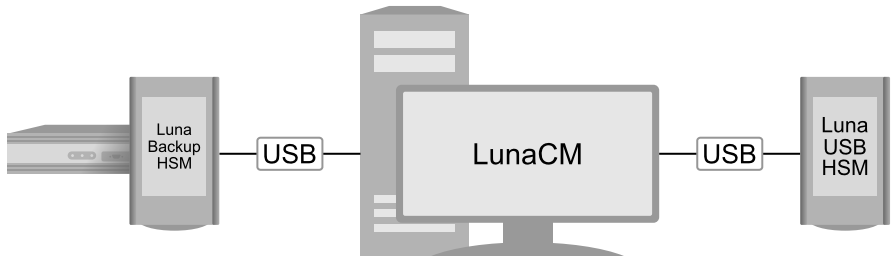


See ["Cloning Objects to Another Application Partition" on page 31](#).

Backup HSM Connected to the Client Workstation

In this configuration, the Luna Backup HSM is connected to a USB port on the client workstation. It is useful in deployments where the partition Crypto Officer keeps backups at the client. This allows you to perform backup/restore operations for all application partitions that appear as visible slots in LunaCM. You can restore a

partition backup to the original source partition or to another existing Luna application partition that shares the same cloning domain.

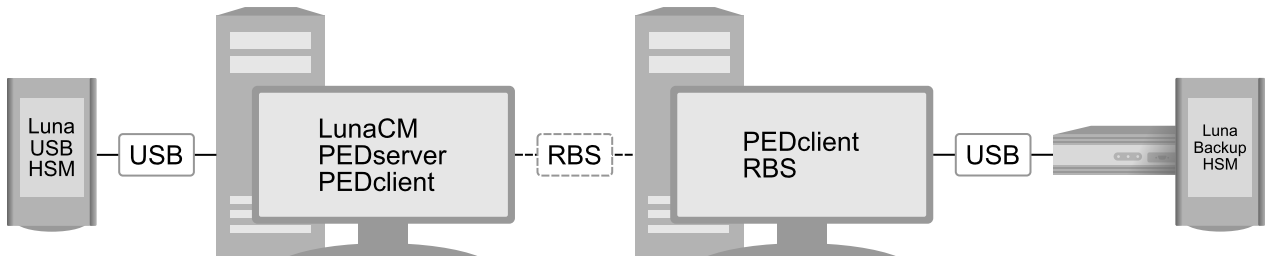


Depending on your Luna Backup HSM and Luna HSM Client version, refer to:

Hardware/Software Requirements	Available Procedures
<ul style="list-style-type: none"> > "Luna Backup HSM 7" on page 113 v2 > Luna HSM Client 10.4.0 or newer 	<ul style="list-style-type: none"> > "Luna Backup HSM 7 Using Direct Multifactor Quorum Authentication" on page 121 > "Luna Backup HSM 7 Using Remote Multifactor Quorum Authentication" on page 129 > "Luna Backup HSM 7 Using Password Authentication" on page 139
<ul style="list-style-type: none"> > "Luna Backup HSM 7" on page 113 v1 or v2 > Luna HSM Client 10.1.0 or newer 	<ul style="list-style-type: none"> > "Luna Backup HSM 7 Using Remote Multifactor Quorum Authentication" on page 129 > "Luna Backup HSM 7 Using Password Authentication" on page 139
<ul style="list-style-type: none"> > "Luna Backup HSM G5" on page 146 	<ul style="list-style-type: none"> > "Backup/Restore Using Luna Backup HSM G5" on page 159

Backup HSM Installed Using Remote Backup Service

In this configuration, the Luna Backup HSM is connected to a remote client workstation that communicates with the Luna USB HSM 7 client via the Remote Backup Service (RBS). It is useful in deployments where backups are stored in a separate location from the Luna USB HSM 7, to mitigate the consequences of catastrophic loss (fire, flood, etc).



Refer to ["Configuring a Remote Backup Server" on page 164](#).

Backup and Restore Best Practices

To ensure that your data is protected in the event of a failure or other catastrophic event, Thales recommends that you use the following best practices as part of a comprehensive backup strategy:

CAUTION! Failure to develop and exercise a comprehensive backup and recovery plan may prevent you from being able to recover from a catastrophic event. Although Thales provides a robust set of backup hardware and utilities, we cannot guarantee the integrity of your backed-up key material, especially if stored for long periods. Thales strongly recommends that you exercise your recovery plan at least semi-annually (every six months) to ensure that you can fully recover your key material.

Develop and document a backup and recovery plan

This plan should include the following:

- > What is being backed up
- > The backup frequency
- > Where the backups are stored
- > Who is able to perform backup and restore operations
- > Frequency of exercising the recovery test plan

Make multiple backups

To ensure that your backups are always available, build redundancy into your backup procedures.

Use off-site storage

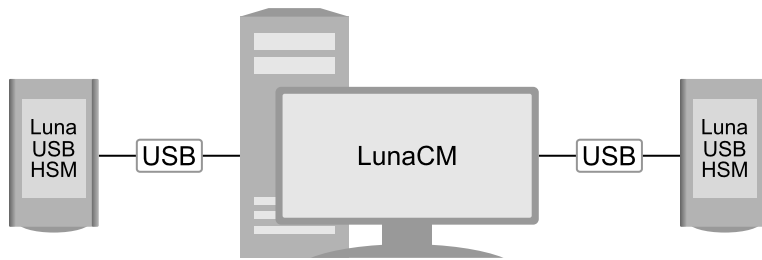
In the event of a local catastrophe, such as a flood or fire, you might lose both your working HSMs and locally-stored backup HSMs. To fully protect against such events, always store a copy of your backups at a remote location.

Regularly exercise your disaster recovery plan

Execute your recovery plan at least semi-annually (every six months) to ensure that you can fully recover your key material. This involves retrieving your stored Backup HSMs and restoring their contents to a test partition, to ensure that the data is intact and that your recovery plan works as documented.

Backup to Another Luna USB HSM 7

Thales recommends backing up your Luna USB HSM 7 to a second Luna USB HSM 7 unit. This way, the second unit can immediately resume production in case of failure. The partitions on both HSMs must be initialized with the same authentication method (password or iKey) and cloning domain, and they must be connected to the same Luna HSM Client computer. You can clone objects from one Luna USB HSM 7 to the other using direct slot-to-slot cloning, or set up an HA group to synchronize your partition contents between the two HSMs.



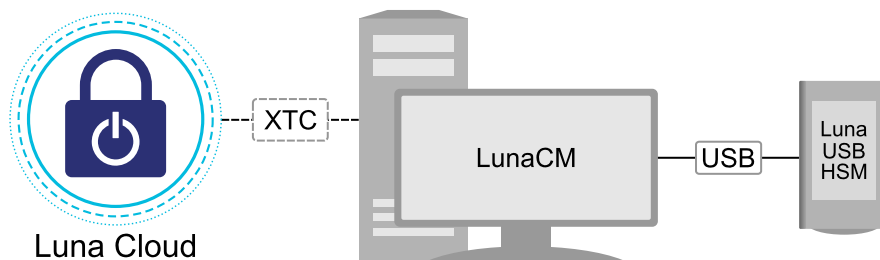
Refer to the following sections to set up and use this backup method:

- > [Installing the Luna USB HSM 7 Hardware](#) for instructions on setting up your second Luna USB HSM 7.
- > ["Cloning Objects to Another Application Partition" on page 31](#) for instructions on cloning objects to another Luna USB HSM 7 manually.
- > ["Configuring a High-Availability Group" on page 90](#) for instructions on setting up two Luna USB HSM 7s in an HA group to be synchronized automatically.

NOTE To use a second Luna USB HSM 7 as a backup, both HSMs must be connected to the same Luna HSM Client computer. If you want to keep backups remotely, you must use a Luna Backup HSM with Remote Backup Service (RBS).

Backup to Luna Cloud HSM

Luna Cloud HSM services allow you to back up your partition objects securely in the cloud, with no additional HSM hardware. With Universal Cloning(includes Extended Domain Management and CPv4, and requires Luna HSM Firmware 7.8.0 or newer for the on-premises HSM and [Luna HSM Client 10.5.0](#) or newer), Luna Cloud HSM can now clone keys from both password-and multifactor quorum-authenticated Luna HSMs. You can create Luna Cloud HSM backups using slot-to-slot cloning, or set up an HA group to synchronize your partition contents with Luna Cloud HSM.



Refer to the following sections to set up and use this backup method:

- > [Adding a Luna Cloud HSM Service](#) for instructions on adding and initializing Luna Cloud HSM for use with your deployment.
- > ["Cloning Objects to Another Application Partition" on page 31](#) for instructions on creating Luna Cloud HSM backups.
- > ["Configuring a High-Availability Group" on page 90](#) for instructions on setting up a synchronized Luna Cloud HSM service.
- > ["Cloning Keys Between Luna 6, Luna 7, and Luna Cloud HSM" on page 32](#) for additional information about mixed-environment cloning.

Luna Backup HSM 7

The Luna Backup HSM 7 is a full-featured, hand-held, USB-attached backup HSM that includes an informational full-color display. The Luna Backup HSM 7 connects easily to a client workstation using the included USB 3.0 Type C cable, and includes a universal 5V external power supply, which may be required to power the device in some instances.

The refreshed v2 model includes a USB-C port, which, combined with a USB-A to USB-C adapter, allows you to insert iKeys directly into the HSM, greatly simplifying the multifactor quorum authentication procedure and, depending on your configuration, eliminating the need for a Luna PED in backup/restore operations.



The Luna Backup HSM 7 is available in the following models. All models can be initialized in multifactor quorum or password-authenticated mode for backing up either multifactor quorum or password authenticated partitions. In-field storage upgrades are not available.

B700	32 MB storage, up to 100 partitions of the same authentication type
B750	128 MB storage, up to 100 partitions of the same authentication type
B790	256 MB storage, up to 100 partitions of the same authentication type

For setup, management, and backup/restore procedures, refer to the following sections:

- > ["Luna Backup HSM 7 Hardware Installation" on the next page](#)
- > ["Managing the Luna Backup HSM 7" on page 117](#)
- > ["Configuring a Remote Backup Server" on page 164](#)

Refer to the following procedures depending on your authentication method, Luna Backup HSM 7 hardware, and Luna HSM Client versions:

Multifactor Quorum Authentication

- > ["Luna Backup HSM 7 Using Direct Multifactor Quorum Authentication" on page 121](#) (requires "Luna Backup HSM 7" above **v2** and Luna HSM Client 10.4.0 or newer)
- > ["Luna Backup HSM 7 Using Remote Multifactor Quorum Authentication" on page 129](#) (requires Luna HSM Client 10.1.0 or newer)

Password Authentication

- > ["Luna Backup HSM 7 Using Password Authentication" on page 139](#) (requires Luna HSM Client 10.1.0 or newer)

Luna Backup HSM 7 Hardware Installation

The following topics describe how to install and connect a Luna Backup HSM 7:

- > ["Luna Backup HSM 7 Required Items" below](#)
- > ["Luna Backup HSM 7 Hardware Functions" on the next page](#)
- > ["Installing the Luna Backup HSM 7 Hardware" on page 116](#)



The Luna Backup HSM 7 complies with the following:



Luna Backup HSM 7 Required Items

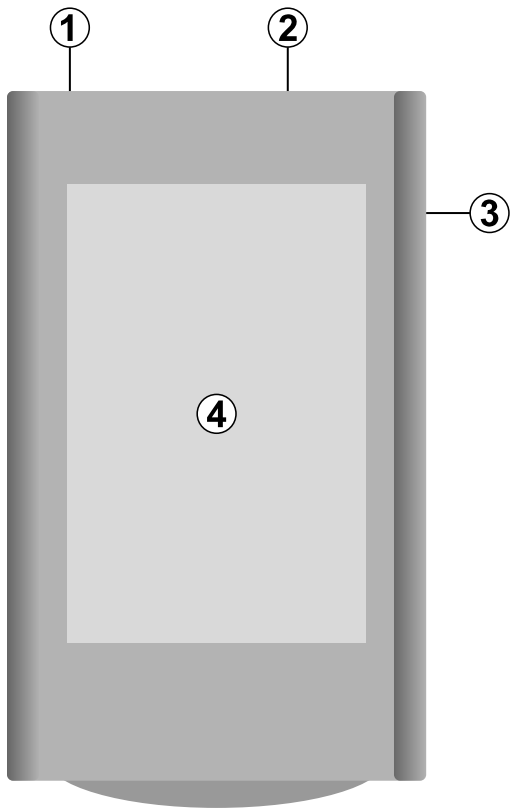
This section provides a list of the components you should have received with your Luna Backup HSM 7 order.

Qty	Item
1	Luna Backup HSM 7 v1 or v2 <div>   </div>

Qty	Item
1	USB 3.0 Cable: Type A to Type C 
1	5V Power Supply with replaceable plug modules for international use. 
1	USB-A to USB-C adapter (included with v2 only)  Used to connect iKeys to the Luna Backup HSM 7 v2.

Luna Backup HSM 7 Hardware Functions

The Luna Backup HSM 7 hardware is illustrated below, with important features labeled.



1	5V power supply connector. Required only if the USB port connected to (2) does not supply adequate power to the Luna Backup HSM 7.
2	USB-C connector. Used for USB power and connecting to the client computer.
3	USB-C connector (v2 only). Used for connecting iKeys to authenticate roles on the HSM. Requires the included USB-A to USB-C adapter.
4	LED touchscreen. Displays information about the Luna Backup HSM 7 and is used to input role-specific information like PINs.

The Luna Backup HSM 7 does not contain an internal battery, and maintains the integrity of its stored key material without being connected to power.

Installing the Luna Backup HSM 7 Hardware

The backup HSM is a USB device. To install the backup HSM, connect it to a USB port on a Luna HSM Client workstation using the included USB cable. The workstation must be running Luna HSM Client software that supports the backup HSM and provides the required drivers.

NOTE On most Luna HSM Client computers, the USB port provides adequate power, and connecting the provided power supply is not recommended. Wait and confirm that the HSM boots properly. If the HSM fails to boot up:

1. Disconnect the HSM from the USB port.
2. Connect the HSM to power using the provided power supply. Wait for it to boot completely.
3. Reconnect the HSM to the USB port on the client.

If the HSM is connected to the USB port before the power supply, you may encounter an issue where the HSM occasionally loses contact with the client, and must be power cycled.

NOTE If you are installing Luna HSM Client on Windows, the driver may not be installed unless the Luna device is connected to the computer first; refer to [Backup/USB/PCIe Drivers Not Installed on Windows 10 or Windows Server 2022 Unless Device is Connected](#).

Managing the Luna Backup HSM 7

This section contains the following procedures for maintaining and using the Luna Backup HSM 7:

- > ["Recovering the Luna Backup HSM 7 from Secure Transport Mode" below](#)
- > ["Configuring the Luna Backup HSM 7 for FIPS Compliance" on the next page](#)
- > ["Updating the Luna Backup HSM 7 Firmware" on the next page](#)
- > ["Rolling Back the Luna Backup HSM 7 Firmware" on page 120](#)

Recovering the Luna Backup HSM 7 from Secure Transport Mode

The Luna Backup HSM 7 is shipped in [Secure Transport Mode](#) (STM). STM provides a logical check on the firmware and critical security parameters (such as configuration, keys, policies, roles, etc.) so that the authorized recipient can determine if these have been altered while the HSM was in transit.

NOTE This procedure requires connection to a client machine with [Luna HSM Client 10.1.0](#) or newer installed. This operation is not possible while the Backup HSM is connected to the Luna Network HSM 7 appliance.

To recover the Luna Backup HSM 7 from STM

1. Connect the Luna Backup HSM 7 to a USB port on a Luna HSM Client workstation with the **Backup** option installed (refer to [Luna HSM Client Software Installation](#) for your client operating system).
2. Launch LunaCM on the client workstation.
3. Select the slot assigned to the Luna Backup HSM 7 Admin partition.
lunacm:> **slot set -slot** <slot_id>
4. Recover the HSM from Secure Transport Mode. See [Secure Transport Mode](#) for more information about the Random User String:
lunacm:> **stm recover -randomuserstring** <string>

NOTE Recovering a Luna Backup HSM 7 from STM may take up to three minutes.

Configuring the Luna Backup HSM 7 for FIPS Compliance

Luna Backup HSM Firmware 7.7.1 and newer uses the same updated cloning protocol as Luna HSM Firmware 7.7.0 and newer. For the Luna Backup HSM 7 to be FIPS-compliant, it must restrict restore operations to application partitions that use the new protocol. This restriction is applied by setting **HSM policy 55: Enable Restricted Restore** to **1** on the backup HSM. The Luna Backup HSM 7 must be initialized and connected to a Luna HSM Client computer to set this policy.

When this policy is enabled on the Luna Backup HSM 7, objects that have been backed up from partitions using firmware older than Luna HSM Firmware 7.7.0 can be restored to Luna HSM Firmware 7.7.0 or newer (V0 or V1) partitions only.

CAUTION! FIPS compliance requires that objects are never cloned or restored to an HSM using less secure firmware, and this includes restoring from Luna Backup HSM 7 firmware. If you have backups already stored on the Luna Backup HSM 7 that were taken from pre-7.7.0 partitions, turning this policy ON will prevent you from restoring them to the same source partition. You must update the HSM containing the source partition to Luna HSM Firmware 7.7.0 or newer before restoring from backup.

NOTE **HSM policy 12: Allow non-FIPS algorithms**, which is used to set FIPS-compliant mode on other Luna HSMs, does not apply to the Luna Backup HSM 7. Attempts to change this policy will fail with the error `CKR_CANCEL`.

To configure the Luna Backup HSM 7 for FIPS compliance

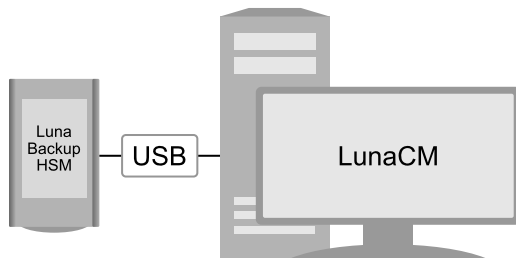
1. On the Luna HSM Client computer, run LunaCM.
2. Set the active slot to the Luna Backup HSM 7.
`lunacm:> slot set -slot <slot_id>`
3. Log in as Backup HSM SO.
`lunacm:> role login -name so`
4. Set **HSM policy 55: Enable Restricted Restore** to **1**.
`lunacm:> hsm changehsm policy -policy 55 -value 1`
5. [Optional] Check that the Luna Backup HSM 7 is now in FIPS approved operation mode.
`lunacm:> hsm showinfo`

```
*** The HSM is in FIPS 140-2 approved operation mode. ***
```

Updating the Luna Backup HSM 7 Firmware

To update the Luna Backup HSM 7, download the desired firmware version from the Thales Support Portal.

Use the following procedure to update the Luna Backup HSM 7 firmware using LunaCM. The Backup HSM SO must complete this procedure.



Prerequisites

- > Luna Backup HSM 7 firmware update file (<filename>.fuf)
- > firmware update authentication code file (<filename>.txt)
- > If you have backups currently stored on the Backup HSM, they must take up less than 60% of storage capacity, or the firmware upgrade will not proceed.

NOTE If you are updating from [Luna Backup HSM 7 Firmware 7.3.2](#), objects and partitions must be re-sized to include additional object overhead associated with the new V1 partitions - this action is automatic (see ["V0 and V1 Partitions" on page 67](#)). This conversion can take a long time, depending on the number of objects stored on the Backup HSM (a few minutes to several hours). Ensure that you can leave the update operation uninterrupted for this amount of time. Do not interrupt the procedure even if the operation appears to have stalled.

To update the Luna Backup HSM 7 firmware using LunaCM

1. Copy the firmware file (<filename>.fuf) and the authentication code file (<filename>.txt) to the Luna HSM Client root directory.
 - Windows: C:\Program Files\SafeNet\LunaClient
 - Linux: /usr/safenet/lunaclient/bin
 - Solaris: /opt/safenet/lunaclient/bin

NOTE On some Windows configurations, you might not have authority to copy or unzip files directly into **C:\Program Files\...** If this is the case, put the files in a known location that you can reference in a LunaCM command.

2. Launch LunaCM.
3. If more than one HSM is installed, set the active slot to the Admin partition of the HSM you wish to update.


```
lunacm:> slot set -slot <slot_number>
```
4. [Multifactor Quorum-Authenticated]
 - If you are updating a Luna Backup HSM 7 v2, you will insert iKeys directly into the Backup HSM; skip to step 5.
 - If you are updating a Luna Backup HSM 7 v1, connect to the Remote PED server.


```
lunacm:> ped connect [-ip <IP_address>] [-port <port#>]
```
5. Log in as HSM SO.

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

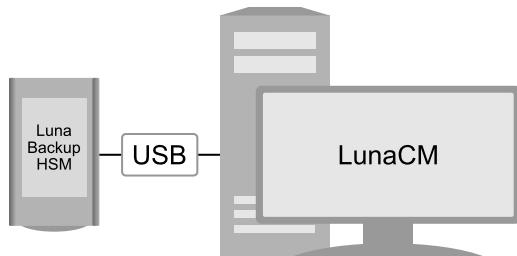
6. Apply the new firmware update by specifying the update file and the authentication code file. If the files are not located in the Luna HSM Client root directory, specify the full filepaths.

```
lunacm:> hsm updatefw -fuf <filename>.fuf -authcode <filename>.txt
```

The previous version of the firmware is stored in reserve on the HSM. To restore the previous firmware version, see ["Rolling Back the Luna Backup HSM 7 Firmware" below](#).

Rolling Back the Luna Backup HSM 7 Firmware

When you update the Luna Backup HSM 7 firmware, the previous version of the firmware is stored in reserve on the HSM. If required, you can use the following procedure to roll back the HSM firmware to the previous version.



CAUTION! Firmware rollback is destructive; earlier firmware versions might have older mechanisms and security vulnerabilities that a new version does not. Ensure that you do not have any important backups stored on the HSM before you proceed. This procedure zeroizes the HSM and all backups are erased.

Prerequisites

- > Connect the Luna Backup HSM 7 to a Luna HSM Client workstation.

To roll back the Luna Backup HSM 7 firmware to the previous version

1. At the LunaCM prompt, set the active slot to the Backup HSM.

```
lunacm:> slot set -slot <slot_number>
```
2. Check the previous firmware version that is available on the HSM.

```
lunacm:> hsm showinfo
```
3. [Multifactor Quorum-Authenticated]
 - If you are rolling back a Luna Backup HSM 7 v2, you can insert iKeys directly into the Backup HSM; skip to step 5.
 - If you are rolling back a Luna Backup HSM 7 v1, connect to the Remote PED server.

```
lunacm:> ped connect [-ip <IP_address>] [-port <port#>]
```
4. Log in as HSM SO.

```
lunacm:> role login -name so
```
5. Roll back the Backup HSM firmware.

```
lunacm:> hsm rollbackfw
```


Luna Backup HSM 7 Using Direct Multifactor Quorum Authentication

In this configuration, you connect the Luna Backup HSM 7 to a USB port on the Luna HSM Client, and insert iKeys directly into the Luna Backup HSM 7. This allows you to perform backup/restore operations for all application partitions that can be accessed by the client. You can restore a partition backup to the original source partition or to another existing Luna application partition that shares the same cloning domain. To use this method, you require:

- > ["Luna Backup HSM 7" on page 113 v2](#)
- > [Luna HSM Client 10.4.0](#) or newer

This section provides instructions for the following procedures:

- > ["Initializing the Luna Backup HSM 7" below](#)
- > ["Configuring the Luna Backup HSM 7 for FIPS Compliance" on page 123](#)
- > ["Backing Up a Multifactor Quorum-Authenticated Partition" on page 123](#)
- > ["Restoring To a Multifactor Quorum-Authenticated Partition" on page 127](#)

Initializing the Luna Backup HSM 7

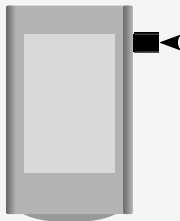
You must initialize the Luna Backup HSM 7 prior to first use. You can initialize the backup HSM by connecting it to a Luna HSM Client and using LunaCM commands to perform the initialization.

Prerequisites

You need the following iKeys:

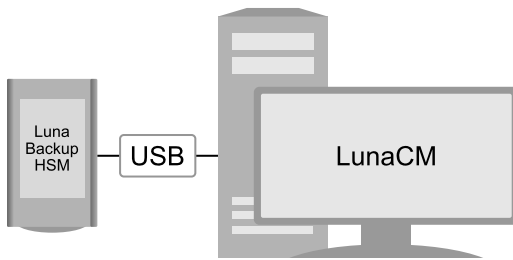
- > N number of blue (HSM SO) iKeys, as defined by the M of N scheme you choose for the HSM SO role, plus the number required to create duplicate iKeys as necessary.
- > Blank or reused red (Domain) iKey(s)

NOTE Use the USB-C adapter in the USB port on the right side of the Luna Backup HSM 7 to insert iKeys:



To initialize the Luna Backup HSM 7

1. Connect your Luna Backup HSM 7 to a workstation:



- a. Install the required Luna HSM Client software on the workstation, including the **Backup** option. See ["Client Software Required to Perform Backup and Restore Operations"](#) on page 108 for details.

NOTE If you are installing Luna HSM Client on Windows, the driver may not be installed unless the Luna device is connected to the computer first; refer to [Backup/USB/PCIe Drivers Not Installed on Windows 10 or Windows Server 2022 Unless Device is Connected](#).

- b. Connect the backup HSM directly to the Luna HSM Client workstation using the included USB cable.

NOTE On most Luna HSM Client computers, the USB port provides adequate power, and connecting the provided power supply is not recommended. Wait and confirm that the HSM boots properly. If the HSM fails to boot up:

1. Disconnect the HSM from the USB port.
2. Connect the HSM to power using the provided power supply. Wait for it to boot completely.
3. Reconnect the HSM to the USB port on the client.

If the HSM is connected to the USB port before the power supply, you may encounter an issue where the HSM occasionally loses contact with the client, and must be power cycled.

2. Launch LunaCM on the client workstation.
3. Select the slot assigned to the backup HSM Admin partition.

```
lunacm:> slot set -slot <slot_id>
```

4. If necessary, recover the HSM from Secure Transport Mode. See [Secure Transport Mode](#) for more information:

```
lunacm:> stm recover -randomuserstring <string>
```

NOTE Recovering a Luna Backup HSM 7 from secure transport mode may take up to three minutes.

5. Initialize the selected backup HSM, specifying a label and the **-iped** authentication mode.

```
lunacm:> hsm init -iped -label <label>
```

- > You are prompted by the touchscreen for the blue HSM SO iKey(s) and red Domain iKey(s). Respond to the prompts and insert and set the PINs on the required keys when requested. Ensure that you label any new iKeys that you create during this process.

Configuring the Luna Backup HSM 7 for FIPS Compliance

[Luna Backup HSM Firmware 7.7.1](#) and newer uses the same updated cloning protocol as Luna HSM Firmware 7.7.0 and newer. For the Luna Backup HSM 7 to be FIPS-compliant, it must restrict restore operations to application partitions that use the new protocol. This restriction is applied by setting **HSM policy 55: [Enable Restricted Restore](#)** to **1** on the backup HSM. The Luna Backup HSM 7 must be initialized and connected to a Luna HSM Client computer to set this policy.

When this policy is enabled on the Luna Backup HSM 7, objects that have been backed up from partitions using firmware older than Luna HSM Firmware 7.7.0 can be restored to Luna HSM Firmware 7.7.0 or newer (V0 or V1) partitions only.

CAUTION! FIPS compliance requires that objects are never cloned or restored to an HSM using less secure firmware, and this includes restoring from Luna Backup HSM 7 firmware. If you have backups already stored on the Luna Backup HSM 7 that were taken from pre-7.7.0 partitions, turning this policy ON will prevent you from restoring them to the same source partition. You must update the HSM containing the source partition to Luna HSM Firmware 7.7.0 or newer before restoring from backup.

NOTE **HSM policy 12: [Allow non-FIPS algorithms](#)**, which is used to set FIPS-compliant mode on other Luna HSMs, does not apply to the Luna Backup HSM 7. Attempts to change this policy will fail with the error `CKR_CANCEL`.

To configure the Luna Backup HSM 7 for FIPS compliance

1. On the Luna HSM Client computer, run LunaCM.
2. Set the active slot to the Luna Backup HSM 7.
lunacm:> **slot set -slot** <slot_id>
3. Log in as Backup HSM SO.
lunacm:> **role login -name so**
4. Set **HSM policy 55: [Enable Restricted Restore](#)** to **1**.
lunacm:> **hsm changehsm policy -policy 55 -value 1**
5. [Optional] Check that the Luna Backup HSM 7 is now in FIPS approved operation mode.
lunacm:> **hsm showinfo**

*** The HSM is in FIPS 140-2 approved operation mode. ***

Backing Up a Multifactor Quorum-Authenticated Partition

Backups are created and stored as partitions within the Admin partition on the Luna Backup HSM 7. A new backup partition is created on initial backup. For subsequent backups, you can choose to replace the contents of the existing backup partition with the current source partition objects, or add new objects in the source partition to the existing backup partition. Like all cloning operations, the source and target backup partitions must be initialized with the same domain.

Prerequisites

> You have the required credentials:

If the source partition is not activated:

- [Remote PED authentication] The Remote PED Vector (orange) iKey(s) for the source HSM
- The Crypto Officer (black) iKey(s) for the source partition

TIP If the source partition is activated, only the source partition Crypto Officer's challenge secret is required. To simplify the backup process and minimize interactions with the PED, it is recommended that you activate the CO role on the user partitions you want to backup. See [Activation on Multifactor Quorum-Authenticated Partitions](#) for more information.

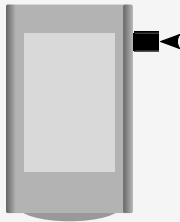
If you are creating a new backup partition:

- New or reused Partition SO (blue) iKey(s) to initialize the backup partition
- The Domain (red) iKey(s) for the source partition, to initialize the domain on the backup
- New or reused Crypto Officer (black) iKey(s) to initialize the CO role on the backup partition

If you are backing up to an existing backup partition whose domain matches the source partition:

- The existing Partition SO (blue) iKey(s) for the backup partition, to log in
- The existing Crypto Officer (black) iKey(s) for the backup partition

NOTE Use the USB-C adapter in the USB port on the right side of the Luna Backup HSM 7 to insert iKeys:



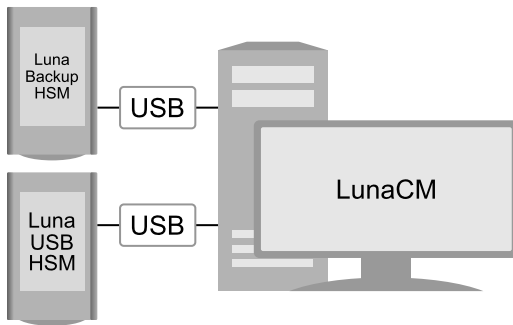
> The following policies are set:

- **HSM policy 16:** [Allow network replication](#) must be set to **1 (ON)** on the HSM that hosts the user partition.
- [V0 partitions] **Partition policy 0:** ["Allow private key cloning" on page 50](#) is set to **1 (ON)** on the user partition.
- [V0 partitions] **Partition policy 4:** ["Allow secret key cloning" on page 51](#) is set to **1 (ON)** on the user partition.

NOTE HSS (PQC) private keys cannot be cloned, due to inherent restrictions of that key type. This means that backup and HA synchronization fail if HSS private keys are encountered in your application partition.

To back up a multifactor quorum-authenticated partition

1. Configure your Luna HSM Client workstation:



- a. If you have not already done so, install the required client software on the Luna HSM Client workstation. See ["Client Software Required to Perform Backup and Restore Operations"](#) on page 108 for details.

NOTE If you are installing Luna HSM Client on Windows, the driver may not be installed unless the Luna device is connected to the computer first; refer to [Backup/USB/PCIe Drivers Not Installed on Windows 10 or Windows Server 2022 Unless Device is Connected](#).

- b. Connect the backup HSM directly to the Luna HSM Client workstation using the included USB cable.

NOTE On most Luna HSM Client computers, the USB port provides adequate power, and connecting the provided power supply is not recommended. Wait and confirm that the HSM boots properly. If the HSM fails to boot up:

1. Disconnect the HSM from the USB port.
2. Connect the HSM to power using the provided power supply. Wait for it to boot completely.
3. Reconnect the HSM to the USB port on the client.

If the HSM is connected to the USB port before the power supply, you may encounter an issue where the HSM occasionally loses contact with the client, and must be power cycled.

2. Launch LunaCM on the workstation that hosts the Luna USB HSM 7 partition slots.
3. Identify the slot assignments for:
 - The Luna USB HSM 7 partition you want to back up.
 - The Luna Backup HSM 7 admin partition (where all backups are stored).

lunacm:> **slot list**

If you cannot see both slots, check your connections or configure your client as required.

4. Select the Luna USB HSM 7 partition:

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

5. Log in to the partition as Crypto Officer (CO):

- If the partition is activated, use the following command and provide the Crypto Officer (CO) challenge secret as prompted:

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

- If the partition is not activated, log in to the selected Luna USB HSM 7 partition as the Crypto Officer (CO):

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

Respond to the prompts on the Luna USB HSM 7 touchscreen to provide the black (CO) key(s) and PIN.

6. Initiate the backup:

```
lunacm:> partition archive backup -slot <backup_HSM_admin_slot> [-partition <target_backup_label>] [-append] [-replace] [-smkonly]
```

If you omit the **-partition** option when creating a new backup, the partition is assigned a default name (<source_partition_name>_<YYYYMMDD>) based on the source HSM's internally-set time and date.

If you are backing up a V1 partition, include **-smkonly** to back up the SMK only. By default, the SMK and any encrypted cryptographic material on the partition are backed up.

The backup begins once you have completed the authentication process. Objects are backed up one at a time. If you are backing up to an existing backup partition, you can use the following options to define how individual objects are backed up:

-append	Add only new objects to an existing backup.
-replace	Delete the existing objects in a target backup partition and replace them with the contents of the source user partition. This is the default.
-append -replace	Add new objects and replace existing objects that have the same OUID but a different fingerprint (such as would occur if any of the object attributes were changed since the previous backup).

NOTE If the backup operation is interrupted (if the Backup HSM is unplugged, or if you fail to respond to PED prompts, for example), the Backup HSM's full available space can become occupied with a single backup partition. If this occurs, delete the backup partition with `lunacm:> partition archive delete` before reattempting the backup operation.

7. Respond to the prompts on the Luna Backup HSM 7 touchscreen to insert the following iKeys:

If you are creating a new backup partition:

- i. The blue HSM SO iKey(s) for the backup HSM.
- ii. You are prompted to initialize the backup Partition SO role by creating a new blue iKey or reusing an existing key. After you initialize the role, you are prompted to insert the key again to log in as Partition SO.
- iii. The red Domain iKey(s). This must be the same iKey(s) used for the Luna USB HSM 7 partition, otherwise the backup will fail.
- iv. The blue Partition SO iKey(s) for the backup partition, to log in again.
- v. You are prompted to initialize the Crypto Officer role for the backup by creating a new black iKey or reusing an existing key. After you initialize the role, you are prompted to insert the key again to log in as Crypto Officer.

If you are backing up to an existing backup partition whose domain matches the source partition:

- i. The blue HSM SO iKey(s) for the backup HSM.
- ii. The blue Partition SO iKey(s) for the backup.
- iii. The black Crypto Officer iKey(s) for the backup.

Restoring To a Multifactor Quorum-Authenticated Partition

You can restore the objects from a multifactor quorum-authenticated backup partition to the same partition that was originally backed up, or to another partition that has been initialized with the same domain (red iKey).

Prerequisites

- > The target partition must be initialized using the same domain (red iKey) as the backup partition, the Crypto Officer role must be initialized and the CO role credential changed from its initial value.
- > You require the Crypto Officer challenge secret for the target partition.

If the target partition is not activated, you also require:

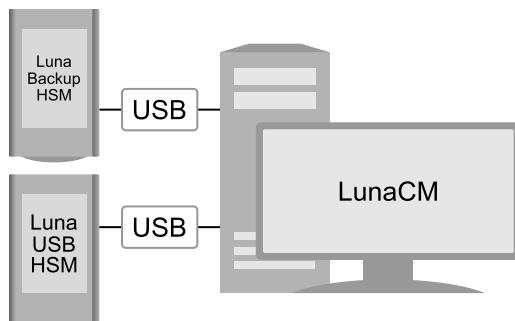
- [Remote PED authentication] The Remote PED Vector (orange) iKey(s) for the target HSM
- The Crypto Officer (black) iKey(s) for the target partition

TIP If the target partition is activated, only the Crypto Officer's challenge secret is required. To simplify the backup process and minimize interactions with the PED, it is recommended that you activate the CO role on the user partitions you want to restore from backup. See [Activation on Multifactor Quorum-Authenticated Partitions](#) for more information.

- > The following policies are set:
 - **HSM policy 16: Allow network replication** must be set to **1 (ON)** on the HSM that hosts the user partition you want to restore to.
 - [V0 partitions] **Partition policy 0: "Allow private key cloning" on page 50** is set to **1 (ON)** on the user partition you want to restore to.
 - [V0 partitions] **Partition policy 4: "Allow secret key cloning" on page 51** is set to **1 (ON)** on the user partition you want to restore to.

To restore a multifactor quorum-authenticated partition

1. Configure your Luna HSM Client workstations:



- a. If you have not done so already, install the required client software on the Luna HSM Client workstation. See [Luna HSM Client Software Installation](#) for details.

NOTE If you are installing Luna HSM Client on Windows, the driver may not be installed unless the Luna device is connected to the computer first; refer to [Backup/USB/PCIe Drivers Not Installed on Windows 10 or Windows Server 2022 Unless Device is Connected](#).

- b. Connect the backup HSM directly to the Luna HSM Client workstation using the included USB cable.

NOTE On most Luna HSM Client computers, the USB port provides adequate power, and connecting the provided power supply is not recommended. Wait and confirm that the HSM boots properly. If the HSM fails to boot up:

1. Disconnect the HSM from the USB port.
2. Connect the HSM to power using the provided power supply. Wait for it to boot completely.
3. Reconnect the HSM to the USB port on the client.

If the HSM is connected to the USB port before the power supply, you may encounter an issue where the HSM occasionally loses contact with the client, and must be power cycled.

2. Launch LunaCM on the workstation that hosts the Luna USB HSM 7 and backup partition slots.
3. Identify the slot assignments for:

- the Luna USB HSM 7 partition you want to restore to.
- the backup HSM admin partition (where all backups are stored).

lunacm:> **slot list**

If you cannot see both slots, check your connections or configure your client as required.

4. Select the Luna USB HSM 7 partition you want to restore from backup:

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

5. Log in to the partition as Crypto Officer (CO):

- If the partition is activated, use the following command and provide the Crypto Officer (CO) challenge secret as prompted:

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

- If the partition is not activated, log in to the selected Luna USB HSM 7 partition as the Crypto Officer (CO):

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

Respond to the prompts on the Luna USB HSM 7 touchscreen to provide the black (CO key(s) and PIN.

6. List the available backups on the Backup HSM by specifying the Backup HSM's slot number. You will require the backup partition label to perform the restore operation.

lunacm:> **partition archive list-slot <backup_HSM_admin_slot>**

7. Initiate the restore operation. Respond to the prompts on the Luna Backup HSM 7 touchscreen to insert the required iKeys.

lunacm:> **partition archive restore -slot <backup_HSM_admin_slot> -partition <backup_partition_label> [-smkonly]**

CAUTION! The **-replace** option is deprecated and has been removed in [Luna HSM Client 10.7.0](#) and newer. If you wish to restore an earlier version of an object, Thales recommends deleting the object(s) manually before restoring the partition from backup.

Ensure that the target partition can receive objects from the backup HSM before deleting objects or using [partition archive restore](#) with the **-replace** option; the cloning protocol may prevent objects from being restored, even if LunaCM states that `x objects will be restored`. This may occur if **HSM policy 55: Enable Restricted Restore** was enabled on the Luna Backup HSM 7 since the original backup was taken. If your partition is on an HSM with firmware older than Luna HSM Firmware 7.7.0, you must update to 7.7.0 or newer to restore objects from this backup.

NOTE If you are restoring a V1 backup to a V1 partition, include **-smkonly** to restore the SMK only (see ["V0 and V1 Partitions" on page 67](#) for more information). By default, the SMK and any cryptographic material on the backup are restored.

The restore operation begins once you have completed the authentication process. Objects are restored one at a time.

Luna Backup HSM 7 Using Remote Multifactor Quorum Authentication

In this configuration, you connect the Luna Backup HSM 7 to a USB port on the Luna HSM Client, and insert iKeys into a Remote Luna PED. This allows you to perform backup/restore operations for all application partitions that can be accessed by the client. You can restore a partition backup to the original source partition or to another existing Luna application partition that shares the same cloning domain. To use this method, you require:

- > ["Luna Backup HSM 7" on page 113 v1 or v2](#)
- > [Luna HSM Client 10.1.0](#) or newer

This section provides instructions for the following procedures:

- > ["Initializing the Luna Backup HSM 7 for Multifactor Quorum Authentication" below](#)
- > ["Configuring the Luna Backup HSM 7 for FIPS Compliance" on page 132](#)
- > ["Backing Up a Multifactor Quorum-Authenticated Partition" on page 133](#)
- > ["Restoring To a Multifactor Quorum-Authenticated Partition" on page 136](#)

Initializing the Luna Backup HSM 7 for Multifactor Quorum Authentication

You must initialize the Luna Backup HSM 7 prior to first use. You can initialize the backup HSM by connecting it to a Luna HSM Client and using LunaCM commands to perform the initialization.

Prerequisites

You will need the following iKeys:

- > A blank orange (PED vector) iKey, plus the number required to create duplicate iKeys as necessary.

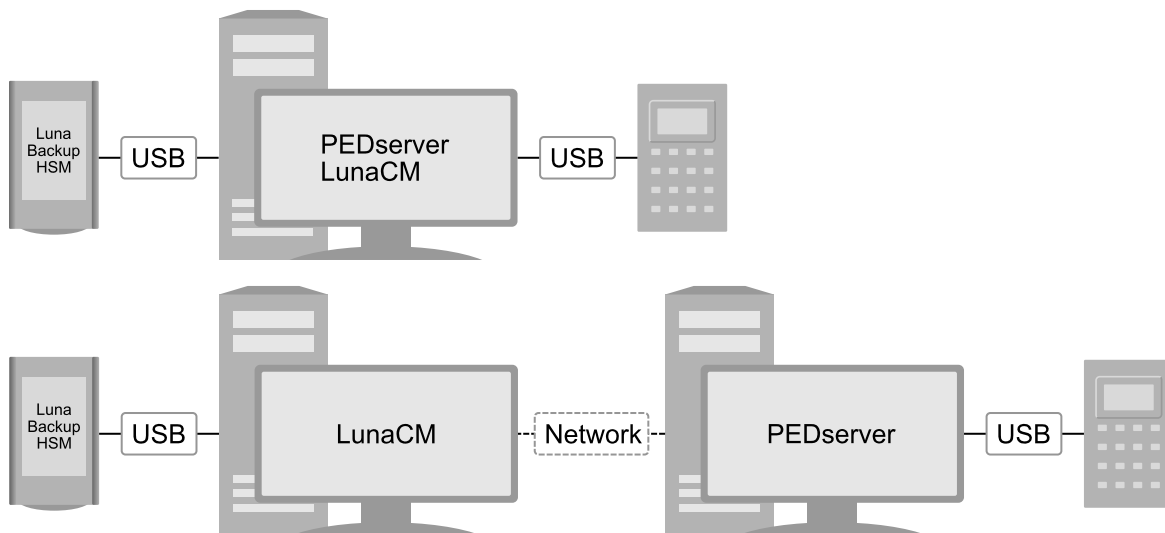
CAUTION! Always make copies of your orange iKeys, or declare MofN as one-of-several, and store at least one safely. For the Luna Backup HSM 7 v1, *the orange iKey is as important as the HSM SO blue key or the Domain red key.*

The orange iKey is required for all Luna Backup HSM 7 v1 operations. If this key is lost, your backups will become irretrievable. Thales recommends keeping multiple backups of all iKeys stored in a secure location.

- > N number of blue (HSM SO) iKeys, as defined by the M of N scheme you choose for the HSM SO role, plus the number required to create duplicate iKeys as necessary.
- > Blank or reused red (Domain) iKey(s)
- > [Luna Backup HSM 7 Firmware 7.7.1 and newer only] Set the value of **-pedwritedelay** to **2000** to avoid experiencing frequent CKR_CALLBACK_ERRORS, which will prevent you from completing the procedure below.

To initialize a Luna Backup HSM 7 for multifactor quorum authentication

1. Configure your Luna HSM Client workstation using one of the following configurations:



- a. Install the required client software on the Luna HSM Client workstation. See ["Client Software Required to Perform Backup and Restore Operations"](#) on page 108 for details.

NOTE If you are installing Luna HSM Client on Windows, the driver may not be installed unless the Luna device is connected to the computer first; refer to [Backup/USB/PCIe Drivers Not Installed on Windows 10 or Windows Server 2022 Unless Device is Connected.](#)

- b. Connect the backup HSM directly to the Luna HSM Client workstation using the included USB cable.

NOTE On most Luna HSM Client computers, the USB port provides adequate power, and connecting the provided power supply is not recommended. Wait and confirm that the HSM boots properly. If the HSM fails to boot up:

1. Disconnect the HSM from the USB port.
2. Connect the HSM to power using the provided power supply. Wait for it to boot completely.
3. Reconnect the HSM to the USB port on the client.

If the HSM is connected to the USB port before the power supply, you may encounter an issue where the HSM occasionally loses contact with the client, and must be power cycled.

- c. Connect the Luna PED to the Luna HSM Client workstation used to host the remote PED, using the PED USB cable.

NOTE You connect to the remote PED using the IP address of the workstation used to host the PED. This can be the same workstation that hosts the user and backup partition slots, or a different workstation. The workstation used to host the PED must be running pedServer.

2. Start the **pedserver** service on the workstation used to host the remote PED:

Windows	C:\Program Files\Safenet\LunaClient> pedserver -mode start
Linux	/usr/safenet/lunaclient> pedserver -mode start

3. Launch LunaCM on the workstation that hosts the user and backup partition slots.

4. Select the slot assigned to the backup HSM Admin partition.

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

5. If necessary, recover the HSM from Secure Transport Mode. See [Secure Transport Mode](#) for more information:

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

NOTE Recovering a Luna Backup HSM 7 from secure transport mode may take up to three minutes.

6. Connect to the Luna HSM Client workstation that hosts the PED. If defaults are not set using lunacm:> **ped set**, specify an IP address (and port if required; 1503 is default).

lunacm:> **ped connect -ip <ip_address> -pwd**

LunaCM generates and displays a one-time password that is used to set up a secure channel between the backup HSM and the PED, allowing you to securely initialize the orange (Remote PED Vector) iKey. Enter the displayed password on the PED when prompted to complete setup of the secure channel.

7. Create an orange (Remote PED vector) iKey for the backup HSM. The PED vector key is required for subsequent multifactor quorum-authenticated sessions to the HSM. Ensure that you label any new iKeys that you create during this process.

lunacm:> **ped vector init**

CAUTION! The orange iKey is required for all Luna Backup HSM 7 v1 operations. If this key is lost, your backups will become irretrievable. Thales recommends keeping multiple backups of all iKeys stored in a secure location.

8. Tear down the one-time, password-protected secure channel between the backup HSM and the PED you used to create the orange (Remote PED vector) iKey.

lunacm:> **ped disconnect**

You are prompted to enter the one-time password that was generated when you performed **ped connect**. Enter the password and press Enter to proceed.

9. Set up a new secure channel between the backup HSM and the PED. If defaults are not set using lunacm:> **ped set**, specify an IP address (and port if required; 1503 is default). You are prompted to insert the orange iKey you created in step 7.

lunacm:> **ped connect**

10. Initialize the selected backup HSM in multifactor quorum-authenticated mode. You are prompted by the PED for the red Domain iKey(s) and blue HSM SO iKey(s). Respond to the PED prompts and insert and set the PINs on the required keys when requested. Ensure that you label any new iKeys that you create during this process.

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

11. Use the **Duplicate** function on the PED to create and label duplicates of the new iKeys, as required.

12. Disconnect the PED when done.

lunacm:> **ped disconnect**

Configuring the Luna Backup HSM 7 for FIPS Compliance

Luna Backup HSM Firmware 7.7.1 and newer uses the same updated cloning protocol as Luna HSM Firmware 7.7.0 and newer. For the Luna Backup HSM 7 to be FIPS-compliant, it must restrict restore operations to application partitions that use the new protocol. This restriction is applied by setting **HSM policy 55: Enable Restricted Restore** to **1** on the backup HSM. The Luna Backup HSM 7 must be initialized and connected to a Luna HSM Client computer to set this policy.

When this policy is enabled on the Luna Backup HSM 7, objects that have been backed up from partitions using firmware older than Luna HSM Firmware 7.7.0 can be restored to Luna HSM Firmware 7.7.0 or newer (V0 or V1) partitions only.

CAUTION! FIPS compliance requires that objects are never cloned or restored to an HSM using less secure firmware, and this includes restoring from Luna Backup HSM 7 firmware. If you have backups already stored on the Luna Backup HSM 7 that were taken from pre-7.7.0 partitions, turning this policy ON will prevent you from restoring them to the same source partition. You must update the HSM containing the source partition to Luna HSM Firmware 7.7.0 or newer before restoring from backup.

NOTE HSM policy 12: [Allow non-FIPS algorithms](#), which is used to set FIPS-compliant mode on other Luna HSMs, does not apply to the Luna Backup HSM 7. Attempts to change this policy will fail with the error `CKR_CANCEL`.

To configure the Luna Backup HSM 7 for FIPS compliance

1. On the Luna HSM Client computer, run LunaCM.
2. Set the active slot to the Luna Backup HSM 7.
lunacm:> **slot set -slot** <slot_id>
3. Log in as Backup HSM SO.
lunacm:> **role login -name so**
4. Set **HSM policy 55:** [Enable Restricted Restore](#) to 1.
lunacm:> **hsm changehsmpolicy -policy 55 -value 1**
5. [Optional] Check that the Luna Backup HSM 7 is now in FIPS approved operation mode.
lunacm:> **hsm showinfo**

*** The HSM is in FIPS 140-2 approved operation mode. ***

Backing Up a Multifactor Quorum-Authenticated Partition

Backups are created and stored as partitions within the Admin partition on the Luna Backup HSM 7. A new backup partition is created on initial backup. For subsequent backups, you can choose to replace the contents of the existing backup partition with the current source partition objects, or add new objects in the source partition to the existing backup partition. Like all cloning operations, the source and target backup partitions must be initialized with the same domain.

Prerequisites

- > You have the required credentials:

If you are creating a new backup partition:

- The Remote PED Vector (orange) iKey(s) for the Backup HSM
- New or reused Partition SO (blue) iKey(s) to initialize the backup partition
- New or reused Crypto Officer (black) iKey(s) to initialize the CO role on the backup partition
- The Domain (red) iKey(s) for the source partition, to initialize the domain on the backup

If you are backing up to an existing backup partition whose domain matches the source partition:

- The Remote PED Vector (orange) iKey(s) for the Backup HSM
- The existing Partition SO (blue) iKey(s) for the backup partition, to log in
- The existing Crypto Officer (black) iKey(s) for the backup partition

- > The following policies are set:

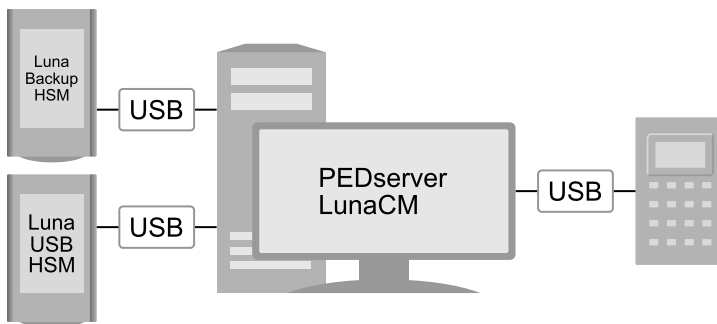
- **HSM policy 16:** [Allow network replication](#) must be set to **1 (ON)** on the HSM that hosts the user partition.

- [V0 partitions] **Partition policy 0: "Allow private key cloning" on page 50** is set to **1 (ON)** on the user partition.
 - [V0 partitions] **Partition policy 4: "Allow secret key cloning" on page 51** is set to **1 (ON)** on the user partition.
- > [Luna Backup HSM 7 Firmware 7.7.1 and newer only] Set the value of **-pedwritedelay** to **2000** to avoid experiencing frequent CKR_CALLBACK_ERRORS, which will prevent you from completing the procedure below.

NOTE HSS (PQC) private keys cannot be cloned, due to inherent restrictions of that key type. This means that backup and HA synchronization fail if HSS private keys are encountered in your application partition.

To back up a multifactor quorum-authenticated partition

1. Configure your Luna HSM Client workstation using one of the following configurations:



- a. Install the required client software on the Luna HSM Client workstation. See ["Client Software Required to Perform Backup and Restore Operations" on page 108](#) for details.

NOTE If you are installing Luna HSM Client on Windows, the driver may not be installed unless the Luna device is connected to the computer first; refer to [Backup/USB/PCIe Drivers Not Installed on Windows 10 or Windows Server 2022 Unless Device is Connected](#).

- b. Connect the backup HSM directly to the Luna HSM Client workstation using the included USB cable.

NOTE On most Luna HSM Client computers, the USB port provides adequate power, and connecting the provided power supply is not recommended. Wait and confirm that the HSM boots properly. If the HSM fails to boot up:

1. Disconnect the HSM from the USB port.
2. Connect the HSM to power using the provided power supply. Wait for it to boot completely.
3. Reconnect the HSM to the USB port on the client.

If the HSM is connected to the USB port before the power supply, you may encounter an issue where the HSM occasionally loses contact with the client, and must be power cycled.

- c. Connect the PED to the Luna HSM Client workstation used to host the remote PED, using the PED USB cable.

NOTE You connect to the remote PED using the IP address of the workstation used to host the PED. This can be the same workstation that hosts the user and backup partition slots, or a different workstation. The workstation used to host the PED must be running PEDserver.

2. Start the **pedserver** service on the workstation used to host the remote PED:

Windows	C:\Program Files\Safenet\LunaClient> pedserver -mode start
Linux	/usr/safenet/lunaclient> pedserver -mode start

3. Launch LunaCM on the workstation that hosts the Luna USB HSM 7 partition slots.

4. Identify the slot assignments for:

- The Luna USB HSM 7 partition you want to backup.
- The Luna Backup HSM 7 admin partition (where all backups are stored).

lunacm:> **slot list**

If you cannot see both slots, check your connections or configure your client as required.

5. Select the Luna USB HSM 7 partition:

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

6. Log in to the partition as Crypto Officer (CO):

- If the partition is activated, use the following command and present the black Crypto Officer iKey(s) to the Luna USB HSM 7 as directed:

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

- If the partition is not activated, log in to the selected Luna USB HSM 7 partition as the Crypto Officer (CO):

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

Respond to the prompts on the Luna USB HSM 7 touchscreen to provide the black CO key(s) and PIN.

7. Select the backup HSM Admin partition:

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

8. Connect to the Luna HSM Client workstation that hosts the PED. If defaults are not set using lunacm:> **ped set**, specify an IP address (and port if required; 1503 is default):

lunacm:> **ped connect [-ip <pedserver_host_ip>]**

9. Select the Luna USB HSM 7 partition:

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

10. Initiate the backup:

lunacm:> **partition archive backup -slot <backup_HSM_admin_slot> [-partition <target_backup_label>] [-append] [-replace] [-smkonly]**

If you omit the **-partition** option when creating a new backup, the partition is assigned a default name (<source_partition_name>_<YYYYMMDD>) based on the source HSM's internally-set time and date.

If you are backing up a V1 partition, include **-smkonly** to back up the SMK only. By default, the SMK and any encrypted cryptographic material on the partition are backed up.

The backup begins once you have completed the authentication process. Objects are backed up one at a time. For existing backups, you can use the following options to define how individual objects are backed up:

-append	Add only new objects to an existing backup.
-replace	Delete the existing objects in a target backup partition and replace them with the contents of the source user partition. This is the default.
-append -replace	Add new objects and replace existing objects that have the same OUID but a different fingerprint (such as would occur if any of the object attributes were changed since the previous backup).

NOTE If the backup operation is interrupted (if the Backup HSM is unplugged, or if you fail to respond to PED prompts, for example), the Backup HSM's full available space can become occupied with a single backup partition. If this occurs, delete the backup partition with `lunacm:> partition archive delete` before reattempting the backup operation.

11. You are prompted for the following credentials in the following order: Respond to the prompts on the Luna USB HSM 7 touchscreen and Luna PED to insert the following iKeys:

If you are creating a new backup partition:

- i. The blue HSM SO iKey(s) for the backup HSM.
- ii. You are prompted to initialize the backup Partition SO role by creating a new blue iKey or reusing an existing key. After you initialize the role, you are prompted to insert the key again to log in as Partition SO.
- iii. The red Domain iKey(s). This must be the same iKey(s) used for the Luna USB HSM 7 partition, otherwise the backup will fail.
- iv. The blue Partition SO iKey(s) for the backup partition, to log in again.
- v. You are prompted to initialize the Crypto Officer role for the backup by creating a new black iKey or reusing an existing key. After you initialize the role, you are prompted to insert the key again to log in as Crypto Officer.

If you are backing up to an existing backup partition whose domain matches the source partition:

- i. The blue HSM SO iKey(s) for the backup HSM.
- ii. The blue Partition SO iKey(s) for the backup.
- iii. The black Crypto Officer iKey(s) for the backup.

12. Disconnect the PED from the Luna Backup HSM 7:

`lunacm:> ped disconnect`

13. If this is the first backup to the backup partition, use the **Duplicate** function on the PED to create and label a set of backup keys for the new backup partition PO (blue) and CO (black) iKeys.

Restoring To a Multifactor Quorum-Authenticated Partition

You can restore the objects from a multifactor quorum-authenticated backup partition to the same partition that was originally backed up, or to another partition that has been initialized with the same domain (red iKey).

Prerequisites

- > The target partition must be initialized using the same domain (red iKey) as the backup partition, the Crypto Officer role must be initialized and the CO role credential changed from its initial value.
- > You have the required credentials:
 - The Remote PED Vector (orange) iKey(s) for the backup HSM
 - The Crypto Officer challenge secret for the target partition
 - The Crypto Officer (black) iKey(s) for the backup partition

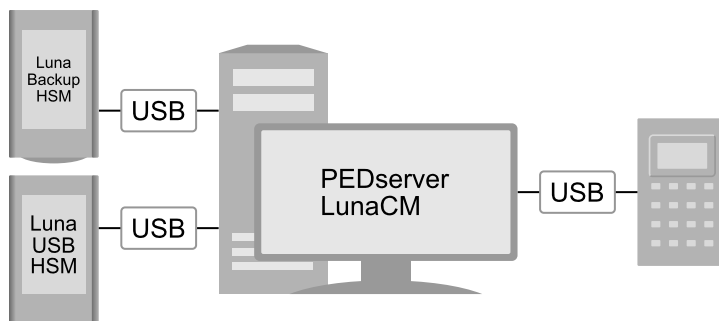
TIP If the target partition is activated, only the Crypto Officer's challenge secret is required. To simplify the backup process and minimize interactions with the PED, it is recommended that you activate the CO role on the user partitions you want to restore from backup. See [Activation on Multifactor Quorum-Authenticated Partitions](#) for more information.

If the target partition is not activated, you also need:

- The Remote PED Vector (orange) iKey(s) for the target HSM
 - The Crypto Officer (black) iKey(s) for the target partition
- > The following policies are set:
- **HSM policy 16: Allow network replication** must be set to **1 (ON)** on the HSM that hosts the user partition you want to restore to.
 - [V0 partitions] **Partition policy 0: "Allow private key cloning"** on page 50 is set to **1 (ON)** on the user partition you want to restore to.
 - [V0 partitions] **Partition policy 4: "Allow secret key cloning"** on page 51 is set to **1 (ON)** on the user partition you want to restore to.
- > [Luna Backup HSM 7 Firmware 7.7.1 and newer only] Set the value of **-pedwritedelay** to **2000** to avoid experiencing frequent CKR_CALLBACK_ERRORS, which will prevent you from completing the procedure below.

To restore a multifactor quorum-authenticated partition

1. Configure your Luna HSM Client workstation using one of the following configurations:



- a. Install the required client software on the Luna HSM Client workstation. See [Luna HSM Client Software Installation](#) for details.

NOTE If you are installing Luna HSM Client on Windows, the driver may not be installed unless the Luna device is connected to the computer first; refer to [Backup/USB/PCIe Drivers Not Installed on Windows 10 or Windows Server 2022 Unless Device is Connected](#).

- b. Connect the backup HSM directly to the Luna HSM Client workstation using the included USB cable.

NOTE On most Luna HSM Client computers, the USB port provides adequate power, and connecting the provided power supply is not recommended. Wait and confirm that the HSM boots properly. If the HSM fails to boot up:

1. Disconnect the HSM from the USB port.
2. Connect the HSM to power using the provided power supply. Wait for it to boot completely.
3. Reconnect the HSM to the USB port on the client.

If the HSM is connected to the USB port before the power supply, you may encounter an issue where the HSM occasionally loses contact with the client, and must be power cycled.

- c. Connect the PED to the Luna HSM Client workstation used to host the remote PED, using the PED USB cable.

NOTE You connect to the remote PED using the IP address of the workstation used to host the PED. This can be the same workstation that hosts the user and backup partition slots, or a different workstation. The workstation used to host the PED must be running **pedserver**.

2. Start the **pedserver** service on the workstation used to host the remote PED:

Windows	C:\Program Files\Safenet\LunaClient> pedserver -mode start
Linux	/usr/safenet/lunaclient> pedserver -mode start

3. Launch LunaCM on the workstation that hosts the Luna USB HSM 7 and backup partition slots.

4. Identify the slot assignments for:

- the Luna USB HSM 7 partition you want to restore to.
- the backup HSM admin partition (where all backups are stored).

lunacm:> **slot list**

If you cannot see both slots, check your connections or configure your client as required.

5. Select the Luna USB HSM 7 partition you want to restore from backup:

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

6. Log in to the partition as Crypto Officer (CO):

- If the partition is activated, use the following command and enter the Crypto Officer challenge secret:

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

- If the partition is not activated, log in to the selected Luna USB HSM 7 partition as the Crypto Officer (CO):

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

Respond to the prompts on the Luna USB HSM 7 touchscreen to provide the black CO key(s) and PIN.

7. Connect the PED to the backup HSM. If defaults are not set using `lunacm:> ped set`, specify an IP address (and port if required; 1503 is default):

```
lunacm:> ped connect [-ip <pedserver_host_ip>]
```

8. List the available backups on the Backup HSM by specifying the Backup HSM's slot number. You will require the backup partition label to perform the restore operation.

```
lunacm:> partition archive list-slot <backup_HSM_admin_slot>
```

9. Initiate the restore operation. Respond to the prompts on the PED to insert the required PED keys.

```
lunacm:> partition archive restore -slot <backup_HSM_admin_slot> -partition <backup_partition_label> [-smkonly]
```

CAUTION! The **-replace** option is deprecated and has been removed in [Luna HSM Client 10.7.0](#) and newer. If you wish to restore an earlier version of an object, Thales recommends deleting the object(s) manually before restoring the partition from backup.

Ensure that the target partition can receive objects from the backup HSM before deleting objects or using `partition archive restore` with the **-replace** option; the cloning protocol may prevent objects from being restored, even if LunaCM states that *x objects will be restored*. This may occur if **HSM policy 55: Enable Restricted Restore** was enabled on the Luna Backup HSM 7 since the original backup was taken. If your partition is on an HSM with firmware older than Luna HSM Firmware 7.7.0, you must update to 7.7.0 or newer to restore objects from this backup.

NOTE If you are restoring a V1 backup to a V1 partition, include **-smkonly** to restore the SMK only (see ["V0 and V1 Partitions" on page 67](#) for more information). By default, the SMK and any cryptographic material on the backup are restored.

The restore operation begins once you have completed the authentication process. Objects are restored one at a time.

Luna Backup HSM 7 Using Password Authentication

In this configuration, you connect the Luna Backup HSM 7 to a USB port on the Luna HSM Client, and enter passwords in LunaCM. This configuration allows you to perform backup/restore operations for all application partitions that can be accessed by the client. You can restore a partition backup to the original source partition or to another existing Luna application partition that shares the same cloning domain. To use this method, you require:

- > ["Luna Backup HSM 7" on page 113 v1 or v2](#)
- > [Luna HSM Client 10.1.0](#) or newer

This section provides instructions for the following procedures:

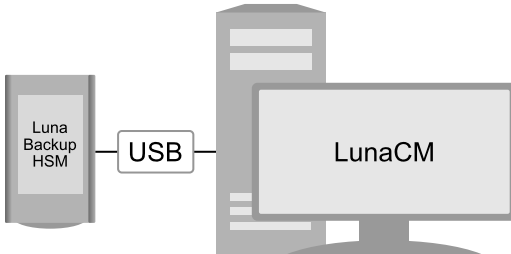
- > ["Initializing the Luna Backup HSM 7 for Password Authentication" on the next page](#)
- > ["Configuring the Luna Backup HSM 7 for FIPS Compliance" on page 141](#)
- > ["Backing Up a Password-Authenticated Partition" on page 142](#)
- > ["Restoring to a Password-Authenticated Partition" on page 144](#)

Initializing the Luna Backup HSM 7 for Password Authentication

You must initialize the Luna Backup HSM 7 prior to first use. You can initialize the backup HSM by connecting it to a Luna HSM Client and using LunaCM commands to perform the initialization.

To initialize a Luna Backup HSM 7 for password authentication

1. Configure your Luna HSM Client workstation as illustrated below:



- a. Install the required client software on the Luna HSM Client workstation. See ["Client Software Required to Perform Backup and Restore Operations"](#) on page 108 for details.

NOTE If you are installing Luna HSM Client on Windows, the driver may not be installed unless the Luna device is connected to the computer first; refer to [Backup/USB/PCIe Drivers Not Installed on Windows 10 or Windows Server 2022 Unless Device is Connected](#).

- b. Connect the backup HSM directly to the Luna HSM Client workstation using the included USB cable.

NOTE On most Luna HSM Client computers, the USB port provides adequate power, and connecting the provided power supply is not recommended. Wait and confirm that the HSM boots properly. If the HSM fails to boot up:

1. Disconnect the HSM from the USB port.
2. Connect the HSM to power using the provided power supply. Wait for it to boot completely.
3. Reconnect the HSM to the USB port on the client.

If the HSM is connected to the USB port before the power supply, you may encounter an issue where the HSM occasionally loses contact with the client, and must be power cycled.

2. Launch LunaCM on the workstation that hosts the user and backup partition slots.
3. Select the slot assigned to the backup HSM Admin partition:

```
lunacm:> slot set -slot <slot_id>
```
4. If necessary, recover the HSM from Secure Transport Mode. See [Secure Transport Mode](#) for more information:

```
lunacm:> stm recover
```

NOTE Recovering a Luna Backup HSM 7 from secure transport mode may take up to three minutes.

5. Initialize the selected backup HSM in password-authenticated mode.

```
lunacm:> hsm init -ipwd -label <label>
```

You are prompted for the new HSM SO password and the HSM domain string (existing or new):

Configuring the Luna Backup HSM 7 for FIPS Compliance

[Luna Backup HSM Firmware 7.7.1](#) and newer uses the same updated cloning protocol as Luna HSM Firmware 7.7.0 and newer. For the Luna Backup HSM 7 to be FIPS-compliant, it must restrict restore operations to application partitions that use the new protocol. This restriction is applied by setting **HSM policy 55: [Enable Restricted Restore](#)** to **1** on the backup HSM. The Luna Backup HSM 7 must be initialized and connected to a Luna HSM Client computer to set this policy.

When this policy is enabled on the Luna Backup HSM 7, objects that have been backed up from partitions using firmware older than Luna HSM Firmware 7.7.0 can be restored to Luna HSM Firmware 7.7.0 or newer (V0 or V1) partitions only.

CAUTION! FIPS compliance requires that objects are never cloned or restored to an HSM using less secure firmware, and this includes restoring from Luna Backup HSM 7 firmware.

If you have backups already stored on the Luna Backup HSM 7 that were taken from pre-7.7.0 partitions, turning this policy ON will prevent you from restoring them to the same source partition. You must update the HSM containing the source partition to Luna HSM Firmware 7.7.0 or newer before restoring from backup.

NOTE **HSM policy 12: [Allow non-FIPS algorithms](#)**, which is used to set FIPS-compliant mode on other Luna HSMs, does not apply to the Luna Backup HSM 7. Attempts to change this policy will fail with the error `CKR_CANCEL`.

To configure the Luna Backup HSM 7 for FIPS compliance

1. On the Luna HSM Client computer, run LunaCM.

2. Set the active slot to the Luna Backup HSM 7.

```
lunacm:> slot set -slot <slot_id>
```

3. Log in as Backup HSM SO.

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

4. Set **HSM policy 55: [Enable Restricted Restore](#)** to **1**.

```
lunacm:> hsm changehsm policy -policy 55 -value 1
```

5. [Optional] Check that the Luna Backup HSM 7 is now in FIPS approved operation mode.

```
lunacm:> hsm showinfo
```

```
*** The HSM is in FIPS 140-2 approved operation mode. ***
```

NOTE HSS (PQC) private keys cannot be cloned, due to inherent restrictions of that key type. This means that backup and HA synchronization fail if HSS private keys are encountered in your application partition.

Backing Up a Password-Authenticated Partition

Backups are created and stored as partitions within the Admin partition on the Luna Backup HSM 7. A new backup partition is created on initial backup. For subsequent backups, you can choose to replace the contents of the existing backup partition with the current source partition objects, or add new objects in the source partition to the existing backup partition. Like all cloning operations, the source and target backup partitions must be initialized with the same domain.

Prerequisites

Before you begin, ensure that you have satisfied the following prerequisites:

- > You have the required credentials:

If you are creating a new backup:

- The Crypto Officer password and domain string for the source partition
- The HSM SO password for the backup HSM

If you are adding to an existing backup initialized with the same domain string as the source partition:

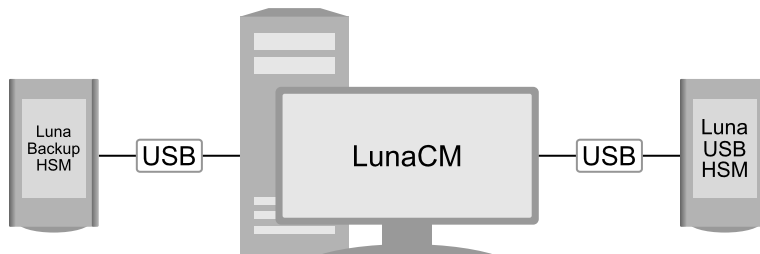
- The Crypto Officer password for the source partition
- The Crypto Officer password for the existing backup
- The HSM SO password for the backup HSM

- > The following policies are set:

- **HSM policy 16:** [Allow network replication](#) must be set to **1 (ON)** on the HSM that hosts the user partition.
- [V0 partitions] **Partition policy 0:** ["Allow private key cloning" on page 50](#) is set to **1 (ON)** on the user partition.
- [V0 partitions] **Partition policy 4:** ["Allow secret key cloning" on page 51](#) is set to **1 (ON)** on the user partition.

To back up a password-authenticated partition

1. Configure your Luna HSM Client workstation as illustrated below:



- a. If you have not already done so, install the required client software on the Luna HSM Client workstation and start LunaCM. See ["Client Software Required to Perform Backup and Restore Operations" on page 108](#) for more information.

NOTE If you are installing Luna HSM Client on Windows, the driver may not be installed unless the Luna device is connected to the computer first; refer to [Backup/USB/PCIe Drivers Not Installed on Windows 10 or Windows Server 2022 Unless Device is Connected](#).

- b. Connect the backup HSM directly to the Luna HSM Client workstation using the included USB cable.

NOTE On most Luna HSM Client computers, the USB port provides adequate power, and connecting the provided power supply is not recommended. Wait and confirm that the HSM boots properly. If the HSM fails to boot up:

1. Disconnect the HSM from the USB port.
2. Connect the HSM to power using the provided power supply. Wait for it to boot completely.
3. Reconnect the HSM to the USB port on the client.

If the HSM is connected to the USB port before the power supply, you may encounter an issue where the HSM occasionally loses contact with the client, and must be power cycled.

2. Launch LunaCM on the workstation that hosts the user and backup partition slots.
3. Identify the slots assigned to:
 - The Luna USB HSM 7 partition slot (to be backed up).
 - The Luna Backup HSM 7 admin slot (where all backups are stored).

lunacm:> **slot list**

If you cannot see both slots, check your connections or configure your client as required.

4. Select the Luna USB HSM 7 partition:
5. Log in to the Luna USB HSM 7 partition as the Crypto Officer (CO):

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

6. Initiate backup of the Luna USB HSM 7 partition to the backup partition:

lunacm:> **partition archive backup -slot <backup_hsm_admin_partition_slot_id> [-partition <target_backup_partition_label>] [-append] [-replace] [-smkonly]**

If you omit the **-partition** option when creating a new backup, the backup is assigned a default name (<source_partition_name>_<YYYYMMDD>) based on the source HSM's internally-set time and date.

If you are backing up a V1 partition, include **-smkonly** to back up the SMK only. By default, the SMK and any encrypted cryptographic material on the partition are backed up.

The backup begins once you have completed the authentication process. Objects are backed up one at a time. For existing backups, you can use the following options to define how individual objects are backed up:

-replace	Delete the target backup partition and replace it with a new backup with the same label, with the contents of the source partition. This is the default.
-append	Add only new objects to the existing backup.
-append -replace	Add new objects and replace existing objects that have the same OUID but a different fingerprint (such as would occur if any of the object attributes were changed since the previous backup).

NOTE If the backup operation is interrupted (if the Backup HSM is unplugged, for example), the Backup HSM's full available space can become occupied with a single backup partition. If this occurs, delete the backup partition with `lunacm:> partition archive delete` before reattempting the backup operation.

7. You are prompted for the following passwords, unless you specified them in the **partition archive backup** options:
 - a. The HSM SO password for the backup HSM. This is required to create or access the backup partition in the Admin slot.
 - b. The Crypto Officer password for the target partition on the backup HSM (if you specified an existing backup). If you are creating a new backup, you must set its CO password now.
 - c. [If creating a new backup] The domain string for the backup partition. The domain must match the domain configured on the source partition.

Restoring to a Password-Authenticated Partition

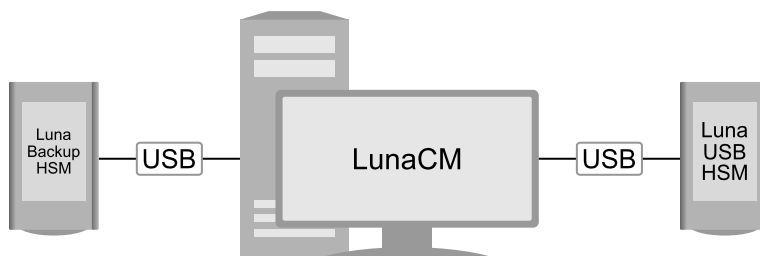
You can restore the objects from a password-authenticated backup to the same partition that was originally backed up, or to another partition that has been initialized with the same domain string.

Prerequisites

- > The backup and the partition you want to restore to must be members of the same domain.
- > You need the following credentials:
 - The Crypto Officer password for the target partition.
 - The Crypto Officer password for the backup
- > The following policies are set:
 - **HSM policy 16: Allow network replication** must be set to **1 (ON)** on the HSM that hosts the user partition.
 - [V0 partitions] **Partition policy 0: "Allow private key cloning"** on page 50 is set to **1 (ON)** on the partition you want to restore to.
 - [V0 partitions] **Partition policy 4: "Allow secret key cloning"** on page 51 is set to **1 (ON)** on the partition you want to restore to.

To restore a password-authenticated partition

1. Configure your Luna HSM Client workstation as illustrated below:



- a. Install the required client software on the Luna HSM Client workstation and start LunaCM. See ["Client Software Required to Perform Backup and Restore Operations"](#) on page 108 for more information.

NOTE If you are installing Luna HSM Client on Windows, the driver may not be installed unless the Luna device is connected to the computer first; refer to [Backup/USB/PCIe Drivers Not Installed on Windows 10 or Windows Server 2022 Unless Device is Connected](#).

- b. Connect the backup HSM directly to the Luna HSM Client workstation using the included USB cable.

NOTE On most Luna HSM Client computers, the USB port provides adequate power, and connecting the provided power supply is not recommended. Wait and confirm that the HSM boots properly. If the HSM fails to boot up:

1. Disconnect the HSM from the USB port.
2. Connect the HSM to power using the provided power supply. Wait for it to boot completely.
3. Reconnect the HSM to the USB port on the client.

If the HSM is connected to the USB port before the power supply, you may encounter an issue where the HSM occasionally loses contact with the client, and must be power cycled.

2. Identify the slots assigned to:

- The Luna USB HSM 7 partition slot (to be restored).
- The Luna Backup HSM 7 admin slot (where all backups are stored).

lunacm:> **slot list**

If you cannot see both slots, check your connections or configure your client as required.

3. Select the Luna USB HSM 7 partition you want to restore to:

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

4. Log in to the partition as Crypto Officer (CO):

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

5. List the available backups on the Backup HSM by specifying the Backup HSM's slot number. You will require the backup partition label to perform the restore operation.

lunacm:> **partition archive list -slot <backup_HSM_slot>**

6. Initiate the restore operation. Respond to the prompts to provide the required passwords, as detailed in the summary above.

lunacm:> **partition archive restore -slot <backup_HSM_admin_slot> -partition <backup_partition_label> [-smkonly]**

You are prompted for the Crypto Officer password for the backup. The restore operation begins once you have completed the authentication process. Objects are restored one at a time.

CAUTION! The **-replace** option is deprecated and has been removed in [Luna HSM Client 10.7.0](#) and newer. If you wish to restore an earlier version of an object, Thales recommends deleting the object(s) manually before restoring the partition from backup.

Ensure that the target partition can receive objects from the backup HSM before deleting objects or using [partition archive restore](#) with the **-replace** option; the cloning protocol may prevent objects from being restored, even if LunaCM states that `x objects will be restored`. This may occur if **HSM policy 55: Enable Restricted Restore** was enabled on the Luna Backup HSM 7 since the original backup was taken. If your partition is on an HSM with firmware older than Luna HSM Firmware 7.7.0, you must update to 7.7.0 or newer to restore objects from this backup.

NOTE If you are restoring a V1 backup to a V1 partition, include **-smkonly** to restore the SMK only (see ["V0 and V1 Partitions" on page 67](#) for more information). By default, the SMK and any encrypted cryptographic material on the backup are restored.

Luna Backup HSM G5

The Luna Backup HSM G5 allows you to safeguard your important cryptographic objects by making secure backups, and restoring those backups to an application partition.



For setup, management and backup/restore procedures, refer to the following sections:

- > ["Luna Backup HSM G5 Hardware Installation" on the next page](#)
- > ["Backup/Restore Using Luna Backup HSM G5" on page 159](#)
- > ["Managing the Luna Backup HSM G5" on page 150](#)
- > ["Configuring a Remote Backup Server" on page 164](#)

The Luna Backup HSM G5 can be configured to back up either password- or multifactor quorum-authenticated partitions. You must specify the authentication method when you initialize the Luna Backup HSM G5. Once initialized, the backup HSM can only be used with partitions sharing the same authentication type. The only way to change the authentication method is to restore the backup HSM to factory condition and re-initialize it.

The storage capacity and maximum number of backup partitions allowed on the backup HSM is determined by the firmware. You can check the capacity using `lunacm:> hsm showinfo`. To update the backup HSM firmware to a version that allows more backups, see ["Updating the Luna Backup HSM G5 Firmware" on page 152](#).

NOTE Objects stored on a Backup HSM may be smaller than their originals. For example, symmetric keys are 8 bytes smaller when stored on a Backup HSM. This size difference has no effect on backup and restore operations.

Considerations when Performing Cloning and Backup-Restore Operations, when SKS is Involved

If you invoked scalable key storage (SKS) for your applications to create and store large numbers of keys, then the partition is V1. If you perform cloning operations (including HA) or Backup and Restore, see ["Backup/Restore and SKS" on page 74](#).

Luna Backup HSM G5 Hardware Installation



This section contains instructions for installing your Luna Backup HSM G5.

- > ["Luna Backup HSM G5 Required Items" below](#)
- > ["Physical Features" on page 149](#)
- > ["Installing the Luna Backup HSM G5" on page 150](#)

Luna Backup HSM G5 Required Items

This section provides a list of the components you should have received with your Luna Backup HSM G5 order.

Qty	Item
1	<div>Luna Backup HSM G5</div> 

Qty	Item
1	<p>External Power Supply</p> <p>The Luna Backup HSM G5 now ships with an external power supply. Previously, these HSMs relied on an internal power supply, requiring the HSM to be periodically powered on to recharge internal capacitors. Failure to charge the capacitors could result in an inability to power on the HSM.</p> <p>With the introduction of external power supplies, periodically powering on the HSM is no longer required. A failed external power supply can be replaced and there is no need to return the HSM for repair (RMA).</p> <div> <p>NOTE External power supplies do contain capacitors which may be affected by extended periods of being unpowered, but they are more easily replaced in the event of failure.</p> </div>
1	<p>Power Supply Cord</p> <p>Your order should include one power supply cord for the Luna Backup HSM G5. The actual cord received depends on the country for which you ordered the Luna Backup HSM G5.</p> 
1	<p>USB cable (USB A to USB mini B)</p>  <p>Your order should include one USB A to 5-pin (Mini-B) cable.</p>

Optional Items

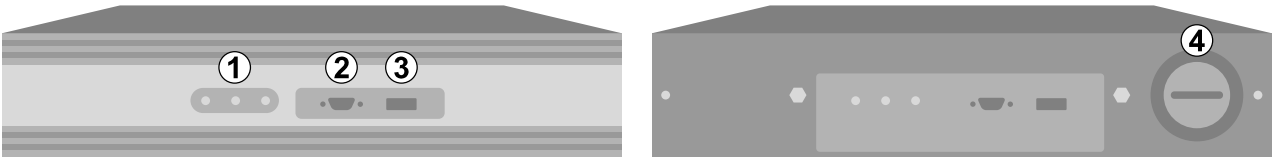
Your order may also include the following optional item

Luna Backup HSM G5 Rack-Mount Shelf

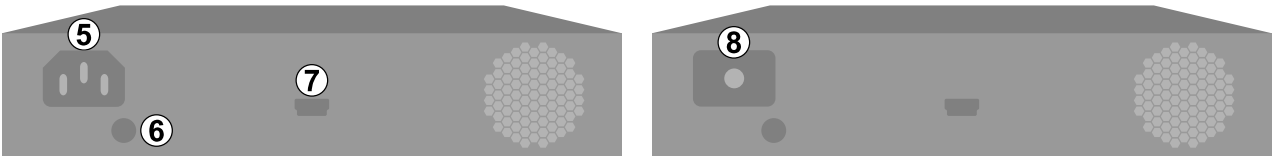
The Luna Backup HSM G5 rack-mount shelf (available by separate order) fits a standard 19-inch equipment rack, allowing you to install up to two Luna Backup HSM G5 units side-by-side in server-room racks. For office use, without rack mounting, Luna Backup HSM G5 units can be placed on a desktop and are stackable.

Physical Features

The front panel of the Luna Backup HSM G5 is illustrated below, with important features labeled. In the second image, the front bezel has been removed, exposing the battery enclosure.



The rear panel of the Luna Backup HSM G5 is illustrated below, with important features labeled. The first image depicts a backup HSM with an internal power supply. The second image depicts one that ships with an external power supply.



1	<p>Status LEDs. When illuminated, they indicate:</p> <ul style="list-style-type: none"> > Active: The backup HSM is performing a procedure. Do not disconnect or unplug the device when this light is illuminated. > Tamper: The backup HSM is in a tamper state. You must clear the tamper state before backing up or restoring partitions. > Error: HSM device driver error. Contact Thales Customer Support (see "Support Contacts" on page 12).
2	Serial port for attaching a local Luna PED using a 9-pin Micro-D to Micro-D cable.
3	USB port. Not applicable to backup/restore functions.
4	Battery enclosure. See "Installing or Replacing the Luna Backup HSM G5 Battery" on page 154 .
5	Power connector for a Luna Backup HSM with an internal power supply. See "Storage and Maintenance" on the next page for more information.
6	Index hole. Engages with the index post on a Luna Backup HSM rack shelf.
7	Mini-USB port for connecting the Luna Backup HSM G5 to a Luna HSM or client workstation. See "Installing the Luna Backup HSM G5" on the next page .
8	Power source connector for a Luna Backup HSM G5 with an external power supply (included).

Installing the Luna Backup HSM G5

You can connect the Luna Backup HSM to a Luna Network HSM, a Luna HSM Client workstation, or a host machine containing a Luna PCIe HSM. Refer to ["Planning Your Backup HSM Deployment" on page 108](#) for detailed descriptions of the configuration options.

To install the Luna Backup HSM G5

1. Connect the Luna Backup HSM G5 to power using the external power source or a standard power cable.
2. If you are connecting the Luna Backup HSM G5 to a client workstation or Luna PCIe HSM 7 host, ensure that you have installed the **Backup** option in the Luna HSM Client installer (see [Luna HSM Client Software Installation](#) for details).
3. [Local PED] If you plan to authenticate the Luna Backup HSM G5 with a local Luna PED, connect the PED using a 9-pin Micro-D to Micro-D cable (see ["Physical Features" on the previous page](#)).
4. Connect the Luna Backup HSM G5 using the included Mini-USB to USB cable. If you are connecting the Backup HSM to:
 - **Luna Network HSM 7:** Connect to one of the USB ports on the front or rear panel of the appliance.
 - **Luna HSM Client:** Connect to a USB port on the client workstation. Run LunaCM on the client workstation to confirm that the Luna Backup HSM G5 appears in a slot.
 - **Luna PCIe HSM 7 host:** Connect to a USB port on the host workstation. Run LunaCM on the host workstation to confirm that the Luna Backup HSM G5 appears in a slot.
5. If your Backup HSM was shipped in Secure Transport Mode, see ["Recovering From a Tamper Event or Secure Transport Mode" on page 158](#).

Managing the Luna Backup HSM G5

This section contains the following procedures for maintaining and using the Luna Backup HSM G5:

- > ["Storage and Maintenance" below](#)
- > ["Initializing the Luna Backup HSM G5 Remote PED Vector" on the next page](#)
- > ["Updating the Luna Backup HSM G5 Firmware" on page 152](#)
- > ["Resetting the Luna Backup HSM G5 to Factory Conditions" on page 153](#)
- > ["Installing or Replacing the Luna Backup HSM G5 Battery" on page 154](#)
- > ["About Luna Backup HSM G5 Secure Transport and Tamper Recovery" on page 156](#)
 - ["Creating a Secure Recovery Key" on page 157](#)
 - ["Setting Secure Transport Mode" on page 158](#)
 - ["Recovering From a Tamper Event or Secure Transport Mode" on page 158](#)
 - ["Disabling Secure Recovery" on page 159](#)

Storage and Maintenance

The Luna Backup HSM G5 can be safely stored, containing backups, when not in use. When stored properly, the hardware has a lifetime of 10+ years. Newer Luna Backup HSM G5s ship with an external power supply.

CAUTION! The internal power supply on older Luna Backup HSM G5s uses capacitors that may be affected if they are left unpowered for extended periods of time. If your Luna Backup HSM G5 has an internal power supply, power it on occasionally to recharge the capacitors. If the capacitors lose function, the Luna Backup HSM G5 will no longer receive power.

With the introduction of external power supplies, this is no longer a requirement. If the external power supply fails from being left unpowered, it can be easily replaced.

The Luna Backup HSM G5 Battery

The battery powers the NVRAM and Real-Time-Clock (RTC), and must be installed for use. The battery can be removed for storage, and this is generally good practice. Thales uses high-quality, industrial-grade batteries that are unlikely to leak and damage the HSM hardware, but an externally-stored battery will last longer. The battery must be stored in a clean, dry area (less than 30% Relative Humidity). Temperature should not exceed +30 °C. When properly stored, the battery has a shelf life of 10 years.

If the battery dies or is removed, and the main power is not connected, NVRAM and the RTC lose power. Battery removal triggers a tamper event. After replacing the battery, the HSM SO must clear the tamper event before operation can resume. The working copy of the Master Tamper Key (MTK) is lost (see ["About Luna Backup HSM G5 Secure Transport and Tamper Recovery" on page 156](#)). Backup objects are stored in non-volatile memory, so they are preserved and remain uncorrupted.

There is no low battery indicator, or other provision for checking the battery status. The voltage remains constant until the very end of battery life.

Your stored (backed-up) content is in long-term memory and is not affected by the state of the battery. A failure or removal of the battery does cause a tamper event, but this is intended as an alert to bring the condition to your attention for action, and does not affect the stored content. A situation where battery removal *could* affect your ability to recover archived data from the Luna Backup HSM G5 is where you have previously extracted a portion of the MTK onto an iKey (PED Key) and then have lost/destroyed/overwritten all copies of that key, leaving the MTK unrecoverable.

Initializing the Luna Backup HSM G5 Remote PED Vector

The Remote PED (via PEDserver) authenticates itself to the Luna Backup HSM G5 with a randomly-generated encrypted value stored on an orange iKey. The orange key proves to the HSM that the Remote PED is authorized to perform authentication. The HSM SO can create this key using LunaCM.

If the Luna Backup HSM G5 is already initialized, the HSM SO must log in to complete this procedure.

Prerequisites

- > Luna PED with firmware 2.7.1 or newer
- > USB mini-B to USB-A connector cable
- > Luna PED DC power supply (if included with your Luna PED)
- > Blank or reusable orange iKey (or multiple keys, if you plan to make extra copies or use an M of N security scheme).
- > Install the Luna Backup HSM G5 at the client and connect it to power (see ["Luna Backup HSM G5 Hardware Installation" on page 147](#)).

- > Connect the PED to the Luna Backup HSM G5 using a 9-pin Micro-D to Micro-D cable. Set the PED to **Local PED-SCP** mode.

To initialize the RPV and create the orange iKey

1. Launch LunaCM on the client workstation.
2. Set the active slot to the Luna Backup HSM G5.
lunacm:> **slot set -slot** <slotnum>
3. If the Luna Backup HSM G5 is initialized, log in as HSM SO. If not, continue to the next step.
lunacm:> **role login -name so**
4. Ensure that you have the orange iKey(s) ready. Initialize the RPV.
lunacm:> **ped vector init**
5. Attend to the Luna PED and respond to the on-screen prompts.

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

- If you have an orange iKey with an existing RPV that you wish to use for this HSM, press **Yes**.
- If you are creating a new RPV, press **No**.

```
SLOT
SETTING RPV...
Insert a
RPK / Remote
PED Key (ORANGE)
Press ENTER.
```

Continue following the prompts for PED PIN, M of N, and duplication options.

To set up a Remote PED server, see "[Configuring a Remote Backup Server](#)" on page 164.

Updating the Luna Backup HSM G5 Firmware

To update Luna Backup HSM G5 firmware, use LunaCM on a client computer that is connected to the Luna Backup HSM G5. You require:

- > Luna Backup HSM G5 firmware update file (<filename>.**fuf**)
- > the firmware update authentication code file(s) (<filename>.**txt**)

CAUTION! Use an uninterruptible power supply (UPS) to power your HSM. There is a small chance that a power failure during an update could leave your HSM in an unrecoverable condition.

NOTE To perform backup operations on Luna HSM Firmware 7.7.0 or newer (V0 or V1 partitions) you require at minimum:

- > [Luna Backup HSM 7 Firmware 7.7.1](#)
- > [Luna Backup HSM G5 Firmware 6.28.0](#)

You can use a Luna Backup HSM with older firmware to restore objects to a V0 or V1 partition, but this is supported for purposes of getting your objects from the older partitions onto the newer V0 or V1 partitions only. V0 and V1 partitions are considered more secure than partitions at earlier firmware versions - any attempt to restore from a higher-security status to lower-security status fails gracefully.

To update the Luna Backup HSM G5 firmware

1. Copy the firmware file (<filename>.fuf) and the authentication code file (<filename>.txt) to the Luna HSM Client root directory.
 - Windows: **C:\Program Files\SafeNet\LunaClient**
 - Linux: **/usr/safenet/lunaclient/bin**

NOTE On some Windows configurations, you might not have authority to copy or unzip files directly into **C:\Program Files\...** If this is the case, put the files in a known location that you can reference in a LunaCM command.

2. Launch LunaCM.
3. If more than one HSM is installed, set the active slot to the Admin partition of the HSM you wish to update.


```
lunacm:> slot set -slot <slot_number>
```
4. Log in as HSM SO. Depending on the currently-installed firmware version, use one of the following two commands:
 - lunacm:> **role login -name so**
 - lunacm:> **hsm login**
5. Apply the new firmware update by specifying the update file and the authentication code file. If the files are not located in the Luna HSM Client root directory, specify the filepaths.


```
lunacm:> hsm updatefw -fuf <filename>.fuf -authcode <filename>.txt
```

Resetting the Luna Backup HSM G5 to Factory Conditions

These instructions will allow you to restore your Luna Backup HSM G5 to its original factory conditions, erasing its contents. This could be necessary if you have old backups that you do not wish to keep, or if you want to re-initialize the Backup HSM to store backups using a different authentication method (password or multifactor quorum). If you have performed firmware updates, they are unaffected. Factory reset can be performed via LunaCM.

To reset the Luna Backup HSM G5 to factory conditions

1. Launch LunaCM on the Luna Backup HSM G5 workstation.
2. Set the active slot to the Luna Backup HSM G5.

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

3. Reset the Backup HSM.

```
lunacm:> hsm factoryreset
```

Installing or Replacing the Luna Backup HSM G5 Battery

The Luna Backup HSM G5 must have a functioning battery installed to preserve the NVRAM and RTC in case of primary power loss. You can purchase a replacement battery from any supplier who can match the following specifications:

- > 3.6 V Primary lithium-thionyl chloride (Li-SOCl₂)
- > Fast voltage recovery after long term storage and/or usage
- > Low self discharge rate
- > 10 years shelf life
- > Operating temperature range -55 °C to +85 °C
- > U.L. Component Recognition, MH 12193

Prerequisites

- > Removing the battery causes a tamper event. If you have created a Secure Recovery Vector (purple iKey) and enabled Secure Recovery, you will need this key to clear the tamper after replacing the battery.

To install or replace the Luna Backup HSM G5 battery

1. Remove the front bezel. It is held in place by two spring clips.



2. The battery compartment is spring-loaded and can be removed without much pressure. Use a coin or your fingers to press in the compartment cover and turn counter-clockwise to remove it.



3. If you are replacing the old battery, remove it from the battery compartment.



4. Insert the new battery, negative end first. The positive end should be visible.



5. Use the battery compartment cover to push the battery into the compartment, aligning the tabs on the cover with the compartment slots. Twist the cover clockwise to lock the compartment.



6. Replace the front bezel by aligning the clips with their posts and pushing it into place.
Removing the battery causes a tamper event on the Luna Backup HSM G5.
7. To clear the tamper, see ["Recovering From a Tamper Event or Secure Transport Mode" on page 158](#).

About Luna Backup HSM G5 Secure Transport and Tamper Recovery

The Luna Backup HSM G5 recognizes a similar list of tamper conditions to the Luna USB HSM 7 (see [Tamper Events](#)). When a tamper event occurs, a tamper state is reported in the **HSM Status** field in LunaCM's list of slots.

By default, tamper events are cleared automatically when you reboot the Luna Backup HSM G5 and log in as HSM SO. However, you can choose to prevent any further operations on the Luna Backup HSM G5. The following procedures will allow you to create a purple Secure Recovery Key (SRK) that the Backup HSM SO must present to unlock the HSM after a tamper event. This key contains part of the Master Tamper Key (MTK), which encrypts all sensitive data stored on the Backup HSM. By splitting the MTK and storing part of it on an SRK (purple iKey), you ensure that none of the stored material can be accessible until the SRK is presented.

You can create the purple SRK even for a Luna Backup HSM G5 that is initialized for password authentication. There is no password-based SRK equivalent; you must have a Luna PED and a purple iKey to use Secure Tamper Recovery and Secure Transport Mode.

Initializing the SRK also allows you to place the Luna Backup HSM G5 in Secure Transport Mode (STM). STM on the Luna Backup HSM G5 functions differently from STM on the Luna USB HSM 7 (see [Secure Transport Mode](#) for comparison). When the SRK is initialized and secure recovery enabled, STM on the Backup HSM is effectively a voluntary tamper state, where no operations are possible until you present the purple iKey.

CAUTION! Always keep a securely-stored backup copy of the purple iKey. If you lose this key, the Backup HSM is permanently locked and you will have to obtain an RMA for the Backup HSM.

This section provides directions for the following procedures:

- > ["Creating a Secure Recovery Key" below](#)
- > ["Setting Secure Transport Mode" on the next page](#)
- > ["Recovering From a Tamper Event or Secure Transport Mode" on the next page](#)
- > ["Disabling Secure Recovery" on page 159](#)

Creating a Secure Recovery Key

To enable secure recovery, you must create the Secure Recovery Key (purple iKey). This procedure will zeroize the SRK split on the Backup HSM, so that you must present the purple iKey to recover from a tamper event or Secure Transport Mode.

Prerequisites

- > Install the Backup HSM at the client and connect it to power (see ["Luna Backup HSM G5 Hardware Installation" on page 147](#)).
- > You require the Backup HSM SO credential (blue iKey).
- > Ensure that the Backup HSM can access PED service (Local or Remote PED), and that you have enough blank or rewritable purple iKeys available for your desired authentication scheme.
 - [Local PED] Connect the PED using a 9-pin Micro-D to Micro-D cable. Set the PED to **Local PED-SCP** mode.
 - [Remote PED] Set up a Remote PED server to authenticate the Backup HSM (see ["Configuring a Remote Backup Server" on page 164](#)).
 - [Remote PED] Initialize the Backup HSM RPV (see ["Initializing the Luna Backup HSM G5 Remote PED Vector" on page 151](#)). You require the orange iKey.

To create a Secure Recovery Key

1. Launch LunaCM on the client workstation.
2. Set the active slot to the Luna Backup HSM.
lunacm:> **slot set -slot** <slotnum>
3. [Remote PED] Connect the Luna Backup HSM G5 to the Remote PED server.
lunacm:> **ped connect -ip** <PEDserver_IP> **-port** <portnum>
4. Create a new split of the MTK on the Luna Backup HSM G5.
lunacm:> **srk generate**
5. Log in as Backup HSM SO.
lunacm:> **role login -name so**
6. Enable secure recovery.
lunacm:> **srk enable**

Attend to the Luna PED prompts to create the purple iKey. Secure Recovery is now enabled on the Luna Backup HSM G5.

Setting Secure Transport Mode

The following procedure will allow you to set Secure Transport Mode on the Luna Backup HSM G5.

Prerequisites

- > Ensure the Luna Backup HSM G5 can access PED services.
- > Secure Recovery must be enabled on the Backup HSM (see ["Creating a Secure Recovery Key" on the previous page](#)). You require the Secure Recovery Key (purple iKey) for the Luna Backup HSM G5.

To set Secure Transport Mode on the Luna Backup HSM G5

1. Launch LunaCM on the client workstation.
2. Set the active slot to the Luna Backup HSM G5.
lunacm:> **slot set -slot** <slotnum>
3. [Remote PED] Connect the Luna Backup HSM G5 to the Remote PED server.
lunacm:> **ped connect -ip** <PEDserver_IP> **-port** <portnum>
4. Set Secure Transport Mode.
lunacm:> **srk transport**
 - a. You are prompted for the SRK (purple iKey). This is to ensure that you have the key that matches the SRK split on the HSM.
 - b. The Luna PED displays a 16-digit verification code. Write this code down as an additional optional check. The SRK is zeroized on the Luna Backup HSM G5 and STM is now active.

Recovering From a Tamper Event or Secure Transport Mode

With Secure Recovery Mode enabled, the procedure to recover from a tamper event or to exit STM is the same.

Prerequisites

- > Ensure the Luna Backup HSM G5 can access PED services.
- > You require the Secure Recovery Key (purple iKey) for the Luna Backup HSM G5.
- > If you are recovering from a tamper event, reboot the Backup HSM and LunaCM before recovering.

lunacm:> **hsm restart**

lunacm:> **clientconfig restart**

To recover from a tamper event or exit STM

1. Launch LunaCM on the client workstation.
2. Set the active slot to the Luna Backup HSM G5.
lunacm:> **slot set -slot** <slotnum>
3. [Remote PED] Connect the Luna Backup HSM G5 to the Remote PED server.
lunacm:> **ped connect -ip** <PEDserver_IP> **-port** <portnum>

4. Recover the Luna Backup HSM G5 from the tamper event or STM.

lunacm:> **srk recover**

Attend to the Luna PED prompts:

- a. You are prompted for the SRK (purple iKey).
- b. [STM] The Luna PED displays a 16-digit verification code. If this code matches the one that was presented when you set STM, you can be assured that the Luna Backup HSM G5 has remained in STM since then.

The Luna Backup HSM G5 is recovered from the tamper/STM state and you can resume backup/restore operations.

Disabling Secure Recovery

To disable secure recovery, you must present the Secure Recovery Key (purple iKey) so that it can be stored on the Luna Backup HSM G5. You will no longer need to present the purple key to recover from a tamper event.

Prerequisites

- > Ensure the Luna Backup HSM G5 can access PED services.
- > You require the Secure Recovery Key (purple iKey) for the Luna Backup HSM G5.

To disable secure recovery

1. Launch LunaCM on the client workstation.
2. Set the active slot to the Luna Backup HSM G5.
3. [Remote PED] Connect the Luna Backup HSM G5 to the Remote PED server.

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

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

4. Log in as Backup HSM SO.

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

5. Disable secure recovery.

lunacm:> **srk disable**

You are prompted for the SRK (purple iKey).

Backup/Restore Using Luna Backup HSM G5

You can connect the Luna Backup HSM G5 to a USB port on the client workstation. This configuration allows you to perform backup/restore operations for all application partitions that appear as visible slots in LunaCM. It is useful in deployments where the partition Crypto Officer wants to keep backups at the client. You can restore a partition backup to the original source partition or to another existing Luna application partition that shares the same cloning domain.

This section provides instructions for the following procedures using this kind of deployment:

- > ["Initializing the Luna Backup HSM G5" on the next page](#)
- > ["Backing Up an Application Partition" on page 161](#)

> ["Restoring an Application Partition from Backup" on page 163](#)

NOTE To perform backup operations on Luna HSM Firmware 7.7.0 or newer (V0 or V1 partitions) you require at minimum:

- > [Luna Backup HSM 7 Firmware 7.7.1](#)
- > [Luna Backup HSM G5 Firmware 6.28.0](#)

You can use a Luna Backup HSM with older firmware to restore objects to a V0 or V1 partition, but this is supported for purposes of getting your objects from the older partitions onto the newer V0 or V1 partitions only. V0 and V1 partitions are considered more secure than partitions at earlier firmware versions - any attempt to restore from a higher-security status to lower-security status fails gracefully.

NOTE The size of the partition header is different for a Luna USB HSM 7 partition and its equivalent backup partition stored on a Luna Backup HSM G5. As a result, the value displayed in the `Used` column in the output of the **partition list** command (for the backed-up Luna USB HSM 7 partition) is different than the value displayed in the `Used` column in the output of the **token backup partition list** command (for the backup partition on the Backup HSM).

Initializing the Luna Backup HSM G5

Before you can use the Luna Backup HSM G5 to back up your partition objects, it must be initialized. This procedure is analogous to the standard HSM initialization procedure.

Prerequisites

- > Install the Luna Backup HSM G5 at the client and connect it to power (see ["Installing the Luna Backup HSM G5" on page 150](#)).
- > Ensure that the Backup HSM is not in Secure Transport Mode and that any tamper events are cleared (see ["Recovering From a Tamper Event or Secure Transport Mode" on page 158](#)).
- > [Multifactor Quorum Authentication] Ensure that you have enough blank or rewritable blue and red iKeys available for your desired authentication scheme.
 - [Local PED] Connect the Luna PED using a 9-pin Micro-D to Micro-D cable. Set the PED to **Local PED-SCP** mode.
 - [Remote PED] Initialize the Backup HSM RPV (see ["Initializing the Luna Backup HSM G5 Remote PED Vector" on page 151](#)). You require the orange iKey.
 - [Remote PED] Set up a Remote PED server to authenticate the Luna Backup HSM G5.

To initialize a client-connected Luna Backup HSM G5

1. Launch LunaCM on the client workstation.
2. Set the active slot to the Luna Backup HSM G5.


```
lunacm:> slot set -slot <slotnum>
```
3. [Remote PED] Connect the Luna Backup HSM G5 to the Remote PED server.


```
lunacm:> ped connect -ip <PEDserver_IP> -port <portnum>
```


4. Initialize the Luna Backup HSM G5, specifying a label and the method of authentication (**-initwithped** or **-initwithpwd**). You must initialize the HSM with the same authentication method as the partition(s) you plan to back up.

```
lunacm:> hsm init -label <label> {-initwithped |-initwithpwd}
```

You are prompted to set an HSM SO credential and cloning domain for the Backup HSM.

NOTE After initializing a client-connected Luna Backup HSM G5 to use PED authentication, the HSM erroneously requests a password to log in with any role. This issue occurs when [Luna HSM Client 10.3.0](#) or newer is used with HSM firmware 6.10.9 or older.

Workaround: Press ENTER to bypass the password prompt, and present the iKey as usual. Alternatively, use an older client or upgrade to [Luna Backup HSM G5 Firmware 6.24.7](#) or newer to avoid this.

Backing Up an Application Partition

You can use LunaCM to back up the contents of an application partition to the client-connected Luna Backup HSM G5. You can use this operation to create a backup on the Backup HSM, or add objects from the source partition to an existing backup.

Prerequisites

- > The Luna Backup HSM G5 must be initialized (see ["Initializing the Luna Backup HSM G5" on the previous page](#)).
- > The following policies are set:
 - **HSM policy 16: Allow network replication** must be set to **1 (ON)** on the HSM that hosts the user partition.
 - [V0 partitions] **Partition policy 0: "Allow private key cloning" on page 50** is set to **1 (ON)** on the user partition.
 - [V0 partitions] **Partition policy 4: "Allow secret key cloning" on page 51** is set to **1 (ON)** on the user partition.
- > You must have the Crypto Officer credential (black iKey) and domain (red iKey) for the source partition.
- > You must have the Backup HSM SO credential (blue iKey).
- > [Multifactor Quorum Authentication] This procedure is simpler if the source partition is activated (see ["Activation on Multifactor Quorum-Authenticated Partitions" on page 44](#)), since you require a Luna PED only for the Backup HSM.
 - [Local PED] Connect the PED to the Backup HSM using a 9-pin Micro-D to Micro-D cable.
 - [Remote PED] You must have the orange iKey for the Backup HSM (see ["Initializing the Luna Backup HSM G5 Remote PED Vector" on page 151](#)). If the source partition is not activated, you may need the orange iKey for the Luna USB HSM 7 as well.
 - [Remote PED] Set up Remote PED on the workstation you plan to use for multifactor quorum authentication. If the partition is not activated, you must connect to PEDserver with **ped connect** before logging in, and disconnect with **ped disconnect** before initiating the backup.

If you invoked scalable key storage (SKS) for your applications to create and store large numbers of keys, then the partition is V1. If you perform cloning operations (including HA) or Backup and Restore, see ["Backup/Restore and SKS" on page 74](#).

To back up an application partition to a client-connected Luna Backup HSM G5

1. Launch LunaCM on the client workstation.
2. Set the active slot to the source partition and log in as Crypto Officer.

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

```
lunacm:> role login -name co
```
3. [Multifactor Quorum Authentication] Connect the Luna Backup HSM G5 to the Luna PED.
 - [Local PED] Set the mode on the Luna PED to **Local PED-SCP**.
 - [Remote PED] Connect the Luna Backup HSM G5 slot to PEDserver.

```
lunacm:> ped connect -slot <Backup_HSM_slotnum> -ip <PEDserver_IP> -port <portnum>
```
4. Back up the partition, specifying the Luna Backup HSM G5 slot and a label for the backup (either a new or existing label). If you specify an existing backup label, include the **-append** option to add only new objects to the backup (duplicate objects will not be cloned). By default, the existing backup will be overwritten with the current contents of the source partition.

```
lunacm:> partition archive backup -slot <Backup_HSM_slotnum> [-partition <backup_label>] [-append] [-replace] [-smkonly]
```

If you omit the **-partition** option when creating a new backup, the partition is assigned a default name (<source_partition_name>_<YYYYMMDD>) based on the source HSM's internally-set time and date.

If you are backing up a V1 partition, include **-smkonly** to back up the SMK only. By default, the SMK and any encrypted cryptographic material on the partition are backed up.

The backup begins once you have completed the authentication process.

Objects are backed up one at a time. For existing backups, you can use the following options to define how individual objects are backed up:

-append	Add only new objects to an existing backup.
-replace	Delete the existing objects in a target backup partition and replace them with the contents of the source user partition. This is the default.
-append -replace	Add new objects and replace existing objects that have the same OUID but a different fingerprint (such as would occur if any of the object attributes were changed since the previous backup).

You are prompted to present or set the following credentials:

- [Remote PED] Backup HSM Remote PED vector (orange iKey)
- Backup HSM SO (password or blue iKey)
- Crypto Officer (password or black iKey) for the backup (can be the same as the source partition)
- Cloning domain (string or red iKey) for the backup (must be the same as the source partition)

The partition contents are cloned to the backup.

5. [Remote PED] Disconnect the Backup HSM from PEDserver.

```
lunacm:> ped disconnect
```

Restoring an Application Partition from Backup

You can use LunaCM to restore the contents of a backup to the original application partition, or any other Luna application partition that shares the same cloning domain.

Prerequisites

- > The target partition must be initialized with the same cloning domain as the backup partition.
- > The following policies are set:
 - **HSM policy 16: Allow network replication** must be set to **1 (ON)** on the HSM that hosts the user partition you want to restore to.
 - [V0 partitions] **Partition policy 0: "Allow private key cloning" on page 50** is set to **1 (ON)** on the user partition you want to restore to.
 - [V0 partitions] **Partition policy 4: "Allow secret key cloning" on page 51** is set to **1 (ON)** on the user partition you want to restore to.
- > You must have the Crypto Officer credentials for the backup partition and the target partition.
- > [Multifactor Quorum Authentication] This procedure is simpler if the application partition is activated (see ["Activation on Multifactor Quorum-Authenticated Partitions" on page 44](#)), since you require a Luna PED only for the Backup HSM.
 - [Local PED] Connect the PED to the Backup HSM using a 9-pin Micro-D to Micro-D cable.
 - [Remote PED] Set up Remote PED on the workstation you plan to use for multifactor quorum authentication. If the partition is not activated, you must connect to PEDserver with **ped connect** before logging in, and disconnect with **ped disconnect** before initiating the backup.

To restore the contents of a backup to an application partition

1. Launch LunaCM on the client workstation.
2. Set the active slot to the target partition and log in as Crypto Officer.


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

```
lunacm:> role login -name co
```
3. [PED Authentication] Connect the Luna Backup HSM G5 to the Luna PED.
 - [Local PED] Set the mode on the Luna PED to **Local PED-SCP**.
 - [Remote PED] Connect the Luna Backup HSM G5 slot to PEDserver.

```
lunacm:> ped connect -slot <Backup_HSM_slotnum> -ip <PEDserver_IP> -port <portnum>
```
4. [Optional] Display the available backups by specifying the Luna Backup HSM G5 slot. Each available backup also appears as a slot in LunaCM.


```
lunacm:> partition archive list -slot <Backup_HSM_slotnum>
```
5. [Optional] Display the contents of a backup by specifying the Luna Backup HSM G5 slot and the backup partition label in LunaCM.


```
lunacm:> partition archive contents -slot <backup_slotnum> -partition <backup_label>
```

6. Restore the partition contents, specifying the Luna Backup HSM G5 slot and the backup you wish to use. By default, duplicate backup objects with the same OUID as objects currently existing on the partition are not restored.

If you have changed attributes of specific objects since your last backup and you wish to revert these changes, include the **-replace** option.

If you are restoring a V1 partition and you only want to restore the SMK, include the **-smkonly** option.

```
lunacm:> partition archive restore -slot <Backup_HSM_slotnum> -partition <backup_label> [-replace] [-smkonly]
```

You are prompted for the backup's Crypto Officer credential.

The backup contents are cloned to the application partition.

Configuring a Remote Backup Server

The Remote Backup Service (RBS) is an optional Luna client component that allows you to connect one or more backup HSMs to a remote Luna HSM Client workstation to back up slots on any local Luna HSM Client workstations that are registered with the RBS server. RBS is useful in deployments where backups are stored in a separate location from the Luna USB HSM 7, to protect against catastrophic loss (fire, flood, etc).

RBS is a utility, included with the Luna HSM Client software, that runs on a workstation hosting one or more Backup HSMs. When RBS is configured and running, other clients or HSMs registered to it can see its Backup HSM(s) as slots in LunaCM.

Installing and Configuring the Remote Backup Service

RBS is installed using the Luna HSM Client installer. You must create a certificate for the RBS workstation and register it on all clients/appliances that will use the remote Backup HSMs. These instructions will allow you to install and configure RBS.

NOTE The Luna HSM Client version installed on the RBS workstation must be the same version installed on the client workstation(s). Ensure that you use a client version that is compatible with your Backup HSM firmware.

Prerequisites

- > Install the following Luna HSM Client components on any Luna USB HSM 7 client workstation that hosts slots for the partitions you want to backup using RBS (see [Luna HSM Client Software Installation](#)):
 - **Network:** The Network component includes utilities that are required for remote backups.
 - **Remote PED:** if you are backing up multifactor quorum-authenticated partitions.
- > Connect the backup HSM(s) directly to the Luna HSM Client workstation that will host RBS using the included USB cable.

NOTE On most workstations, the USB 3.0 connection provides adequate power to the backup HSM and it will begin the boot sequence. If you are using a low-power workstation, such as a netbook, the USB connection may not provide adequate power, in which case you will also need to connect the external power supply. It is recommended that you use the power supply for all backup HSMs connected to the RBS host workstation. If you are connecting multiple backup HSMs, you can use an external USB 3.0 hub if required.

- > Initialize the backup HSMs if necessary.
- > **HSM policy 16: Allow network replication** must be set to **1 (ON)** on the HSMs that hosts the user partitions to be backed up.

To install and configure RBS

1. On the workstation hosting the Backup HSM(s), install the **Backup** component of the Luna HSM Client (see [Luna HSM Client Software Installation](#)). If this workstation will also host a Remote PED, install the **Remote PED** component as well (Windows only).
2. Navigate to the Luna HSM Client home directory (`/usr/safenet/lunaclient/rbs/bin` on Linux/Unix) and generate a certificate for the RBS host.

> **rbs --genkey**

You are prompted to enter and confirm an RBS password. The certificate is generated in:

- Linux/UNIX: `<LunaClient_install_directory>/rbs/server/server.pem`
- Windows: `<LunaClient_install_directory>\cert\server\server.pem`

3. Specify the Backup HSM(s) that RBS will make available to clients.

> **rbs --config**

RBS displays a list of Backup HSMs currently connected to the workstation. Select the ones you want to provide remote backup services. When you have specified your selection, enter **X** to exit the configuration tool.

4. Launch the RBS daemon (Linux/UNIX) or console application (Windows).

- Linux/UNIX: `# rbs --daemon`
- Windows: Double-click the **rbs** application. A console window will remain open.

You are prompted to enter the RBS password.

5. Securely transfer the RBS host certificate (**server.pem**) to your Luna HSM Client workstation using **pscp** or **sftp**.
6. On the client workstation, register the RBS host certificate to the server list.

> **vtl addServer -n <Backup_host_IP> -c server.pem**

7. [Optional] Launch LunaCM on the client to confirm that the Backup HSM appears as an available slot.

NOTE If you encounter issues, try changing the RBS and PEDclient ports from their default values. Check that your firewall is not blocking ports used by the service.

You can now use the Backup HSM(s) as though they were connected to the client workstation locally, using Remote PED.