THALES

OpenStack Barbican: Integration Guide

THALES LUNA HSM AND LUNA CLOUD HSM

Document Information

Document Part Number	007-013570-001
Revision	C
Release Date	21 October 2021

Trademarks, Copyrights, and Third-Party Software

CONTENTS

Overview

This document guides security administrators through the steps for integrating OpenStack Barbican with a Thales Luna HSM or Thales Luna Cloud HSM. OpenStack Barbican is a REST API designed for the secure storage, provisioning, and management of secrets. Secrets are encrypted and decrypted on retrieval by a project specific Key Encryption Key (KEK), which in its turn encrypted with a Master Key (MKEK) and signed with an HMAC key. Luna HSMs can be used to securely store the Barbican MKEK and HMAC keys. OpenStack Barbican crypto components allow users to encrypt and decrypt cryptographic information using Thales Luna HSM.

The benefits of securing the cryptographic keys with Luna HSMs include:

- Secure generation, storage, and protection of the keys on FIPS 140-2 level 3 validated hardware.
- Full life cycle management of the keys.
- HSM audit trail*.
- Significant performance improvements by off-loading cryptographic operations from application servers.

*Luna Cloud HSM service does not have access to the secure audit trail

Certified Platforms

Certified platforms for Luna HSM

Luna HSM: Luna HSM appliances are purposefully designed to provide a balance of security, high performance, and usability that makes them an ideal choice for enterprise, financial, and government organizations. Luna HSMs physically and logically secure cryptographic keys and accelerate cryptographic processing. The Luna HSM on premise offerings include the Luna Network HSM, Luna PCIe HSM, and Luna USB HSMs. Luna HSMs are also available for access as an offering from cloud service providers such as IBM cloud HSM and AWS cloud HSM classic.

> **NOTE:** OpenStack Barbican Stein and onwards version is supported in FIPS mode for Luna HSM Firmware version 7.4 or below.

Certified platforms for Luna Cloud HSM

Luna Cloud HSM: Luna Cloud HSM platform provides on-demand, cloud-based HSM and Key Management services through a simple graphical user interface. With Luna Cloud HSM, security is simple, cost effective and easy to manage because there is no hardware to buy, deploy and maintain. As an Application Owner, you click and deploy services, generate usage reports and maintain just the services you need.

> **NOTE:** OpenStack Barbican Stein and onwards version is supported in Non-FIPS mode for Luna Cloud HSM.

Prerequisites

Before beginning the integration, ensure you have completed the following tasks:

> Configure Luna HSM

> Configure Luna Cloud HSM service

> Set up OpenStack Barbican

Configure Luna HSM

To configure Luna HSM:

1. Ensure the HSM is setup, initialized, provisioned, and ready for deployment.
2. Create a partition on the HSM for use by OpenStack Barbican.
3. Create and exchange certificate between the Luna Network HSM and Client system. Register client and assign partition to create an NTLS connection. Initialize Crypto Officer and Crypto User roles for the registered partition.
4. Verify that the partition is successfully registered and configured. The command to see the registered partition is:

```
# /usr/safenet/lunaclient/bin/lunacm

lunacm (64-bit) v10.3.0-275. Copyright (c) 2020 SafeNet. All rights
reserved.

        Available HSMs:
        Slot Id ->              0
        Label ->                Barbican
        Serial Number ->        1238696044950
        Model ->                LunaSA 7.4.0
        Firmware Version ->     7.4.0
        Configuration ->        Luna User Partition With SO (PW) Key Export
                                With Cloning Mode
        Slot Description ->     Net Token Slot
        FM HW Status ->         Non-FM
        Current Slot Id: 0
```

5. For PED-authenticated HSM, enable partition policies 22 and 23 to allow activation and auto-activation.

> **NOTE:** Follow the Luna Network Luna HSM documentation for detailed steps for creating NTLS connection, initializing the partitions, and managing various user roles.

Set up Luna HSM High-Availability

Refer to the Luna HSM documentation for HA steps and details regarding configuring and setting up two or more HSM boxes on host systems. You must enable the HAOnly setting in HA for failover to work so that if the primary goes down due to any reason all calls automatically route to the secondary until the primary recovers and starts up.

Configure Luna Cloud HSM service

You can configure Luna Cloud HSM Service in the following ways:

> Standalone Cloud HSM service using minimum client package

> Standalone Cloud HSM service using full Luna client package

> Luna HSM and Luna Cloud HSM service in hybrid mode

NOTE: Luna Client v10.x or higher is required for configuring Luna HSM device and Luna Cloud HSM service in hybrid mode.

Standalone Cloud HSM service using minimum client package

To configure Luna Cloud HSM service using minimum client package:

1. Transfer the downloaded .zip file to your Client workstation using pscp, scp, or other secure means.
2. Extract the .zip file into a directory on your client workstation.
3. Extract or untar the appropriate client package for your operating system. Do not extract to a new subdirectory; place the files in the client install directory.

   ```
   [Linux]

   cvclient-min.tar

   # tar -xvf cvclient-min.tar
   ```

4. Run the setenv script to create a new configuration file containing information required by the Luna Cloud HSM service.

   ```
   [Linux]

   Source the setenv script.

   # source ./setenv
   ```

5. Run the LunaCM utility and verify that the Cloud HSM service is listed.

NOTE: Follow the Luna Cloud HSM documentation for detailed steps for creating service, client, and initializing various user roles.

Standalone Cloud HSM service using full Luna client package

To configure Luna Cloud HSM service using full Luna client package:

1. Transfer the downloaded .zip file to your Client workstation using pscp, scp, or other secure means.
2. Extract the .zip file into a directory on your client workstation.
3. Extract or untar the appropriate client package for your operating system. Do not extract to a new subdirectory; place the files in the client install directory.

   ```
   [Linux]

   cvclient-min.tar

   # tar -xvf cvclient-min.tar
   ```

4. Run the setenv script to create a new configuration file containing information required by the Luna Cloud HSM service.

   ```
   [Linux]

   Source the setenv script.

   # source ./setenv
   ```

5. Copy the server and partition certificates from the Cloud HSM service client directory to Luna client certificates directory:

   > **NOTE:** Skip this step for Luna Client v10.2 or higher.

   Cloud HSM Certificates

   ```
   server-certificate.pem

   partition-ca-certificate.pem

   partition-certificate.pem
   ```

   LunaClient Certificate Directory

   ```
   [Linux default location for Luna Client]

   /usr/safenet/lunaclient/cert/
   ```

6. Open the configuration file from the Cloud HSM service client directory and copy the XTC and REST section.

   Linux

   ```
   Chrystoki.conf
   ```

7. Edit the Luna Client configuration file and add the XTC and REST sections copied from Cloud HSM service client configuration file.
8. Change server and partition certificates path from step 5 in XTC and REST sections. Do not change any other entries provided in these sections.

   > **NOTE:** Skip this step for Luna Client v10.2 or higher.

   [XTC]

   ```
   . . .
   PartitionCAPath=<LunaClient_cert_directory>\partition-ca-certificate.pem
   PartitionCertPath00=<LunaClient_cert_directory>\partition-certificate.pem
   . . .

   [REST]
   . . .
   SSLClientSideVerifyFile=<LunaClient_cert_directory>\server-certificate.pem
   . . .
   ```

9. Edit the following entry from the Misc section and update the correct path for the plugins directory:

   ```
   Misc]
   PluginModuleDir=<LunaClient_plugins_directory>

   [Linux Default]
   ```

```
/usr/safenet/lunaclient/plugins/
```

10. Save the configuration file. If you wish, you can now safely delete the extracted Cloud HSM service client directory.
11. Reset the ChrystokiConfigurationPath environment variable and point back to the location of the Luna Client configuration file.

    Linux

    Either open a new shell session, or export the environment variable for the current session pointing to the location of the Chrystoki.conf file:

    ```
    # export ChrystokiConfigurationPath=/etc/
    ```

12. Run the LunaCM utility and verify that the Cloud HSM service is listed. In hybrid mode, both Luna and Cloud HSM service will be listed.

> **NOTE:** Refer to Luna Cloud HSM documentation for detailed steps for creating service, client, and initializing various user roles.

Luna HSM and Luna Cloud HSM service in hybrid mode

To configure Luna HSM and Luna Cloud HSM service in hybrid mode, follow the steps mentioned under the Standalone Cloud HSM service using full Luna client package section above.

> **NOTE:** Luna Client v10.x or higher is required for configuring Luna HSM device and Luna Cloud HSM service in hybrid mode.

To use Luna Cloud HSM Service in FIPS mode

Cloud HSM service operates in both FIPS and non-FIPS mode. If your organization requires non-FIPS algorithms for your operations, enable the Allow non-FIPS approved algorithms check box when configuring your Cloud HSM service. The FIPS mode is enabled by default. Refer to the Mechanism List in the SDK Reference Guide for more information about available FIPS and non-FIPS algorithms.

Set up OpenStack Barbican

It is recommended that you familiarize yourself with OpenStack Barbican before beginning the integration. Refer to the OpenStack Barbican Documentation for more information about installation and pre-installation requirements. Complete the installation of OpenStack Barbican on the target machine for integration with Thales HSM.

Configuring OpenStack Barbican with Thales Luna HSM or Luna Cloud HSM

To configure OpenStack Barbican to use Luna HSM or Luna Cloud HSM:

1. Add the user `barbican` to the hsmusers group.

```
# gpasswd --add barbican hsmusers
```

> **NOTE:** Skip this step if you are using Standalone Cloud HSM service using minimum client package.

2. Open the OpenStack Barbican configuration file /etc/barbican/barbican.conf and make the following changes in the Crypto plugin section.

```
# ================= Secret Store Plugin =============
[secretstore]
namespace = barbican.secretstore.plugin
enabled_secretstore_plugins = store_crypto
# ================= Crypto plugin ===================
[crypto]
enabled_crypto_plugins = p11_crypto
[p11_crypto_plugin]
library_path = '<path_to_cyptoki_library>'
login = '<partition_password>'
mkek_label = '<mkek_label>'
mkek_length = 32
hmac_label = '<hmac_label>'
slot_id = <partition_slot_id>
```

> **NOTE:** Update the `barbican.conf` file with correct information for your Thales Luna HSM or Luna Cloud HSM.

3. Generate the Master Key Encryption Key (MKEK). The MKEK gets generated on the registered Luna HSM partition or Luna Cloud HSM service.

```
# barbican-manage hsm gen_mkek --library-path '<path_to_cyptoki_library>' --passphrase '<partition_password>' --slot-id <partition_slot_id>
--label '<mkek_label>' --length 32
```

4. Generate the HMAC key using the following command. The HMAC key gets generated on the registered HSM partition.

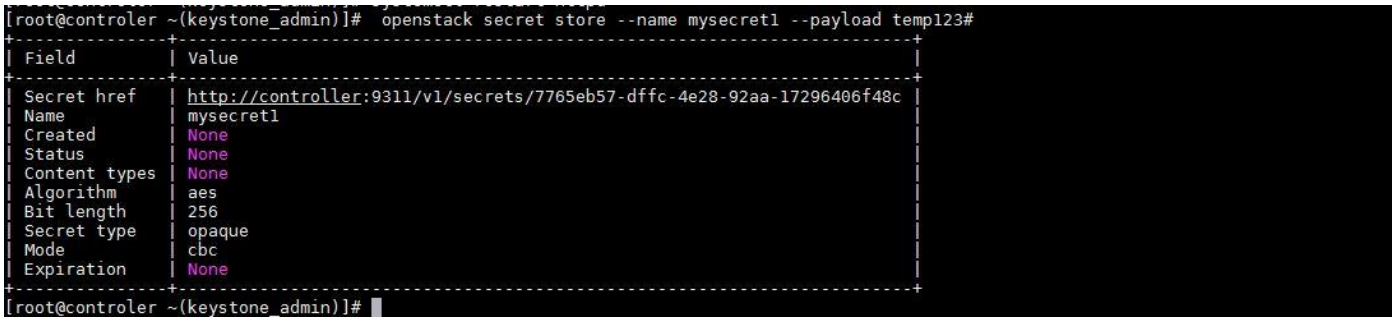
```
# barbican-manage hsm gen_hmac --library-path '<path_to_cyptoki_library>' --passphrase '<partition_password>' --slot-id <partition_slot_id> --label '<hmac_label>' --length 32
```

5. Restart the OpenStack Barbican API and the httpd service.

```
# systemctl restart openstack-barbican-api.service
# systemctl restart httpd.service
```

6. Use the OpenStack CLI to store a secret.

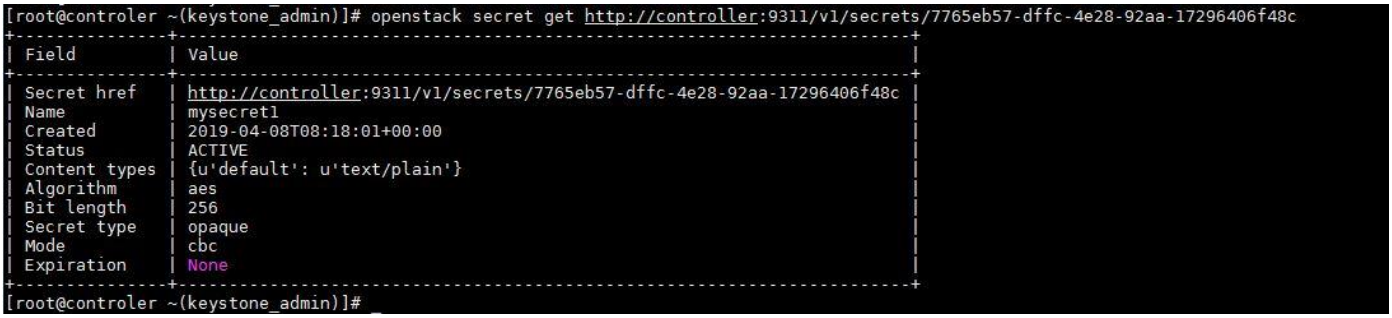
```
# openstack secret store --name mysecret1 --payload temp123#
```



> **NOTE:** If the command fails with the error `CKR_INVALID_ATTRIBUTE`, open your `pkcs11.py` file at `/usr/lib/python2.7/site-packages/barbican/plugin/crypto/pkcs11.py` and set `CKA_SENSITIVE = True`.

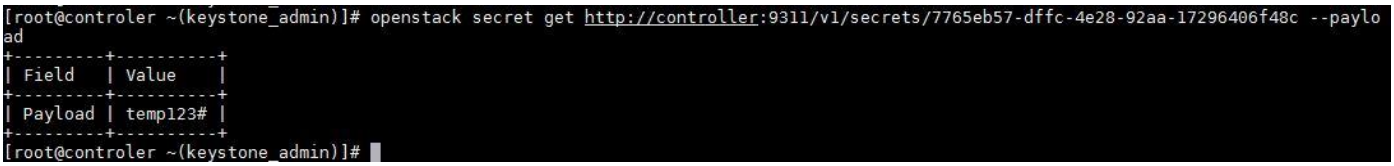
7. Confirm that the secret was stored by retrieving it without using the secret payload.

```
# openstack secret get http://controller:9311/v1/secrets/7765eb57-dffc-4e28-92aa-17296406f48c
```



8. Retrieve the secret payload.

```
# openstack secret get http://controller:9311/v1/secrets/7765eb57-dffc-4e28-92aa-17296406f48c --payload
```



You should see the original decrypted secret in the response.

This completes the Integration of OpenStack Barbican with a Thales Luna HSM or Luna Cloud HSM.

Contacting customer support

If you encounter a problem during this integration, contact your supplier or Thales Customer Support. Thales Customer Support operates 24 hours a day, 7 days a week. Your level of access to this service is governed by the support plan arrangements made between Thales and your organization. Please consult this support plan for further information about 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 a database where you can find solutions for most common problems. The Customer Support Portal is a comprehensive, fully searchable repository 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. You can also use the portal to create and manage support cases.

> **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 support

If you have an urgent problem, or cannot access the Customer Support Portal, you can contact Thales Customer Support by telephone at +1 410-931-7520. Additional local telephone support numbers are listed on the support portal.

Email support

You can also contact technical support by email at technical.support.DIS@thalesgroup.com.