THALES

Red Hat Certificate System: Integration Guide

THALES LUNA HSM AND LUNA CLOUD HSM

Document Information

Document Part Number	007-012317-001
Revision	D
Release Date	8 April 2024

Trademarks, Copyrights, and Third-Party Software

CONTENTS

Overview

This guide provides step-by-step instructions for seamlessly integrating Red Hat Certificate System with a Luna HSM device or Luna Cloud HSM service. Red Hat Certificate System is comprehensive security framework that handles user identity management, protects digital communication against threats, and streamlines the integration of essential encryption and authentication technologies.

Red Hat Certificate System utilizes Luna HSMs to safeguard private signing keys, thereby offloading cryptographic tasks from the host server to the HSM. Integration with Red Hat Certificate System relies on the widely adopted PKCS#11 interface. This interface enables Red Hat Certificate System to create RSA/ECDSA keys directly on Luna HSMs. These keys, used for encryption and signing, are crucial for various subsystems within Red Hat Certificate System, including CA, KRA, OCSP, TPS, or TKS. Supported key sizes for RSA and ECC algorithms on Luna HSMs are as follows:

Algorithms	Supported Key Sizes
RSA	> 2048 > 3072 > 4096
ECC	> nistp256 > nistp384 > nistp521

This guide illustrates the integration process by demonstrating the use of a signing key generated on a Luna HSM. The integration of Red Hat Certificate System with Luna HSMs offers the following benefits:

- Secure generation, storage, and protection of the identity signing private keys using FIPS 140-2 level 3 validated hardware.
- Full life cycle management of the keys to ensure their integrity and reliability throughout their usage.
- Maintenance of a comprehensive HSM audit trail for transparency and accountability in key operations. It's important to note that Luna Cloud HSM service does not have access to this secure audit trail.
- Significant performance enhancements by offloading cryptographic operations from application servers.

Certified Platforms

This integration has been tested and verified on the following platforms:

HSM Type	Operating System	Red Hat Certificate System	Red Hat Directory Server
Luna HSM f/w v7.7.x / v7.8.x Luna Client v10.7	Red Hat Enterprise Linux 8.6 (64-bit)	RHCS v10.4	RHDS v11.7

HSM Type	Operating System	Red Hat Certificate System	Red Hat Directory Server
Luna HSM f/w 7.3.x Luna Client v7.x	Red Hat Enterprise Linux 7.6 (64-bit)	RHCS v9.5	RHDS v10.4

NOTE: This integration is tested using Luna Client in both High Availability (HA) and FIPS-compliant modes.

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. Luna HSM on premise offerings include the Luna Network HSM, Luna PCIe HSM, and Luna USB HSMs. Luna HSM are also available for access as an offering from cloud service providers such as IBM cloud HSM and AWS cloud HSM classic.

HSM Type	Operating System	Red Hat Certificate System	Red Hat Directory Server
Luna Cloud HSM	Red Hat Enterprise Linux 7.6 (64-bit)	RHCS v9.5	RHDS v10.4

Luna Cloud HSM: Luna Cloud HSM provides on-demand 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.

Prerequisites

Before you begin this integration, complete the following tasks:

Configure Luna HSM

If you are using Luna HSM:

1. Verify that the HSM is set up, initialized, provisioned, and ready for deployment. Refer to the Luna HSM documentation for more information.
2. Create a partition that will be later used by Red Hat Certificate System.
3. If using a Luna Network HSM, register a client for the system and assign the client to the partition to create an NTLS connection. Initialize the Crypto Officer and Crypto User roles for the registered partition.
4. Ensure that the partition is successfully registered and configured. The command to view the registered partitions is:

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

lunacm (64-bit) v10.7.0-255. Copyright (c) 2023 Thales Group. All rights
reserved.


        Available HSMs:
```

```
Slot Id ->              0
Label ->                TPA02
Serial Number ->        1238696044901
Model ->                LunaSA 7.8.4
Firmware Version ->     7.8.4
Bootloader Version ->   1.1.5
Configuration ->        Luna User Partition With SO (PED) Key Export
                        With Cloning Mode
Slot Description ->     Net Token Slot
FM HW Status ->         FM Ready

Current Slot Id: 0
```

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

> **NOTE:** Refer to Luna HSM documentation for detailed steps to create NTLS connection and initialize the partitions and various user roles.

Configuring PED Authenticated Luna HSM

If you are using a PED-based Luna HSM, ensure the policy `ProtectedAuthenticationPathFlagStatus` is set to "`1`" in the *`Misc`* section of the `Chrystoki.conf` file.

```
Misc = {
ProtectedAuthenticationPathFlagStatus = 1;
}
```

Set up Luna HSM High-Availability

Follow the instructions provided in the Luna HSM documentation to configure and set up two or more HSM boxes on host systems for high availability. Ensure that the HAOnly setting is enabled to enable failover functionality. In the event of the primary HSM going down, all calls will automatically route to the secondary HSM until the primary recovers and restarts.

Controlling user access to HSM

> **NOTE:** This section is applicable only for Linux users.

By default, only the root user has access to the HSM. You can specify a set of non-root users that are permitted to access the HSM by adding them to the `hsmusers` group. The client software installation automatically creates the `hsmusers` group. The `hsmusers` group is retained when you uninstall the client software, allowing you to upgrade the software while retaining your `hsmusers` group configuration.

Add a user to hsmusers group

To allow non-root users or applications access to the HSM, assign the users to the `hsmusers` group. The users you assign to the `hsmusers` group must exist on the client workstation.

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

   ```
   # sudo gpasswd --add <username> hsmusers
   ```

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

Removing a user from hsmusers group

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

   ```
   # sudo gpasswd -d <username> hsmusers
   ```

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

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

Configure Luna Cloud HSM service.

Follow these steps to set up your Luna Cloud HSM:

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 using the following command:

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

   > **NOTE:** Do not extract to a new subdirectory. Place the files in the client install directory.

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

   ```
   # source ./setenv
   ```

   > **NOTE:** To add the configuration to an already installed UC client, use the --addcloudhsm option when running the setenv script.

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

   > **NOTE:** If your organization requires non-FIPS algorithms for your operations, ensure that the Allow non-FIPS approved algorithms check box is checked. For more information, refer to Supported Mechanisms.

Set up Red Hat Certificate System

To set up Red Hat Certificate System, complete these procedures:

- Set up a new directory server
- Install a Red Hat Certificate System

NOTE: Before installing RHCS v10.4, ensure that your system is running on RHEL 8.6 and that the EUS update is enabled by executing the following commands:

```
# subscription-manager release --set 8.6
# subscription-manager repos --enable rhel-8-for-x86_64-baseos-eus-rpms
# subscription-manager repos --enable rhel-8-for-x86_64-appstream-eus-rpms
```

Set up a new directory server

NOTE: Before you begin, ensure you've followed the guidelines outlined in the Red Hat Directory Server Installation Documentation and adhered to the provided instructions. Additionally, enable the necessary repository by executing the following command, replacing `dirsrv-11-for-rhel-8-x86_64-rpms` with the actual repository name:

```
# subscription-manager repos --enable=dirsrv-11-for-rhel-8-x86_64-rpms
```

To set up a new directory server:

1. Use the following command to install the Red Hat Directory Server module, such as redhat-ds:11, for Red Hat Directory Server 11. This command automatically handles all necessary dependencies:

   ```
   # dnf module install redhat-ds:11
   ```

2. Verify that the `firewalld` service is running.

   ```
   # systemctl status firewalld
   ```

3. Open the necessary ports using the firewall-cmd utility. For example, to open the default LDAP and LDAPS ports in the default firewall zone, run:

   ```
   # firewall-cmd --permanent --add-port={389/tcp,636/tcp}
   ```

4. Reload the firewall configuration to ensure the change takes effect.

   ```
   # firewall-cmd --reload
   ```

5. Initiate the interactive installer to set up a new instance by running the following command:

   ```
   # dscreate interactive
   ```

   Follow the prompts and provide the required values to complete the setup of the Red Hat Directory Server instance, which will be used with the Red Hat Certificate System.

Install a Red Hat Certificate System

NOTE: Before proceeding, ensure you've followed the guidelines outlined in the Red Hat Certificate System Documentation and adhered to the provided instructions. Additionally, enable the necessary repository by executing the following command, replacing `certsys-10.4-for-rhel-8-x86_64-rpms` with the actual repository name:

```
# subscription-manager repos --enable certsys-10.4-for-rhel-8-x86_64-rpms
```

To install a Red Hat Certificate System:

1. Enable FIPS mode on the RHEL host. To verify if FIPS mode is enabled, run the following command:

   ```
   # sysctl crypto.fips_enabled
   ```

   If the returned value is 1, FIPS mode is enabled. If not, refer to the Red Hat Linux Documentation to enable the FIPS mode.

2. Ensure SELinux is set to enforcing mode. By default, SELinux is enabled and running in enforcing mode after installing Red Hat Enterprise Linux. To display the current SELinux mode, use the following command:

   ```
   # getenforce
   ```

3. Open the required ports using the firewall-cmd utility. For example, to open the Certificate System default ports in the default firewall zone, run:

   ```
   # firewall-cmd --permanent --add-port={8080/tcp,8443/tcp,8009/tcp,8005/tcp}
   ```

4. Reload the firewall configuration to ensure that the change takes into effect.

   ```
   # firewall-cmd --reload
   ```

5. Enable and install the Red Hat Certificate System.

   ```
   # dnf module enable redhat-pki
   ```

   ```
   # dnf install redhat-pki
   ```

   The `redhat-pki` module installs subsystems of Red Hat Certificate System along with all required dependencies. Alternatively, you can install packages separately following the Red Hat Certificate System Documentation.

Integrate Luna HSM with Red Hat Certificate System

Integration of Luna HSM with Red Hat Certificate system involves two steps:

> Add Luna HSM as an external token

> Install and configure Red Hat Certificate System with Luna HSM

Add Luna HSM as an external token

Before Integrating Red Hat Certificate System, ensure that Luna HSM is working as an External Token. Red Hat Certificate System uses to generate and store its key pairs and certificates PKCS #11-compliant external tokens. To test this, add the Luna HSM as a module in a test database.

1. Create an empty test database.

   ```
   # mkdir ~/test_nssdb/
   # certutil -N -d ~/test_nssdb/ --empty-password
   ```

2. Add the Luna HSM as a PKCS#11 module in test database.

   ```
   # modutil -dbdir ~/test_nssdb/ -add LUNAHSM -libfile
   /usr/safenet/lunaclient/lib/libCryptoki2_64.so
   ```

3. Verify that LUNAHSM is added as a module and displaying the token information.

   ```
   # modutil -list -dbdir ~/test_nssdb
   ```

4. Delete the test database.

   ```
   # rm -rf ~/test_nssdb
   ```

This confirms that Luna HSM library is loading successfully as Red Hat Certificate System uses it as an entry point while installing and configuring the Certificate subsystem.

Install and configure Red Hat Certificate System with Luna HSM

Follow these steps to install and configure Red Hat Certificate System for Luna HSM integration:

1. Create the `default_luna.txt` file by using the template provided below.

> **NOTE:** During the installation of Red Hat Certificate System, an automatic generation of a default configuration file occurs. However, to enable Luna HSM integration, you must create a dedicated PKI configuration file named default_luna.txt. This file will supersede the default configuration, containing specific Luna HSM-related information for PKI subsystems.

Ensure to replace all placeholder values such as passwords and HSM parameters with the appropriate values specific to your environment. Passwords and HSM parameter values requiring customization are highlighted in bold.

By default, the RSA algorithm is used to generate all subsystem certificates. If you prefer ECC keys for the subsystem, you can uncomment the ECC parameters defined in the respective section in `default_luna.txt`. However, please note that while ECC keys can be used for other certificates, the audit signing certificate must always use an RSA key.

```
#############################################################################
#############################################################################
#############################################################################
##                                                                         ##
## EXAMPLE: Configuration File used to override '/etc/pki/default.cfg'     ##
## when using a LunaSA Hardware Security Module (HSM):                     ##
##                                                                         ##
## # modutil -dbdir . -list                                                ##
## Listing of PKCS #11 Modules                                             ##
## -----------------------------------------------------------             ##
## 1. NSS Internal PKCS #11 Module                                         ##
## slots: 2 slots attached                                                 ##
## status: loaded                                                          ##
##                                                                         ##
## slot: NSS Internal Cryptographic Services                               ##
## token: NSS Generic Crypto Services                                      ##
##                                                                         ##
## slot: NSS User Private Key and Certificate Services                     ##
## token: NSS Certificate DB                                               ##
##                                                                         ##
## 2. lunasa                                                               ##
## library name: /usr/safenet/lunaclient/lib/libCryptoki2_64.so            ##
```

```
## slots: 4 slots attached                                                    ##
## status: loaded                                                             ##
##                                                                            ##
## slot: LunaNet Slot                                                         ##
## token: rhcs-pki                                                            ##
##                                                                            ##
## slot: Luna UHD Slot                                                        ##
## token:                                                                     ##
##                                                                            ##
## slot: Luna UHD Slot                                                        ##
## token:                                                                     ##
## ----------------------------------------------------------                 ##
##                                                                            ##
## Based on the example above, substitute all password values,                ##
## as well as the following values:                                           ##
##                                                                            ##
## <hsm_libfile>=/usr/safenet/lunaclient/lib/libCryptoki2_64.so               ##
## <hsm_modulename>=lunasa                                                    ##
## <hsm_token_name>=rhcs-pki                                                  ##
##                                                                            ##
## Where hsm_modulename is user-defined value for Luna HSM.                   ##
##                                                                            ##
################################################################################
################################################################################
################################################################################

[DEFAULT]
############################
# Provide HSM parameters #
############################
pki_hsm_enable=True
pki_hsm_libfile=<hsm_libfile>
pki_hsm_modulename=<hsm_modulename>
pki_token_name=<hsm_token_name>
```

```
pki_token_password=<pki_token_password>

###################################
# Remove Old Directory Server Data #
###################################
pki_ds_remove_data=True

########################################
# Provide PKI-specific HSM token names #
########################################
pki_audit_signing_token=<hsm_token_name>
pki_ssl_server_token=<hsm_token_name>
pki_subsystem_token=<hsm_token_name>

#################################
# Provide PKI-specific passwords #
#################################
pki_admin_password=<pki_admin_password>
pki_client_pkcs12_password=<pki_client_pkcs12_password>
pki_ds_password=<pki_ds_password>

#####################################
# Provide non-CA-specific passwords #
#####################################
pki_client_database_password=<pki_client_database_password>

########################################
# Only required, if ECC keys are desired #
########################################
#pki_admin_key_algorithm=SHA384withEC
#pki_admin_key_size=nistp384
#pki_admin_key_type=ecc
#pki_admin_signing_algorithm=SHA384withEC
#pki_ssl_server_key_algorithm=SHA384withEC
```

```
#pki_ssl_server_key_size=nistp384
#pki_ssl_server_key_type=ecc
#pki_ssl_server_signing_algorithm=SHA384withEC
#pki_subsystem_key_algorithm=SHA384withEC
#pki_subsystem_key_size=nistp384
#pki_subsystem_key_type=ecc
#pki_subsystem_signing_algorithm=SHA384withEC

#############################################################
# ONLY required if specifying a non-default PKI instance name #
#############################################################
#pki_instance_name=<pki_instance_name>

#############################################################
# ONLY required if specifying non-default PKI instance ports #
#############################################################
#pki_http_port=<pki_http_port>
#pki_https_port=<pki_https_port>

#########################################################################
# ONLY required if specifying non-default 389 Directory Server ports #
#########################################################################
#pki_ds_ldap_port=<pki_ds_ldap_port>
#pki_ds_ldaps_port=<pki_ds_ldaps_port>

#########################################################################
# ONLY required if PKI is using a Security Domain on a remote system #
#########################################################################
#pki_ca_hostname=<pki_ca_hostname>
#pki_issuing_ca_hostname=<pki_issuing_ca_hostname>
#pki_issuing_ca_https_port=<pki_issuing_ca_https_port>
#pki_security_domain_hostname=<pki_security_domain_hostname>
#pki_security_domain_https_port=<pki_security_domain_https_port>
```

```
############################################################
# ONLY required for PKI using an existing Security Domain #
############################################################
# NOTE: pki_security_domain_password == pki_admin_password
# of CA Security Domain Instance
pki_security_domain_password=<pki_admin_password>

[Tomcat]
################################################################
# ONLY required if specifying non-default PKI instance ports #
################################################################
#pki_ajp_port=<pki_ajp_port>
#pki_tomcat_server_port=<pki_tomcat_server_port>

[CA]
#######################################
# Provide CA-specific HSM token names #
#######################################
pki_ca_signing_token=<hsm_token_name>
pki_ocsp_signing_token=<hsm_token_name>

#################################################
# Include keyflag options for all core CA certs #
#################################################
pki_ca_signing_opsFlag=encrypt,decrypt,sign,verify,wrap,unwrap
pki_subsystem_opsFlag=encrypt,decrypt,sign,verify,wrap,unwrap
pki_sslserver_opsFlag=encrypt,decrypt,sign,verify,wrap,unwrap
pki_ocsp_signing_opsFlag=encrypt,decrypt,sign,verify,wrap,unwrap
pki_audit_signing_opsFlag=encrypt,decrypt,sign,verify,wrap,unwrap

#####################################################
# Include keyflag mask options for all core CA certs #
#####################################################
pki_ca_signing_opsFlagMask=encrypt,decrypt,sign,verify,wrap,unwrap
```

```
pki_subsystem_opsFlagMask=encrypt,decrypt,sign,verify,wrap,unwrap
pki_sslserver_opsFlagMask=encrypt,decrypt,sign,verify,wrap,unwrap
pki_ocsp_signing_opsFlagMask=encrypt,decrypt,sign,verify,wrap,unwrap
pki_audit_signing_opsFlagMask=encrypt,decrypt,sign,verify,wrap,unwrap

########################################
# Only required, if ECC keys are desired #
########################################
#pki_ca_signing_key_algorithm=SHA384withEC
#pki_ca_signing_key_size=nistp384
#pki_ca_signing_key_type=ecc
#pki_ca_signing_signing_algorithm=SHA384withEC
#pki_ocsp_signing_key_algorithm=SHA384withEC
#pki_ocsp_signing_key_size=nistp384
#pki_ocsp_signing_key_type=ecc
#pki_ocsp_signing_signing_algorithm=SHA384withEC

#############################################################################
# ONLY required if 389 Directory Server for CA resides on a remote system #
#############################################################################
#pki_ds_hostname=<389 hostname>

[KRA]
########################################
# Provide KRA-specific HSM token names #
########################################
pki_storage_token=<hsm_token_name>
pki_transport_token=<hsm_token_name>

################################################
# Include keyflag options for all core KRA certs #
################################################
pki_storage_opsFlag=encrypt,decrypt,sign,verify,wrap,unwrap
pki_transport_opsFlag=encrypt,decrypt,sign,verify,wrap,unwrap
```

```
pki_audit_signing_opsFlag=encrypt,decrypt,sign,verify,wrap,unwrap

#######################################################
# Include keyflag mask options for all core KRA certs #
#######################################################
pki_storage_opsFlagMask=encrypt,decrypt,sign,verify,wrap,unwrap
pki_transport_opsFlagMask=encrypt,decrypt,sign,verify,wrap,unwrap
pki_audit_signing_opsFlagMask=encrypt,decrypt,sign,verify,wrap,unwrap

#########################################
# Only required, if ECC keys are desired #
#########################################
#pki_storage_key_algorithm=SHA384withEC
#pki_storage_key_size=nistp384
#pki_storage_key_type=ecc
#pki_storage_signing_algorithm=SHA384withEC
#pki_transport_key_algorithm=SHA384withEC
#pki_transport_key_size=nistp384
#pki_transport_key_type=ecc
#pki_transport_signing_algorithm=SHA384withEC

#############################################################################
# ONLY required if 389 Directory Server for KRA resides on a remote system #
#############################################################################
#pki_ds_hostname=<389 hostname>

[OCSP]
#########################################
# Provide OCSP-specific HSM token names #
#########################################
pki_ocsp_signing_token=<hsm_token_name>

#####################################################
# Include keyflag options for all core OCSP certs #
```

```
##################################################
pki_ocsp_signing_opsFlag=encrypt,decrypt,sign,verify,wrap,unwrap
pki_audit_signing_opsFlag=encrypt,decrypt,sign,verify,wrap,unwrap

##########################################################
# Include keyflag mask options for all core OCSP certs #
##########################################################
pki_ocsp_signing_opsFlagMask=encrypt,decrypt,sign,verify,wrap,unwrap
pki_audit_signing_opsFlagMask=encrypt,decrypt,sign,verify,wrap,unwrap

########################################
# Only required, if ECC keys are desired #
########################################
#pki_ocsp_signing_key_algorithm=SHA384withEC
#pki_ocsp_signing_key_size=nistp384
#pki_ocsp_signing_key_type=ecc
#pki_ocsp_signing_signing_algorithm=SHA384withEC

#############################################################################
# ONLY required if 389 Directory Server for OCSP resides on a remote system #
#############################################################################
#pki_ds_hostname=<389 hostname>

[TKS]
######################################
# Provide TKS-specific HSM token names #
######################################

##################################################
# Include keyflag options for all core TKS certs #
##################################################
pki_audit_signing_opsFlag=encrypt,decrypt,sign,verify,wrap,unwrap

########################################################
```

```
# Include keyflag mask options for all core TKS certs #
#######################################################
pki_audit_signing_opsFlagMask=encrypt,decrypt,sign,verify,wrap,unwrap

#############################################################################
# ONLY required if 389 Directory Server for TKS resides on a remote system #
#############################################################################
#pki_ds_hostname=<389 hostname>

[TPS]
###################################
# Provide TPS-specific parameters #
###################################
pki_authdb_basedn=<dnsdomainname where hostname.b.c.d is dc=b,dc=c,dc=d>

#########################################
# Provide TPS-specific HSM token names #
#########################################

#################################################
# Include keyflag options for all core TPS certs #
#################################################
pki_audit_signing_opsFlag=encrypt,decrypt,sign,verify,wrap,unwrap

#####################################################
# Include keyflag mask options for all core TPS certs #
#####################################################
pki_audit_signing_opsFlagMask=encrypt,decrypt,sign,verify,wrap,unwrap

#############################################################################
# ONLY required if 389 Directory Server for TPS resides on a remote system #
#############################################################################
#pki_ds_hostname=<389 hostname>
```

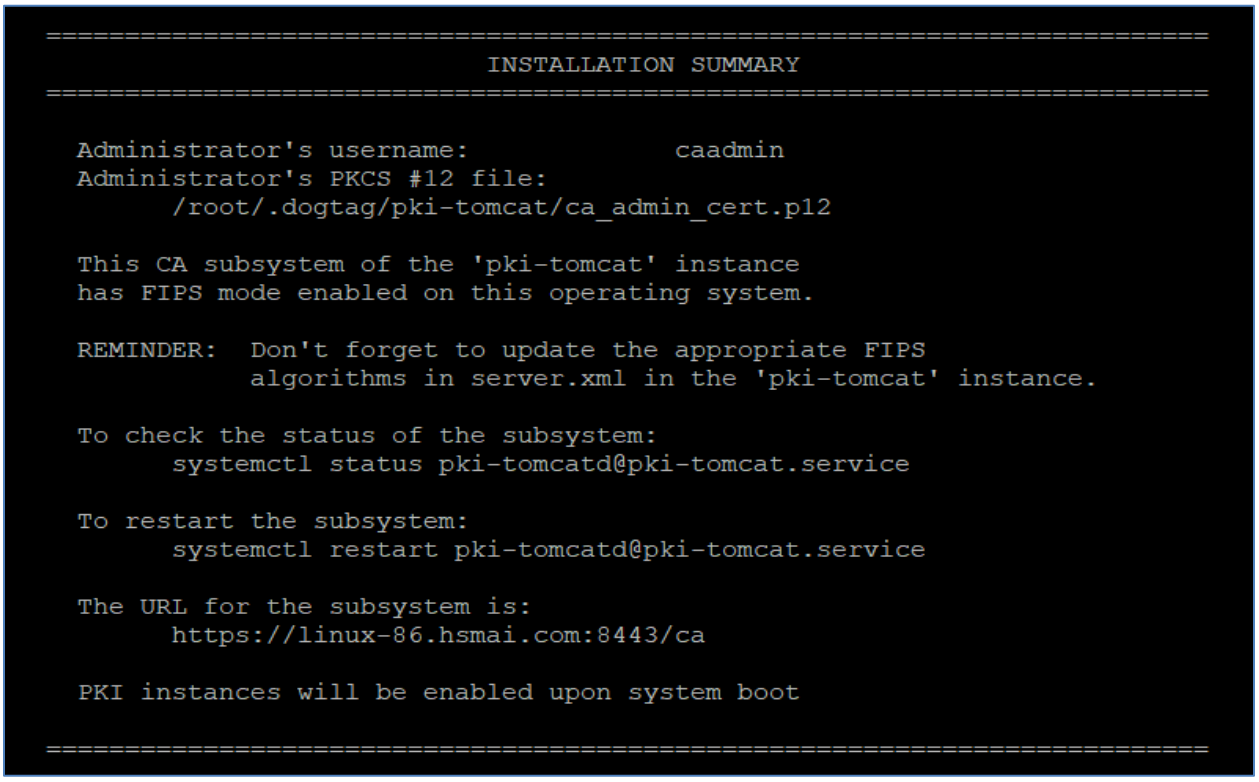
```
############################################################
# ONLY required if TPS requires a CA on a remote machine #
############################################################
#pki_ca_uri=https://<pki_ca_hostname>:<pki_ca_https_port>

######################################
# ONLY required if TPS requires a KRA #
######################################
#pki_enable_server_side_keygen=True

############################################################
# ONLY required if TPS requires a KRA on a remote machine #
############################################################
#pki_kra_uri=https://<pki_kra_hostname>:<pki_kra_https_port>

############################################################
# ONLY required if TPS requires a TKS on a remote machine #
############################################################
#pki_tks_uri=https://<pki_tks_hostname>:<pki_tks_https_port>
```

2. Install the Certificate Authority:

```
# pkispawn -s CA -f ./default_luna.txt --debug
```

NOTE: Before proceeding with the installation and configuration of any dependent subsystems, it's imperative to first install and configure the Certificate Authority. This ensures a smooth setup and operation of the Red Hat Certificate System with Luna HSM.

NOTE: It is recommended to review the installation summary to confirm the successful execution of the above command and all subsequent commands.



3. Install the Key Recovery Authority (KRA).

```
# pkispawn -s KRA -f ./default_luna.txt --debug
```

```
==========================================================================
                      INSTALLATION SUMMARY
==========================================================================

  Administrator's username:             kraadmin

  This KRA subsystem of the 'pki-tomcat' instance
  has FIPS mode enabled on this operating system.

  REMINDER:  Don't forget to update the appropriate FIPS
             algorithms in server.xml in the 'pki-tomcat' instance.

  To check the status of the subsystem:
        systemctl status pki-tomcatd@pki-tomcat.service

  To restart the subsystem:
        systemctl restart pki-tomcatd@pki-tomcat.service

  The URL for the subsystem is:
        https://linux-86.hsmai.com:8443/kra

  PKI instances will be enabled upon system boot

==========================================================================
```

4. Install the Online Certificate Responder Service (OCSP):

```
# pkispawn -s OCSP -f ./default_luna.txt --debug
```

```
==========================================================================
                      INSTALLATION SUMMARY
==========================================================================

  Administrator's username:             ocspadmin

  This OCSP subsystem of the 'pki-tomcat' instance
  has FIPS mode enabled on this operating system.

  REMINDER:  Don't forget to update the appropriate FIPS
             algorithms in server.xml in the 'pki-tomcat' instance.

  To check the status of the subsystem:
        systemctl status pki-tomcatd@pki-tomcat.service

  To restart the subsystem:
        systemctl restart pki-tomcatd@pki-tomcat.service

  The URL for the subsystem is:
        https://linux-86.hsmai.com:8443/ocsp

  PKI instances will be enabled upon system boot

==========================================================================
```

5. Install the Token Key Service (TKS):

```
# pkispawn -s TKS -f ./default_luna.txt --debug
```

```
==========================================================================
                      INSTALLATION SUMMARY
==========================================================================

  Administrator's username:             tksadmin

  This TKS subsystem of the 'pki-tomcat' instance
  has FIPS mode enabled on this operating system.

  REMINDER:  Don't forget to update the appropriate FIPS
             algorithms in server.xml in the 'pki-tomcat' instance.

  To check the status of the subsystem:
        systemctl status pki-tomcatd@pki-tomcat.service

  To restart the subsystem:
        systemctl restart pki-tomcatd@pki-tomcat.service

  The URL for the subsystem is:
        https://linux-86.hsmai.com:8443/tks

  PKI instances will be enabled upon system boot

==========================================================================
```

6. Install the Token Processing System (TPS):

```
# pkispawn -s TPS -f ./default_luna.txt --debug
```

```
==========================================================================
                      INSTALLATION SUMMARY
==========================================================================

  Administrator's username:             tpsadmin

  This TPS subsystem of the 'pki-tomcat' instance
  has FIPS mode enabled on this operating system.

  REMINDER:  Don't forget to update the appropriate FIPS
             algorithms in server.xml in the 'pki-tomcat' instance.

  To check the status of the subsystem:
        systemctl status pki-tomcatd@pki-tomcat.service

  To restart the subsystem:
        systemctl restart pki-tomcatd@pki-tomcat.service

  The URL for the subsystem is:
        https://linux-86.hsmai.com:8443/tps

  PKI instances will be enabled upon system boot

==========================================================================
```

7. Confirm the presence of all subsystem certificates in the key database:

```
# certutil -L -d /etc/pki/pki-tomcat/alias -h <token_name>
```

```
[root@linux-86 ~]# certutil -L -d /etc/pki/pki-tomcat/alias -h TPA02

Certificate Nickname                                         Trust Attributes
                                                             SSL,S/MIME,JAR/XPI

Enter Password or Pin for "TPA02":
TPA02:auditSigningCert cert-pki-tomcat TPS                   u,u,Pu
TPA02:auditSigningCert cert-pki-tomcat TKS                   u,u,Pu
TPA02:auditSigningCert cert-pki-tomcat OCSP                  u,u,Pu
TPA02:ocspSigningCert cert-pki-tomcat OCSP                   u,u,u
TPA02:auditSigningCert cert-pki-tomcat KRA                   u,u,Pu
TPA02:storageCert cert-pki-tomcat KRA                        u,u,u
TPA02:transportCert cert-pki-tomcat KRA                      u,u,u
TPA02:Server-Cert cert-pki-tomcat                            u,u,u
TPA02:auditSigningCert cert-pki-tomcat CA                    u,u,Pu
TPA02:subsystemCert cert-pki-tomcat                          u,u,u
TPA02:ocspSigningCert cert-pki-tomcat CA                     u,u,u
TPA02:caSigningCert cert-pki-tomcat CA                       CTu,Cu,Cu
[root@linux-86 ~]#
```

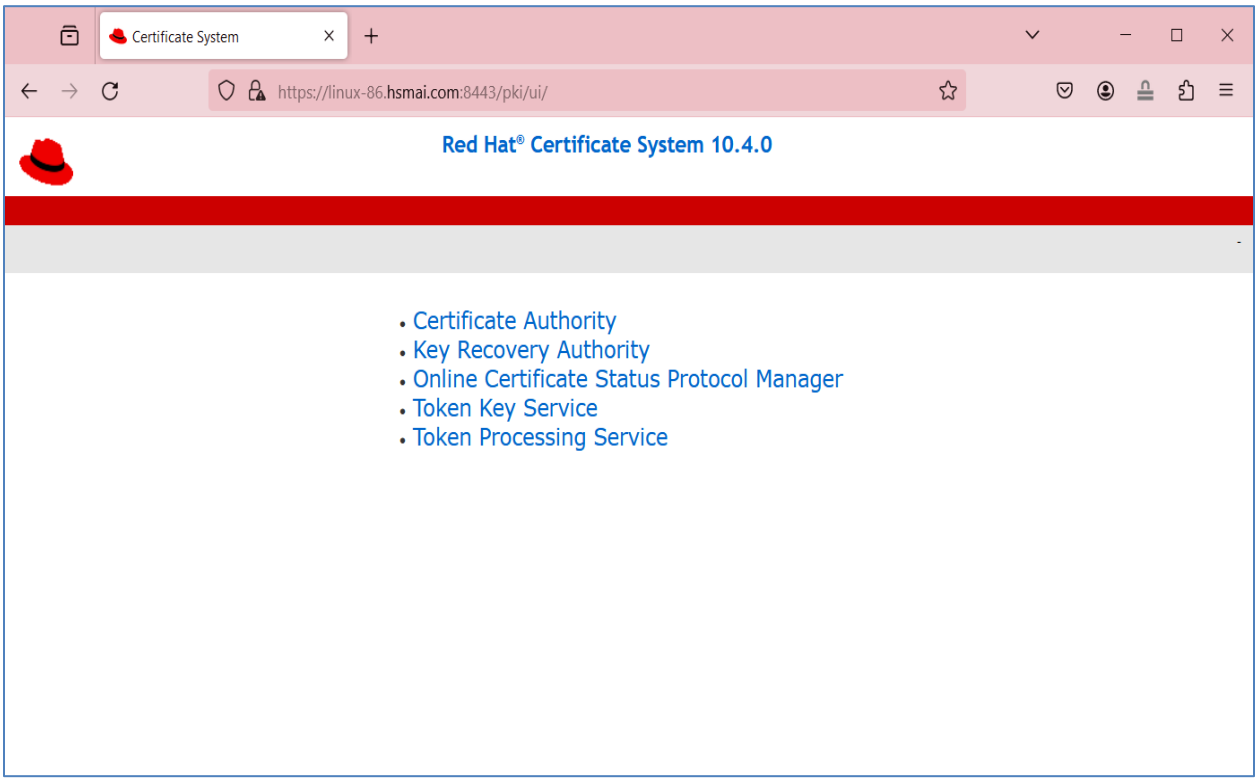
8. Verify that all keys are created on Luna HSM by checking the partition contents.

```
# /usr/safenet/lunaclient/bin/cmu list
```

```
[root@linux-86 ~]# /usr/safenet/lunaclient/bin/cmu list
Certificate Management Utility (64-bit) v10.7.0-255. Copyright (c) 2023 Thales Group. A
ll rights reserved.

Please enter password for token in slot 0 : ********

handle=148      label=auditSigningCert cert-pki-tomcat TPS
handle=154      label=auditSigningCert cert-pki-tomcat TPS
handle=153      label=
handle=116      label=auditSigningCert cert-pki-tomcat TKS
handle=147      label=auditSigningCert cert-pki-tomcat TKS
handle=146      label=
handle=128      label=auditSigningCert cert-pki-tomcat OCSP
handle=140      label=auditSigningCert cert-pki-tomcat OCSP
handle=139      label=
handle=126      label=ocspSigningCert cert-pki-tomcat OCSP
handle=132      label=ocspSigningCert cert-pki-tomcat OCSP
handle=131      label=
handle=115      label=auditSigningCert cert-pki-tomcat KRA
handle=125      label=auditSigningCert cert-pki-tomcat KRA
handle=124      label=
handle=106      label=storageCert cert-pki-tomcat KRA
handle=104      label=transportCert cert-pki-tomcat KRA
handle=118      label=storageCert cert-pki-tomcat KRA
handle=117      label=
handle=111      label=transportCert cert-pki-tomcat KRA
handle=110      label=
handle=100      label=Server-Cert cert-pki-tomcat
handle=93       label=auditSigningCert cert-pki-tomcat CA
handle=99       label=auditSigningCert cert-pki-tomcat CA
handle=98       label=
handle=56       label=subsystemCert cert-pki-tomcat
handle=92       label=subsystemCert cert-pki-tomcat
handle=90       label=
handle=88       label=
handle=87       label=
handle=34       label=ocspSigningCert cert-pki-tomcat CA
handle=70       label=ocspSigningCert cert-pki-tomcat CA
handle=63       label=
handle=9        label=caSigningCert cert-pki-tomcat CA
handle=69       label=caSigningCert cert-pki-tomcat CA
handle=64       label=
[root@linux-86 ~]#
```

9. Access the Red Hat Certificate Subsystem console using the following URL:

   https://<fully qualified domain name>:8443



This completes the Red Hat Certificate System integration with Luna HSM by securing all Subsystem core certificate keys on Luna HSM.

Contact Customer Support

If you encounter a problem during this integration, refer to the documentation. 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 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.