Cloudera Manager

Ranger KMS

Date published: 2024-12-10
Date modified: 2024-12-10

https://docs.cloudera.com/

Legal Notice

Contents

Ranger KMS overview

Apache Ranger Key Management Service (KMS) provides a centralized key management service that allows you to create, manage, and store encryption keys used for data encryption and decryption across various Hadoop ecosystem components. By integrating with Apache Ranger, it offers robust security features, including fine-grained authorization, auditing, and policies management for encryption keys.

Key features of Apache Ranger KMS include:

- Centralized key management

  Simplifies the management of encryption keys across your Hadoop ecosystem.
- Fine-grained authorization

  Allows administrators to define detailed access policies for encryption keys, ensuring only authorized users and applications can use them.
- Comprehensive auditing

  Tracks all key-related activities, providing detailed logs and reports for compliance and security audits.
- Seamless integration

  Works seamlessly with Hadoop Distributed File System (HDFS) encryption and other Hadoop ecosystem components, enhancing overall data security.

Using the Ranger Key Management Service

Ranger Key Management Service (KMS) can be accessed by logging into the Ranger web UI as the KMS administrator.

Role Separation

Ranger uses separate admin users for Ranger and Ranger KMS.

- The Ranger admin user manages Ranger access policies.
- The Ranger KMS admin user (keyadmin by default) manages access policies and keys for Ranger KMS, and has access to a different set of UI features than the Ranger admin user.

Using separate administrator accounts for Ranger and Ranger KMS separates encryption work (encryption keys and policies) from cluster management and access policy management.

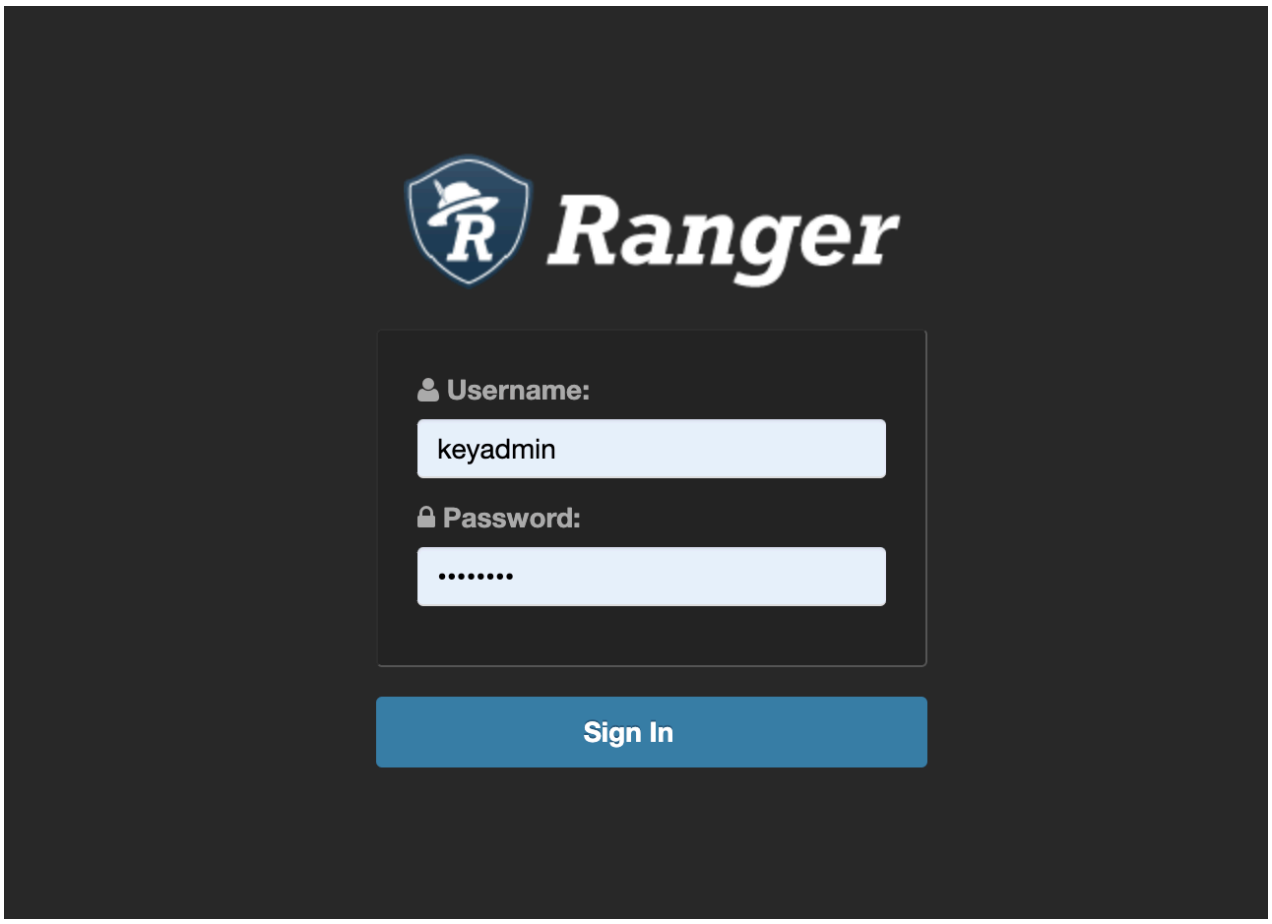
Note:

For more information about creating, deleting, listing, and rolling over existing keys using Ranger REST APIs, see https://ranger.apache.org/apidocs/resource_XKeyREST.html.

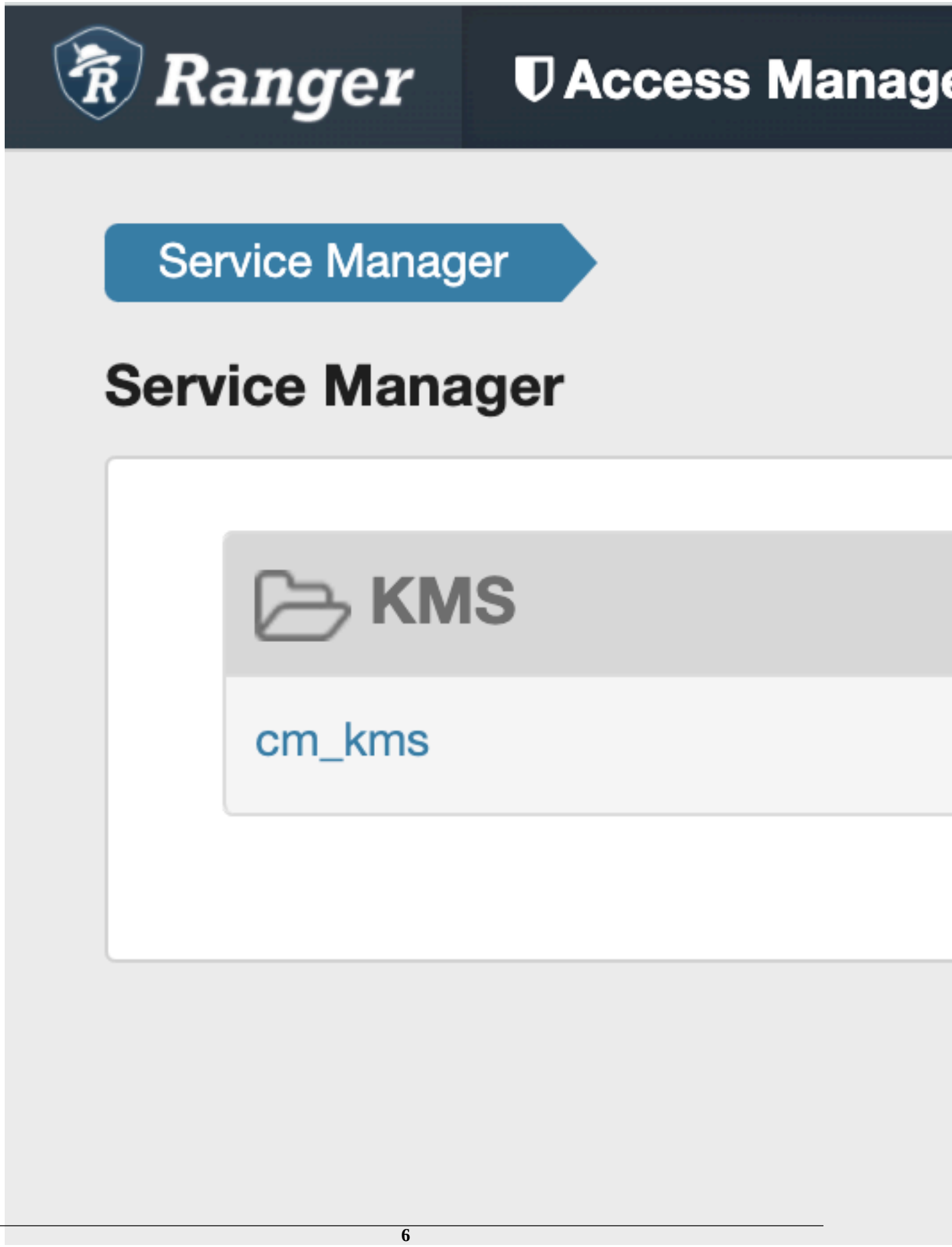
Accessing the Ranger KMS Web UI

How to access the Ranger Key Management Service (KMS) Web UI.

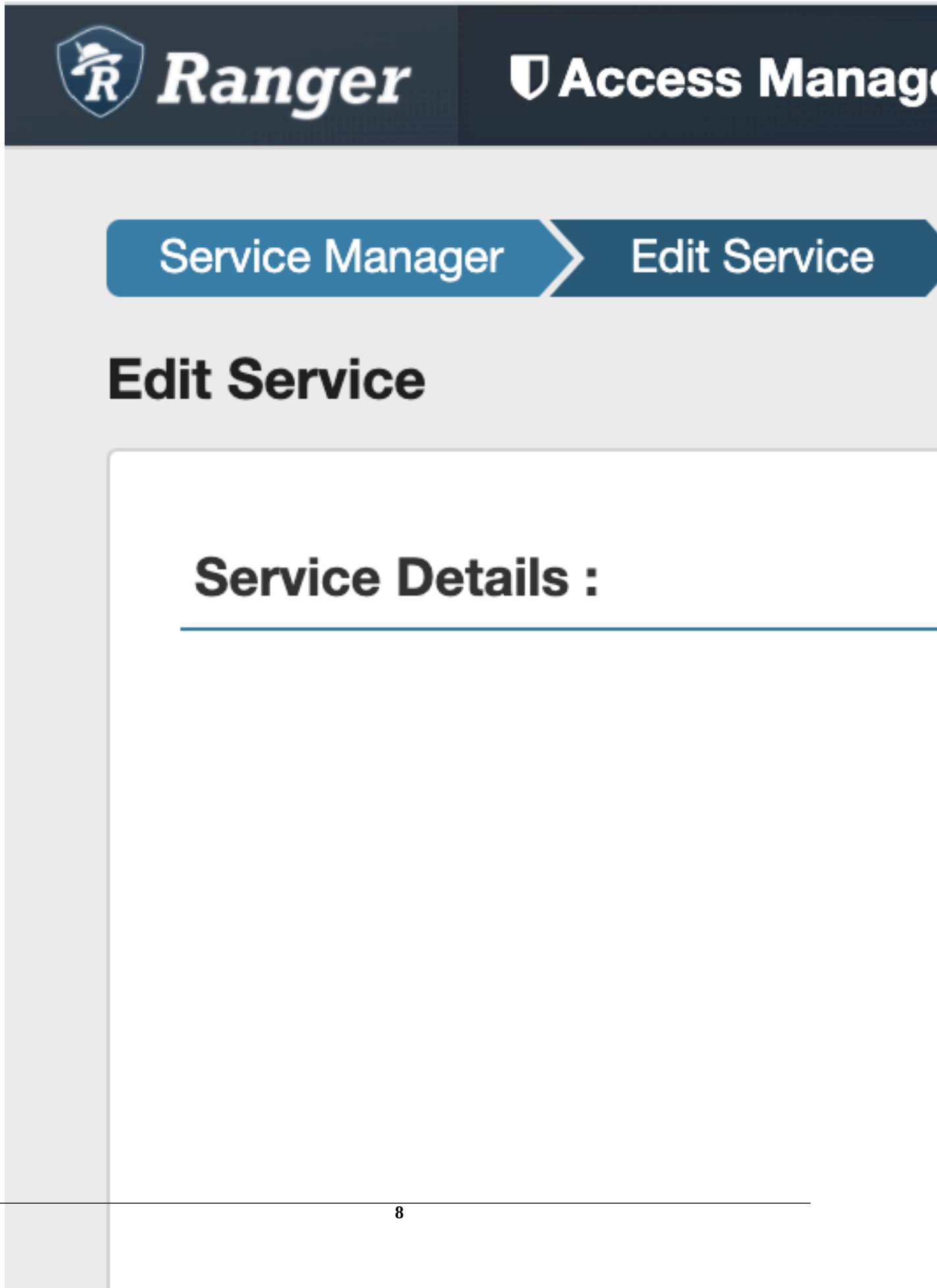
To access Ranger KMS, click the Ranger Admin Web UI link, enter your Ranger KMS admin user name and password, then click Sign In.



After logging in, the **Service Manager** page appears.


Ranger
Access Manage
Service Manager
Service Manager
KMS
cm_kms

To edit Ranger KMS repository properties, click the Edit icon for the service and update the settings on the **Edit Service** page.

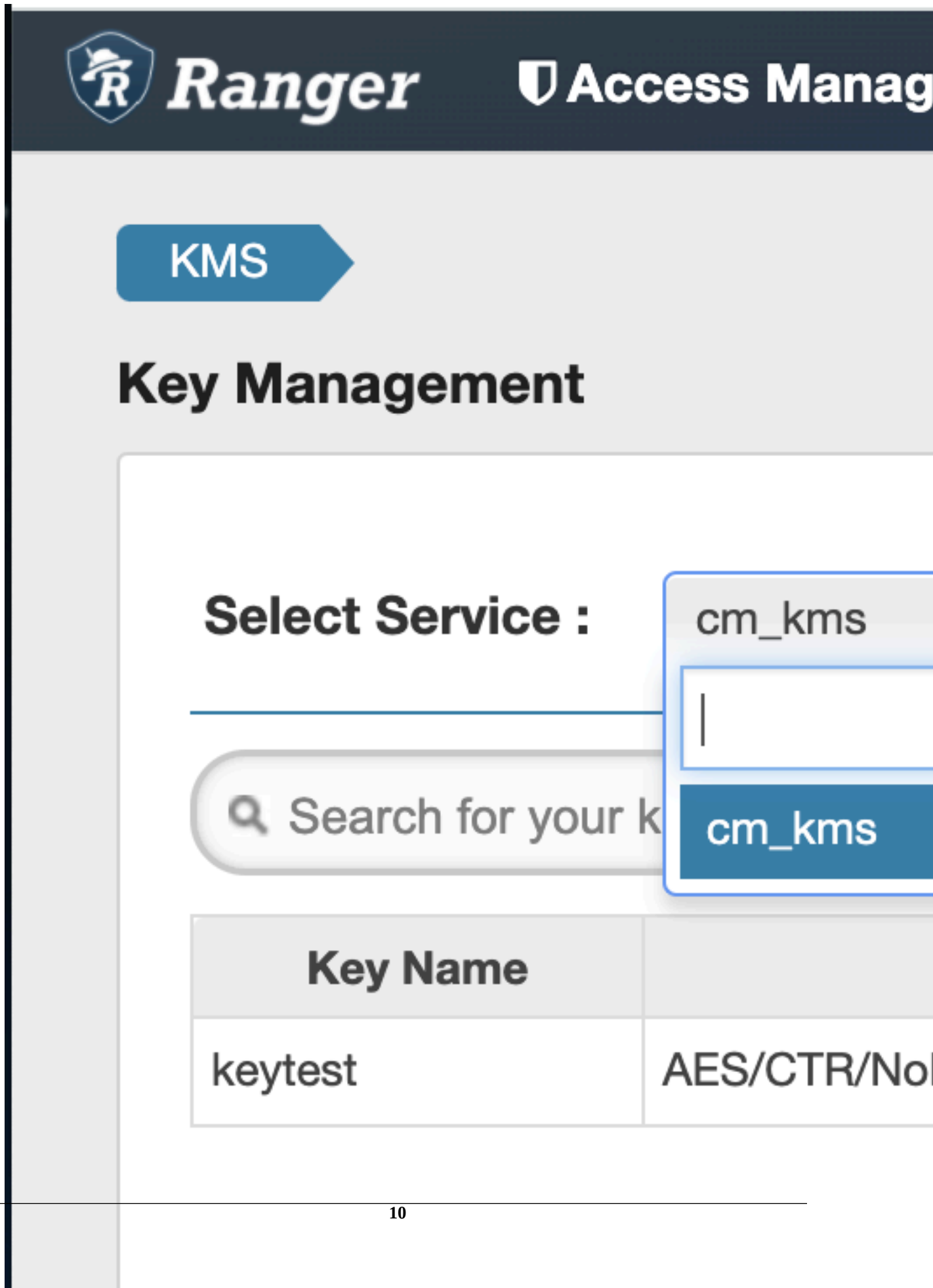

Ranger
Access Manag
Service Manager
Edit Service
Edit Service
Service Details :

List and Create Keys

How to list and create Ranger Key Management Service (KMS) keys.

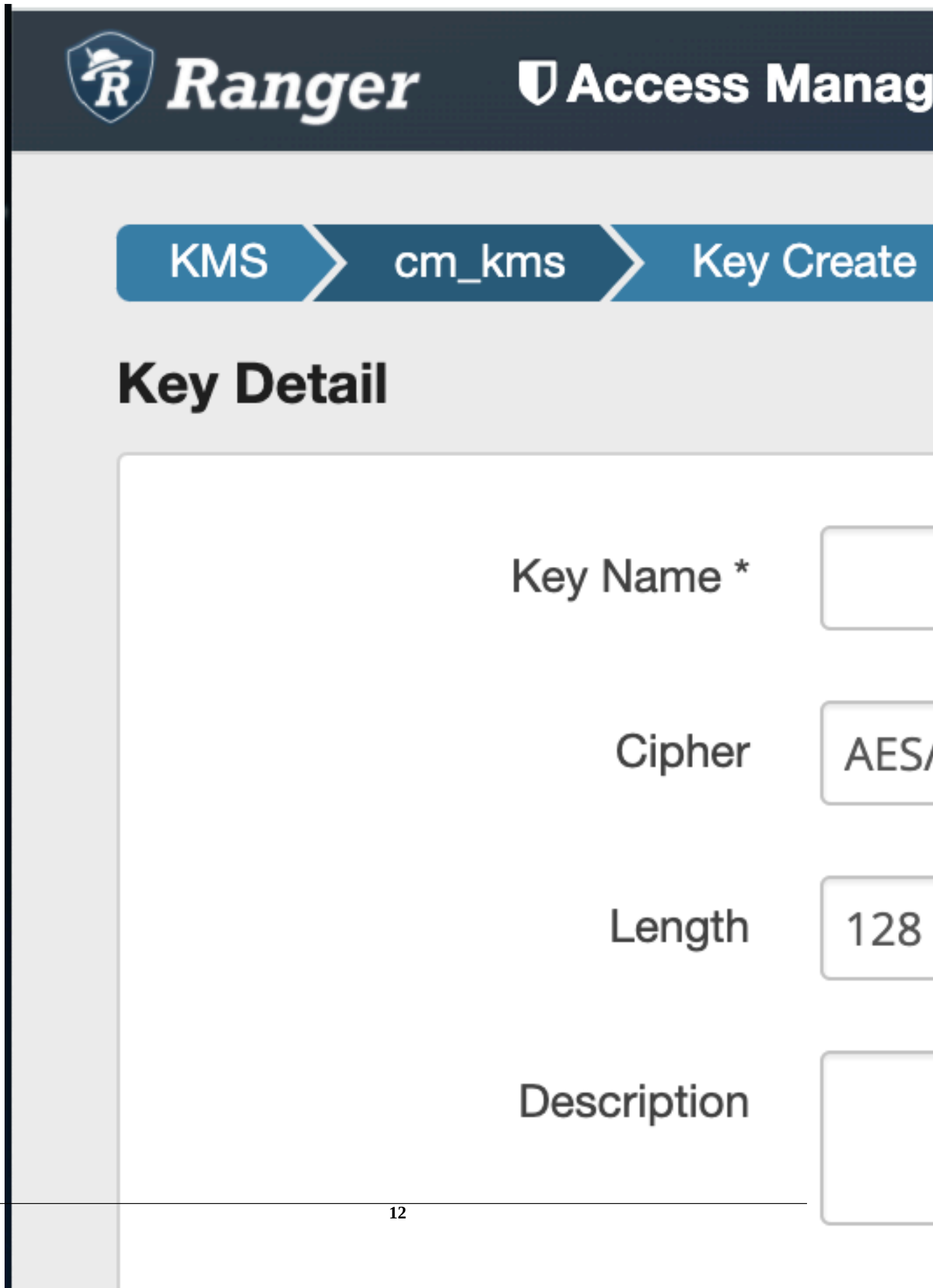
List existing keys

1. Log in to Ranger as the Ranger KMS admin user.
2. Click Encryption in the top menu to display the **Key Management** page.
3. Use the Select Service box to select a Ranger KMS service. The keys for the service appear.


Ranger
Access Manag
KMS
Key Management
Select Service :
cm_kms
cm_kms
Search for your k
Key Name
keytest
AES/CTR/No

Create a new key

1. Click Add New Key.
2. On the Key Detail page, add a valid key name.
3. Specify a cipher. Ranger KMS supports AES/CTR/NoPadding as the cipher suite.
4. Specify the key length: 128 or 256 bits.
5. Add other attributes as needed, then click Save.


Ranger
Access Manag
KMS
cm_kms
Key Create
Key Detail
Key Name *
Cipher
AES
Length
128
Description

Roll Over an Existing Key

How to roll over an existing Ranger Key Management Service (KMS) key.

About this task

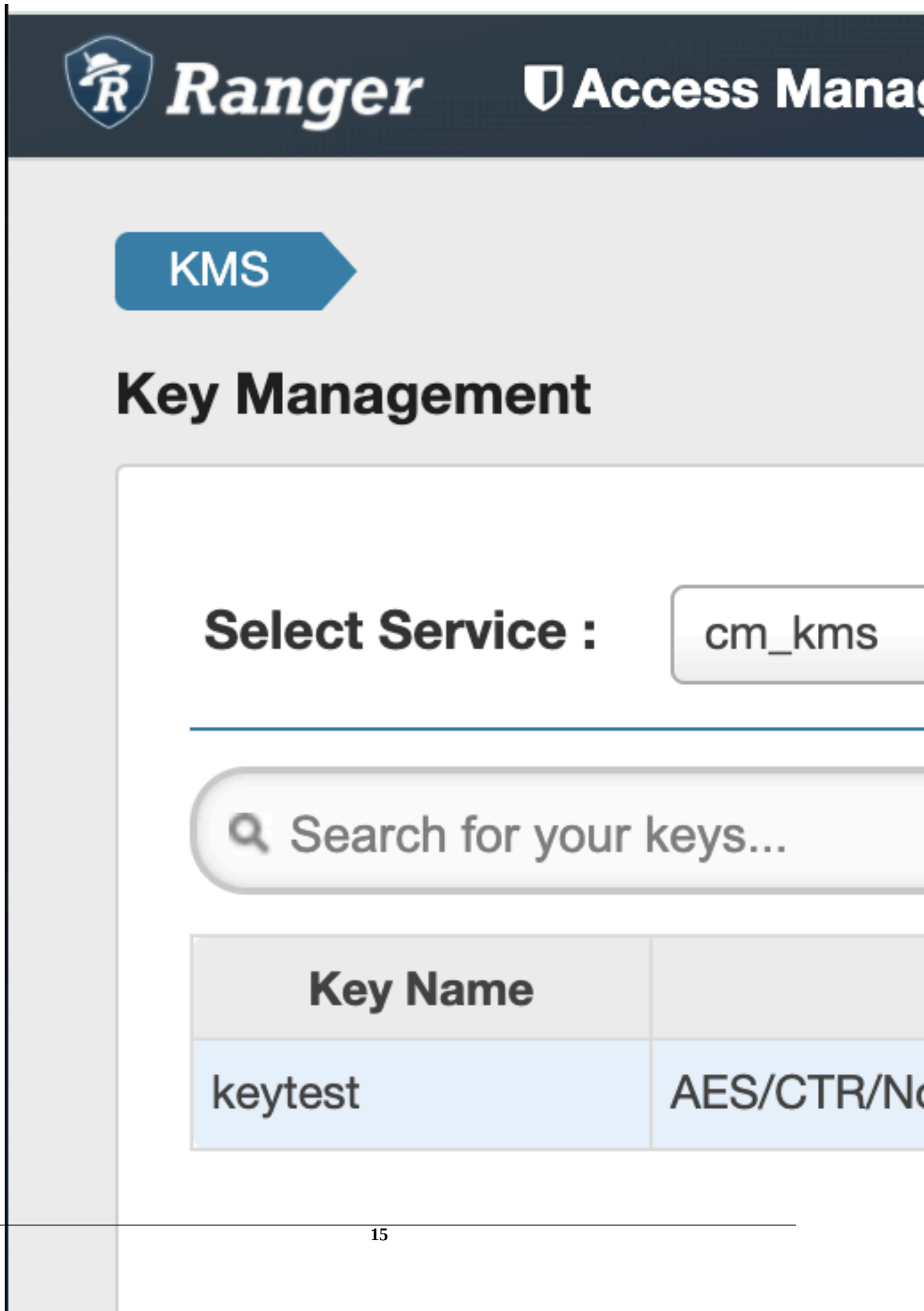
When you roll over (or rotate) a key, the key retains the same key name, but creates a new version of the key. This newly versioned key becomes the currentKey. After the key rotation, new files will have the file key encrypted by the current encryption zone (EZ) key for the encryption zone.

Note: Ranger KMS does not delete older versions of the key, as the older versions are used to decrypt the file keys that were encrypted with them.

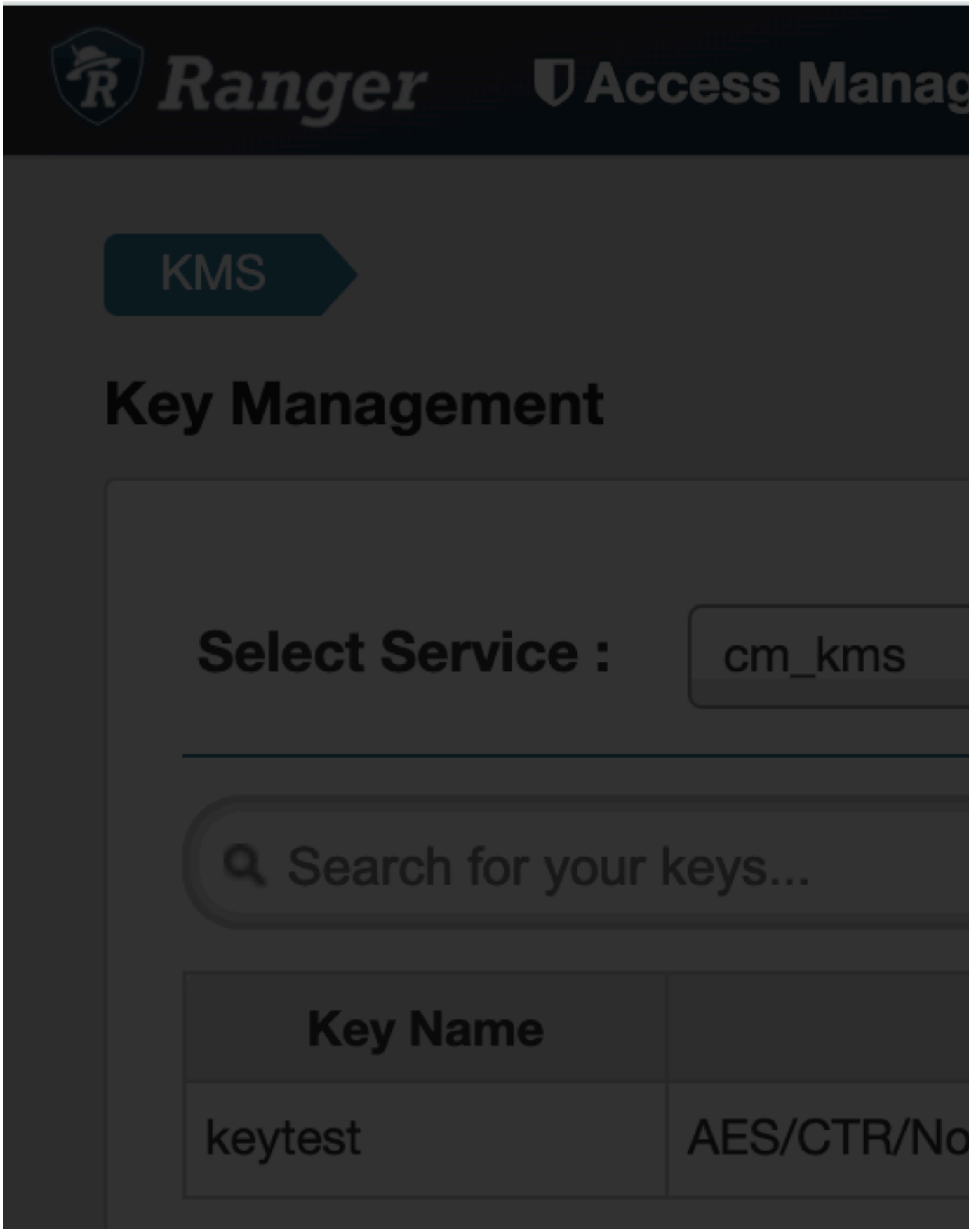
Procedure

1. Log in to Ranger as the Ranger KMS admin user, click Encryption in the top menu, then select a Ranger KMS service.

2. To rotate a key, click the Rollover icon for the key in the Action column.


Ranger
Access Manag
KMS
Key Management
Select Service :
cm_kms
Search for your keys...
Key Name
keytest
AES/CTR/N

3. Click OK on the confirmation pop-up.



Delete a Key

How to delete a Ranger Key Management Service (KMS) key.

About this task

Important:

Deleting a key associated with an existing encryption zone will result in data loss.

Note:

- Encryption zone keys should be deleted from the Ranger UI or Hadoop Command line.
- Encryption keys should NOT be deleted in the HSM before deleting from the Ranger UI or Hadoop command line.

Procedure

1. Log in to Ranger as the Ranger KMS admin user, click Encryption in the top menu, then select a Ranger KMS service.
2. Click on the Delete icon for the key in the Action column.
3. Click OK on the confirmation pop-up.

Securing the Key Management System (KMS)

Cloudera provides the following Key Mangement System (KMS) implementations: Ranger KMS with database and Ranger KMS with HSM. You can secure Ranger KMS using Kerberos, TLS/SSL communication, and access control lists (ACLs) for operations on encryption keys.

Cloudera Manager supports wizard-driven instructions for installing Ranger KMS with a database.

Enabling Kerberos Authentication for the KMS

You can use Cloudera Manager to enable Kerberos authentication for the KMS.

About this task

Minimum Required Role: Full Administrator

Procedure

1. Open the Cloudera Manager Admin Console and go to the KMS service.
2. Click Configuration.
3. Set the Authentication Type property to kerberos.
4. Click Save Changes.
5. Because Cloudera Manager does not automatically create the principal and keytab file for the KMS, you must run the Generate Credentials command manually.

   On the top navigation bar, go to Administration Security Kerberos Credentials and click Generate Missing Credentials

   Note: This does not create a new Kerberos principal if an existing HTTP principal exists for the KMS host.
6. Return to the home page by clicking the Cloudera Manager logo.
7. Click the ⏻ icon that is next to any stale services to invoke the cluster restart wizard.

8. Click Restart Stale Services.
9. Click Restart Now.
10. Click Finish.

Configuring TLS/SSL for the KMS

You must configure specific TLS/SSL properties associated with the KMS.

About this task

Minimum Required Role: Configurator (also provided by Cluster Administrator, Full Administrator)

Procedure

1. Go to the KMS service.
2. Click Configuration.
3. In the Search field, type TLS/SSL to show the KMS TLS/SSL properties (in the Key Management Server Default Group Security category).
4. Edit the following TLS/SSL properties according to your cluster configuration.

Property	Description
Enable TLS/SSL for Key Management Server	Encrypt communication between clients and Key Management Server using Transport Layer Security (TLS) (formerly known as Secure Socket Layer (TLS/SSL)).
Key Management Server TLS/SSL Server JKS Keystore File Location	The path to the TLS/SSL keystore file containing the server certificate and private key used for TLS/SSL. Used when Key Management Server is acting as a TLS/SSL server. The keystore must be in JKS format.
Key Management Server TLS/SSL Server JKS Keystore File Password	The password for the Key Management Server JKS keystore file.
Key Management Server Proxy TLS/SSL Certificate Trust Store File	The location on disk of the truststore, in .jks format, used to confirm the authenticity of TLS/SSL servers that Key Management Server Proxy might connect to. This is used when Key Management Server Proxy is the client in a TLS/SSL connection. This truststore must contain the certificates used to sign the services connected to. If this parameter is not provided, the default list of well-known certificate authorities is used instead.
Key Management Server Proxy TLS/SSL Certificate Trust Store Password	The password for the Key Management Server Proxy TLS/SSL Certificate Trust Store File. This password is not required to access the truststore; this field can be left blank. This password provides optional integrity checking of the file. The contents of truststores are certificates, and certificates are public information.

5. Click Save Changes.
6. Return to the home page by clicking the Cloudera Manager logo.
7. Click the ⏻ icon that is next to any stale services to invoke the cluster restart wizard.
8. Click Restart Stale Services.
9. Click Restart Now.
10. Click Finish.

Migrating Ranger Key Management Server Role Instances to a New Host

You can move the Ranger Admin and Ranger KMS database role instances for an existing Ranger KMS service from one host to another, using Cloudera Manager.

Note: This procedure applies only to the Ranger Key Management Server role instances.

In some cases–for example, after upgrading your servers–you may want to migrate a Ranger KMS Server role instance to a new host. This procedure describes how to move a Ranger KMS role instance from an existing cluster host to another cluster host.

Migrate the Ranger Admin role instance to a new host

To migrate the Ranger KMS role instances to a new host, first migrate the Ranger Admin role instance.

Procedure

1. Add a new Ranger Admin role instance on another node.

   Note: If you enabled manual SSL on this cluster, you must update the SSL configs when adding a new role.

2. Start the new Ranger Admin role instance.
3. Stop the initial Ranger Admin instance.
4. Delete the initial Ranger Admin instance.
5. Restart the cluster.

   Restarting the cluster removes the "stale" changes.

Migrate the Ranger KMS db role instance to a new host

After migrating the Ranger Admin role instance to a new host, migrate the Ranger KMS db role instance.

About this task

Only if Ranger KMS has a backend database for key storage, should you migrate the Ranger KMS db role instance.

Procedure

1. Add a new Ranger KMS db role instance on another node.

   Note: If you enabled manual SSL on this cluster, you must update the SSL configs when adding a new role.

2. Start the new Ranger KMS db role instance.
3. Stop the initial Ranger KMS db instance.
4. Delete the initial Ranger KMS db instance.
5. Restart the cluster.
6. Login to Ranger Admin UI using keyadmin credentials.
7. Update the cm_kms service to use the kms url that refers to the new hostname.

Working with an HSM

How to integrate Cloudera Data Encryption components to provide enterprise data encryption solutions.

Ranger Key Mangement System (KMS)

Consists of Ranger KMS Ranger KMS providing enterprise-grade key management with a backend database that provides key storage.

1. Install Ranger KMS using CM Administration Security HDFS Encryption Wizard .

2. Install a seperate database to store keys.

   For more information, see related links.

Ranger KMS and HSM

Consists of Ranger KMS and database integrated with a backend hardware security module (HSM). In this solution, Ranger KMS provides enterprise-grade key management, HSM provides encryption zone key protection. HSM stores only the encryption master key.

1. Install Ranger KMS using CM Administration Security HDFS Encryption Wizard .
2. Install a seperate database to store keys.
3. Obtain and integrate one of the following hardware security modules (HSM) supplied by a vendor.
   - Luna 7
   - CipherTrust
   - GCP Cloud HSM
   - Azure Key Vault

   For more information, see related links.

Set up Luna 7 HSM for Ranger KMS

How to integrate Cloudera Ranger Key Management System (KMS) software with the Luna 7 HSM appliance supplied by SafeNet.

About this task

This task describes how to set up the Luna 7 hardware security moudule (HSM) supplied by SafeNet. The process inlcudes setting up Luna 7 HSM on a client (host) and using Cloudera Manager to add configuration properties that enable Ranger KMS and Luna 7 HSM to interact.

Note: Ranger KMS should have separate HSM partition per KMS setup. If the same HSM partition is configured in the second KMS cluster, it overwrites the existing master key in the HSM partition.

Before you begin

You must:

- Acquire the Luna 7 HSM from SafeNet.
- If the Luna HSM module is configured for FIPS mode, you must add the following additional configuration option to the Luna client:

  ```
  /usr/safenet/lunaclient/bin/configurator setValue -s Misc -e RSAKeyGenMe
  chRemap -v 1
  ```

- Have both Ranger KMS and a backend database to store keys installed in your environment.

See related topics for more information about installing Ranger KMS and a database to store keys.

Procedure

Set Up the Luna 7 Client

1. Download Luna 7 client on the host where Ranger KMS service resides.

   ```
   610-013144-006_SW_Client_SDK_SafeNet_HSM_7.3.0_Linux_RevA.tar
   ```

2. Untar the Luna 7 client.

```
tar -xf 610-013144-006_SW_Client_SDK_SafeNet_HSM_7.3.0_Linux_RevA.tar
```

the LunaClient_7.3.0-165_Linux/ folder gets created.

3. Navigate to the Luna client folder.

```
cd LunaClient_7.3.0-165_Linux/64/
```

4. In the Luna client folder, install Luna products and components.

```
bash install.sh
```

a) At the (y/n) prompt, choose y.

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

b) At the Products prompt, choose Luna products to be installed:

- [1]: Luna Network HSM
- [2]: Luna PCIe HSM
- [3]: Luna USB HSM
- [4]: Luna Backup HSM
- [N|n]: Next
- [Q|q]: Quit

Enter selection: 1, then enter selection n.

c) At the Components prompt, choose Luna Components to be installed

- [1]: Luna SDK
- [2]: Luna JSP (Java)
- [3]: Luna JCProv (Java)
- [B|b]: Back to Products selection
- [I|i]: Install
- [Q|q]: Quit

Enter selection: i, then enter selection Q.
Enter selection: 1,2,and 3 then type i.

5. Navigate to the Luna SA command directory.

```
cd /usr/safenet/lunaclient/bin
```

You should see the following:

ls

ckdemo cmu common configurator lunacm multitoken openssl.cnf plink pscp

salogin uninstall.sh vtl

6. Add a user to the hsmusers group.

```
sudo gpasswd --add kms hsmusers
```

7. Copy the Luna appliance server certificate to the client.

```
scp admin@<LunaBoxHostname>:server.pem
```

```
 scp e02paruser115@elab2.safenet-inc.com:server.pem .
(grant permission chmod 777 and chown kms:kms)
```

```
The authenticity of host 'elab2.safenet-inc.com (192.43.161.62)' can't be
established.
ECDSA key fingerprint is SHA256:Lz36zjWHh3BMtI9TVHUBGoHffxgA6azFtPSGRBC
kiYU.

Are you sure you want to continue connecting (yes/no)? yes

Warning: Permanently added 'elab2.safenet-inc.com,192.43.161.62' (ECDSA) t
o the list of known hosts.
e02paruser115@elab2.safenet-inc.com's password:  SafeNetPSG95 (given by
 the luna hsm team)
  press enter
server.pem                                        100% 1155      1.1KB/s
           00:00
```

8. Confirm that server.pem is added to the client.

```
ls
```

```
ckdemo cmu common configurator lunacm multitoken openssl.cnf plink pscp
 salogin server.pem uninstall.sh vtl
```

server.pem is added

9. As the KMS user, register the server with the client.

```
su -l kms
./vtl addServer -n <LunaBoxHostname> -c server.pem
```

```
 ./vtl addserver -n elab2.safenet-inc.com -c server.pem
```

vtl (64-bit) v7.3.0-165. Copyright (c) 2018 SafeNet. All rights reserved.

New server elab2.safenet-inc.com successfully added to server list.

10. Generate a client certificate.

```
./vtl  createCert -n  <ClientHostname>
```

```
 ./vtl createcert -n e02paruser115
```

vtl (64-bit) v7.3.0-165. Copyright (c) 2018 SafeNet. All rights reserved.

Private Key created and written to: /usr/safenet/lunaclient/cert/client/e02paruser115Key.pem. Certificate created and written to: /usr/safenet/lunaclient/cert/client/e02paruser115.pem .

(grant permission chmod 777 and chown kms:kms)

11. Copy the client certificate to the server.

```
scp /usr/safenet/lunaclient/cert/client/<ClientHostname>.pem admin@<Luna
BoxHostname>:
```

```
scp /usr/safenet/lunaclient/cert/client/e02paruser115.pem e02paruser115@
elab2.safenet-inc.com:
e02paruser115@elab2.safenet-inc.com's password: SafeNetPSG95
e02paruser115.pem                                                 100%
1172   201.7KB/s   00:00
```

12. Login to luna hsm.

```
ssh admin@<lunaboxhostname>
```

```
ssh e02paruser115@elab2.safenet-inc.com
e02paruser115@elab2.safenet-inc.com's password: SafeNetPSG95
Last login: Fri Jul 19 03:59:38 2019 from 114.143.87.94
Luna Network HSM Command Line Shell v7.3.0-165.
Copyright (c) 2018 SafeNet. All rights reserved.
[elab2] lunash:>
```

13. Register the client with the server, then assign the client to a server partition.

```
lunash:> client register -client <ClientHostname> -hostname <ClientHostn
ame>
```

```
client register -client e02paruser115 -hostname e02paruser115
```

14. Check the existing partitions.

```
lunash:> partition list
```

```
lunash:> partition list
                        Storage (bytes)
                        ----------------------------
                        Partition            Name                   Obj
ects   Total    Used    Free
                        ================================================
============================
                        1254277068838        elab2par058
 0  325896       0  325896
```

15. Assign client to the partition.

```
lunash:> client assignPartition -client <ClientHostname> -partition <Gat
ewayPartition>
```

```
lunash:> client assignPartition -client e02paruser115 -partition elab2pa
r058
```

16. client show -client e02paruser115

```
ClientID:     e02paruser115
Hostname:     e02paruser115
Partitions:   "elab2par058"
```

17. Log out from the Luna HSM.

```
lunash:> exit
```

18. Set the read permissions for the certificate files in the following directories.

Note: Make sure to log in as root user.

```
chmod a+r /usr/safenet/lunaclient/cert/server/*.pem
chmod a+r /usr/safenet/lunaclient/cert/client/*.pem
```

```
(grant permission chmod 777 and chown kms:kms to above .pem files)
```

19. Verify that the client is connected to its assigned partition.

Note: Make sure to log in as kms user.

```
cd /usr/safenet/lunaclient/bin/
./vtl verify
```

```
[root@os-mv-711-1 bin]# ./vtl verify
vtl (64-bit) v7.3.0-165. Copyright (c) 2018 SafeNet. All rights reserved.

The following Luna SA Slots/Partitions were found:

Slot            Serial #                   Label
==========================================================================
===
0                1254277068842             elab2par115
```

20. ./lunacm

```
./lunacm
```

```
[root@os-mv-711-1 bin]# ./lunacm
lunacm (64-bit) v7.3.0-165. Copyright (c) 2018 SafeNet. All rights rese
rved.

Available HSMs:

Slot ID ->           0
Label ->             elab2par115
Serial # ->          1254277068842
Model ->             LunaSA 7.3.0
Firmware version ->  7.3.0
Configuration ->     Luna User Partition with SD (PW) Key Export with Cl
eaning Mode
Slot Description ->  Net Token Slot

Current Slot ID:  0
```

21. role login -n co

```
enter password: pwd123
```

22. par con

If Master Key RangerKMSKey exists, then the following will be visible:

```
lunacm:>par con
    The 'Crypto Officer' is currently logged in.
    Looking for objects accessible to the 'Crypto Officer'.

    Object List:

    Label:             RangerKMSKey
    Handle:            131
    Object Type:       Symmentric Key
    Object UID:        ba8e00002e00000554380800
    Number of Objects: 1
```

```
Command Result: No Error
Else
lunacm:>par con

    The 'Crypto Officer' is currently logged in.
    Looking for objects accessible to the 'Crypto Officer'.

    No objects viewable to 'Crypto Officer' are currently stored in the
 partition.
Command Result: No Error
```

23. Navigate to the following directory on the Gateway.

```
# cd /usr/safenet/lunaclient/jsp/lib/
(grant permission chmod 777 and chown kms:kms to all the at this location)
```

24. Copy the Luna .JAR files over to the Gateway.

```
cp libLunaAPI.so Luna*.jar {JAVA_HOME}/jre/lib/ext/
```

```
cp libLunaAPI.so Luna*.jar /usr/java/jdk1.8.0_232-cloudera/jre/lib/ext
```

25. Set the file permissions for the JDK library as follows:

```
chmod a+r {JAVA_HOME}/jre/lib/
```

```
chmod a+r /usr/java/jdk1.8.0_232-cloudera/jre/lib/
```

26. Open the following file in a text editor:

```
vim {JAVA_HOME}/jre/lib/security/java.security
```

```
vim /usr/java/jdk1.8.0_232-cloudera/jre/lib/security/java.security
```

a) Add these two lines:

```
security.provider.6=com.safenetinc.luna.provider.LunaProvider
com.safenetinc.luna.provider.createExtractableKeys=true
```

replacing the line highlighted below:

```
Java SDK/JRE 1.6.x or 1.7.x installation to read as follows:
security.provider.1=sun.security.provider.Sun
security.provider.2=sun.security.rsa.SunRsaSign
security.provider.3=sun.security.ec.SunEC
security.provider.4=com.sun.net.ssl.internal.ssl.Provider
security.provider.5=com.sun.crypto.provider.SunJCE
security.provider.6=com.safenetinc.luna.provider.LunaProvider
security.provider.7=sun.security.jgss.SunProvider
security.provider.8=com.sun.security.sasl.Provider
security.provider.9=org.jcp.xml.dsig.internal.dom.XMLDSigRI
security.provider.10=sun.security.smartcardio.SunPCSC
```

27. Set the file permissions for the Luna client as follows:

```
chmod -R 777 /usr/safenet
chown kms:kms
```

Set KMS Configuration Properties.

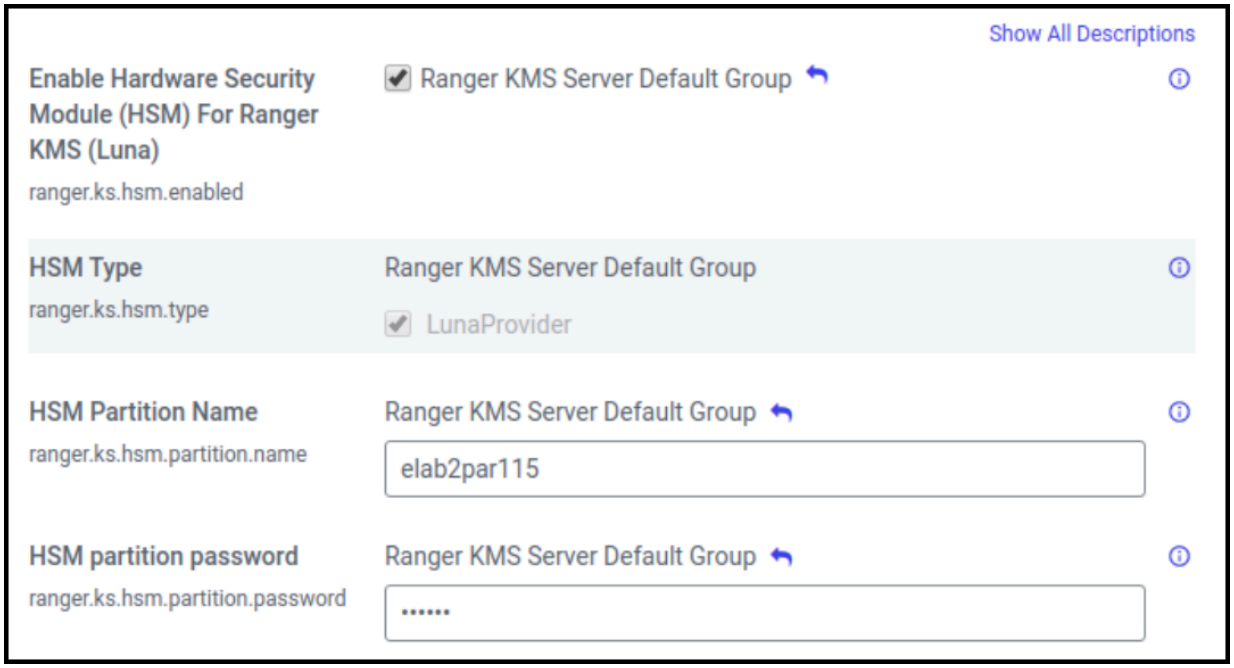
28. In Cloudera Manager Ranger KMS Configs edit the following properties:

Note: For CM-7.1.1 and CM-7.1.2 you must add properties to the dbks-site.xml, also known as Ranger KMS Server Advanced Configuration Snippet (Safety Valve) for conf/dbks-site.xml .

```
ranger.ks.hsm.type = LunaProvider
ranger.ks.hsm.enabled = true
ranger.ks.hsm.partition.name=elab2par115
ranger.ks.hsm.partition.password=pwd123
(CM-7.1.1 & CM-7.1.2 password will be in plain text)
```

Note: For CM-7.1.3 and higher, update the configuration as shown in:

Figure 1: Adding Ranger KMS Configuration for Luna 7 HSM



29. Restart Ranger KMS from Cloudera Manager.

30. Login to Luna client and validate whether the master key is successfully created.

```
cd /usr/safenet/lunaclient/bin/
./lunacm
role login -n co

enter password: pwd123

par con
```

```
lunacm:>par con
    The 'Crypto Officer' is currently logged in.
    Looking for objects accessible to the 'Crypto Officer'.

    Object List:

    Label:              RangerKMSKey
    Handle:             131
```

```
    Object Type:       Symmentric Key
    Object UID:        ba8e00002e00000554380800
    Number of Objects: 1

Command Result: No Error
```

Results

Ranger KMS is successfully started.

What to do next

You can now create Encryption zone keys using hadoop command or from Ranger UI using credentials of keyadmin user.

Set up Luna 10.5 HSM Client for Ranger KMS

How to integrate Cloudera Ranger Key Management System (KMS) software with the Luna 10.5 HSM appliance supplied by SafeNet.

About this task

This task describes how to set up the Luna 10.5 hardware security moudule (HSM) supplied by SafeNet. The process inlcudes setting up Luna 10.5 HSM on a client (KMS host) and using Cloudera Manager to add configuration properties that enable Ranger KMS and Luna 10.5 HSM to interact.

Note: Ranger KMS should have separate HSM partition per KMS setup. If the same HSM partition is configured in the second KMS cluster, it overwrites the existing master key in the HSM partition.

Before you begin

You must:

- Acquire the Luna v10.5 client, HSM Software Version v7.3.0, and HSM Firmware v7.3.0 from SafeNet.
- Have both Ranger KMS and a backend database to store keys installed in your environment.

See related topics for more information about installing Ranger KMS and a database to store keys.

Procedure

Set Up the Luna 10.5 Client

1. Download Luna 10.5 client on the host where Ranger KMS service resides.

```
610-000397-006_SW_Linux_Luna_Client_V10.5.0_RevA.tar
```

2. Untar the Luna 10.5 client.

```
tar -xf 610-000397-006_SW_Linux_Luna_Client_V10.5.0_RevA.tar
```

   The LunaClient_10.5.0-*_Linux/ folder gets created.

3. Navigate to the Luna client folder.

```
cd LunaClient_10.5.0-*_Linux/64/
```

4. In the Luna client folder, install Luna products and components.

```
bash install.sh
```

   a) At the (y/n) prompt, choose y.

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

   b) At the Products prompt, choose Luna products to be installed:

      - [1]: Luna Network HSM
      - [2]: Luna PCIe HSM
      - [3]: Luna USB HSM
      - [4]: Luna Backup HSM
      - [N|n]: Next
      - [Q|q]: Quit

      Enter selection: 1, then enter selection n.

   c) At the Components prompt, choose Luna Components to be installed

      - [1]: Luna SDK
      - [2]: Luna JSP (Java)
      - [3]: Luna JCProv (Java)
      - [B|b]: Back to Products selection
      - [I|i]: Install
      - [Q|q]: Quit

      Enter selection: 1,2,and 3 then type i.

5. Navigate to the Luna SA command directory.

```
cd /usr/safenet/lunaclient/bin
```

   You should see the following:

   ls

   ckdemo cmu common configurator lunacm multitoken openssl.cnf plink pscp

   salogin uninstall.sh vtl

6. Add a user to the hsmusers group.

```
sudo gpasswd --add kms hsmusers
```

7. Copy the Luna appliance server certificate to the client.

```
scp admin@<LunaBoxHostname>:server.pem
```

   Example :

```
scp e02paruser115@elab2.safenet-inc.com:server.pem .
(grant permission chmod 777 and chown kms:kms)
The authenticity of host 'elab2.safenet-inc.com (192.43.161.62)' can't be
 established.
ECDSA key fingerprint is SHA256:Lz36zjWHh3BMtI9TVHUBGoHffxgA6azFtPSGRBCk
iYU.

Are you sure you want to continue connecting (yes/no)? yes

Warning: Permanently added 'elab2.safenet-inc.com,192.43.161.62' (ECDSA) t
o the list of known hosts.
```

```
e02paruser115@elab2.safenet-inc.com's password:  SafeNetPSG95 (given by
 the luna hsm team)
  press enter
server.pem                                       100% 1155      1.1KB/s
           00:00
```

8. Confirm that server.pem is added to the client.

```
ls
```

Example:

```
ckdemo cmu common configurator lunacm multitoken openssl.cnf plink pscp
 salogin server.pem uninstall.sh vtl
```

server.pem is added

9. As the KMS user, register the server with the client.

```
./vtl addServer -n <LunaBoxHostname> -c server.pem
```

Example :

```
 ./vtl addserver -n elab2.safenet-inc.com -c server.pem
```

The new server elab2.safenet-inc.com is successfully added to server list.

10. Generate a client certificate.

```
./vtl  createCert -n  <ClientHostname>
```

Example :

```
 ./vtl createcert -n e02paruser115
```

Private Key created and written to: /usr/safenet/lunaclient/cert/client/e02paruser115Key.pem. Certificate created and written to: /usr/safenet/lunaclient/cert/client/e02paruser115.pem .

(grant permission chmod 777 and chown kms:kms)

11. Copy the client certificate to the server.

```
scp /usr/safenet/lunaclient/cert/client/<ClientHostname>.pem admin@<Luna
BoxHostname>:
```

Example :

```
scp /usr/safenet/lunaclient/cert/client/e02paruser115.pem e02paruser115@
elab2.safenet-inc.com:
e02paruser115@elab2.safenet-inc.com's password: SafeNetPSG95
e02paruser115.pem                                                 100%
1172   201.7KB/s   00:00
```

12. Login to luna hsm.

```
ssh admin@<lunaboxhostname>
```

Example :

```
ssh e02paruser115@elab2.safenet-inc.com
e02paruser115@elab2.safenet-inc.com's password: SafeNetPSG95
```

```
[elab2] lunash:>
```

13. Register the client with the server, then assign the client to a server partition.

```
lunash:> client register -client <ClientHostname> -hostname <ClientHostn
ame>
```

Example :

```
client register -client e02paruser115 -hostname e02paruser115
```

14. Check the existing partitions.

```
lunash:> partition list
```

Example:

```
lunash:> partition list
                        Storage (bytes)
                        ----------------------------
                        Partition            Name                   Obj
ects   Total    Used    Free
                        ================================================
============================
                        1254277068838        elab2par058
 0  325896       0  325896
```

15. Assign client to the partition.

```
lunash:> client assignPartition -client <ClientHostname> -partition <Gat
ewayPartition>
```

Example :

```
lunash:> client assignPartition -client e02paruser115 -partition elab2pa
r058
```

16. client show -client e02paruser115
Example:

```
ClientID:     e02paruser115
Hostname:     e02paruser115
Partitions:   "elab2par058"
```

17. Log out from the Luna HSM.

```
lunash:> exit
```

18. Set the read permissions for the certificate files in the following directories.

Note: Make sure to log in as root user.

Example :

```
chmod a+r /usr/safenet/lunaclient/cert/server/*.pem
chmod a+r /usr/safenet/lunaclient/cert/client/*.pem
(grant permission chmod 777 and chown kms:kms to above .pem files)
```

19. Verify that the client is connected to its assigned partition.

Note: Make sure to log in as kms user.

```
cd /usr/safenet/lunaclient/bin/
./vtl verify
```

```
[root@os-mv-711-1 bin]# ./vtl verify


The following Luna SA Slots/Partitions were found:

Slot            Serial #                   Label
==========================================================================
==
0                1254277068842             elab2par115
```

Troubleshooting : If you get following error : Application "vtl" has detected "locale::facet::_S_create_c_locale name not valid" , then

```
export LC_ALL="C"
```

and re-execute the command.

20. ./lunacm

```
./lunacm
```

```
[root@os-mv-711-1 bin]# ./lunacm


Available HSMs:

Slot ID ->           0
Label ->             elab2par115
Serial # ->          1254277068842
Model ->             LunaSA 7.3.0
Firmware version ->  7.3.0
Configuration ->     Luna User Partition with SD (PW) Key Export with Cl
eaning Mode
Slot Description ->  Net Token Slot

Current Slot ID:  0
```

21. role login -n co

```
enter password: passwrd123
```

22. par con

If Master Key RangerKMSKey exists, then the following will be visible:

```
lunacm:>par con
    The 'Crypto Officer' is currently logged in.
    Looking for objects accessible to the 'Crypto Officer'.

    Object List:

    Label:             RangerKMSKey
    Handle:            131
```

```
    Object Type:       Symmentric Key
    Object UID:        ba8e00002e00000554380800
    Number of Objects: 1

Command Result: No Error
Else
lunacm:>par con

    The 'Crypto Officer' is currently logged in.
    Looking for objects accessible to the 'Crypto Officer'.

    No objects viewable to 'Crypto Officer' are currently stored in the
 partition.
Command Result: No Error
```

23. Navigate to the following directory on the Gateway.

```
# cd /usr/safenet/lunaclient/jsp/lib/
(grant permission chmod 777 and chown kms:kms to all the at this location)
```

24. Copy the Luna .JAR files over to the Gateway.

For JDK 8:

```
cp libLunaAPI.so Luna*.jar {JAVA_HOME}/jre/lib/ext/
```

Example:

```
cp libLunaAPI.so Luna*.jar /usr/java/jdk1.8.0_232-cloudera/jre/lib/ext
```

For JDK 11:

```
cp libLunaAPI.so Luna*.jar {JAVA_HOME}/lib
```

Example:

```
cp libLunaAPI.so Luna*.jar /usr/java/default/lib/
```

25. Set the file permissions for the JDK library as follows:

```
chmod a+r {JAVA_HOME}/jre/lib/
```

Example :

```
chmod a+r /usr/java/jdk1.8.0_232-cloudera/jre/lib/
```

26. Open the following file in a text editor:

```
vim {JAVA_HOME}/jre/lib/security/java.security
```

Example :

```
vim /usr/java/jdk1.8.0_232-cloudera/jre/lib/security/java.security
```

a) Add these two lines:

```
security.provider.6=com.safenetinc.luna.provider.LunaProvider
```

```
com.safenetinc.luna.provider.createExtractableKeys=true
```

replacing the entry for security.provider.6:

```
Java SDK/JRE 1.6.x or 1.7.x installation to read as follows:
security.provider.1=sun.security.provider.Sun
security.provider.2=sun.security.rsa.SunRsaSign
security.provider.3=sun.security.ec.SunEC
security.provider.4=com.sun.net.ssl.internal.ssl.Provider
security.provider.5=com.sun.crypto.provider.SunJCE
security.provider.6=com.safenetinc.luna.provider.LunaProvider
security.provider.7=sun.security.jgss.SunProvider
security.provider.8=com.sun.security.sasl.Provider
security.provider.9=org.jcp.xml.dsig.internal.dom.XMLDSigRI
security.provider.10=sun.security.smartcardio.SunPCSC
```

27. Set the file permissions for the Luna client as follows:

```
chmod -R 777 /usr/safenet
chown kms:kms
```

Set KMS Configuration Properties in CM.

28. In Cloudera Manager Ranger KMS Configs edit the following properties:

Note: For CM-7.1.1 and CM-7.1.2 you must add properties to the dbks-site.xml, also known as Ranger KMS Server Advanced Configuration Snippet (Safety Valve) for conf/dbks-site.xml .

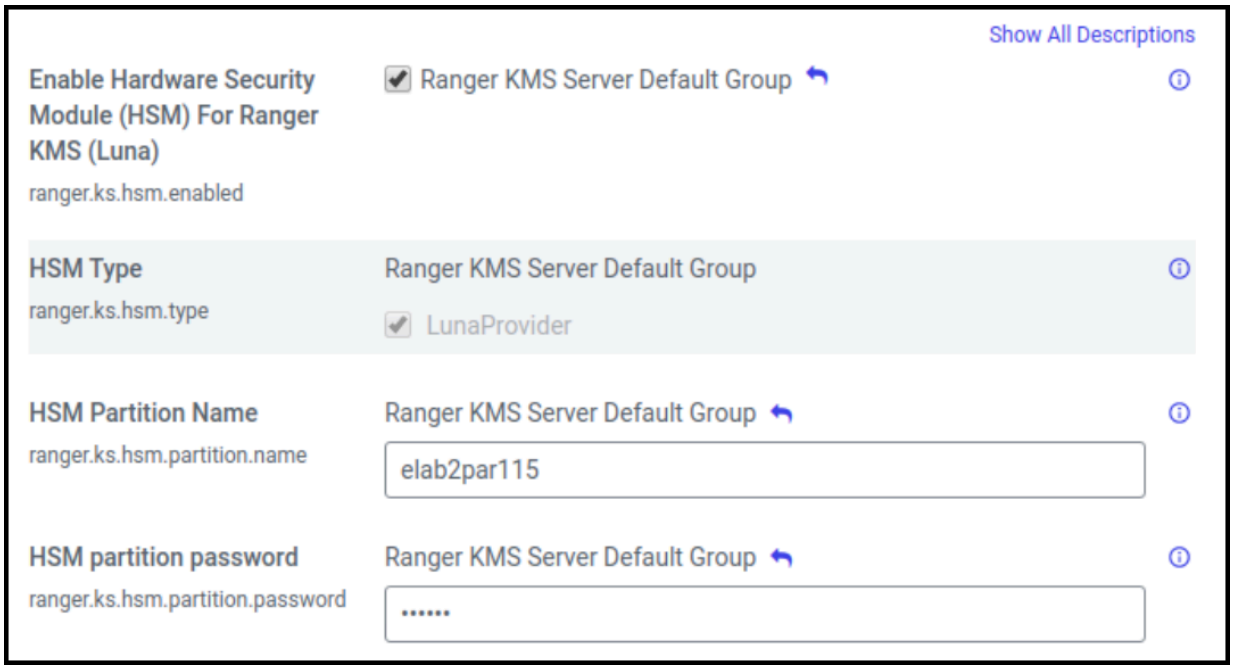
```
ranger.ks.hsm.type = LunaProvider
ranger.ks.hsm.enabled = true
ranger.ks.hsm.partition.name=elab2par115
ranger.ks.hsm.partition.password=passwrd123
```

```
(CM-7.1.1 & CM-7.1.2 password will be in plain text)
```

Note: For CM-7.1.3 and higher, update the configuration as shown in:

Example :

Figure 2: Adding Ranger KMS Configuration for Luna 10.5 HSM



29. Restart Ranger KMS from Cloudera Manager.
30. Login to Luna client and validate whether the master key is successfully created.

```
cd /usr/safenet/lunaclient/bin/
./lunacm
role login -n co

enter password: passwrd123

par con
```

Example :

```
lunacm:>par con
    The 'Crypto Officer' is currently logged in.
    Looking for objects accessible to the 'Crypto Officer'.

    Object List:

    Label:              RangerKMSKey
    Handle:             131
    Object Type:        Symmentric Key
    Object UID:         ba8e00002e00000554380800
    Number of Objects: 1

Command Result: No Error
```

Results

Ranger KMS is successfully started.

What to do next

You can now create Encryption zone keys using hadoop command or from Ranger UI using credentials of keyadmin user.

Integrating Ranger KMS DB with Google Cloud HSM

How to integrate Ranger KMS DB with Google Cloud HSM

About this task

This task describes how to integrate Ranger KMS DB with Google Cloud Platform (GCP) Hardware Security Module (HSM). This process includes setting up the GCP HSM service on a client (host), configuring Ranger KMS with GCP, or migrating the Master Key storage from the KMS database to the Google Cloud HSM.

Before you begin

- Ensure you can log in to the Google cloud console using your accout. (Requires Google account access).
- Ensure you have Java (jdk1.8.0.232) installed.

Procedure

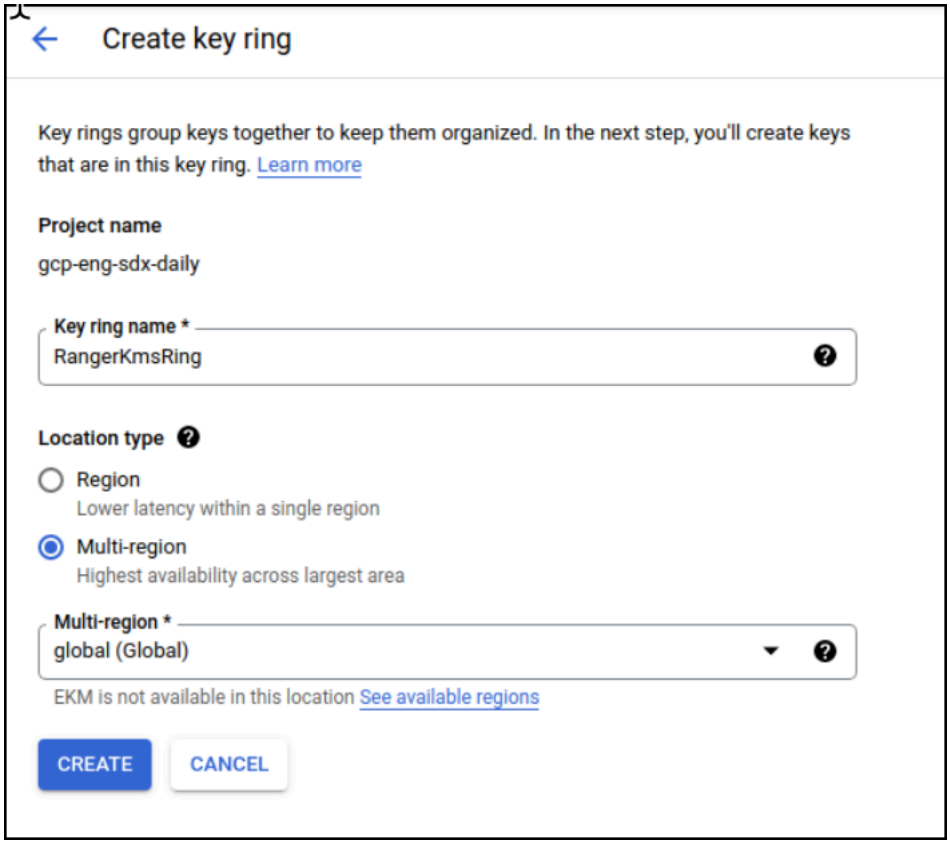
Set Up Google Cloud HSM

1. Log in to Google Cloud console using Cloudera account.
2. Create the service account by selecting or creating the Project.
3. Create the key.
4. Download and save the key in JSON format.

Note: Record the project ID, Location ID and save the JSON file.

5. In GCP Console Key Management create the key ring.

Figure 3: Creating a key ring in Google Cloud Platform

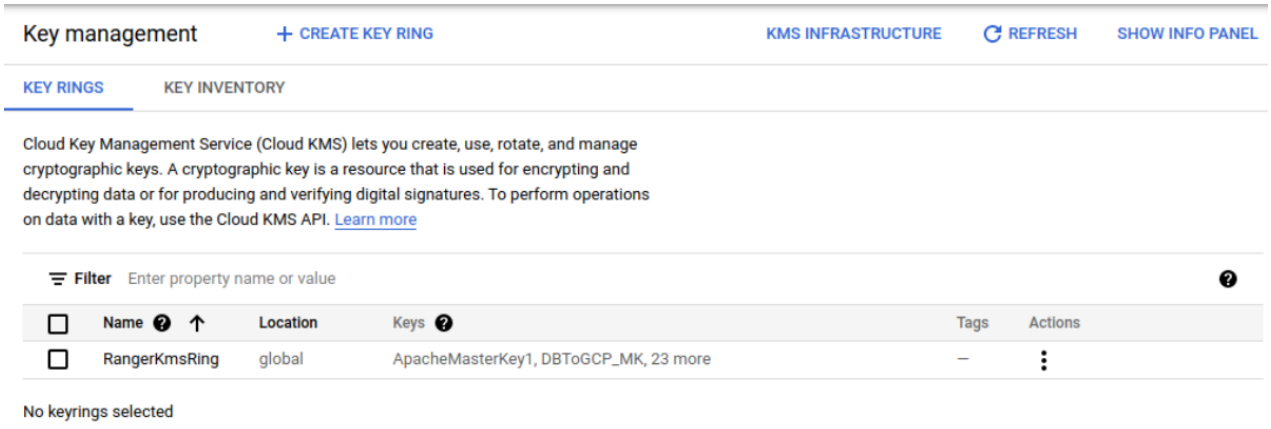


This example shows a project gcp-eng-sdx-daily, region Global, and key ring RangerKMSRing.

Results

The key ring is created.

Figure 4: RangerKMSRing created



Integrating Ranger KMS DB with CipherTrust Manager HSM

How to integrate Ranger KMS DB with CipherTrust Manager HSM.

About this task

This task describes how to integrate Ranger KMS DB with CipherTrust Manager Hardware Security Module (HSM). This process includes configuring the NAE port in Thales Cipher Trust Manager, configuring Ranger DB KMS to interact with Thales CipherTrust HSM, or, migrating Ranger KMS DB Master Key To CipherTrust Manager HSM, and migrating the master key from CipherTrust Manager HSM to Ranger KMS DB.

Before you begin

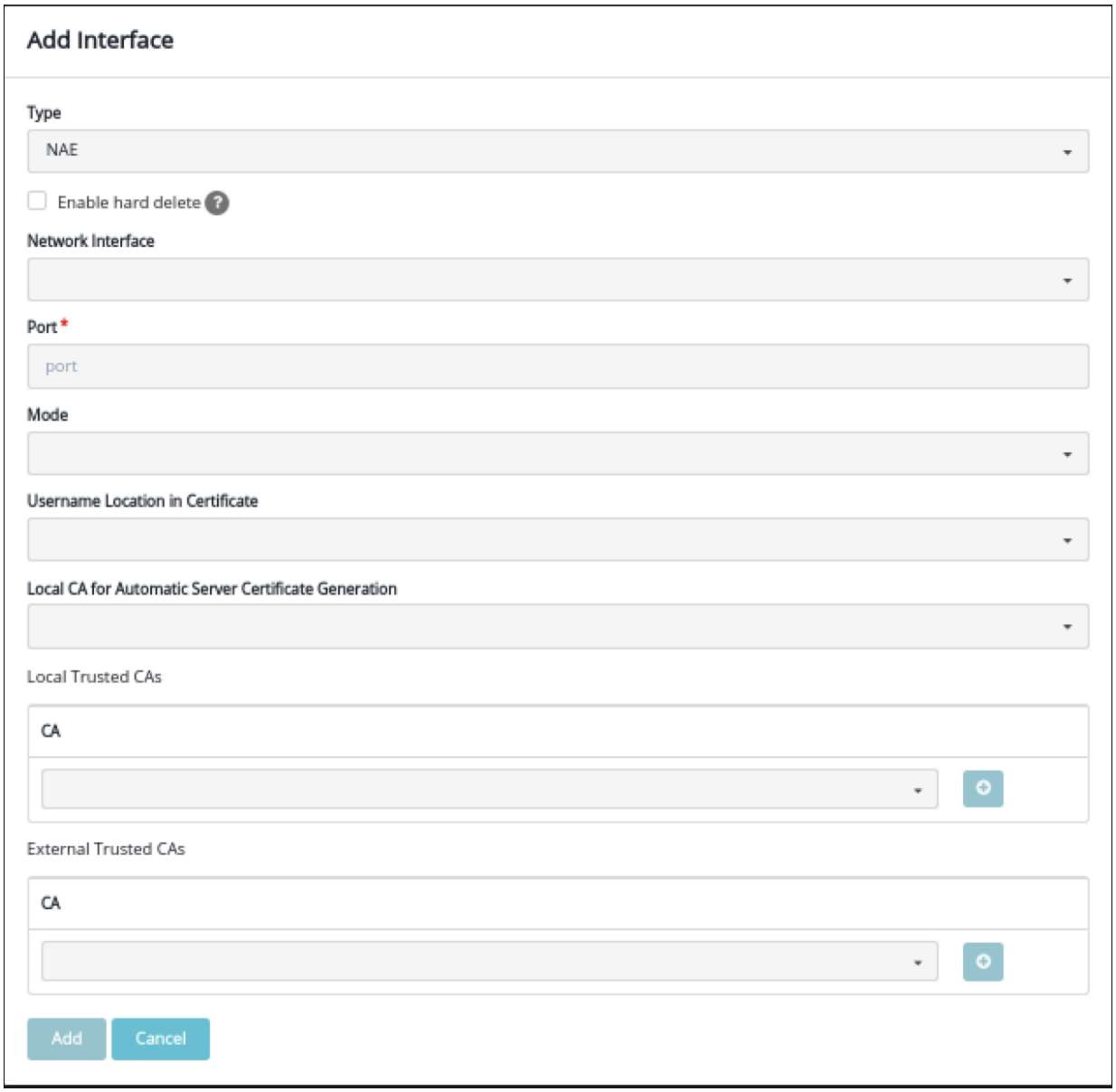
- Ensure you have Thales CipherTrust Manger installed in your enivronment.
- Ensure you have Java (jdk1.8.0.232) installed.

Procedure

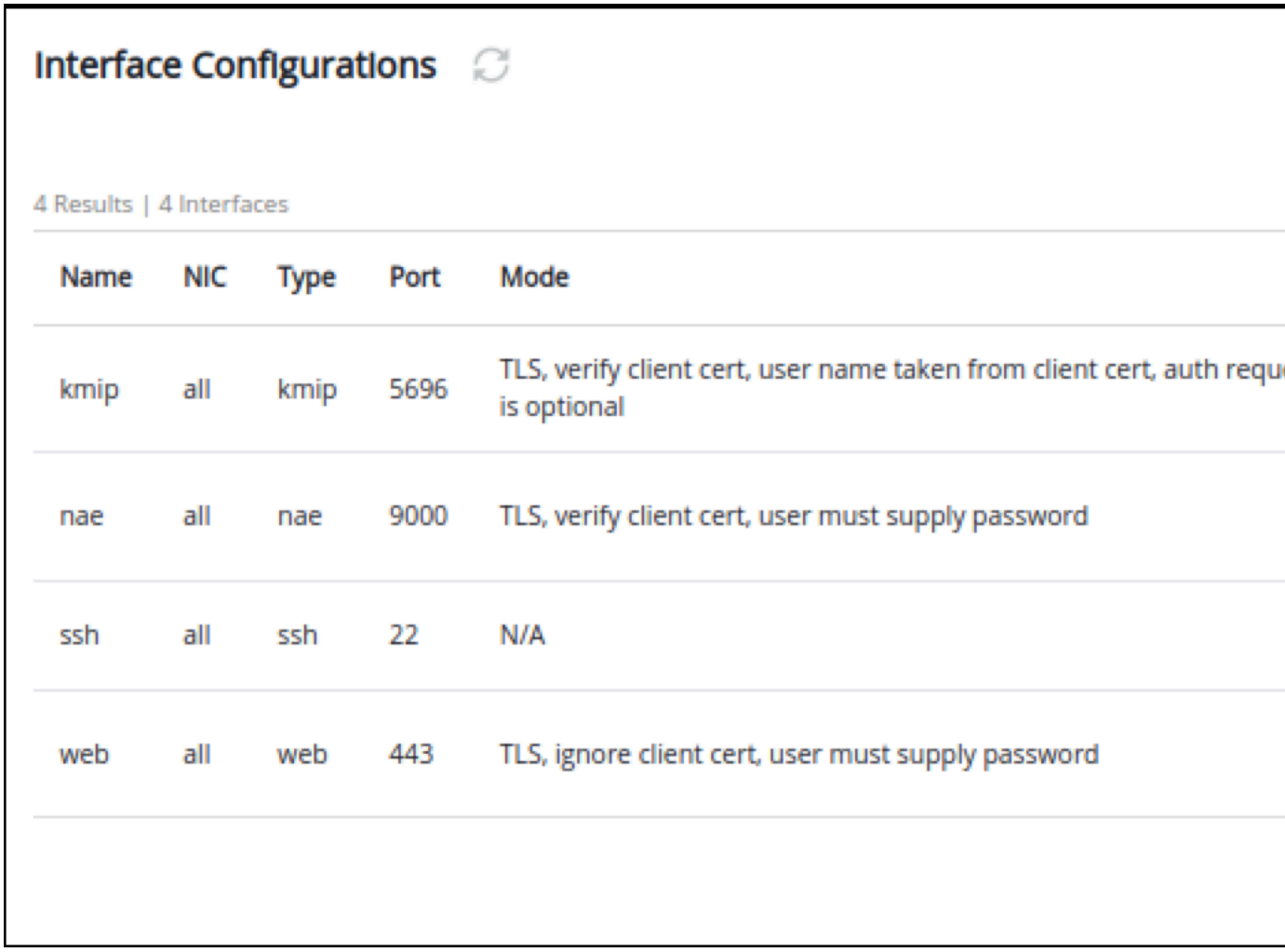
Configure NAE port in Thales CipherTrust Manager

1. Log in to Thales CipherTrust Manager.
2. In CipherTrust Manager Admin Settings , select Add Interface.
3. In Type, Select NAE (default).
4. In Network Interface, selectAll.
5. In Port, type a value for the port number.
   9000
6. In Mode, select one of the following options to match your environment:
   - No TLS,user must supply password.
   - TLS, Ignore client cert. user must supply password.

7. Click Add.


Add Interface
Type
NAE
Enable hard delete
Network Interface
Port *
port
Mode
Username Location in Certificate
Local CA for Automatic Server Certificate Generation
Local Trusted CAs
CA
External Trusted CAs
CA
Add
Cancel

8. If selected mode is TLS, ignore client cert, user must supply password while adding interface, then click Edit and Download Current Certificate as shown in the images below. Else, skip this step.



Interface Configurations

4 Results | 4 Interfaces

Name	NIC	Type	Port	Mode
kmip	all	kmip	5696	TLS, verify client cert, user name taken from client cert, auth requ is optional
nae	all	nae	9000	TLS, verify client cert, user must supply password
ssh	all	ssh	22	N/A
web	all	web	443	TLS, ignore client cert, user must supply password

Configure NAE

Enable hard delete

Mode

TLS, ignore client cert, user must supply passwo

Username Location in Certificate

CN

Local CA for Automatic Server Certificate Generation

/C=US/ST=MD/L=Belcamp/O=Gemalto/CN=KeyS

Disabled cipher suites (9)

- TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA
- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA25
- TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA
- TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA
- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA
- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA
- TLS_RSA_WITH_AES_128_CBC_SHA256

Local Trusted CAs

CA

/C=US/ST=MD/L=Belcamp/O=Gemalto/CN=KeySecur

9. After the certificate is downloaded (e.g -Certificate_nae.txt) copy it to Ranger KMS server
   Create a directory on Ranger KMS serverhost under /etc/security.

   ```
   mkdir etc/security/serverKeys
   ```

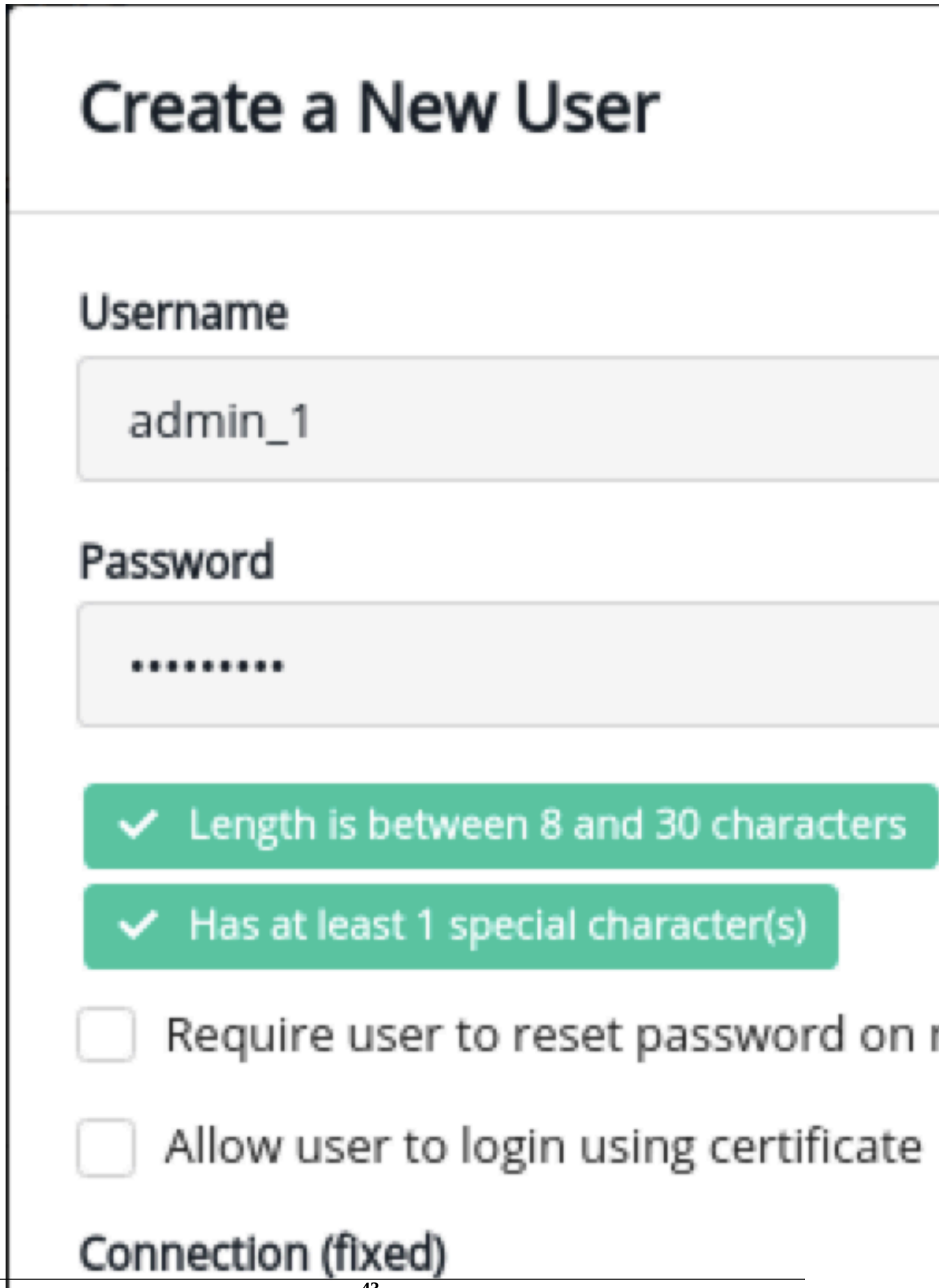
   and scp the downloaded certificate to this directory. Ensure that the user has required access to the file

   ```
   chown kms:kms etc/security/serverKeys/Certificate_nae.txt
   ```

   ```
   chmod 755 etc/security/serverKeysCertificate_nae.txt
   ```

10. Create a user.

a) Go to Access Management Users , click Create New User .
b) In Create a New User, provide a username, password, and any required information.
c) Click Create.


Create a New User
Username
admin_1
Password
Length is between 8 and 30 characters
Has at least 1 special character(s)
Require user to reset password on
Allow user to login using certificate
Connection (fixed)
local_account

Integrating Ranger KMS DB with SafeNet Keysecure HSM

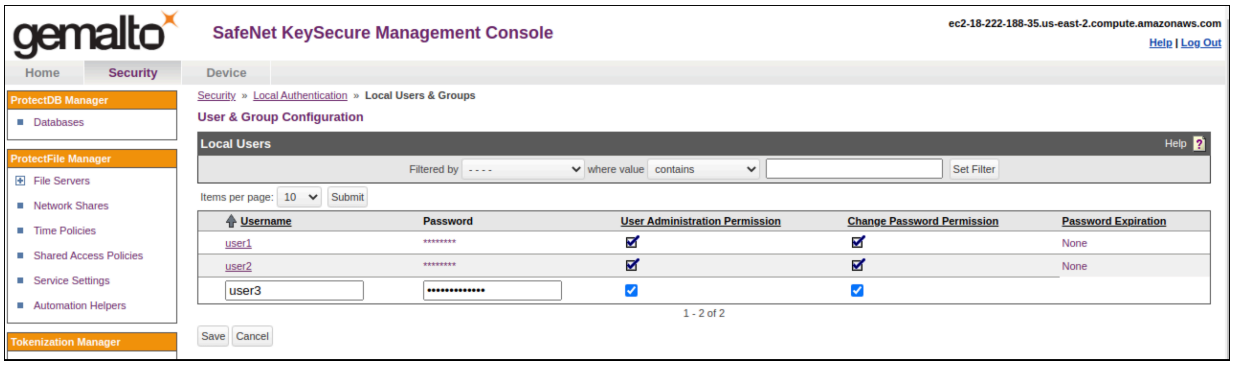
How to integrate Ranger KMS DB with SafeNet Keysecure HSM.

About this task

This task describes how to integrate Ranger KMS DB with Safenet Keysecure Hardware Security Module (HSM). This process includes setting up the SafeNet KeySecure Management Console, and configuring Ranger KMS to communicate with the Keysecure instance.

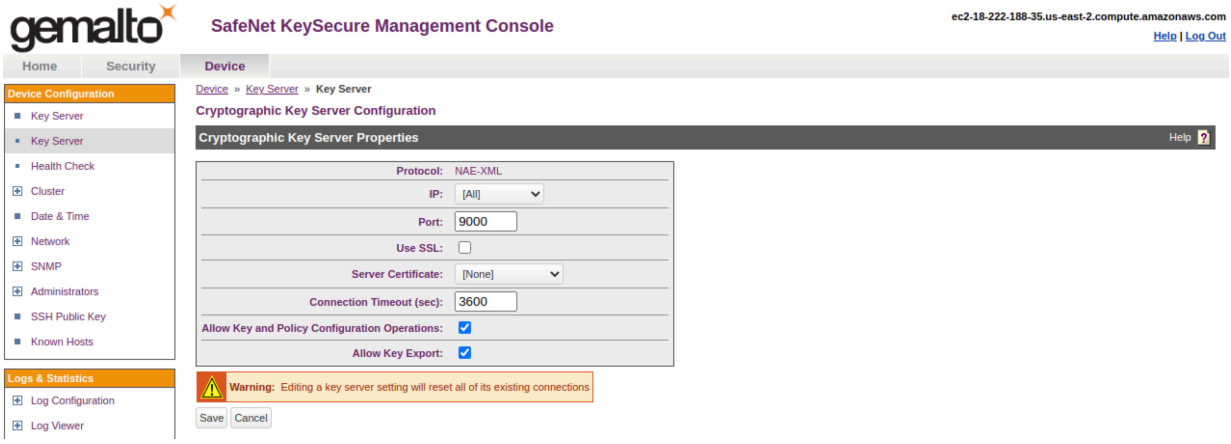
Creating the user on SafeNet keysecure

1. Log in to keysecure as an user with admin privileges.
2. Go to the Security tab.
3. Go to the Users & Groups section.
4. Click Local Authentication, and click Add to add a new user.
5. Check both ‘User Administration Permission’ and ‘Change Password Permission’ when adding the new user.
6. Save changes.



Creating device on SafeNet KeySecure

1. Log in to Keysecure with user having admin privilges.
2. Go to Device NAE-XML protocol.
3. Click Properties Edit.
4. Select Allow Key and Policy Configuration Operations and Allow Key Export .

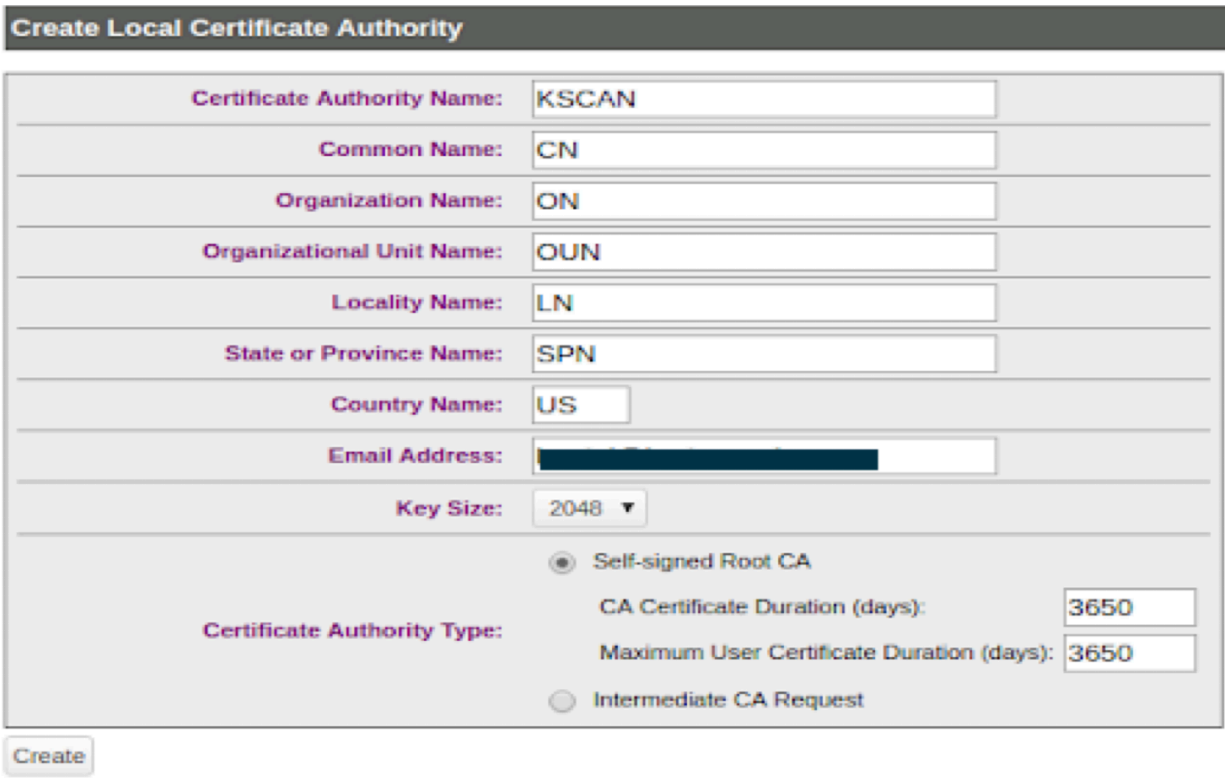


5. Save changes.

Configure SSL on Safenet Keysecure (NAE-XML)

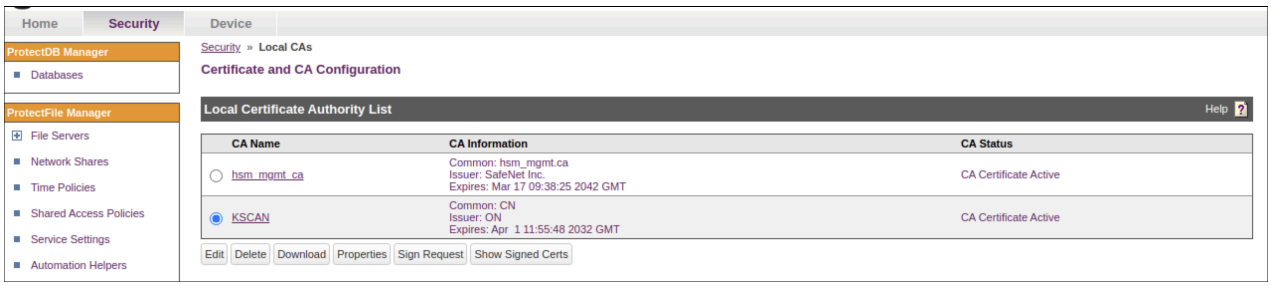
Creating a local CA

1. Log in to the Management Console as an administrator with Certificate Authority (CA) access control.
2. Navigate to the Security, CAs & SSL Certificates section and click o Local CA's.
3. Enter the required details and select Self-signed Root CA as the Certificate Authority Type.



4. Click Create.

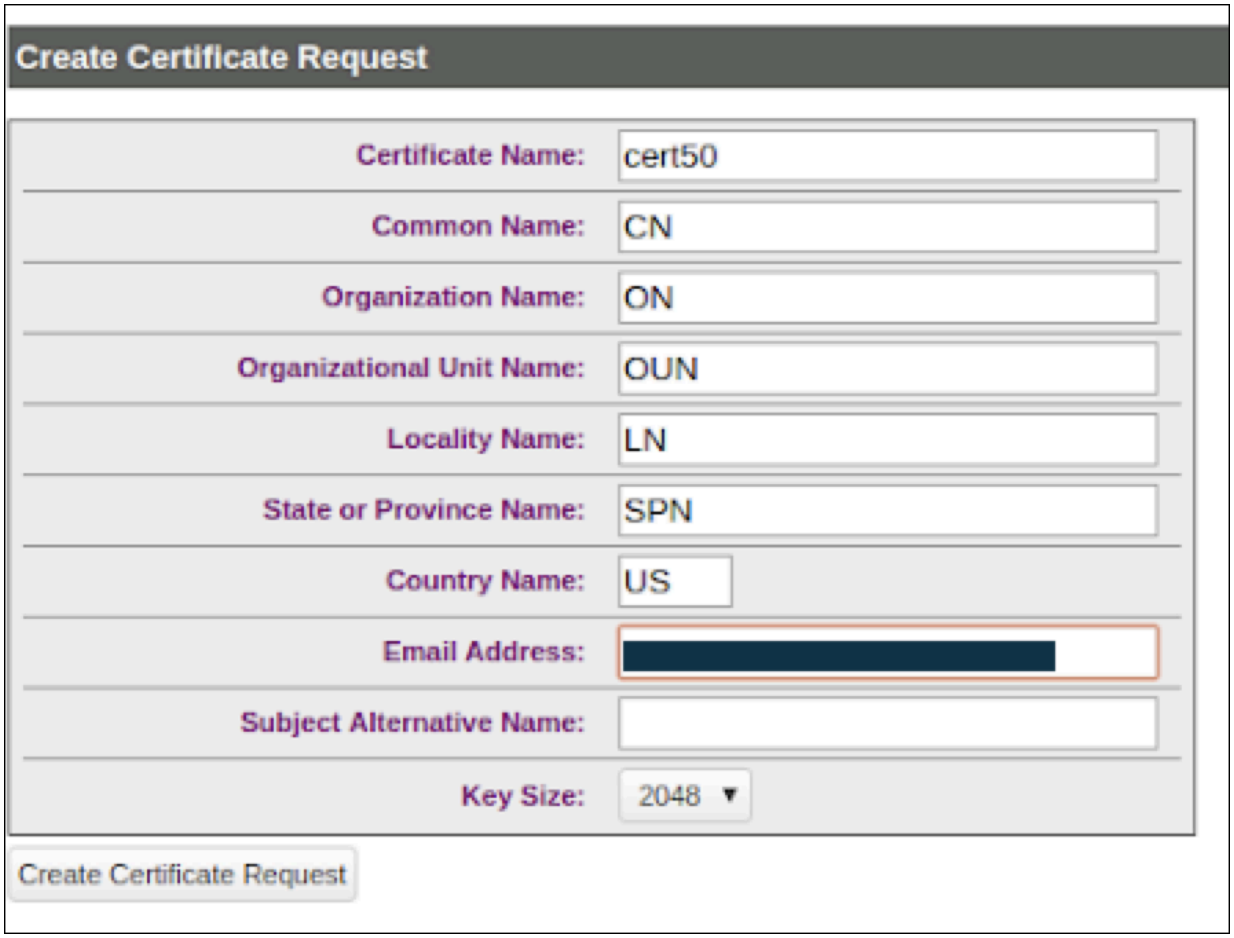
The Local CA is visble.



Creating a Server Certificate Request on the Management Console

1. Log on to the Management Console as an administrator with Certificate Authority (CA) access control.
2. Go to the Security tab and on the left side panel .
3. Navigate to the Device CAs & SSL Certificates section.

4. Click SSL certificates and modify the fields as needed.



5. Click Create Certificate Request.

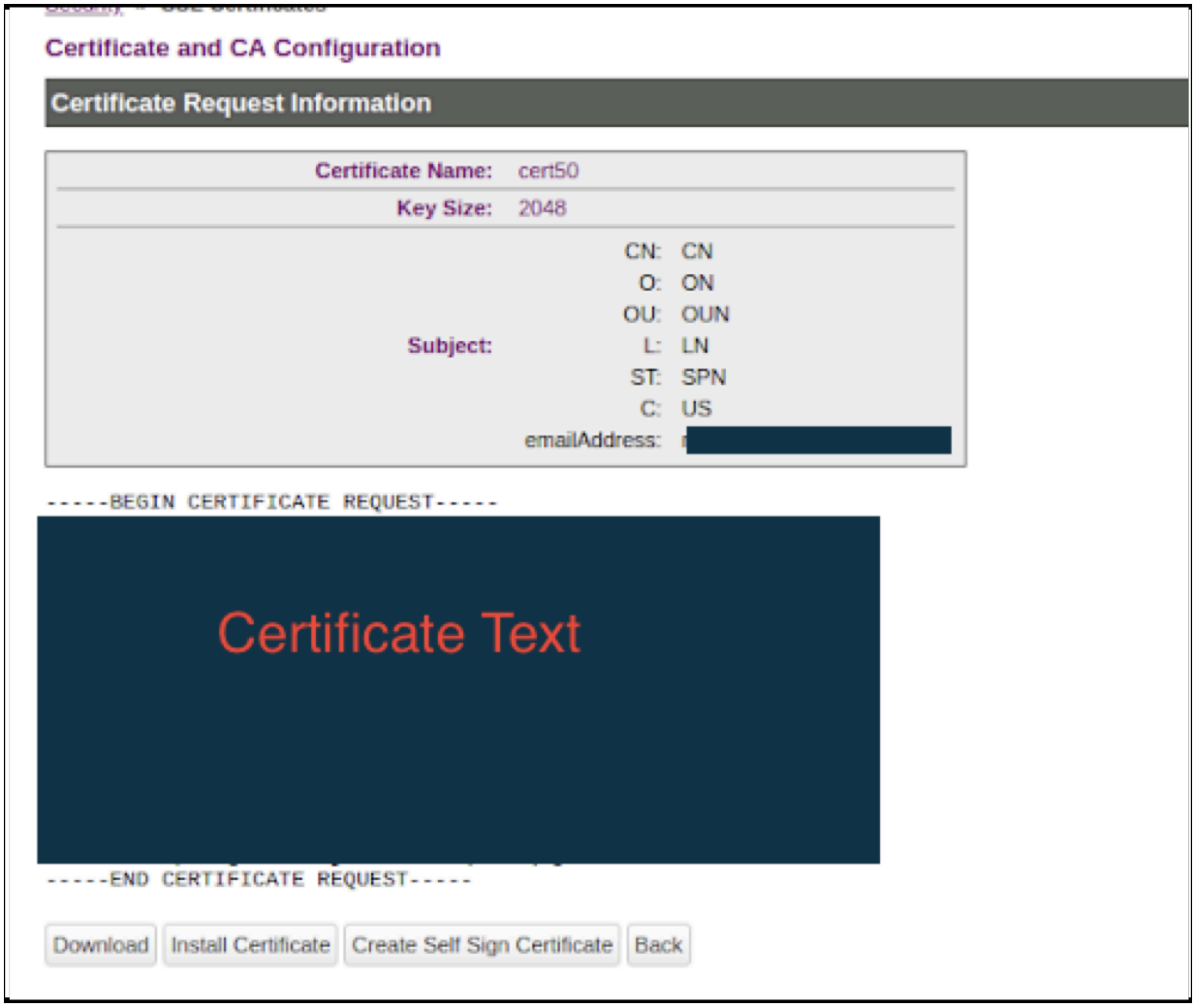
   This creates the certificate request and places it in the Certificate List section of the Certificate and CA Configuration page. The new entry shows that the Certificate Purpose is Certificate Request and that the Certificate Status is Request Pending.

nae_kmip_server	Common: nae_kmip_server Issuer: SafeNet Inc. Expires: Mar 16 09:38:26 2042 GMT	Server	Active
cert50_	Common: CN	Certificate Request	Request Pending

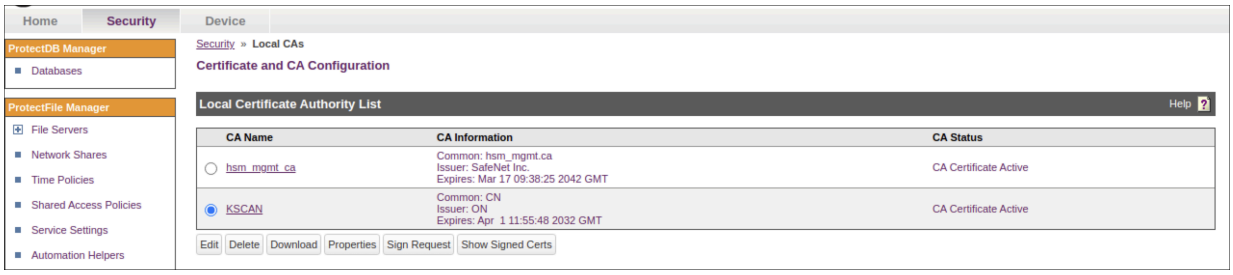
Signing a Server Certificate Request with a Local CA

1. Log on to the Management Console as an Administrator with Certificates and Certificate Authorities (CA) access controls.
2. Navigate to the Security Tab -> Device, CAs and SSL Certificates section.
3. Click SSL Certificates .

4. Select the certificate request (cert50) and click Properties.

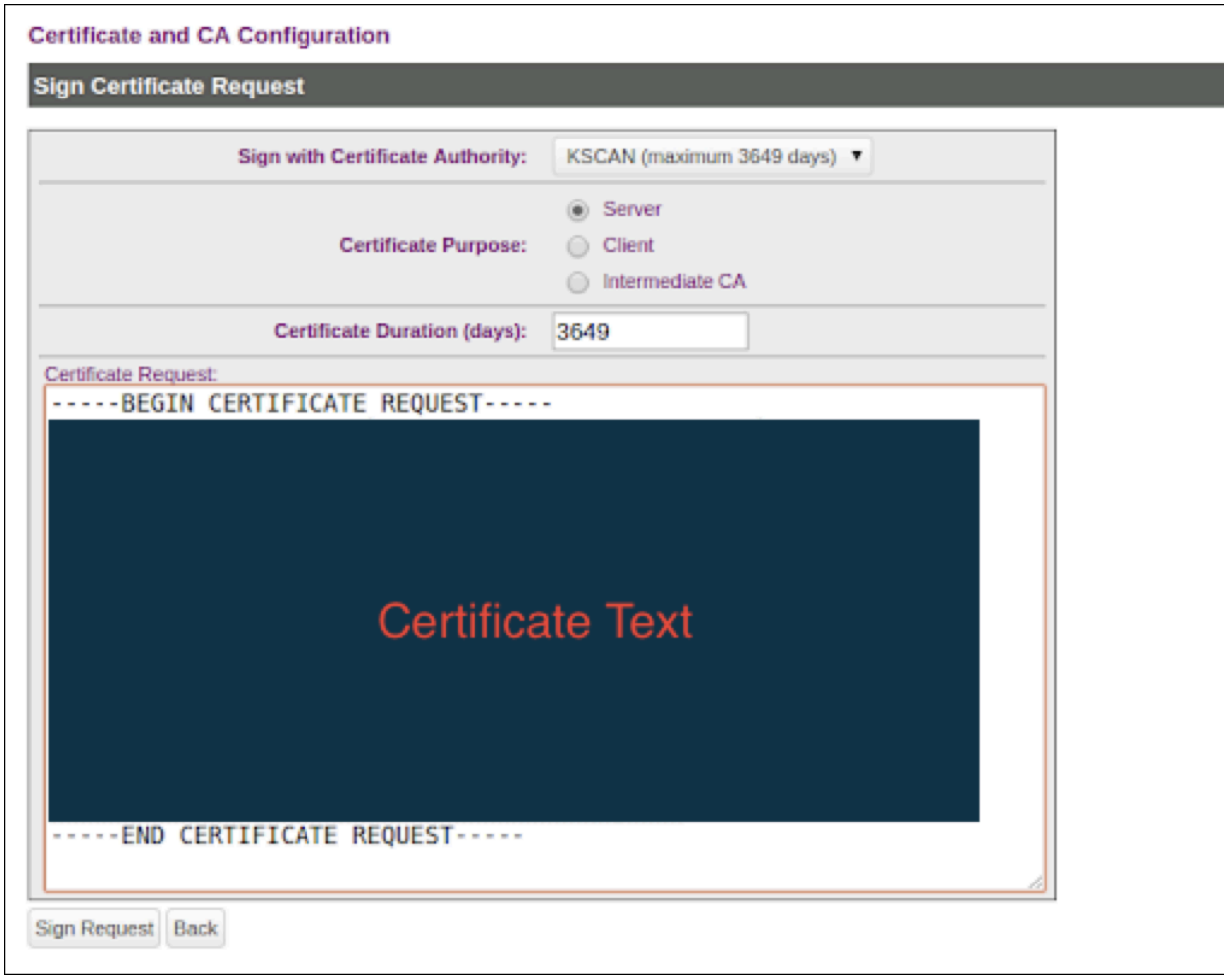


5. Copy the text of the certificate request. The copied text must include the header (-----BEGIN CERTIFICATE REQUEST-----) and footer (-----END CERTIFICATE REQUEST-----).
6. Navigate to the Security Tab -> Device, CAs & SSL Certificates section.
7. Click Local CAs and select the CA name from the list.
8. Click Sign Request to access the Sign Certificate Request section.



9. On the Sign Certificate Request screen, select Server as certificate Purpose.
10. Enter the validity of the certificate for Certificate Duration (days).

11. Paste the copied text from the server certificate request, including the header and footer in Certificate Request.



12. Click Sign Request. This takes you to the CA Certificate Information section.

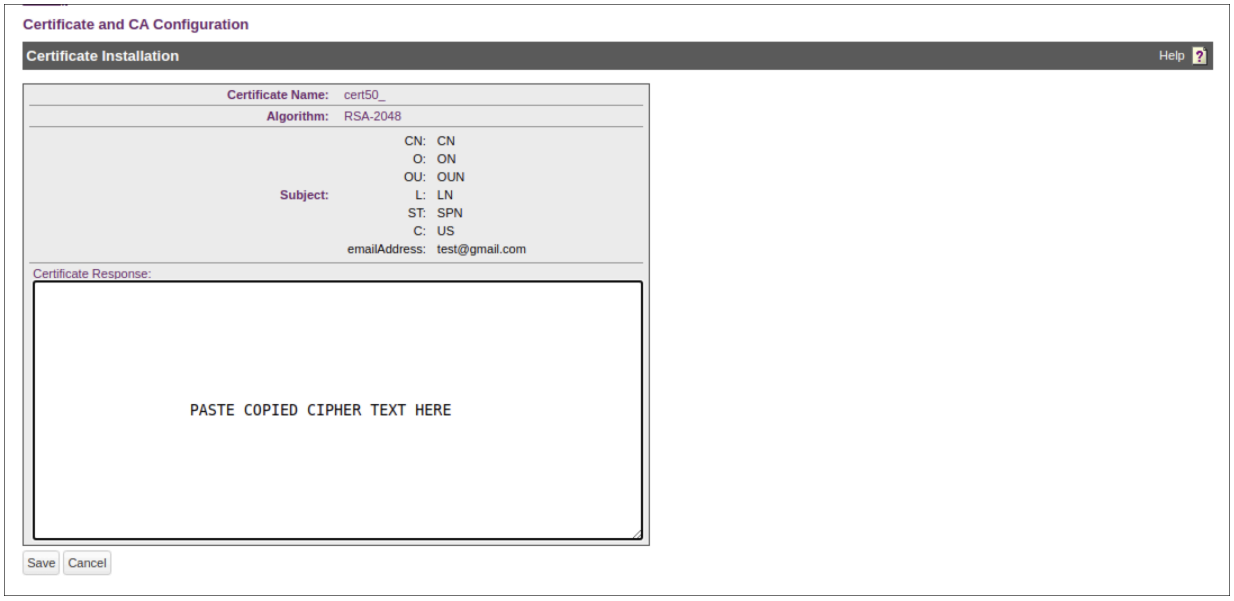
13. Copy the actual (for eaxample,. KSCAN) certificate text. The copied text must include the header (-----BEGIN CERTIFICATE-----) and footer (-----END CERTIFICATE-----).

14. Navigate back to the Certificate List section (Device, CAs & SSL Certificates) and click SSL Certificates.

15. Select your certificate request and click properties.

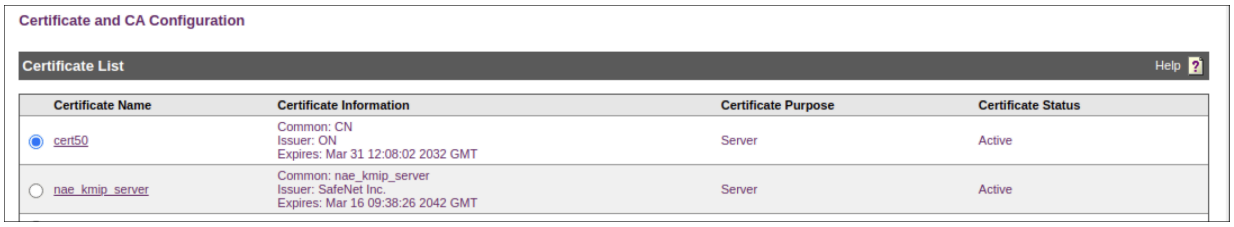
16. Click Install Certificate.

17. Paste the certificate as the Certificate Response.



18. Click Save.

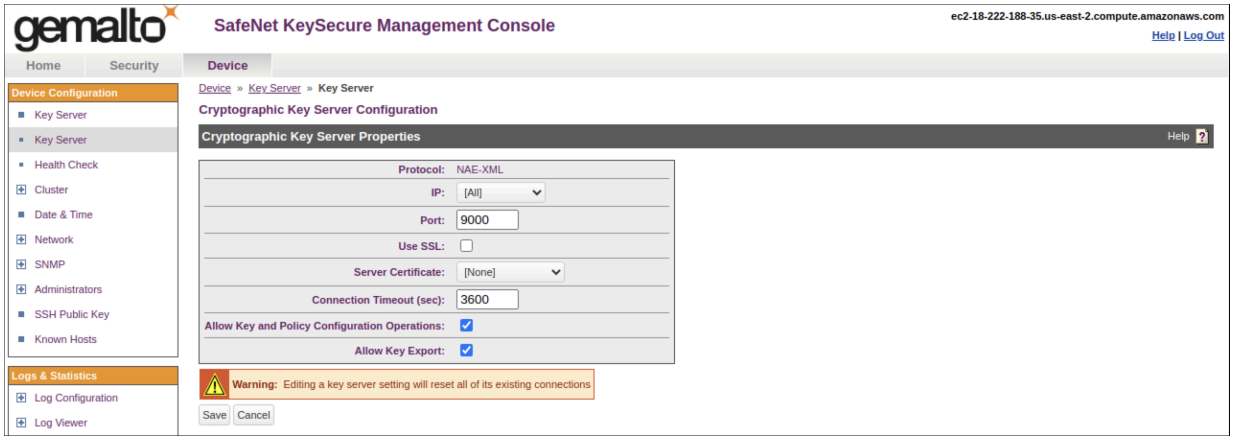
The Management Console takes you to the Certificate List section. The section shows that the Certificate Purpose is Server and that the Certificate Status is Active.



Enable SSL on Keysecure (NAE-XML)

After SSL has been configured in Safenet KeySecure, perform the following steps.

1. Log in to keysecure with admin privileges.
2. Go to the Device tab and click NAE-XML -> properties -> edit.



3. Select Use SSL.
4. Select the Server Certificate from the given drop-down list (for example, cert50).
5. Save changes.

Migrating the Master Key from Ranger KMS DB to Luna HSM

How to migrate the master key from Ranger KMS DB to Luna HSM.

Procedure

1. Go to the Ranger KMS directory.
   Example:

```
cd /opt/cloudera/parcels/CDH/lib/ranger-kms
```

2. Export the below variables

```
export JAVA_HOME=/usr/java/jdk1.8.0_232-cloudera
```

```
export RANGER_KMS_HOME=/opt/cloudera/parcels/CDH/lib/ranger-kms
```

3. Get the active directiory for rangerkms process and copy the conf directory

```
ps -ef | grep rangerkms
```

   From the output of the above command, get the value of the rangerkms conf directory.

```
export RANGER_KMS_CONF=/var/run/cloudera-scm-agent/process/xxxx-ranger_k
ms-RANGER_KMS_SERVER/conf
export SQL_CONNECTOR_JAR=/opt/cloudera/cm/lib/postgresql-42.1.4.jre7.jar
```

4. Get the active directory for rangerkms process and copy the active directory path.

```
ps -ef | grep rangerkms
```

5. Open proc.json and get the value for HADOOP_CREDSTORE_PASSWORD

```
vim /var/run/cloudera-scm-agent/process/xxxx-ranger_kms-RANGER_KMS_SERVER/
proc.json
export HADOOP_CREDSTORE_PASSWORD=hadoop_credstore_pwd
```

6. Run the following command:

```
[root@os-mv-711-1 ranger-kms]# ${JAVA_HOME}/bin/java -cp "${RANGER_KMS_H
OME}/cred/lib/*:${RANGER_KMS_CONF}:${RANGER_KMS_HOME}/ews/webapp/WEB-INF
/classes/lib/*:${SQL_CONNECTOR_JAR}:${RANGER_KMS_HOME}/ews/webapp/config
:${RANGER_KMS_HOME}/ews/lib/*:${RANGER_KMS_HOME}/ews/webapp/lib/*:${RANG
ER_KMS_HOME}/ews/webapp/META-INF:${RANGER_KMS_CONF}/*" org.apache.hadoop
.crypto.key.DB2HSMMKUtil LunaProvider <partition-name>
```

7. Enter the partition password.
8. Login to the Luna client and validate if the master key is successfully migrated.

```
cd /usr/safenet/lunaclient/bin/
./lunacm
role login -n co
         enter password: passwrd123
par con
```

   If Master Key RangerKMSKey exists, then the following will be visible:

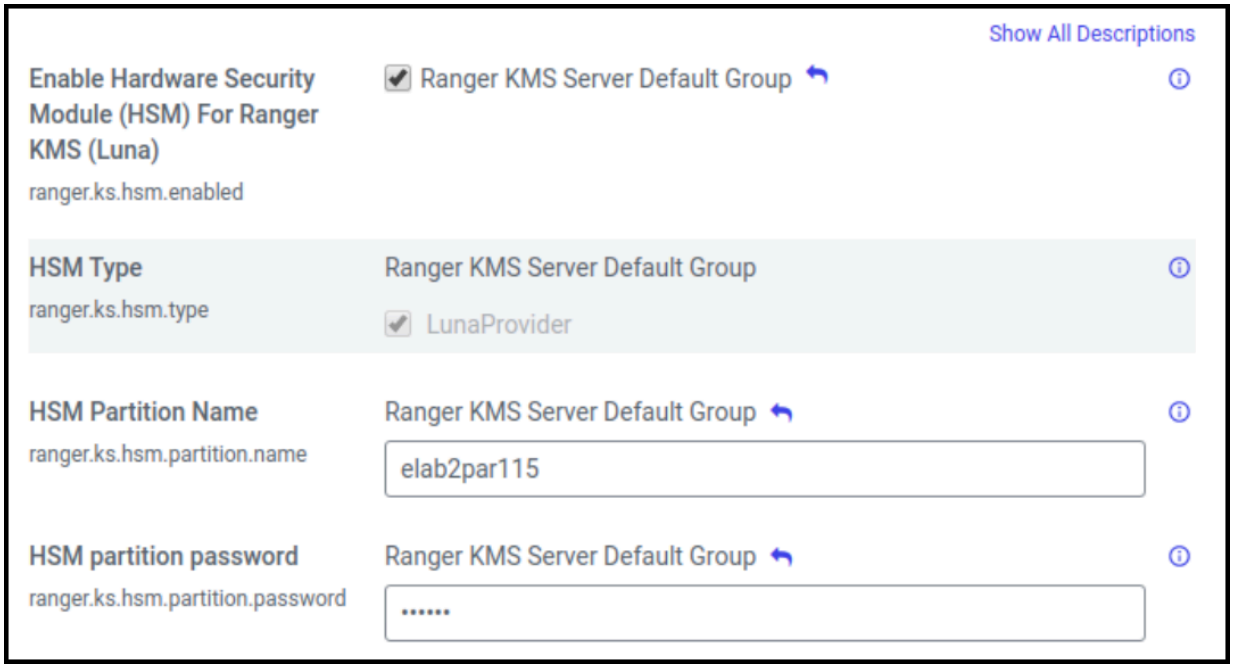
```
lunacm:>par con
```

```
                          The 'Crypto Officer' is currently logged in.
                          Looking for objects accessible to the 'Crypto Off
icer'.

                          Object List:

                          Label:              RangerKMSKey
                          Handle:             131
                          Object Type:        Symmentric Key
                          Object UID:         ba8e00002e00000554380800

                          Number of Objects: 1

                          Command Result: No Error
```

9. In Cloudera Manager Ranger KMS Configs edit the following properties:

```
ranger.ks.hsm.type = LunaProvider
                          ranger.ks.hsm.enabled = true
                          ranger.ks.hsm.partition.name=<partition-name>
                          ranger.ks.hsm.partition.password=<passwrd123>
```

Figure 5: Adding Ranger KMS Configuration for Luna HSM



10. Restart Ranger KMS from Cloudera Manager.

What to do next

Ensure Ranger KMS is running with HSM enabled. If you do not require, delete the master key row from the database table “ranger_masterkey”, as the master key has already been migrated to the HSM.

Migrating the Master Key from HSM to Ranger KMS DB

How to migrate the master key from Luna HSM to Ranger KMS DB.

Procedure

1. Go to the Ranger KMS directory.
   Example:

```
cd /opt/cloudera/parcels/CDH/lib/ranger-kms
```

2. Export the below variables

```
export JAVA_HOME=/usr/java/jdk1.8.0_232-cloudera
```

```
export RANGER_KMS_HOME=/opt/cloudera/parcels/CDH/lib/ranger-kms
```

3. Get the active directiory for rangerkms process and copy the conf directory

```
ps -ef | grep rangerkms
```

   From the output of the above command, get the value of the rangerkms conf directory.

```
export RANGER_KMS_CONF=/var/run/cloudera-scm-agent/process/xxxx-ranger_k
ms-RANGER_KMS_SERVER/conf
export SQL_CONNECTOR_JAR=/opt/cloudera/cm/lib/postgresql-42.1.4.jre7.jar
```

4. Get the active directory for rangerkms process and copy the active directory path.

```
ps -ef | grep rangerkms
```

5. Open proc.json and get the value for HADOOP_CREDSTORE_PASSWORD

```
vim /var/run/cloudera-scm-agent/process/xxxx-ranger_kms-RANGER_KMS_SERVER/
proc.json
export HADOOP_CREDSTORE_PASSWORD=hadoop_credstore_pwd
```

6. Run the following command:

```
[root@os-mv-711-1 ranger-kms]# ${JAVA_HOME}/bin/java -cp "${RANGER_KMS_H
OME}/cred/lib/*:${RANGER_KMS_CONF}:${RANGER_KMS_HOME}/ews/webapp/WEB-INF
/classes/lib/*:${SQL_CONNECTOR_JAR}:${RANGER_KMS_HOME}/ews/webapp/config
:${RANGER_KMS_HOME}/ews/lib/*:${RANGER_KMS_HOME}/ews/webapp/lib/*:${RANG
ER_KMS_HOME}/ews/webapp/META-INF:${RANGER_KMS_CONF}/*" org.apache.hadoop
.crypto.key.HSM2DBMKUtil LunaProvider <partition-name>
```

7. Run the following command:

```
./HSMMK2DB.sh <provider> <HSM_PARTITION_NAME>
```

   Example :

```
./HSMMK2DB.sh LunaProvider
```

8. Enter the partition password when requested.
9. Login to the database that Ranger KMS is using, and validate whether master key is successfully migrated.
   Example : If the Ranger KMS database is Postgres, then

```
su - postgres
psql
Password : cloudera
\l
Find rangerkms db

 \c rangerkms
```

```
 select * from ranger_masterkey;
```

10. Login to CM and disable the HSM

```
ranger.ks.hsm.enabled = false
```

11. Restart Ranger KMS.

12. Delete the master key from the partition.

```
/usr/safenet/lunaclient/bin/
./lunacm
lunacm:>role login -n co
enter password: **********
lunacm:>par con
lunacm:>par clear
proceed
```