Oracle® Key Vault

Integration with Hardware Security Module (HSM)

Release 12.2 BP10

E75841-13

October 2019

Introduction

Release 12.2 Bundle Patch 1 introduced Hardware Security Module (HSM) integration with Oracle Key Vault, where the HSM acts as a “Root of Trust” by storing a top-level encryption key for Oracle Key Vault.

Topics:

- How Oracle Key Vault Works with Hardware Security Modules (page 2)
- Install HSM Client Software on Oracle Key Vault Server (page 3)
- Enroll Oracle Key Vault as a Client of HSM (page 3)
- Protect the Oracle Key Vault TDE Master Key with the HSM (page 3)
- Enable HSM in a High Availability Key Vault Installation (page 5)
- Backup and Restore in HSM Mode (page 7)
- Reverse Migrating to Local Wallet (page 9)
- General Troubleshooting (page 12)
- Vendor Specific Notes for SafeNet (page 14)
- Vendor Specific Notes for nCipher (page 18)
- CNSA Suite Support (page 23)
- Commands used in Oracle Key Vault 12.2.0.5.0 and Earlier (page 25)

Note:

- HSM integration is limited to new installations of Oracle Key Vault. This can be a fresh installation of Oracle Key Vault Release 12.2 BP 1 and later. The latest release is the recommended path as it contains the latest enhancements.
- If you have an existing Oracle Key Vault installation with HSM and you want to upgrade to a later release of Oracle Key Vault with HSM, you must contact Oracle support.

How Oracle Key Vault Works with Hardware Security Modules

Oracle Key Vault is a full-stack software appliance that contains an operating system, database, and key-management application to help organizations store and manage their keys and credentials. The configuration that you perform using this guide also establishes a Root-of-Trust (RoT) for Oracle Key Vault in the HSM. When an HSM is deployed with Oracle Key Vault, the Root of Trust (RoT) remains in the HSM. The HSM RoT protects the wallet password, which protects the TDE master key,which in turn protects all the encryption keys, certificates, and other security artifacts managed by the Oracle Key Vault server. Note that the HSM in this RoT usage scenario does not store any customer encryption keys. The customer keys are stored and managed directly by the Oracle Key Vault server.

Using HSM as a RoT is intended to mitigate attempts to recover keys from an Oracle Key Vault server which has been started in an unauthorized environment. Physical loss of an Oracle Key Vault server from a facility is one example of such a scenario. An unauthorized user attempting to run a lost or stolen Oracle Key Vault server, without authorized access to the HSM,would be prevented from recovering the encryption keys stored on the appliance.

Oracle Key Vault employs a hierarchy of security controls including operating system hardening, database encryption, and data access enforcement using Database Vault. These controls are designed to mitigate the risk of users potentially extracting keys and credentials from systems they can physically access. Administrators do not need to access the internal components of the appliance for normal, day-to-day operations. Oracle Key Vault should be deployed in a secure location, and physical and logical access to the appliance should be controlled and monitored.

Enabling HSM in your Oracle Key Vault installation will not disrupt existing features. You can continue to work with Oracle Key Vault features like high availability, backup, and restore in HSM mode.

HSMs contain tamper-resistant, specialized hardware which is harder to access than normal server memory. Oracle Key Vault can use HSMs to generate and store a Root of Trust (RoT) that protects encryption keys used by Oracle Key Vault to safeguard users' keys and credentials. When using Oracle Key Vault with an HSM, keys and credentials can be read if the RoT stored in the HSM is available. Since HSMs are designed to make the RoT very difficult to extract, this significantly mitigates the risk of compromise of users' keys and credentials. In addition, the HSM can be used in FIPS 140-2 Level 2 or Level 3 mode which can help meet certain compliance requirements.

> **Note:**
>
> Oracle Key Vault can function only if the RoT stored in the HSM is available.

The HSM vendors currently integrated with Oracle Key Vault are: SafeNet (a Thales company) Luna SA 7000 and nCipher nShield Connect 6000+.

Install HSM Client Software on Oracle Key Vault Server

You must first install Oracle Key Vault, then install the HSM client software on the Key Vault server. You will need to refer to the HSM documentation from the HSM vendor for more information.

To install an HSM on an Oracle Key Vault server:

1. Install the HSM vendor's client software on the Oracle Key Vault server.
2. Ensure that the vendor's software includes a PKCS#11 library.

> **See Also:**
>
> - Vendor Specific Notes for SafeNet (page 14) for vendor-specific instructions to enroll Key Vault as a client of the SafeNet Luna SA 7000 HSM
> - Vendor Specific Notes for nCipher (page 18) for vendor-specific instructions to enroll Key Vault as a client of the nCipher nShield Connect 6000+ HSM

Enroll Oracle Key Vault as a Client of HSM

You must enroll Oracle Key Vault as a client of HSM and ensure connectivity between the HSM client and the HSM.

You must refer to your specific HSM documentation to complete enrolling Key Vault as an HSM client.

In general you must:

1. Install the HSM vendor's client software on the Oracle Key Vault server.
2. Ensure that the HSM client software can communicate from Oracle Key Vault to the HSM.

> **See Also:**
>
> - Vendor Specific Notes for SafeNet (page 14) for more instructions
> - Vendor Specific Notes for nCipher (page 18) for more instructions

Protect the Oracle Key Vault TDE Master Key with the HSM

To protect the TDE master key with the HSM, do the following:

1. Log into the Oracle Key Vault management console as a user with system administrative privileges.

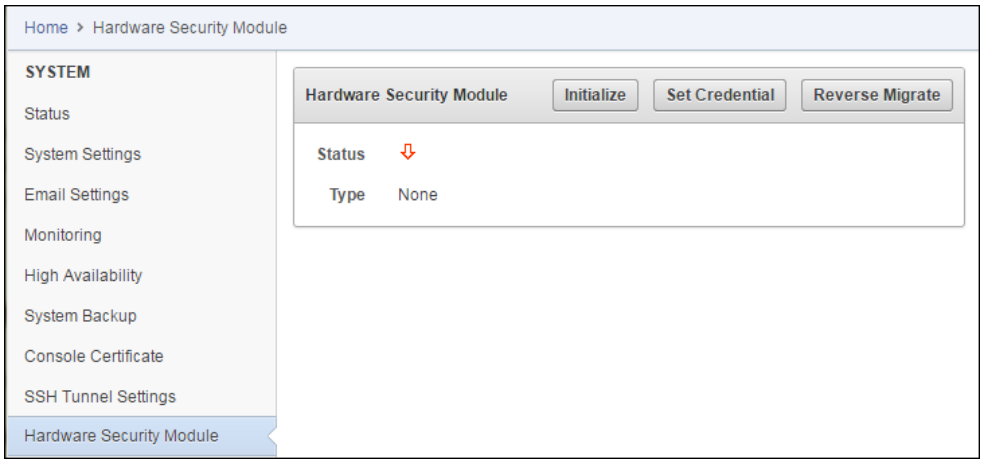
   The Oracle Key Vault **Home** page appears.

2. Click the **System** tab.

   The **Status** page appears.

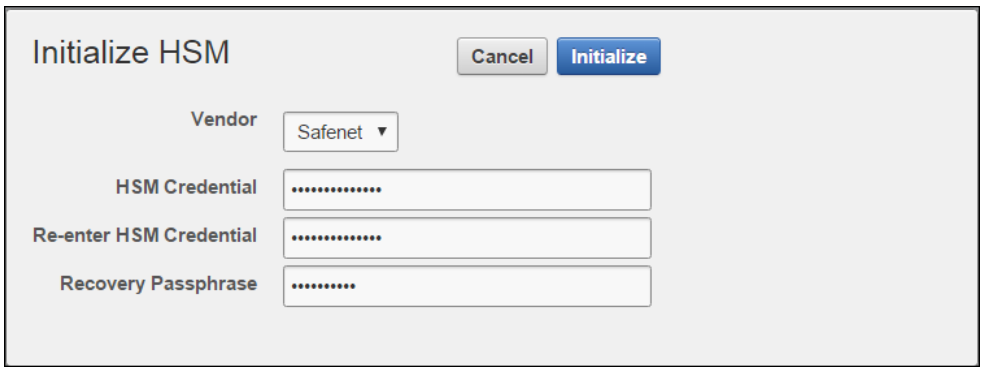
3. Click **Hardware Security Module** in the left sidebar.

   The **Hardware Security Module** page appears. The red downward arrow shows the non-initialized **Status** . The **Type** field displays None.



4. Click **Initialize**.

   The **Initialize HSM** screen appears.



5. Enter the HSM credential two times: first in **HSM Credential** and second in **Re-enter HSM Credential**.
6. Enter the **Recovery Passphrase** for Oracle Key Vault.
7. Click **Initialize**.

At the end of a successful initialize operation, the **Hardware Security Module** page appears. The initialized **Status** is indicated by an upward green arrow. The **Type** field displays details of the HSM in use.

8. If you have implemented nCipher Hardware Security Module (HSM), run the following command as user `oracle`:

```
oracle$ /opt/nfast/bin/rfs-sync --commit
```

If the initialize operation fails you will be redirected to the **Hardware Security Module** page with non-initialized **Status** and **Type** None.

> **Note:**
>
> If you change the HSM credential on the HSM after initialization, you must also update the HSM credential on the Oracle Key Vault server using the **Set Credential** command.

Enable HSM in a High Availability Key Vault Installation

In a high availability Oracle Key Vault installation you must enable HSM separately on the servers you mean to designate as primary and standby before pairing them in a high availability configuration.

> **See Also:**
>
> If you are enabling High Availability using an nCipher HSM, see Vendor Specific Notes for nCipher (page 18) for more instructions.

To enable HSM in a high availability deployment:

1. Install two separate Oracle Key Vault instances.
2. Choose one to be the primary node and the other to be the standby node.
3. Install the HSM client software on both the primary and the standby node.
4. Enroll the primary and standby nodes as clients of HSM.

5. Initialize HSM use on the primary. Log in to the designated primary server through SSH as user `support`, switch user (`su`) to `root`, then switch user (`su`) to `oracle`.

```
$ ssh support@okv_primary_instance
<Enter password when prompted>
$ su root
root# su oracle
```

6. Perform the following manual steps on the primary server as user `oracle`:

```
oracle$ cd /usr/local/okv/hsm/wallet
oracle$ scp cwallet.sso support@okv_standby_instance:/tmp
oracle$ scp enctdepwd support@okv_standby_instance:/tmp
oracle$ cd /usr/local/okv/hsm/restore
oracle$ scp ewallet.p12 support@okv_standby_instance:/tmp
```

> **Note:**
>
> While performing this procedure on Oracle Key Vault 12.2.0.5.0 and earlier, use the commands in Enabling the HSM_ENABLED Parameter in the okv_security.conf File (page 26).

7. Log in to the designated standby server through SSH as user `support`, then switch user (`su`) to `root`.

```
$ ssh support@okv_standby_instance
<Enter password when prompted>
$ su root
```

8. Open the `okv_security.conf` file.

   A sample `okv_security.conf` file before enabling HSM mode:

```
SNMP_ENCRYPTION_PWD="snmp_encryption_password"
SNMP_AUTHENTICATION_PWD="snmp_auth_password"
SNMP_USERNAME="snmpuser"
SMTP_TRUSTSTORE_PWD="smtp_truststore_password"
HSM_ENABLED="0"
```

   In Oracle Key Vault 12.2.0.6.0 and later, the `okv_security.conf` file contains an additional parameter:

```
FIPS_ENABLED="0"
```

9. Enable the `HSM_ENABLED` parameter in the `okv_security.conf` file.

```
$ cd /usr/local/okv/hsm/wallet
$ mv /tmp/enctdepwd .
$ mv /tmp/cwallet.sso .
```

```
$ chown oracle *
$ chgrp oinstall *
$ cd /usr/local/okv/hsm/restore
$ mv /tmp/ewallet.p12 .
$ chown oracle *
$ chgrp oinstall *
$ vi /usr/local/okv/etc/okv_security.conf
   Set HSM_ENABLED="1"
   Set HSM_PROVIDER="<provider value>"
```

Save and quit by entering the following sequence of characters in the vi file: `:wq!`

> **Note:**
>
> While performing this procedure on Oracle Key Vault 12.2.0.5.0 and earlier, use the commands in Enabling the HSM_ENABLED Parameter in the okv_security.conf File (page 26).

After enabling HSM the `okv_security.conf` file should look like this:

```
SNMP_ENCRYPTION_PWD="snmp_encryption_password"
SNMP_AUTHENTICATION_PWD="snmp_auth_password"
SNMP_USERNAME="snmpuser"
SMTP_TRUSTSTORE_PWD="smtp_truststore_password"
HSM_ENABLED="1"
HSM_PROVIDER="<provider value>"
```

In Oracle Key Vault 12.2.0.6.0 and later, the `okv_security.conf` file contains an additional parameter:

```
FIPS_ENABLED="0"
```

Check vendor-specific notes for the specific provider value to use.

10. Then, without restarting the OKV instances, navigate to the primary and standby management consoles and configure high availability.

Backup and Restore in HSM Mode

You can backup and restore Oracle Key Vault with HSM mode enabled.

Backup in HSM Mode

Backing up Key Vault data in HSM mode is the same as backing up in non-HSM mode. So proceed in the usual way to take a backup.

Restore in HSM Mode

Only backups taken in HSM mode can be restored onto an HSM-enabled Oracle Key Vault. Before you restore a backup onto a system, you must ensure that the system can access both the:

- HSM
- Root of Trust used to take the backup

You must therefore have installed the HSM on the Oracle Key Vault server and enrolled Oracle Key Vault as a client of HSM prior to this step.

Prepare the system for restore as follows:

1. Log into the Oracle Key Vault management console as a user with system administrative privileges.

   The Oracle Key Vault **Home** page appears.

2. Click the **System** tab.

   The **Status** page appears.

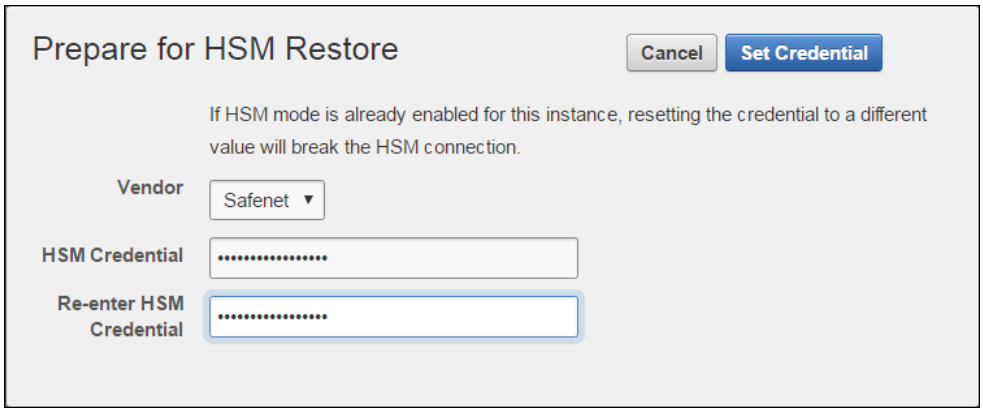
3. Click **Hardware Security Module** in the left sidebar.

   The **Hardware Security Module** page appears. On restore, the **Status** is disabled first, then enabled after the restore completes.

4. Click **Set Credential**.

   The **Prepare for HSM Restore** screen appears.

Figure 1-1 Prepare for HSM Restore Screen



5. Enter the HSM credential two times: first in **HSM Credential** and second in **Re-enter HSM Credential**.
6. Click **Set Credential**.

> **Caution:**
>
> In Oracle Key Vault 12.2.0.5.0 and earlier, to successfully restore HSM, you must enter the HSM credential correctly. If you enter an incorrect credential for the HSM, you will disable the HSM. In this situation you must reset the credential to its proper value immediately, by re-entering the correct HSM credential and clicking **Set Credential**. If the Key Vault server is rebooted before resetting the credential, Key Vault will become inoperable and will need to be restored from backup.
>
> In Oracle Key Vault 12.2.0.6.0 and later with HSM mode enabled, if you enter an incorrect credential for the HSM, the previous credential will continue to be stored and used. If HSM mode is not enabled, and you enter an incorrect credential for the HSM, the incorrect credential is not stored.

The HSM credential will be stored in the system. This HSM credential must be entered manually to do an HSM restore because it is not stored in the backup itself.

7. Go to the **Restore** page via the Key Vault user interface and restore the backup as usual.

Reverse Migrating to Local Wallet

The HSM reverse migrate procedure allows you to disable the HSM and go back to a local wallet protected by the Recovery Passphrase.

The purpose of reverse migrate is to revert back to a local wallet protected by the Recovery Passphrase. This will be necessary if an HSM currently protecting Oracle Key Vault needs to be decommissioned.

Topics:

- Reverse Migrating a Standalone Deployment (page 9)
- Reverse Migrating a High Availability Deployment (page 10)

Reverse Migrating a Standalone Deployment

To reverse migrate a standalone deployment:

1. Log into the Oracle Key Vault management console as a user with system administrative privileges.

   The Oracle Key Vault **Home** page appears.

2. Click the **System** tab.

   The **Status** page appears.

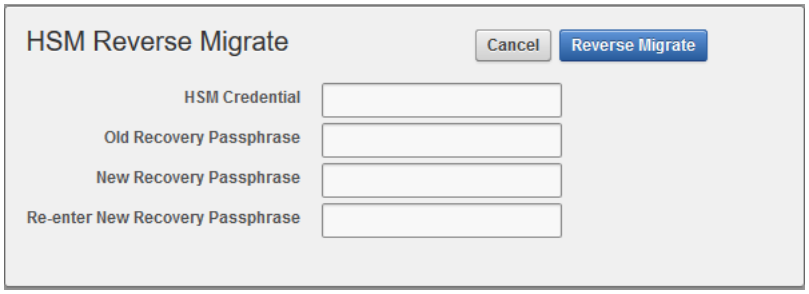
3. Click **Hardware Security Module** in the left sidebar.

The **Hardware Security Module** page appears.

4. Click **Reverse Migrate**.

   The **HSM Reverse Migrate** screen is displayed.

Figure 1-2 HSM Reverse Migrate Screen



On the **HSM Reverse Migrate** screen, enter the following details:

- Enter the HSM credential.
- Enter the old Recovery Passphrase.
- Enter the new Recovery Passphrase in the **New Recovery Passphrase** and **Re-enter New Recovery Passphrase** fields.

5. Click **Reverse Migrate**

   The **Hardware Security Module** page appears. The red downward arrow indicates the **Status**.

Reverse Migrating a High Availability Deployment

Perform the following procedure to reverse migrate a High Availability deployment (Oracle Key Vault 12.2.0.6.0 and later).

To reverse migrate a High Availability deployment (Oracle Key Vault 12.2.0.6.0 and later):

1. On the Primary server, log into the Oracle Key Vault management console as a user with system administrative privileges.

   The Oracle Key Vault **Home** page appears.

2. Click the **System** tab.

   The **Status** page appears.

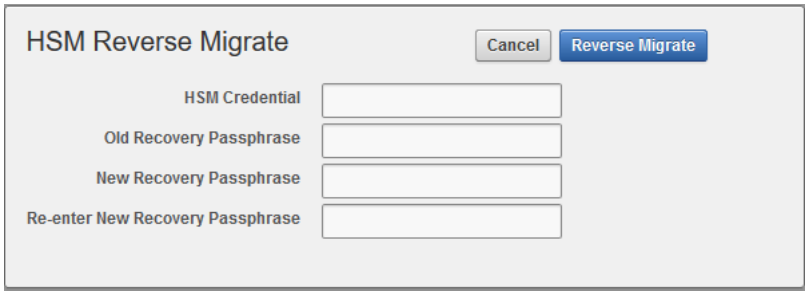
3. Click **Hardware Security Module** in the left sidebar.

   The **Hardware Security Module** page appears.

4. Click **Reverse Migrate**.

   The **HSM Reverse Migrate** screen is displayed.

Figure 1-3 HSM Reverse Migrate Screen



On the **HSM Reverse Migrate** screen, enter the following details:

- Enter the HSM credential.
- Enter the old Recovery Passphrase.
- Enter the new Recovery Passphrase in the **New Recovery Passphrase** and **Re-enter New Recovery Passphrase** fields.

5. Click **Reverse Migrate**

   The **Hardware Security Module** page appears. The red downward arrow indicates the **Status**.

6. On the Standby server, log in to the Oracle Key Vault Server through SSH as user `support`, then switch user (`su`) to `root`.

```
$ ssh support@okv_standby_instance
<Enter password when prompted>
$ su root
```

7. Modify the `okv_security.conf` file.

```
$ vi /usr/local/okv/etc/okv_security.conf
```

   - Delete the line `HSM_PROVIDER="<provider value>"`.
   - Change the value of the parameter `HSM_ENABLED` to `"0"`.

   Save and quit by entering the following sequence of characters in the vi file: `:wq!`

8. On the standby server, remove the following files:

```
$ cd /usr/local/okv/hsm/wallet
$ rm -f cwallet.sso enctdepwd
$ cd /usr/local/okv/hsm/restore
$ rm -f cwallet.sso ewallet.p12
$ cd /mnt/okvram
$ rm -f cwallet.sso ewallet.p12
$ cd /mnt/okvram/restore
$ rm -f cwallet.sso ewallet.p12
```

```
$ cd /usr/local/okv/tde
$ rm -f cwallet.sso
```

9. Switch user (`su`) to `oracle`:

```
$ su oracle
```

10. Run the following command:

```
/var/lib/oracle/dbfw/bin/orapki wallet create -wallet /usr/local/okv/
tde -auto_login
```

11. Enter the new Recovery Passphrase specified in *Step 4*.

The High Availability deployment is successfully reverse migrated.

General Troubleshooting

This section covers general troubleshooting help. Vendor-specific troubleshooting is covered in the vendor-specific notes.

Trace Files for Diagnosing Issues

Oracle Key Vault provides trace files so that you can better diagnose issues that may arise.

Use these trace files to more finely diagnose issues when you attempt hardware security module operations. These trace files are located in the */var/okv/log/hsm/* directory on the Oracle Key Vault server. To see the most recently failed operation, you can sort the trace files by their last modified time. For example, `ls -ltr /var/okv/log/hsm` lists the most recently modified trace files at the bottom of the list.

HSM Alert

Oracle Key Vault provides an alert mechanism that periodically monitors the HSM configuration to check for Root of Trust key availability and file health.

When an Oracle Key Vault server is HSM-enabled, Oracle Key Vault contacts the HSM every five minutes (or whatever you have set the monitoring interval to) to ensure that the Root of Trust key is available and the TDE wallet password can be decrypted. When an alert has been raised, an error saying `HSM configuration error. Please refer to the HSM Alert section in the Oracle Key Vault HSM Integration Guide.` appears.

If this alert appears, then follow these steps:

1. Log in as root as follows:

```
ssh support@okv_instance_IP_address
Enter support user password when prompted.
```

```
su - root
Enter root user password when prompted.
```

2. Back up the SSO wallet. For example:

```
cp /mnt/okvram/cwallet.sso
    /var/lib/oracle/cwallet_hsm_backup.sso
```

3. Contact Oracle Support.

hsm_initialize: Error Running orapki — Check User

This error occurs when the `hsm_initialize command` is run as the wrong OS user. Switch to the `oracle` user and re-run the command.

hsm_initialize: Could Not Get Slot for HSM

This error indicates that Key Vault is not properly enrolled as a client of the HSM. Check vendor-specific instructions for more information.

Key Vault Management Console Does Not Start After Restarting HSM-Enabled Key Vault Server

If the management console does not appear after restarting the HSM-enabled Key Vault server, log into the Key Vault server using SSH as user `support` and try manually opening the wallet as follows:

```
$ ssh support@okv_instance
<Enter password when prompted>
$ su root
root# su oracle
$ cd /usr/local/okv/hsm/bin
$ ./hsmclient open_wallet
```

If the `open_wallet` command succeeds, the database will open and the management console will appear, unless there is another non-HSM problem. If the command does not succeed, check for vendor-specific instructions. Otherwise, copy the output and contact Oracle Support.

High Availability Errors

1. Check that the files have been transported to the standby server:

   Execute the command `ls -l` as root on the standby server:

```
$ ls -l /usr/local/okv/hsm/wallet
-rw------- 1 oracle oinstall 324 May 16 22:57 cwallet.sso
-rw------- 1 oracle oinstall 176 May 16 22:57 enctdepwd
```

```
$ ls -l /usr/local/okv/hsm/restore
-rw------- 1 oracle oinstall 320 May 16 22:57 ewallet.p12
```

You must see `cwallet.sso` and `enctdepwd in` the `/usr/local/okv/hsm/wallet` directory and `ewallet.p12` in the `/usr/local/okv/hsm/restore` directory.

2. Check that the mode is set to HSM on the standby server:

   Open the file `okv_security.conf` as root on the standby server:

```
$ cat /usr/local/okv/etc/okv_security.conf
 Look for the line:
 HSM_ENABLED="1"
```

   You must see the number within double quotes.

3. Check the vendor-specific instructions.

Backup

You must check that the `pre_restore` command has been run on the target as follows:

Execute the command `ls -l` as root on the standby server:

```
$ ls -l /usr/local/okv/hsm/wallet
-rw------- 1 oracle oinstall 324 May 16 22:57 cwallet.sso
```

You must see the wallet file `cwallet.sso` .

You must also check that you have followed the instructions from the HSM vendor.

Restoring an HSM-Enabled Backup

This procedure must only be used in a restore operation *and* you must enter the HSM credential correctly. If you enter an incorrect credential or if Oracle Key Vault is unable to connect to the HSM, the credential will not be stored. Ensure that Oracle Key Vault is enrolled as a client of the HSM and then ensure that the correct credential has been entered.

For more information about enrolling Oracle Key Vault as a client of the HSM, see Enroll Oracle Key Vault as a Client of HSM (page 3).

Vendor Specific Notes for SafeNet

Release 12.2 BP 1 and higher support Oracle Key Vault integration with SafeNet Luna SA Hardware Security Modules from Thales version 7000. The use of a Host Trust Link (HTL) for SafeNet Luna HSM is unsupported at this time.

The following installation and enrollment instructions apply to the SafeNet Luna SA 7000 HSM.

Install the HSM Client Software on the Key Vault Server

To install the HSM client on Key Vault:

1. Obtain the SafeNet client software package, version 6.2 for Linux x64. For the purposes of this document, we will refer to this as "safenet.tar".
2. Transport the SafeNet client software package to the Key Vault machine. Oracle recommends using scp, for example:

   ```
   scp safenet.tar support@[okv hostname]:/tmp
   ```

3. Install the SafeNet client software on Key Vault.
4. Log in to the Oracle Key Vault Server through SSH as user `support`, and switch user (`su`) to `root`:

   ```
   $ ssh support@okv_instance
   $ su root
   $ cd /usr/local/okv/hsm
   $ cp /tmp/safenet.tar /usr/local/okv/hsm
   $ tar -xvf safenet.tar
   $ cd 64
   $ ./install.sh
   ```

5. Accept the SafeNet license by typing ‘y’ at the prompt.
6. Install the Luna SA by entering ‘1’, ‘n’, ‘i’ at the successive prompts:

   This installs the SafeNet software in the directory `/usr/safenet/lunaclient` .
7. Delete the safenet.tar file from /tmp directory.

   ```
   $ rm -f /tmp/safenet.tar
   ```

HSM Credential

The HSM credential is the SafeNet partition password. You choose a partition with the client `assignPartition` command.

Enroll Key Vault as a Client of HSM

To enroll Key Vault as an HSM client:

1. Set the DNS servers for Key Vault via the management console from System —> System Settings. This step is required for the Luna SA to communicate with Key Vault.
2. Exchange certificates between Key Vault and the Luna SA:

Log in to the Oracle Key Vault Server through SSH as user `support`, and switch user (`su`) to `root`:

```
$ ssh support@okv_instance
$ su root
$ cd /usr/safenet/lunaclient/bin
$ scp admin@[hsm hostname]:server.pem .
$ ./vtl addServer -n [hsm hostname] -c server.pem
$ ./vtl createCert -n [okv hostname]
$ scp /usr/safenet/lunaclient/cert/client/[okv hostname].pem admin@[hsm
hostname]:
```

You will need to enter the HSM admin password when using scp with the HSM.

3. Register Key Vault as a client of the Luna SA. This assumes you have a partition set up on the Luna SA. You can use any client name that is not yet taken. Oracle recommends using a descriptive name that will identify the Key Vault instance.

   Access the HSM administrative console by using SSH to admin@[hsm hostname] and providing the admin password:

   ```
   $ client register -client [client name] -hostname [okv hostname]
   $ client hostip map -c [client name] -i [okv IP]
   $ client assignPartition -client [client name] -partition [partition
   name]
   ```

4. Verify enrollment:

   Login to Key Vault as the support user using SSH:

   ```
   $ su root
   $ cd /usr/safenet/lunaclient/bin
   $ ./vtl verify
   ```

   The following output appears:

   ```
   The following Luna SA Slots/Partitions were found:

   Slot    Serial #        Label
   ====    ========        =====
    1      [serial #]      [partition name]
   ```

HSM Provider Value

For Safenet, the provider value is 1. If setting manually for High Availability, set HSM_PROVIDER="1". For more information about enabling HSM in a High Availability deployment, see Enabling HSM in a High Availability Deployment (page 5).

HSM Vendor Specific Checks

The instructions in this section apply to the SafeNet (Gemalto) Luna SA 7000 only.

You can verify the connection to the HSM for every Key Vault server as follows:

Login to the Key Vault server as user `support` using SSH:

```
$ ssh support@okv_instance
$ su root
$ cd /usr/safenet/lunaclient/bin
$ ./vtl verify
```

The following output appears when the HSM is set up properly:

```
The following Luna SA Slots/Partitions were found:

Slot    Serial #        Label
====    ========        =====
 1      [serial #]      [partition name]
```

If you do not see this output, it means that the HSM is not set up properly. You may diagnose further as follows:

1. Log into the Luna SA administrative console.
2. Type the command: `client show -client [client name]`
3. Verify that the expected client exists and is assigned a partition.
4. If it does not exist, register the client with the command:

   ```
   client register -client [client name]-hostname [hostname]
   ```

5. If no partition is assigned, assign a partition with the command:

   ```
   client assignPartition -client [client name] -partition [partition
   name]
   ```

6. Verify that all client IPs are mapped correctly. If entries are missing, run the command:

   ```
   client hostip map -c [client name] -i [ip]
   ```

Troubleshooting

Verify Connection between Key Vault and SafeNet
You can verify that Key Vault can reach the HSM using the `vtl verify` command as follows:

```
$ su root
root# cd /usr/safenet/lunaclient/bin
root# ./vtl verify
```

The following output appears:

```
The following Luna SA Slots/Partitions were found:

Slot    Serial #        Label
====    ========        =====
 1      [serial #]      [partition name]
```

If the command fails, it means that the Key Vault server is unable to contact the HSM. Check the vendor's other troubleshooting sections for instructions to restore `vtl verify` functionality. Contact your HSM administrator and confirm that Key Vault's access to the HSM has not been revoked. If you are unable to resolve the problem, contact Oracle Support.

Vendor Specific Notes for nCipher

Release 12.2 BP 3 and higher support Oracle Key Vault integration with the nCipher HSM. At this time, only the nCipher nShield Connect 6000+ is supported.

The following installation and enrollment instructions apply to the nCipher HSM.

Install the HSM Client Software on the Key Vault Server

The nCipher HSM requires a separate non-HSM computer on the network to use as the Remote File System. You must set up this computer and copy the nCipher software files to it before you start.

To install the nCipher software on the Key Vault server:

1. Log in to the Oracle Key Vault server as support user using SSH:

   ```
   $ ssh support$okv_instance
   <Enter the support user password when prompted>
   ```

2. Switch to root:

   ```
   $ su root
   ```

3. Go to the `root` directory and create the directories `ctls`, `hwsp`, and `pkcs11`:

   ```
   root# cd /root
   root# mkdir ctls
   root# mkdir hwsp
   root# mkdir pkcs11
   ```

4. Transfer the nCipher software installation files using the Secure Copy (SCP) protocol as follows:

   For example:

   ```
   root# scp <user@remote_file_system_machine>:/<your_source_directory>/ncipher/
   nfast/ctls/agg.tar ctls
   root# scp <user@remote_file_system_machine>:/<your_source_directory>/ncipher/
   nfast/hwsp/agg.tar hwsp
   root# scp <user@remote_file_system_machine>:/<your_source_directory>/ncipher/
   nfast/pkcs11/user.tar pkcs11
   ```

5. Install these files as follows:

```
root# cd /
root# tar xvf /root/ctls/agg.tar
root# tar xvf /root/hwsp/agg.tar
root# tar xvf /root/pkcs11/user.tar
root# /opt/nfast/sbin/install
```

6. As `root` perform additional edits on the Key Vault server:

```
root# usermod -a -G nfast oracle
root# cd /etc/rc.d/rc5.d
root# mv S50nc_hardserver S40nc_hardserver
root# cd /etc/rc.d/rc3.d
root# mv S50nc_hardserver S41nc_hardserver
```

7. Switch to user `oracle` and verify the installation:

```
root# su oracle
oracle$ PATH=/opt/nfast/bin:$PATH
oracle$ export PATH
oracle$ enquiry
```

   The state should say “operational” in the output.

8. Reboot the system for the group change to take effect.

HSM Credential

The HSM credential is the Operator Card Set password.

Enroll Key Vault as a Client of HSM

Enroll Key Vault as an HSM client as follows:

1. Add the Key Vault server IP address to the client list on the HSM using the front panel. Select privileged on any port.
2. Switch to user `oracle`:

```
root# su oracle
oracle$ PATH=/opt/nfast/bin:$PATH
oracle$ export PATH
```

3. On the Key Vault server, enroll with the HSM:

```
oracle$ nethsmenroll <HSM IP address> <HSM ESN> <HSM keyhash>
```

4. Configure TCP sockets:

```
oracle$ config-serverstartup --enable-tcp --enable-privileged-tcp
```

5. Switch to root and restart the hardserver (nCipher client process that communicates with the HSM):

```
oracle$ su root
root# /opt/nfast/sbin/init.d-ncipher restart
```

6. On the Remote File System machine run the following command:

```
rfs-setup --gang-client --write-noauth <IP address of your Key Vault server>
```

7. On the Key Vault server as user `oracle` run the commands:

```
oracle$ rfs-sync --setup --no-authenticate <IP address of Remote File System
machine>
oracle$ rfs-sync --update
```

8. Test PKCS#11 access as follows:

```
root# /opt/nfast/bin/ckcheckinst
```

   A prompt appears listing the module. You can confirm or exit.

9. Create the config file `/opt/nfast/cknfastrc` as user `root`. Write the following lines to the file:

```
CKNFAST_NO_ACCELERATOR_SLOTS=1
CKNFAST_OVERRIDE_SECURITY_ASSURANCES=explicitness;tokenkeys;longterm
```

10. Perform the steps described in Protect the Oracle Key Vault TDE Master Key with the HSM (page 3).

11. On the Key Vault server as user `oracle` run the command:

```
oracle$ /opt/nfast/bin/rfs-sync --commit
```

HSM Provider Value

For nCipher, the provider value is 2. If setting manually for HA, set `HSM_PROVIDER="2"`. For more information about enabling HSM in a High Availability deployment, see Enabling HSM in a High Availability Deployment (page 5).

Enable HSM Mode

After installing HSM software and enrolling Key Vault as an HSM client, you can enable HSM mode with nCipher from the Key Vault user interface on the management console. Just select Thales from the vendor drop-down list.

Backup and Restore

To take a backup of the Key Vault server in HSM mode:

1. Install a new Key Vault server.
2. Install the nCipher HSM software as described in a previous section.
3. From the Key Vault user interface add the backup destination on the **System Backup** page, just as you would in non-HSM mode.
4. Perform a backup as usual from the user interface on the management console.

To restore a Key Vault server from a backup:

1. Go to the **Prepare for HSM Restore** page from the user interface.
2. Select Thales from the **Vendor** drop down list and enter the HSM credential twice as requested.

3. Click **Set Credential** .
4. Log in to the Oracle Key Vault Server through SSH as user `support`, switch user (`su`) to `root`, then switch user (`su`) to `oracle`.

```
$ ssh support@okv_instance
<Enter password when prompted>
$ su root
root# su oracle
```

5. Run the following command, which retrieves information from the RFS:

```
oracle$ /opt/nfast/bin/rfs-sync --update
```

6. Restore via the user interface as usual, as in non-HSM mode.

HSM in a High Availability Oracle Key Vault Installation

This procedure shows you how to pair two Key Vault servers in HSM mode in a high availability configuration. You must enable HSM mode in both primary and standby Key Vault servers before pairing them. To configure the HSM for high availability, please see the vendor documentation.

To configure Oracle Key Vault with nCipher HSM in a high availability installation, do:

1. Install Oracle Key Vault on two servers that you mean to designate as primary and standby.
2. Install the nCipher HSM software on each Key Vault server.
3. On the server you mean to designate as primary server do the following:
   - Log in to the designated Oracle Key Vault Primary Server through SSH as user `support`, switch user (`su`) to `root`, then switch user (`su`) to `oracle`:

     ```
     $ ssh support@okv_primary_instance
     <Enter password when prompted>
     $ su root
     root# su oracle
     ```

   - Run the following command:

     ```
     oracle$ /opt/nfast/bin/rfs-sync --update
     ```

4. From the user interface on the Key Vault management console initialize the intended primary server for HSM mode with nCipher.
5. On the server you mean to designate as primary server do the following:
   - Log in to the designated Oracle Key Vault Primary Server through SSH as user `support`, switch user (`su`) to `root`, then switch user (`su`) to `oracle`:

     ```
     $ ssh support@okv_primary_instance
     <Enter password when prompted>
     $ su root
     root# su oracle
     ```

   - Run the following command:

     ```
     oracle$ /opt/nfast/bin/rfs-sync --commit
     ```

6. Repeat Step 3 on the intended standby server.

7. Perform the following manual steps on the intended primary as user `oracle`:

```
$ ssh support@okv_primary_instance
<Enter password when prompted>
$ su root
root# su oracle
oracle$ cd /usr/local/okv/hsm/wallet
oracle$ scp cwallet.sso support@standby:/tmp
oracle$ scp enctdepwd support@standby:/tmp
oracle$ cd /usr/local/okv/hsm/restore
oracle$ scp ewallet.p12 support@standby:/tmp
```

> **Note:**
>
> While performing this procedure on Oracle Key Vault 12.2.0.5.0 and earlier, use the commands in Enabling an HSM in a High Availability Oracle Key Vault Installation (page 25).

8. Perform the following manual steps on the intended standby as user `root`:

```
$ ssh support@okv_standby_instance
<Enter password when prompted>
$ su root
root# cd /usr/local/okv/hsm/wallet
root# mv /tmp/enctdepwd .
root# mv /tmp/cwallet.sso .
root# chown oracle *
root# chgrp oinstall *
root# cd /usr/local/okv/hsm/restore
root# mv /tmp/ewallet.p12 .
root# chown oracle *
root# chgrp oinstall *
```

> **Note:**
>
> While performing this procedure on Oracle Key Vault 12.2.0.5.0 and earlier, use the commands in Enabling an HSM in a High Availability Oracle Key Vault Installation (page 25).

9. Continuing as user `root` open the file `okv_security.conf` for writing:

```
root# vi /usr/local/okv/etc/okv_security.conf
```

10. Make two updates to the file as follows:

    a. Set the variable `HSM_ENABLED` to 1. If the variable does not exist, add it and set its value to 1.

    ```
    HSM_ENABLED="1"
    ```

    b. Add the following line:

    ```
    HSM_PROVIDER="2"
    ```

11. Then proceed to set up high availability as usual via the user interface on the Key Vault management console.

CNSA Suite Support

Oracle Key Vault 12.2 BP 3 and higher offer compliance with the Commercial National Security Algorithm (CNSA) for TLS connections to and from the appliance.

The CNSA suite is a list of strong encryption algorithms and key lengths, that offer greater security and relevance into the future. A link to the full CNSA specification is in the **See Also** section that follows this section.

Note that 12.2 BP3 and higher do not provide complete compliance across every component in the system. You will be able to switch to the CNSA algorithms, where available by means of two scripts that are packaged with the ISO:

1. The first script `/usr/local/okv/bin/okv_cnsa` makes configuration file changes to update as many components as possible to use the enhanced algorithms. It is reversible and will not interfere with existing operations.
2. The second script `/usr/local/okv/bin/okv_cnsa_cert` regenerates CNSA compliant public key pairs and certificates.

> **Note:**
>
> The second script `/usr/local/okv/bin/okv_cnsa_cert` is disruptive because it replaces the old key pairs with new ones. This has consequences for the following operations:
>
> - **Endpoint Enrollment**: Enroll endpoints after running this script when possible. If you had endpoints enrolled before running the CNSA script, you must re-enroll them so that fresh CNSA compliant keys are generated using CNSA algorithms.
> - **High Availability**: Run the CNSA scripts on both Oracle Key Vault instances before pairing them in a high availability configuration when possible. If you had high availability set up prior to running the CNSA scripts, you must re-configure high availability as follows: unpair the primary and standby servers, run the CNSA scripts individually on each server, then pair them again.

Running the CNSA Scripts

To run the CNSA scripts do the following:

1. Install Oracle Key Vault and complete the post-installation tasks. The last post-installation task is to set the support user password, which is needed now.
2. Log into the Key Vault browser-based management console and enable SSH access to the server.

3. SSH into the Key Vault server as the support user. Enter the support user password created during post-installation, when prompted.

```
$ ssh support@okv_instance
<Enter support user password created during post-installation>
```

4. Change to root user:

```
$  su root
```

5. Run the scripts as root user:

```
root#  /usr/local/okv/bin/okv_cnsa
root#  /usr/local/okv/bin/okv_cnsa_cert
```

6. The scripts put data into `/usr/local/okv/etc/okv_security.conf`.

   The line `USE_ENHANCED_ALGORITHMS_ONLY="1"` will be added if the scripts are run.

Backup

After restoring a backup, re-run the first script:`/usr/local/okv/bin/okv_cnsa` to update the configuration to use the enhanced CNSA algorithms as follows:

1. Wait for the system to reboot after the restore operation initiated via the user interface of the Key Vault management console.
2. SSH into the Key Vault server as the support user:

```
$ ssh support@okv_instance
<Enter support user password created during post-installation>
```

3. Switch to root user:

```
$   su root
```

4. Run the first CNSA script :

```
root#  /usr/local/okv/bin/okv_cnsa
```

Upgrade

You must re-run the first script during the upgrade to ensure CSNA compliance as follows:

1. Execute **Step 8** of the upgrade procedure which is to run the ruby script as root:

```
root# /usr/bin/ruby/images/upgrade.rb --format
```

2. Run the first CNSA script :

```
root#  /usr/local/okv/bin/okv_cnsa
```

3. Continue with **Step 9** of upgrade procedure:

```
root# /sbin/reboot
```

Limitations:

- CNSA compliance is not supported for some components in the Oracle Key Vault infrastructure, for example SSH, or for the database encryption via TDE.
- The Firefox browser is not supported for use with the Oracle Key Vault management console when CNSA is enabled. This is because the Firefox browser does not support CNSA-approved cipher suites.

See Also:

- https://cryptome.org/2016/01/CNSA-Suite-and-Quantum-Computing-FAQ.pdf
- Post Installation Tasks
- Setup SSH Access
- Upgrade Procedure

Commands used in Oracle Key Vault 12.2.0.5.0 and Earlier

The following commands are used in Oracle Key Vault 12.2.0.5.0 and earlier:

- Enabling an HSM in a High Availability Oracle Key Vault Installation (page 25)
- Enabling the HSM_ENABLED Parameter in the okv_security.conf File (page 26)

Enabling an HSM in a High Availability Oracle Key Vault Installation

While performing the procedure "*HSM in a High Availability Oracle Key Vault Installation*" under Vendor Specific Notes for nCipher (page 18) on Oracle Key Vault 12.2.0.5.0 and earlier, use the following commands:

- Perform the following manual steps on the intended primary as user `oracle`:

```
$ ssh support@okv_primary_instance
<Enter password when prompted>
$ su root
root# su oracle
oracle$ cd /usr/local/okv/hsm/wallet
```

```
oracle$ scp cwallet.sso support@(standby):/tmp
oracle$ scp enctdepwd support@(standby):/tmp
```

- Perform the following manual steps on the intended standby as user `root`:

```
$ ssh support@okv_standby_instance
<Enter password when prompted>
$ su root
root# cd /usr/local/okv/hsm/wallet
root# mv /tmp/enctdepwd .
root# mv /tmp/cwallet.sso .
root# chown oracle *
root# chgrp oinstall *
```

Enabling the HSM_ENABLED Parameter in the okv_security.conf File

While performing the procedure Enable HSM in a High Availability Key Vault Installation (page 5) on Oracle Key Vault 12.2.0.5.0 and earlier, use the following commands.

- Perform the following manual steps on the primary node as user oracle:

```
$ cd /usr/local/okv/hsm/wallet
$ scp cwallet.sso support@standby:/tmp
$ scp enctdepwd support@standby:/tmp
```

- Enable the `HSM_ENABLED` parameter in the `okv_security.conf` file:

```
$ cd /usr/local/okv/hsm/wallet
$ mv /tmp/enctdepwd .
$ mv /tmp/cwallet.sso .
$ chown oracle *
$ chgrp oinstall *
$ vi /usr/local/okv/etc/okv_security.conf
   Set HSM_ENABLED="1"
   Set HSM_PROVIDER="<provider value>"
```

Save and quit by entering the following sequence of characters in the vi file: `:wq!`

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Oracle® Key Vault Integration with Hardware Security Module (HSM), Release 12.2 BP10
E75841-13

Copyright © 2016, 2019, Oracle and/or its affiliates

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.