# Oracle Key Vault 18.x

## nShield® HSM Integration Guide

2024-10-21

# Table of Contents

# Chapter 1. Introduction

The nShield Hardware Security Module (HSM) can be used to generate and store a Root of Trust (RoT) that protects security objects used by Oracle Key Vault to safeguard users' keys and credentials. The HSM can be used in FIPS 140 Level 2 or Level 3 mode to meet compliance requirements. An Oracle Key Vault cluster node can have multiple HSMs enrolled, as long as the HSMs are in the same Security World.

An existing Oracle Key Vault deployment cannot be migrated to use an HSM as a RoT.

Oracle Key Vault can function only if the RoT stored in the HSM is available.

To restart or restore Key Vault in HSM mode when Operator Card Set (OCS) protection is used, the OCS for the HSM must be in slot 0 of the HSM.

## 1.1. Product configurations

We have successfully tested nShield HSM integration with Oracle Key Vault in the following configurations:

| Product | Version |
|---|---|
| Operating System | Oracle Linux 7 64-bit |
| Oracle Key Vault Version | 18.6 |

## 1.2. Supported nShield hardware and software versions

We have successfully tested with the following nShield hardware and software versions:

### 1.2.1. Connect XC

| Security World Software | Firmware | Image | OCS | Softcard | Module |
|---|---|---|---|---|---|
| 12.60.11 [1] | 12.50.11 (FIPS 140-2 certified) | 12.60.10 | ✓ | ✓ | ✓ |

[1] The 12.60.11 version requires the `redist` package. For installation information, see the *Installation Guide* for the HSM.

## 1.3. Supported nShield functionality

| Feature | Support | Feature | Support | Feature | Support |
|---|---|---|---|---|---|
| Key generation | Yes | 1-of-N Operator Card Set | Yes | FIPS 140 Level 3 support | Yes |
| Key managemen t | Yes | k-of-N Operator Card Set | No | Common Criteria support | Yes |
| Key import | Yes | Softcards | Yes | Load sharing | Yes |
| Key recovery | Yes | Module-Only key | Yes | Fail over | Yes |

## 1.4. Requirements

Before installing these products, read the associated documentation:

- For the nShield HSM: *Installation Guide* and *User Guide*.
- If nShield Remote Administration is to be used: *nShield Remote Administration User Guide*.
- Oracle Key Vault documentation (https://docs.oracle.com/en/database/oracle/key-vault).

In addition, the integration between nShield HSMs and Oracle Key Vault requires:

- A separate non-HSM machine on the network to use as the Remote File System for the HSM. The RFS machine can also be used as a client to the HSM, to allow presentation of Java Cards using nShield Remote Administration. See the *nShield Remote Administration User Guide*.
- PKCS #11 support in the HSM.
- A correct quorum for the Administrator Card Set (ACS).
- Operator Card Set (OCS), Softcard, or Module-Only protection.
  - If OCS protection is to be used, a 1-of-N quorum must be used.
- Firewall configuration with usable ports:
  - 9004 for the HSM (hardserver).
  - 8200 for Key Vault.

Furthermore, the following design decisions have an impact on how the HSM is installed and configured:

- Whether your Security World must comply with FIPS 140 Level 3 standards.
  - If using FIPS 140 Level 3, it is advisable to create an OCS for FIPS authorization. The OCS can also provide key protection for the Vault master key. For information about limitations on FIPS authorization, see the *Installation Guide* of the nShield HSM.
- Whether to instantiate the Security World as recoverable or not.

Entrust recommends that you allow only unprivileged connections unless you are performing administrative tasks.

## 1.5. More information

For more information about OS support, contact your Oracle Key Vault sales representative or Entrust nShield Support, https://nshieldsupport.entrust.com.

Access to the Entrust nShield Support Portal is available to

> customers under maintenance. To request an account, contact nshield.support@entrust.com.

# Chapter 2. Procedures

The high-level procedure to install and configure one or more Oracle Key Vault servers with one or more nShield HSMs is as follows:

1. Install the required number of instances of Oracle Key Vault. For instructions, see the Oracle Key Vault documentation.
2. Install and configure the required number of HSMs and the Security World software, including setting up the Remote File System (RFS) or Remote Administration. For instructions, see the *Installation Guide* for your HSM.
    - nShield HSMs require a separate non-HSM machine on the network to use as the RFS. You must set up this machine and copy the nShield Security World Software files to it before you install the HSM client software on Oracle Key Vault servers.
    - All enrolled HSMs must be in the same Security World and must have access to the OCS in slot 0 if OCS-protection is used. If the HSM whose slot 0 is used is enrolled on each of the Key Vault servers, the Key Vault web user interface has access to all of the HSMs, as long as they are in the same Security World.
    - If dynamic slots are to be used on the HSMs, set up Remote Administration and configure slot mapping.
3. Install the HSM client software on the Oracle Key Vault server(s).
4. Enroll the Key Vault(s) as client(s) of the HSM(s).
5. Enable HSM mode in the Oracle Key Vault web user interface.
6. If you have a high availability Oracle Key Vault environment, enroll your HSM and configure initialization of the HSM in each of the nodes.

## 2.1. Install HSM client software on the Key Vault server

Perform these steps on the Oracle Key Vault server.

If you have a high availability Oracle Key Vault environment, perform these steps:

- In a primary-standby architecture, on both the primary and the standby.
- In a cluster architecture, on each Key Vault instance to be added to the cluster.

To install HSM client software on the Key Vault server:

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

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

2. Switch to root:

```
$ su root
```

3. Install the latest version of the Security World software as described in the *Installation Guide* for the HSM.

We recommend that you uninstall any existing nShield software before installing the new nShield software.

4. Create the Security World as described in the *User Guide*, creating the ACS and OCS that you require.
5. As root, perform additional edits on the Key Vault server:
   a. Add the nfast group to the oracle user

```
root# usermod -a -G nfast oracle
```

6. Switch to the `oracle` user, and verify the installation:

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

The mode should say `operational` in the output. For example:

```
Server:
 enquiry reply flags  none
 enquiry reply level  Six
 serial number
 mode                 operational
```

7. Restart the Oracle Key Vault server for the group change to take effect.

To restart or restore Key Vault in HSM mode when OCS protection is used, the OCS for the HSM must be in slot 0 of the HSM.

8. As the `root` user, set firewall rules to enable port 9004 for the hardserver (the client process in the nShield Security World software that communicates with the HSM).

## 2.2. Enroll Key Vault as a client of the HSM

1. Add the Key Vault server IP address to the client list on the HSM using the front panel or via an update to the Connect configuration file. For instructions, see the *User Guide* for your HSM.
   - Select privileged on any port.
   - If you have a high availability Oracle Key Vault environment, add the IP addresses of all Key Vault servers to the client list on all HSMs.
2. To obtain the ESN and keyhash for the `nethsmenroll` command in the next step, run the `anonkneti` command:

   ```
   anonkneti <HSM IP address>
   ```

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

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

4. Run the `enquiry` command:

   ```
   enquiry
   ```

   Verify that the HSM mode is operational and the hardware status is OK.

5. Configure TCP sockets:

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

6. Switch to root and restart the hardserver:

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

7. On the Remote File System machine, run the following command:

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

8. If OCS protection is intended to be used but the Security World has not been created yet, edit the `cardlist` file to enable Java Cards for use through dynamic slots. If the Security World has been created with this RFS, this configuration is already enabled.

a. Go to the following folder on the RFS:

```
#/opt/nfast/kmdata/config
```

b. Open the `cardlist` file in a text editor.

c. Add an asterisk (*) to authorize all Java Cards for dynamic slots.

If only certain Java Cards are authorized for this use, list them by their serial number. For example:

```
4286005559064791
4286005559064792
4286005559064793
```

d. Copy the updated `cardlist` file from the RFS to all clients.

9. On the Key Vault server as the `oracle` user, run the following commands:

```
oracle$ rfs-sync --setup <IP address of Remote File System machine>
oracle$ rfs-sync --update
```

10. As the `root` user, create the `/opt/nfast/cknfastrc` configuration file for PKCS#11 variables. For information on these variables, see the *User Guide* for your HSM.

a. OCS protection

If you are using OCS or module protection, `cknfastrc` needs the following set:

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

b. Softcard Protection

If you are using softcard protection, then `CKNFAST_LOADSHARING` must be set. This is not supported alongside the Module-only Key protection settings. See also Known issues.

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

c. Module Protection

If you are using Module-Only protection, `cknfastrc` needs the following set:

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

11. On the Key Vault Server, test PKCS#11 access as follows:

    ```
    oracle$ /opt/nfast/bin/ckcheckinst
    ```

    Select slot number to run library test. Various slots are displayed, depending on your configuration.

    Example 1:

    ```
    0 Fixed token "accelerator"
    1 Operator card "OKV_OCS"
    ```

    Example 2:

    ```
    0  Operator card     "OKV_OCS"
    1  Soft token        "OKV_Softcard"
    ```

    Test execution:

    ```
    Test                    Pass/Failed
    ----                    -----------

    1 Generate RSA key pair   Pass
    2 Generate DSA key pair   Pass
    3 Encryption/Decryption   Pass
    4 Signing/Verification    Pass

    Deleting test keys         ok

    PKCS#11 library test successful.
    ```

## 2.3. Enable HSM mode in Key Vault

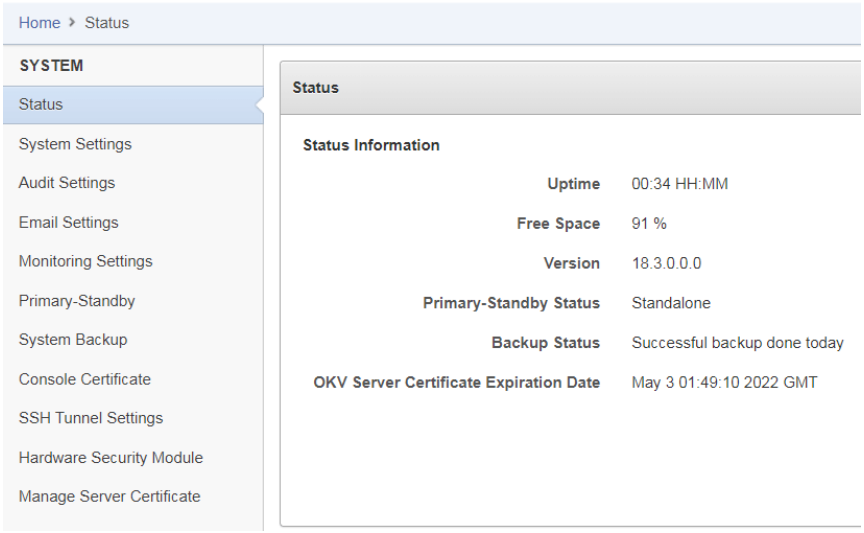
After installing HSM software and enrolling Key Vault as an HSM client, you can enable HSM mode with nShield HSM(s) from the Key Vault web user interface. This will protect the Oracle Key Vault Root of Trust key with the HSM.

1. Log into the Oracle Key Vault web user interface as a Key Administrator.

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

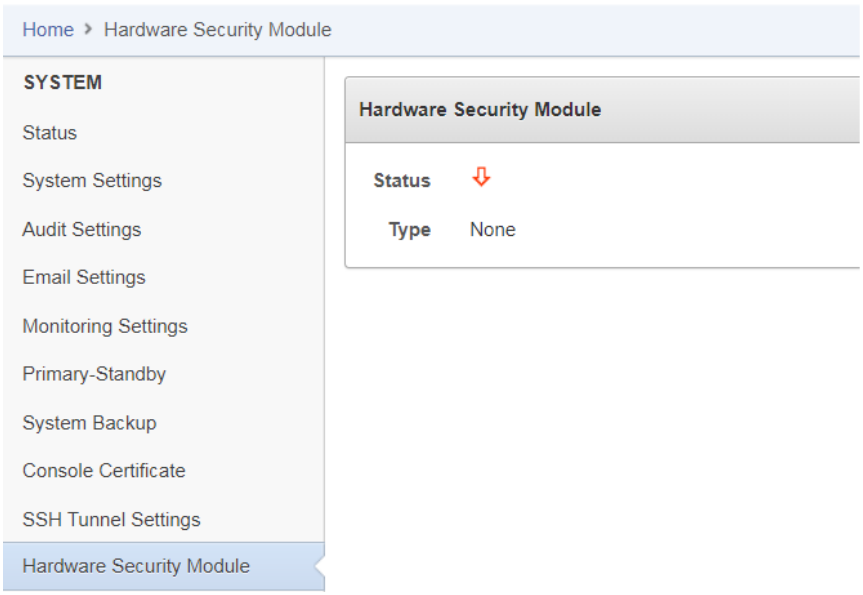
2. Select the **System** tab.

   The **Status** page appears.




3. Select **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. Select **Initialize**.

   The **Initialize HSM screen** appears.

5. From the **Vendor** list, select **nCipher**:

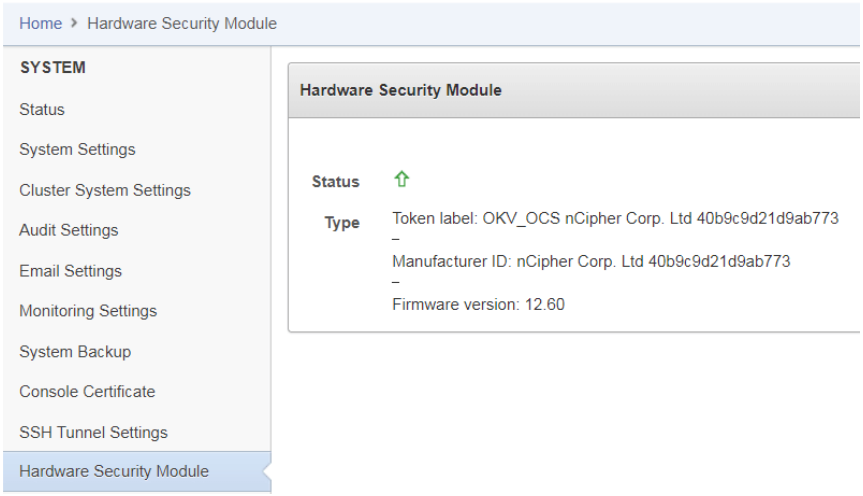
6. Enter a password two times: first in **HSM Credential** and second in **Re-enter HSM Credential**.
    - If you are using OCS protection, then your OCS passphrase needs to be entered in twice with your card presented in slot 0.
    - If you are using Softcard protection, then the Softcard passphrase needs to be entered twice.
    - If you are using Module-Only protection, enter a password that you set up for this credential check.

    The password will be needed in the future, for example for reverse migration.

7. Enter the recovery passphrase for Oracle Key Vault.
8. Select **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.




The **Token** label is **accelerator** if Module-Only protection is used.

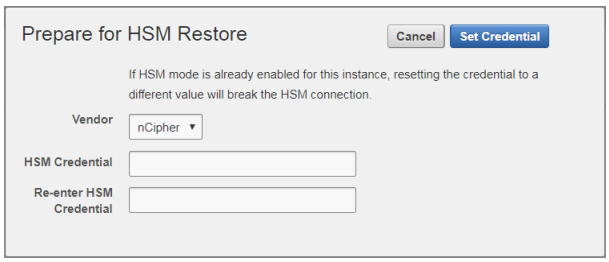
Only the first two numbers of the firmware are displayed.

9. After a successful initialize operation of the nShield HSM, run the following command as the `oracle` user on the Key Vault server:

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

If you change the HSM credential on the HSM after initialization,

> you must also update the HSM credential on the Oracle Key Vault server: In the **Vendor** list select **nCipher**, then select the `Set Credential` button.




## 2.4. Enable the HSM in a primary-standby high availability deployment

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

1. Install Oracle Key Vault on two servers that you mean to designate as primary and standby.
2. Install the nShield Security World Software on each Oracle Key Vault server, see Install HSM client software on the Key Vault server.
3. Enroll the primary and standby nodes as clients of the HSM, see Enroll Key Vault as a client of the HSM.
4. From the Oracle Key Vault web user interface, initialize the intended primary server for HSM mode with nShield HSM(s), see Enable HSM mode in Key Vault.
5. On the primary server, run the following commands as the `oracle` user:

```
$ 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@<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
```

6. On the standby server, run the following commands as the `root` user:

```
$ 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 *
```

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

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

   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"
   FIPS_ENABLED="0"
   HSM_FIPS_ENABLED="1"
   OKV_OCI_INSTALL="DISABLED"
   HSM_TOKEN_LABEL=""
   HSM_KEY_EXTRACTABLE="0"
   ```

8. Make updates to the `okv_security.conf` file as follows:

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

   ```
   HSM_ENABLED="1"
   ```

   Add the following line:

   ```
   HSM_PROVIDER="2"
   ```

   If using softcards or OCS cards, enter the name of the cardset:

   ```
   HSM_TOKEN_LABEL="card_set_name"
   ```

9. On the standby server, run the `rfs-sync --update` command as the `oracle` user:

   ```
   root# su oracle
   oracle$ /opt/nfast/bin/rfs-sync --update
   ```

10. Without restarting the Oracle Key Vault instances, navigate to the web user interfaces of the primary and standby servers, and configure primary-standby

via the Oracle Key Vault web user interface. For information on the configuration and settings, see the Oracle documentation.

## 2.5. Reverse migration operations to a local wallet

Reverse migrating an HSM-enabled Oracle Key Vault server reverts the Key Vault server to using the recovery passphrase to protect the TDE wallet. This operation is necessary if the HSM that protects Oracle Key Vault must be decommissioned.

- Reverse migrating a standalone deployment

  You can reverse migrate a standalone deployment by using the Oracle Key Vault web user interface.

- Reverse migrating a primary-standby deployment

  To reverse migrate a primary-standby deployment, use both the Oracle Key Vault web user interface and the command line.

- Reverse migrating a multi-master cluster

  You can reverse migrate a multi-master cluster by using the Oracle Key Vault web user interface.

### 2.5.1. Reverse migrate a standalone deployment

You can reverse migrate a standalone deployment by using the Oracle Key Vault web user interface.

1. Log into the Oracle Key Vault web user interface as a Key Administrator.

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

2. Select the **System** tab.

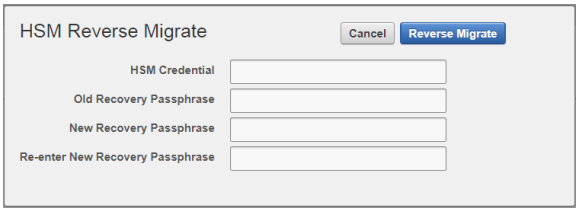
   The **Status** page appears.

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

   The **Hardware Security Module** page appears.

4. Select **Reverse Migrate**.

   The **HSM Reverse Migrate** dialog box is displayed.




5. On the **HSM Reverse Migrate** dialog box, enter the following details:
   a. Enter the HSM credential in the **HSM Credential** field. For nShield HSMs, the credential is what you use for OCS, Softcard, or Module-Only protection.
   b. Enter the old recovery passphrase in the **Old Recovery Passphrase** field.
   c. Enter the new recovery passphrase in the **New Recovery Passphrase** and **Re-enter New Recovery Passphrase** fields.
6. Select **Reverse Migrate**.
7. The **Hardware Security Module** page appears. The red downward arrow indicates the **Status**.

## 2.5.2. Reverse migrate a primary-standby deployment

To reverse migrate a primary-standby deployment, use both the Oracle Key Vault web user interface and the command line.

1. On the primary server, log into the Oracle Key Vault web user interface as a Key Administrator.

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

2. Select the **System** tab.

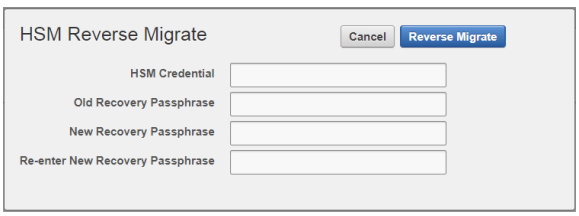
   The **Status** page appears.

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

   The **Hardware Security Module** page appears.

4. Select **Reverse Migrate**.

   The **HSM Reverse Migrate** dialog box is displayed.



5. On the **HSM Reverse Migrate** dialog box, enter the following details:
   a. Enter the HSM credential in the **HSM Credential** field. For nShield HSMs, the credential is what you use for OCS, Softcard, or Module-Only protection.
   b. Enter the old recovery passphrase in the **Old Recovery Passphrase** field.
   c. Enter the new recovery passphrase in the **New Recovery Passphrase** and **Re-enter New Recovery Passphrase** fields.
6. Select **Reverse Migrate**.

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

7. On the standby server, log in through SSH as the `support` user, then, with the `su` command, switch to the `root` user.

```
$ ssh support@<okv_standby_instance>
$ su root
```

   Modify the `okv_security.conf` file.

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

   - Delete the line `HSM_PROVIDER="2"`.
   - Change the value of the parameter `HSM_ENABLED` to 0.
   - Check that the parameter `HSM_TOKEN_LABEL` is set to "".

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 that you specified in Step 5.

```
Enter wallet password:
Operation is successfully completed.
```

The primary-standby deployment is successfully reverse migrated.

### 2.5.3. Reverse migrate a multi-master cluster

You can reverse migrate a multi-master cluster by using the Oracle Key Vault web user interface.

1. Log into the Oracle Key Vault web user interface as a Key Administrator.

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

2. Select the **System** tab.

   The **Status** page appears.

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

   The **Hardware Security Module** page appears.

4. Select **Reverse Migrate**.

   The **HSM Reverse Migrate** dialog box is displayed.


5. In the **HSM Reverse Migrate** dialog box, enter the following details:
   a. Enter the HSM credential. For nShield HSMs, the credential is what you use for OCS, Softcard, or Module-Only protection.
   b. Enter the recovery passphrase.
6. Select **Reverse Migrate**.

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

# 2.6. Configure an HSM for a multi-master cluster

You can configure HSMs in a multi-master cluster with a single node or multiple nodes. In a multi-master Oracle Key Vault installation, any Key Vault node in the cluster can use any HSM. The nodes in the multi-master cluster can use different TDE wallet passwords (recovery passwords), RoT keys, and HSM credentials.

To ensure complete security, you must HSM-enable all Oracle Key Vault nodes in the cluster.

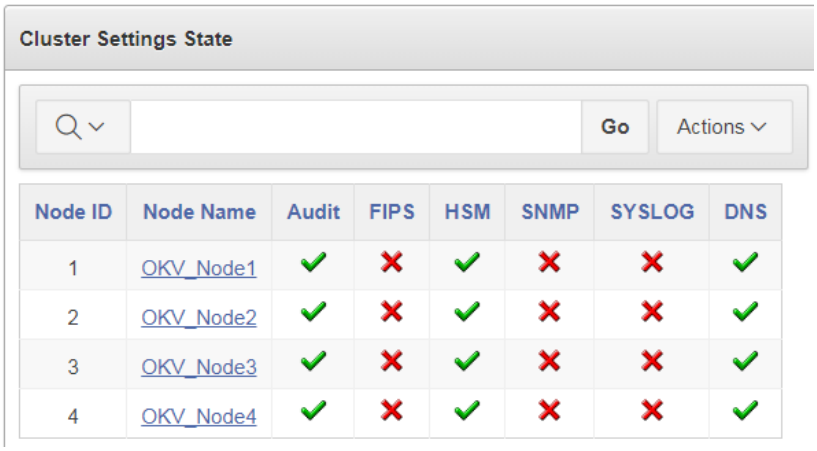
## 2.6.1. Configure an HSM for a multi-master cluster with a single node

To use an HSM with a multi-master cluster, you should start with a single HSM-enabled node and add additional HSM-enabled nodes. Oracle recommends the following steps to configure an HSM for a multi-master cluster with a single node:

1. Configure the first node of the cluster.
2. Configure the HSM on the first node before adding any new nodes. If there is already more than one node in the cluster, then configure the HSM for a multi-master cluster with multiple nodes. See Configure an HSM for a multi-master cluster with multiple nodes.
3. HSM-enable the candidate node before adding it to the cluster.
4. Add the HSM-enabled candidate node to the cluster using a controller node that is also HSM-enabled.

   If any node in the cluster is already HSM-enabled, you cannot add a new node that is not HSM-enabled.

5. Verify that the HSM is enabled:
   a. In the Oracle Key Vault web user interface, select the **Cluster** tab.
   b. Select **Monitoring** in the left sidebar.
   c. Check that the **Cluster Settings State** has all green ticks for **HSM**:



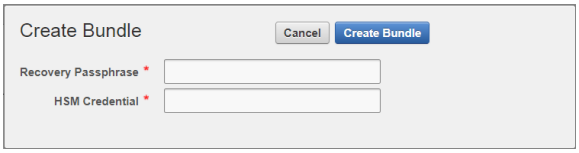

An enabled HSM does not mean that the HSM is active. The status only indicates whether the HSM is enabled for these nodes. To check whether the HSM is active, use the status information on the **Hardware Security Module** page of the web user interface.

## 2.6.2. Configure an HSM for a multi-master cluster with multiple nodes

You can configure HSM for multiple nodes by copying information from an HSM-enabled node to the non-enabled nodes.

For instructions to configure an HSM for a multi-master cluster, see Configure an HSM for a multi-master cluster with a single node. If the first node to be HSM-enabled is in a cluster that already has multiple nodes, then you must manually copy information from that HSM-enabled Oracle Key Vault to the other Oracle Key Vault installations in the cluster before you can enable HSM in any other nodes. If the first node to be HSM-enabled has a read-write peer, then the read-write peer will not be able to decrypt the information from the HSM-enabled node until the bundle is copied and applied successfully to the read-write peer.

1. Log in to the Oracle Key Vault web user interface as a Key Administrator.
2. Select the **System** tab.
3. On the left side of the **System** page, select **Hardware Secure Module**.
4. On the HSM-enabled node, select **Create Bundle** on the **Hardware Security Module** page.
5. Enter the recovery passphrase.




6. Log in to the HSM-enabled node through SSH as the `support` user.

```
ssh support@<hsm_enabled_node>
```

7. Switch to the `root` user.

```
su root
```

8. Copy the bundle to each node using the node IP addresses:

```
scp /usr/local/okv/hsm/hsmbundle support@<ip_address>:/tmp
```

9. Log in to each node in the cluster, except the original HSM-enabled node, using the node IP address:

```
ssh support@<ip_address>
```

10. Switch to the `root` user.

```
su root
```

11. Perform the following steps on each node to copy the bundle to the /usr/local/okv/hsm location and apply user and group ownership:

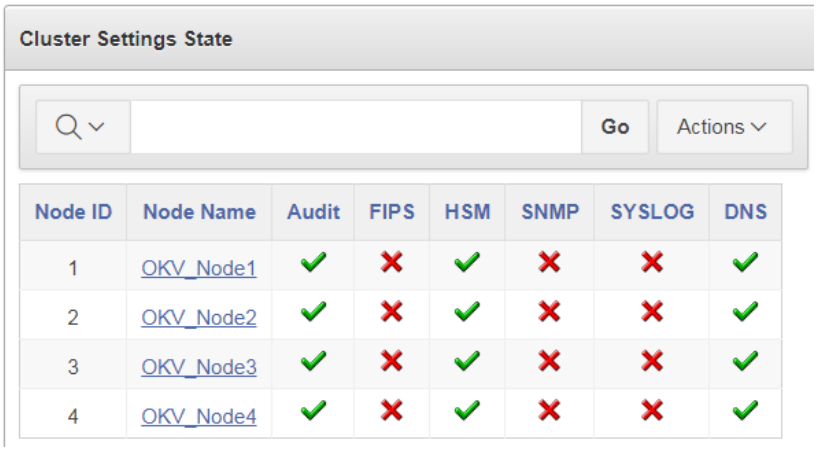
```
cp /tmp/hsmbundle /usr/local/okv/hsm/
chown oracle:oinstall /usr/local/okv/hsm/hsmbundle
```

12. On each node except the original HSM-enabled node, select **Apply Bundle** on the **Hardware Security Module** page. Enter the recovery passphrase.

    If you plan on reverse-migrating the original HSM-enabled node, you must apply the bundle immediately on all nodes first.

13. Proceed to HSM-enable each of these nodes in the same way that you HSM-enabled the first node. Also verify that each HSM is enabled:
    a. In the Oracle Key Vault web user interface, select the **Cluster** tab.
    b. Select **Monitoring** in the left sidebar.

c. Check that the **Cluster Settings State** has all green ticks for **HSM**:




| Node ID | Node Name | Audit | FIPS | HSM | SNMP | SYSLOG | DNS |
|---|---|---|---|---|---|---|---|
| 1 | OKV_Node1 | ✓ | ✗ | ✓ | ✗ | ✗ | ✓ |
| 2 | OKV_Node2 | ✓ | ✗ | ✓ | ✗ | ✗ | ✓ |
| 3 | OKV_Node3 | ✓ | ✗ | ✓ | ✗ | ✗ | ✓ |
| 4 | OKV_Node4 | ✓ | ✗ | ✓ | ✗ | ✗ | ✓ |

An enabled HSM does not mean that the HSM is active. The status only indicates whether the HSM is enabled for these nodes. To check whether the HSM is active, use the status information on the **Hardware Security Module** page of the web user interface.

14. After HSM-enabling a node in a cluster, the `rfs-sync --update` command must be run on all other nodes to ensure that all nodes have up-to-date Security World files.
15. After you have HSM-enabled all nodes and verified the replication between all nodes, remove the `hsmbundle` file from all of the nodes.

## 2.7. Configure backup of the Key Vault server in HSM mode

1. Install a new Key Vault server.
2. Install the nShield Security World Software as described in Install HSM client software on the Key Vault server.
3. From the Key Vault web 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 web user interface.

## 2.8. Restore from a Key Vault backup in HSM mode

To restart or restore Key Vault in HSM mode when OCS protection is used, the OCS for the HSM must be in slot 0 of the

> HSM.

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 the HSM prior to this step.
  1. If OCS protection is used, present the OCS card to the HSM.
  2. Log into the Oracle Key Vault web user interface as a user with system administrative privileges.

     The **Oracle Key Vault Home** page appears.
  3. Select the **System** tab.

     The **Status** page appears.
  4. Select **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.
  5. Select **Set Credential**.

     The **Prepare for HSM Restore** screen appears.
  6. From the **Vendor** list, select **nCipher** and enter the HSM credential twice as requested.
  7. Select **Set Credential**.

     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.
  8. Go to the **Restore** page via the Key Vault web user interface and restore the Key Vault backup.

## 2.9. Restart or restore in HSM mode using nShield Remote Administration

To restart or restore Key Vault in HSM mode when OCS protection is used, the OCS for the HSM must be in slot 0 of the HSM. The `raserv` package of nShield software is only available on the nShield RFS machine, it is not supported on Oracle Key Vault servers. When the Oracle Key Vault server restarts or restores from a backup and Java Cards cannot be presented to the HSMs that are enrolled to that server, the restart or restore will fail. If the HSM is also enrolled to the RFS, you can present Java Cards there when the RFS is operational. This way, when the Oracle Key Vault server comes back up, it can still access the keys from the HSM using the OCS in slot 0.

# Chapter 3. Known issues

| Issue | Action for Integrator |
| --- | --- |
| If you want to use softcards as a means of protection, you need to set `CKNFAST_LOADSHARING=1` in `cknfastrc`, but this causes the firmware version to display as **0.0** when the Oracle Key Vault server is initialized with the HSM. | None. |

# Chapter 4. Migrate from Oracle TDE to Oracle Key Vault

The section describes the following migration from Oracle TDE to Oracle Key Vault:

TDE environment:

- Non-multitenant database.
- The TDE database is configured to use the following key encryption methods:

  **File Keystore**

  KeystorePasswod1

  **HSM Softcard**

  softcard1|scard1

- The TDE database encryption is using the HSM Softcard.
- Set up using the instructions in https://docs.oracle.com/en/database/oracle/key-vault/18.6/okvag/security_objects.html#GUID-4FDDCC75-3AC1-4ABC-83C1-BF437487B574

Oracle Key Vault environment:

- Standalone.

## 4.1. Hardware and software versions

- VM running Oracle Database 19c (TDE Environment)
- VM running Oracle Key Vault 18.6
- Operating System: Red Hat 7 Linux 64

## 4.2. Scenario

A customer has nShield HSMs, probably Connect XCs, nShield Security World software, and is using Oracle TDE that uses that HSM in a Security World. They want to migrate to Oracle Key Vault and use the nShield HSM to protect the OKV master key. They also want to enroll their existing TDE database as an endpoint, along with any other endpoints they might use.

Assumptions:

- nShield software and all Security World files are installed on a separate RFS.
- nShield software is installed and Security World files are synchronized to the Oracle TDE clients.
- Oracle TDE servers are enrolled as clients to the HSMs.

Procedure overview:

1. Reverse migrate the TDE with HSM setup to use an Oracle software wallet only.

   Verify that their data can still be encrypted and decrypted using the software wallet, and can read the columns.

2. Un-enroll the HSM from the TDE server (using `nethsmenrol -r` if using an nShield Connect).
3. Uninstall the Security World software from the TDE server, backing up any Security World files that haven't been synced to the RFS.

   At this stage they have an Oracle TDE setup using only the software wallet.

   There is no nShield software or hardware in the configuration.

   The Security World files are on the RFS and also a backup.

   There is no reason to have nShield software installed or the HSM enrolled on the TDE server at this point as they are not going to be used after this step.

4. Set up a separate server with an Oracle Key Vault installation and enroll the Oracle TDE client that is using a software-based wallet as an endpoint within OKV.
5. Verify that you can still encrypt and decrypt data and read columns on the TDE end.
6. Execute the steps in Install HSM client software on the Key Vault server to install Security World software on the OKV server.
7. Enroll the HSM and configure RFS sync so the Security World files are present. There is the option to create a brand new Security World at this point.
8. Execute the steps in Enable HSM mode in Key Vault to initialize the HSM using the OKV web UI and verify the status is active.
9. Verify that you can still encrypt and decrypt within Oracle TDE and read the columns.

   The nShield HSM is not used to perform encryption or decryption. It only protects the master key protected by OKV master key(s), which in turn is

protected by the HSM.

Some of these steps can be interchangeable. You can initialize OKV with an HSM and then enroll the endpoint afterwards.

## 4.3. Prerequisites

- The OKV system is already set up and is ready to receive the migration.
- The steps below start from migrating back from the HSM wallet to the software wallet, and then migrating the wallet to the OKV environment.
  - Software wallet: KeystorePassword1
  - HSM wallet: softcard1|scard1
  - OKV wallet: okvwallet
- All SQL commands are to be executed as the `oracle` user whose environment is configured.

  The `oracle` user will run the `sqlplus` utility as the `sysdba` user.

```
% sqlplus / as sysdba
```

## 4.4. Migrate back from HSM wallet to Software wallet

1. Set the Keystore configuration to `FILE`:

```
ALTER SYSTEM SET TDE_CONFIGURATION = "KEYSTORE_CONFIGURATION=FILE" SCOPE=BOTH SID='*';
```

   Exit the `sqlplus` section and establish a new section so this takes effect.

2. Show the wallet parameters:

```
SHOW PARAMETER WALLET_ROOT;
SHOW PARAMETER TDE_CONFIGURATION;
```

3. Make sure `KEYSTORE_CONFIGURATION` is set to `FILE`.
4. Migrate from Softcard to keystore:

```
ADMINISTER KEY MANAGEMENT SET ENCRYPTION KEY IDENTIFIED BY "KeystorePassword1" REVERSE MIGRATE USING
"softcard1|scard1" WITH BACKUP USING 'HSM.bkp';
SELECT * FROM sys.v_$encryption_wallet;
```

The expected output is similar to the following:

```
keystore altered.

WRL_TYPE
--------------------
WRL_PARAMETER
------------------------------------------------------------------------------------------
STATUS                         WALLET_TYPE          WALLET_OR KEYSTORE FULLY_BAC     CON_ID
------------------------------ -------------------- --------- -------- --------- ----------
FILE
/opt/oracle/admin/CDB1/keystore-folder/tde/
OPEN                           PASSWORD             PRIMARY   NONE     NO                0

HSM

CLOSED                         UNKNOWN              SECONDARY NONE     UNDEFINED         0

WRL_TYPE
--------------------
WRL_PARAMETER
------------------------------------------------------------------------------------------
STATUS                         WALLET_TYPE          WALLET_OR KEYSTORE FULLY_BAC     CON_ID
------------------------------ -------------------- --------- -------- --------- ----------
```

5. Check the encryption Keys.

```
SELECT * FROM v$encryption_keys;
SELECT KEYSTORE_TYPE FROM V$ENCRYPTION_KEYS;
```

6. Check that you can see the encrypted table content in plaintext.

   If you can see tables in plaintext, the TDE is working with the software wallet. This can be done in various ways and it is going to be based on each environment.

   Example:

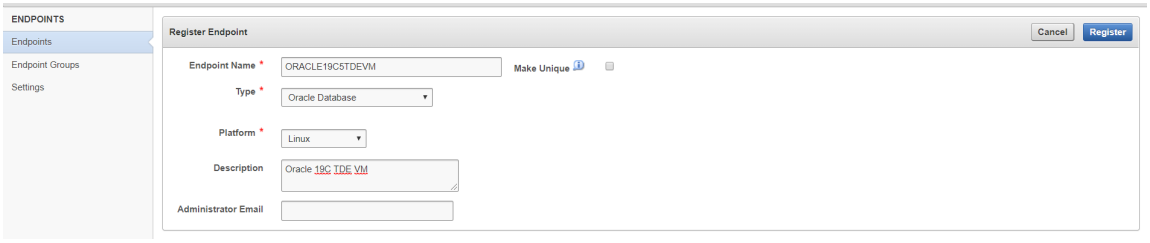
```
SET SERVEROUTPUT ON;

SET FEEDBACK OFF;
CREATE OR REPLACE PROCEDURE display (p_string IN VARCHAR2) IS
BEGIN
    dbms_output.put_line(p_string);
END;
/
SET FEEDBACK ON;

SET FEEDBACK OFF;
exec display('CDB1Table3DES168In3DES168 contents:');
SET FEEDBACK ON;
SELECT FirstName,LastName,FavColor,TeamName FROM CDB1Table3DES168In3DES168;
```

# 4.5. Configure the Oracle TDE VM as an endpoint to the OKV server

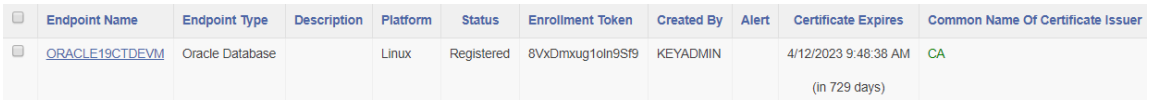
Enrolling using administrator-initiated type of enrollment instead of self-enrollment.

## 4.5.1. Enroll and provision the endpoint

1. Sign in to the OKV web UI.
2. Navigate to **Endpoints**.
3. Select **Add**, and enter the endpoint details, that is, the Oracle TDE Database VM.

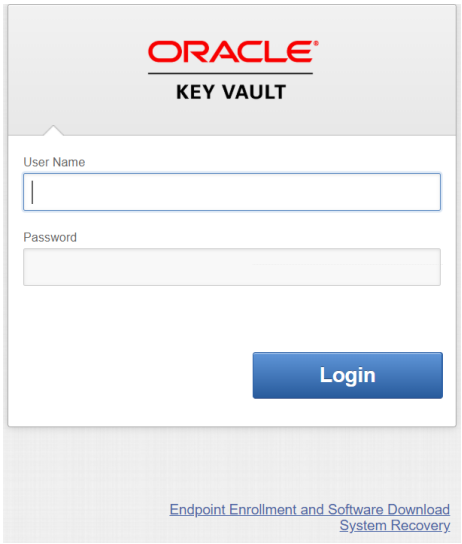



4. Select **Register**.



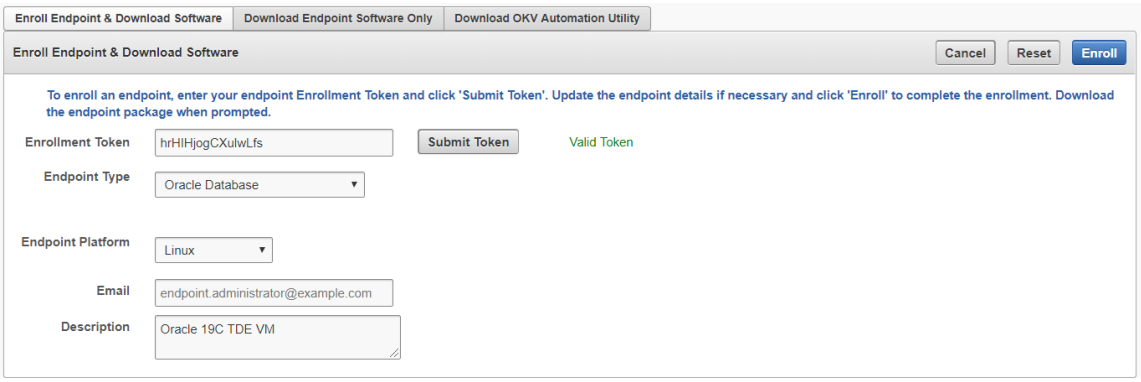
| Endpoint Name | Endpoint Type | Description | Platform | Status | Enrollment Token | Created By | Alert | Certificate Expires | Common Name Of Certificate Issuer |
|---|---|---|---|---|---|---|---|---|---|
| ORACLE19CTDEVM | Oracle Database | | Linux | Registered | 8VxDmxug1oln9Sf9 | KEYADMIN | | 4/12/2023 9:48:38 AM (in 729 days) | CA |

5. Copy the Enrollment Token.
6. Sign out of the OKV web UI.
7. Before entering the credentials, select the **Endpoint Enrollment and Software Download** link at the bottom of the page.



8. Enter in the Enrollment Token as seen in the **Endpoint** section.
9. Select **Submit Token**.

   The response should be **Valid Token**.




10. Select **Enroll**.

    This downloads the `okvclient.jar` file.

## 4.6. Install the Oracle Key Vault Client Software on the Oracle TDE VM endpoint

### 4.6.1. Install the OKV utilities in the TDE Environment

1. Install JDK 1.4 to 7 on the endpoint. In this example, `jdk1.7.0_80` was used. Set `JAVA_HOME` and add it to the `PATH`.
2. Copy the `okvclient.jar` file to the TDE VM endpoint.
3. Create a `$ORACLE_BASE/okvutil` folder. This is the directory location of the Oracle Key Vault software and configuration files. `$OKV_HOME` should be set to this folder.
4. Ensure `ORACLE_HOME1`, `ORACLE_BASE`, and `OKV_HOME` are set. `OKV_HOME` is the folder created above.
5. As the `oracle` user, run:

```
% java -jar okvclient.jar -d $OKV_HOME/okvutil -v

Detected JAVA_HOME: /opt/oracle19c5/product/19.5.0/dbhome_1/jdk
Detected ORACLE_HOME: /opt/oracle19c5/product/19.5.0/dbhome_1
Detected ORACLE_BASE: /opt/oracle19c5
Using OKV_HOME: /opt/oracle19c5/okvutil
Please set environment variables ORACLE_HOME, ORACLE_BASE, and OKV_HOME
consistently across processes.
Enter new Key Vault endpoint password (<enter> for auto-login):
Confirm new Key Vault endpoint password:
```

```
The endpoint software for Oracle Key Vault installed successfully.
```

Password was `okvwallet`. This is the password for the Oracle Key Vault endpoint and is needed later for the migration.

### 4.6.2. Setup communication to the OKV server

Switch to `root` and run `$OKV_HOME/bin/root.sh`.

This copies the `liborapkcs.so` file, which contains the library that the Oracle database uses to communicate with Oracle Key Vault.

```
% sudo su - root
$OKV_HOME/bin/root.sh

Creating directory: /opt/oracle/extapi/64/hsm/oracle/1.0.0/
Copying PKCS library to /opt/oracle/extapi/64/hsm/oracle/1.0.0/
Setting PKCS library file permissions
Installation successful.
```

## 4.7. Create a virtual wallet in Oracle Key Vault

1. In the OKV web UI, select **Keys & Wallets**, and create a new wallet. The example in this guide uses TDE_WALLET.


2. Select **Save**.

   The wallet is created.

3. At the top of the web UI in the menu bar, select **Endpoints**.
4. Edit the endpoint that you created by selecting its name.
5. Make the wallet the default wallet for the endpoint.

## 4.8. Remove Security World Software from the TDE system

Perform these steps as the `root` user.

1. Back up of the `/opt/nfast` folder.

```
% cd /opt; cp -rp nfast nfast.backup
```

2. Uninstall the Security World.

```
% /opt/nfast/sbin/install -u
% rm -rf /opt/nfast
```

## 4.9. Migrate the software TDE wallet to Oracle Key Vault

1. Copy the encryption keys to a wallet in the OKV system.

   As the `oracle` user, run the following:

   Use the wallet name that you created in the OKV web UI.

```
$OKV_HOME/bin/okvutil upload -t WALLET -l SOFTWARE_WALLET_LOCATION -g TDE_WALLET

Enter source wallet password: KeystorePassword1

Enter Oracle Key Vault endpoint password: okvwallet

Upload succeeded
```

   Example:

```
$OKV_HOME/bin/okvutil upload -t WALLET -l /opt/oracle/admin/CDB1/keystore-folder/tde/ -g TDE_WALLET
```

   The Oracle Key Vault web UI shows some activity for a TDE Master Encryption Key:

**Managed Content**

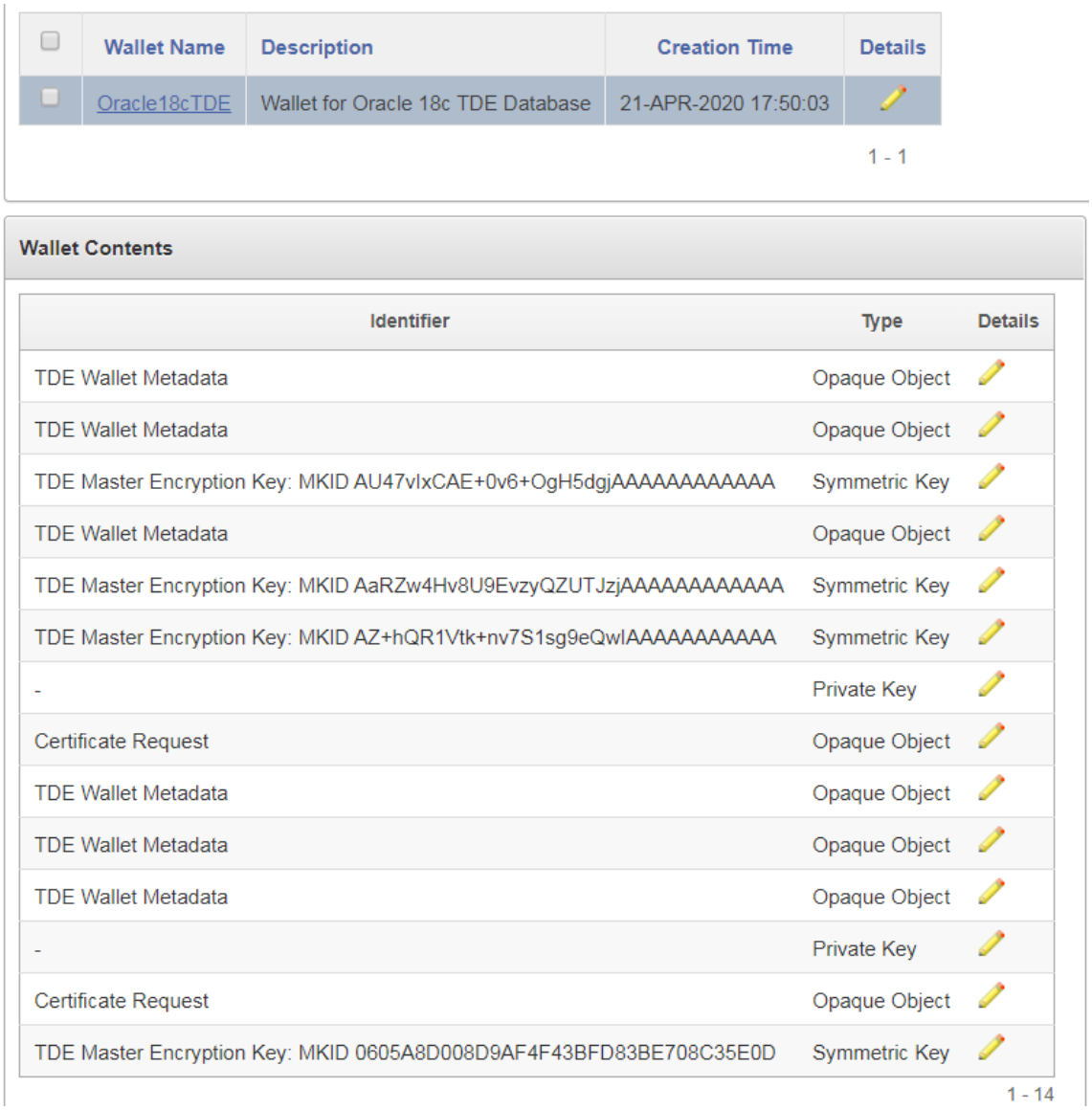
**Item Type**

| Type | Count |
|---|---|
| TDE Master Encryption Key | 9 |

2. In the OKV web UI, select **Keys & Wallets**, and navigate to the wallet that you created.

   You should see all the security objects for the database in the virtual wallet.

If you cannot see them, then tick all of the items which have the owner / wallet membership of the Oracle 19c TDE database and select **Save**. You can now see all security objects for the database in this virtual wallet, including the TDE Master Encryption Keys.



| Wallet Name | Description | Creation Time | Details |
|---|---|---|---|
| Oracle18cTDE | Wallet for Oracle 18c TDE Database | 21-APR-2020 17:50:03 | |

1 - 1

**Wallet Contents**

| Identifier | Type | Details |
|---|---|---|
| TDE Wallet Metadata | Opaque Object | |
| TDE Wallet Metadata | Opaque Object | |
| TDE Master Encryption Key: MKID AU47vlxCAE+0v6+OgH5dgjAAAAAAAAAAAA | Symmetric Key | |
| TDE Wallet Metadata | Opaque Object | |
| TDE Master Encryption Key: MKID AaRZw4Hv8U9EvzyQZUTJzjAAAAAAAAAAAA | Symmetric Key | |
| TDE Master Encryption Key: MKID AZ+hQR1Vtk+nv7S1sg9eQwlAAAAAAAAAAAA | Symmetric Key | |
| - | Private Key | |
| Certificate Request | Opaque Object | |
| TDE Wallet Metadata | Opaque Object | |
| TDE Wallet Metadata | Opaque Object | |
| TDE Wallet Metadata | Opaque Object | |
| - | Private Key | |
| Certificate Request | Opaque Object | |
| TDE Master Encryption Key: MKID 0605A8D008D9AF4F43BFD83BE708C35E0D | Symmetric Key | |

1 - 14

## 4.10. Configure the TDE environment to start using the OKV wallet in the OKV system

Configure the TDE environment to start using the OKV system.

1. Set the keystore configuration to `OKV|FILE`.

```
ALTER SYSTEM SET TDE_CONFIGURATION = "KEYSTORE_CONFIGURATION=OKV|FILE" SCOPE=BOTH SID='*';
```

2. Show the wallet parameters.

```
SHOW PARAMETER WALLET_ROOT;
SHOW PARAMETER TDE_CONFIGURATION;

NAME                                 TYPE        VALUE
------------------------------------ ----------- ------------------------------------
wallet_root                          string      /opt/oracle/admin/CDB1/keystore-folder

NAME                                 TYPE        VALUE
------------------------------------ ----------- ------------------------------------
tde_configuration                    string      KEYSTORE_CONFIGURATION=OKV|FILE
```

3. Make sure that the OKV wallet points to `$OKV_HOME`.

   If `$OKV_HOME` is not `<wallet_root>/okv`, make sure that `<wallet_root>/okv` exists by creating a link to it.

   In the example, *<wallet_root* is set to `/opt/oracle/admin/CDB1/keystore-folder`

```
% cd /opt/oracle/admin/CDB1/keystore-folder
% ln -s $OKV_HOME okv
```

4. Migrate from the software wallet to the OKV wallet:

```
ADMINISTER KEY MANAGEMENT SET ENCRYPTION KEY IDENTIFIED BY "okvwallet" MIGRATE USING "KeystorePassword1"
WITH BACKUP;

keystore altered.
```

5. Check that you can see the encrypted table content in plaintext.

   If you can see tables in plaintext, the TDE is working with OKV.

6. Set the Keystore configuration to OKV.

```
ALTER SYSTEM SET TDE_CONFIGURATION = "KEYSTORE_CONFIGURATION=OKV" SCOPE=BOTH SID='*';
```

7. Remove the software wallet keystore

```
% cd /opt/oracle/admin/CDB1/keystore-folder/
% mv tde tde.removed
```

8. Check that you can see the encrypted table content in plaintext.

   If you can see tables in plaintext, the TDE is working with OKV.

# Chapter 5. Additional resources and related products

## 5.1. nShield Connect

## 5.2. nShield as a Service

## 5.3. Entrust digital security solutions

## 5.4. nShield product documentation