Oracle® Key Vault

Root of Trust HSM Configuration Guide

Release 21.2
F39992-01
July 2021

Oracle Key Vault Root of Trust HSM Configuration Guide, Release 21.2

F39992-01

Copyright © 2014, 2021, Oracle and/or its affiliates.

Primary Author: Mark Doran

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 embedded, installed or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software" or "commercial computer software documentation" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government’s use of Oracle cloud services are defined by the applicable contract for such services. 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 Inside 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, Epyc, and the AMD 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.

Contents

Preface

Welcome to *Oracle Key Vault Root of Trust HSM Configuration Guide* (formerly *Oracle Key Vault Integration with Hardware Security Module*). This guide explains how to integrate a hardware security module (HSM) with Oracle Key Vault.

Audience

Oracle Key Vault is meant for users who are responsible for deploying, maintaining, and managing security within the enterprise. These users can be database, system, or security administrators. This guide can be used by any information security personnel who is responsible for protecting enterprise data residing in database servers, application servers, operating systems, and other information systems.

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.

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of

these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

Related Documents

For more information, see these Oracle resources:

- *Oracle Key Vault Administrator's Guide*
- *Oracle Key Vault RESTful Services Administrator's Guide*
- *Oracle Key Vault Developer's Guide*
- *Oracle Key Vault Licensing Information*
- *Oracle Key Vault Release Notes*
- *Key Management Interoperability Protocol Specification Version 1.1*

To download the product data sheet, frequently asked questions, links to the latest product documentation, product download, and other collateral, visit the Oracle Technology Network (OTN). You must register online before using OTN. Registration is free and can be done at

`https://www.oracle.com/database/technologies/security/key-vault.html`

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Changes in This Release for Oracle Key Vault

This Oracle Key Vault release introduces new features that enhance the use of Oracle Key Vault in a large enterprise.

- Changes for Oracle Key Vault Release 21.2
  Oracle Key Vault release 21.2 introduces several new features.
- Changes for Oracle Key Vault Release 21.1
  Oracle Key Vault release 21.1 introduces several new features.

Changes for Oracle Key Vault Release 21.2

Oracle Key Vault release 21.2 introduces several new features.

- Changes in the Oracle Key Vault Management Console
  In Oracle Key Vault release 21.2, the Oracle Key Vault management console user interface has had minor changes throughout.
- Certificate and Secret Objects Expiration Alerts
  Starting with this release, you can configure alert notifications for the expiration of certificate and secret objects.

Changes in the Oracle Key Vault Management Console

In Oracle Key Vault release 21.2, the Oracle Key Vault management console user interface has had minor changes throughout.

These changes are the result of modified terms, updates to the current release, and enhancements for better usability. The overall interface has not had major changes.

Certificate and Secret Objects Expiration Alerts

Starting with this release, you can configure alert notifications for the expiration of certificate and secret objects.

In previous releases, expiration alerts for all managed objects shared a common configuration under the `Key Rotations` alert. Starting with this release, you can separately configure the expiration alerts for certificate and secret objects. The expiration alerts for the certificate and secret objects are no longer reported as `Key Rotations` alerts. Similar to alerts such as those for cluster components or user password expiration, you can set this type of alert to notify users when the deactivation date for a certificate or secret object is within its threshold value.

The new alerts for certificate and secret objects are as follows:

- `Certificate Object Expiration`
- `Secret Object Expiration`

The object expiration alerts are now raised only when the object is in the `PRE-ACTIVE` or `ACTIVE` state. Previously, they were raised regardless of the object state.

The object expiration alerts are now deleted when an object is revoked or destroyed. Previously, they were deleted when object was destroyed.

Related Topics

- *Oracle Key Vault Administrator's Guide*

Changes for Oracle Key Vault Release 21.1

Oracle Key Vault release 21.1 introduces several new features.

- Dual NIC Network Interface Support
  Starting with this release, Oracle Key Vault supports the use of two network interfaces, referred to as dual NIC configuration.
- LDAP User Authentication and Authorization in Oracle Key Vault
  Starting with this release, you can configure authentication and authorization of Oracle Key Vault users to be centrally managed in a Microsoft Active Directory.
- RESTful Services Utility Command-Line Interface for Appliance Management
  In Oracle Key Vault release 21.2, the the RESTful service command-line interface has been expanded and redesigned to provide more functionality.
- Support for SFTP to Transfer External Backups
  Oracle Key Vault now supports the use of SSH Secure File Transfer Protocol (SFTP) for the transfer of (scheduled) external backups to remote backup destinations.
- Development Using the Java SDK
  This release introduces a new Java language software development kit that you can use to integrate custom endpoints with the Oracle Key Vault server.
- Development Using the C SDK
  This release introduces a new C language software development kit that you can use to integrate custom endpoints with the Oracle Key Vault server.

Dual NIC Network Interface Support

Starting with this release, Oracle Key Vault supports the use of two network interfaces, referred to as dual NIC configuration.

In a dual NIC configuration, Oracle Key Vault combines the two network interfaces into a single logical interface using the Linux NIC bonding mechanism to provide redundancy at the network layer. The dual NIC configuration maintains the network availability of an Oracle Key Vault in case one of the interfaces becomes unavailable. Depending upon the dual NIC configuration mode, load balancing of the network traffic may also be achieved.

This type of configuration is particularly useful in large Oracle Key Vault deployments where need for operational continuity is higher despite physical or software failures. Configuring a dual NIC network interface helps to avoid the scenario where, for example, a network interface associated with an Oracle Key Vault server becomes unavailable, which can result in a loss of communication between the Oracle Key Vault nodes and between endpoints and Oracle Key Vault server.

In previous releases, Oracle Key Vault supported only one network interface. When you install and configure Oracle Key Vault in this release, you have the option of using a single network interface (Classic mode) or using dual NIC mode.

Related Topics

- *Oracle Key Vault Administrator's Guide*

LDAP User Authentication and Authorization in Oracle Key Vault

Starting with this release, you can configure authentication and authorization of Oracle Key Vault users to be centrally managed in a Microsoft Active Directory.

This feature benefits large deployment environments where enterprise users are centrally managed in a Microsoft Active Directory. Centrally managing users, as opposed to creating user accounts in different systems and applications, is not only easier and more efficient for administrators, it improves compliance, control, and security. You enable the Microsoft Active Directory users to authenticate with Oracle Key Vault through the use of their directory credentials. You manage the authorization of the directory users in Oracle Key Vault through mapping definitions between Microsoft Active Directory groups and Oracle Key Vault administrative roles or user groups. When a directory user successfully logs in to Oracle Key Vault the first time, Oracle Key Vault automatically creates an Oracle Key Vault user account for this user.

Related Topics

- *Oracle Key Vault Administrator's Guide*

RESTful Services Utility Command-Line Interface for Appliance Management

In Oracle Key Vault release 21.2, the the RESTful service command-line interface has been expanded and redesigned to provide more functionality.

This redesign includes the following:

- Structured and simplified command-line interface with the following format:

  ```
  okv category resource action configuration-options command-options
  ```

- Profile support in configuration file to centrally administer multiple Oracle Key Vault endpoints.
- JSON support for command input and output.
- New commands to support system management tasks and monitoring of deployments, in addition to the enhancements for the current functionality for endpoints, wallets, and security objects.

In previous releases, the RESTful command-line interface covered only endpoint, wallet, and security object management commands. The addition of system management commands, which include commands for backup operations and server operations for standalone, multi-master, and primary-standby environments, benefits large deployments where the automation of these types of configuration is needed.

The previous RESTful services APIs are still supported.

Related Topics

- *Oracle Key Vault RESTful Services Administrator's Guide*

Support for SFTP to Transfer External Backups

Oracle Key Vault now supports the use of SSH Secure File Transfer Protocol (SFTP) for the transfer of (scheduled) external backups to remote backup destinations.

SFTP enables the use of ZFS Storage Appliance as a backup destination. The use of Secure Copy Protocol (SCP) is also supported.

Related Topics

- *Oracle Key Vault Administrator's Guide*

Development Using the Java SDK

This release introduces a new Java language software development kit that you can use to integrate custom endpoints with the Oracle Key Vault server.

The Java SDK enables developers to create their own custom endpoint integration solutions for Oracle Key Vault.

Related Topics

- *Oracle Key Vault Developer's Guide*Introduction to the Oracle Key Vault Client SDK

Development Using the C SDK

This release introduces a new C language software development kit that you can use to integrate custom endpoints with the Oracle Key Vault server.

The C SDK allows developers to create their own custom endpoint integration solutions for Oracle Key Vault.

1
Getting Started with HSM

To integrate a hardware security module (HSM) with Oracle Key Vault, you must install the HSM client software and enroll Oracle Key Vault as an HSM client.

- How Oracle Key Vault Works with Hardware Security Modules
  This guide explains how to configure Oracle Key Vault to use a supported hardware security module (HSM).
- Installing the HSM Client Software on an Oracle Key Vault Server
  After you install Oracle Key Vault, you can install the HSM client software on the Oracle Key Vault server.
- Enrolling Oracle Key Vault as a Client of the HSM
  You must enroll Oracle Key Vault as a client of HSM and ensure connectivity between the HSM client and the HSM.

1.1 How Oracle Key Vault Works with Hardware Security Modules

This guide explains how to configure Oracle Key Vault to use a supported hardware security module (HSM).

A hardware security module (HSM) contains tamper-resistant, specialized hardware that is designed to protect security objects stored within the HSM. HSMs are physical computing devices that safeguard and manage digital keys, and provide cryptographic processing for clients. HSMs do not usually allow security objects to leave the cryptographic boundary of the HSM.

Oracle Key Vault is a key management platform designed to securely store, manage and share security objects. Unlike an HSM, Oracle Key Vault allows trusted clients to retrieve security objects like decryption keys. Oracle Key Vault is a full-stack software appliance that contains an operating system, database, and key-management application. Oracle Key Vault is designed to help organizations store and manage their keys and credentials.

Your organization may require the use of an HSM to protect encryption keys. Because they are designed to not allow keys to leave the cryptographic boundary of the HSM, in most cases it is not practical to connect databases directly to an HSM. Instead, databases will connect to the Oracle Key Vault which will in turn be protected by the HSM. This configuration establishes a Root-of-Trust (RoT) for Oracle Key Vault in the HSM. When an HSM is deployed with Oracle Key Vault, the RoT remains in the HSM. The HSM RoT protects the Transparent Data Encryption (TDE) 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 theft of the disk images that represent an Oracle Key Vault server that runs as a virtualization guest is

one example of such a scenario. An unauthorized user attempting to run a 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. Therefore, you should disable Secure Shell Protocol (`ssh`) access into Oracle Key Vault at all times, except when you must apply quarterly release upgrades. Oracle Key Vault should be deployed in a secure location, and physical and logical access to the appliance should be controlled and monitored.

If your site uses HSMs from Thales Luna (formerly SafeNet Luna), Entrust (formerly nCipher), or Utimaco, then you can configure these HSM products with Oracle Key Vault in standalone, primary-standby, and multi-master environments.

This guide assumes that you have installed and configured Oracle Key Vault. It also assumes that you have sufficient knowledge of the HSM products that you plan to configure.

The general process that you must follow to configure the HSM with Oracle Key Vault is as follows:

1. Install the HSM client software on the Oracle Key Vault server.
2. Enroll Oracle Key Vault as a client of the HSM.
3. Perform further configuration operations, which are as follows:
   - Configure protection for the TDE master encryption key with the HSM.
   - Use an HSM in a primary-standby Oracle Key Vault installation.
   - Use an HSM in an Oracle Key Vault multi-master cluster environment.
   - Perform backup and restore operations in an HSM-enabled Oracle Key Vault instance.
   - When necessary, perform reverse-migration so that the Oracle Key Vault environment is no longer HSM-enabled.

1.2 Installing the HSM Client Software on an Oracle Key Vault Server

After you install Oracle Key Vault, you can install the HSM client software on the Oracle Key Vault server.

1. Ensure that the vendor's software includes a PKCS#11 library.

   Refer to the HSM documentation from the HSM vendor for more information.
2. Install the HSM vendor's client software on the Oracle Key Vault server.

   You can install Thales Luna, Entrust, or Utimaco HSM products.

Related Topics

- Installing the HSM Client Software on the Oracle Key Vault Server for Thales Luna
  You must use the latest Luna Universal Client version for Linux x64 for the installation.
- Installing the HSM Client Software on the Oracle Key Vault Server for Entrust
  The Entrust HSM requires a separate non-HSM computer on the network to use as the remote file system.
- Installing the HSM Client Software on the Oracle Key Vault Server for Utimaco
  The setup files for Utimaco are provided in the `SecurityServerEvaluation-V4.31.1.0.zip` file from Utimaco.

1.3 Enrolling Oracle Key Vault as a Client of the HSM

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

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.

Related Topics

- Installing the HSM Client Software on an Oracle Key Vault Server
  After you install Oracle Key Vault, you can install the HSM client software on the Oracle Key Vault server.
- Enrolling Oracle Key Vault as a Client of a Thales Luna HSM
  To perform the enrollment, you use the Oracle Key Vault management console and the command-line interface.
- Enrolling Oracle Key Vault as a Client of an Entrust HSM
  You use both the Entrust user interface and the command line to enroll Oracle Key Vault as a client of an Entrust HSM.

2

Configuring an HSM for Oracle Key Vault

Oracle Key Vault can be configured to use the HSM as the Root of Trust in a standalone, primary-standby, or multi-master cluster environment.

- HSM-Enabling in a Standalone Oracle Key Vault Deployment
  You can use the Oracle Key Vault management console to HSM-enable Oracle Key Vault, which configures additional protection for the TDE master encryption key.
- HSM-Enabling in a Primary-Standby Oracle Key Vault Deployment
  In an Oracle Key Vault primary-standby deployment, you must perform the HSM-enabling tasks separately on the Oracle Key Vault servers that will be become primary and standby servers.
- HSMs in a Multi-Master Cluster
  You can configure HSMs in a multi-master cluster with a single node or multiple nodes.
- Backup and Restore Operations in an HSM-Enabled Oracle Key Vault Instance
  You can back up and restore an HSM-enabled Oracle Key Vault instance.
- Reverse Migration Operation
  Reverse migrating an HSM-enabled Oracle Key Vault server reverts the Key Vault server to using the recovery passphrase to protect the TDE wallet.

2.1 HSM-Enabling in a Standalone Oracle Key Vault Deployment

You can use the Oracle Key Vault management console to HSM-enable Oracle Key Vault, which configures additional protection for the TDE master encryption key.

If you plan to use a multi-master cluster, then Oracle recommends that you perform this procedure before you configure the cluster environment. Ensure that you complete the following steps on this server before you start these steps on another Oracle Key Vault server.

1. If you have installed the Entrust client software, then run the following command as user `oracle`:

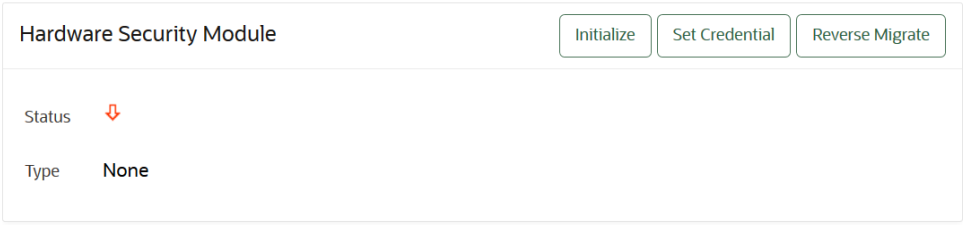
   ```
   oracle$ /opt/nfast/bin/rfs-sync --update
   ```

2. Log into the Oracle Key Vault management console as a user with the System Administrator role.

   If you are using a multi-master cluster environment, then log into the Oracle Key Vault node that you want to HSM-enable.

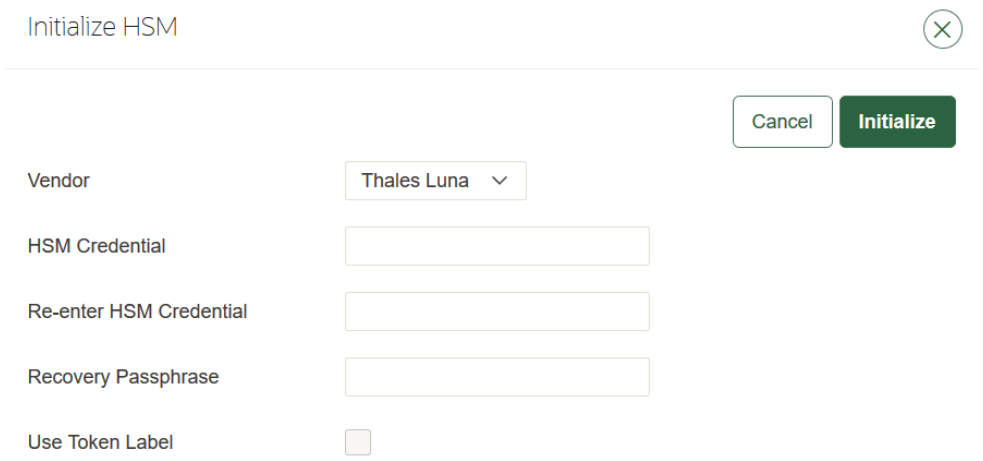
3. Select the **System** tab, then **Settings** in the left navigation bar.
4. In Network Services, click **HSM**.

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



5. Click **Initialize**.

   The Initialize HSM window appears.



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

   Consult the documentation that came with your HSM for this credential. The HSM credential for Thales Luna is the Thales Luna partition password. For Entrust, the credential is the password that is associated with the Operator Card Set or Softcard. For Utimaco, the credential is the PIN that was initialized when the token was configured. Oracle currently only supports HSM credentials up to 79 characters in length.

7. Enter the **Recovery Passphrase** for Oracle Key Vault.

8. If you want Oracle Key Vault to use a specific token to create and use objects in the HSM, then select the **Use Token Label** check box and enter the token label of the token that Oracle Key Vault should use.

   Oracle recommends that you select **Use Token Label** if Oracle Key Vault has access to more than one token. Oracle Key Vault does not support using a token that has the same name as one or more other tokens, nor does Oracle Key Vault support the use of tokens that have names with leading spaces.

9. 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.

10. If you are using an Entrust HSM, then run the following command as user `oracle`:

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

    If you do not perform this step after each initialization when using Entrust, then the multiple features will not be usable, including restoring backups and using the primary-standby configuration.

11. Verify that the operation was successful by checking the most recent initialization log files in the `/var/okv/log/hsm/` directory.

If the initialize operation fails, then you will be redirected to the Hardware Security Module page with non-initialized **Status** and **Type** None. You can find detailed information in the log files in the `/var/okv/log/hsm` directory.

> **Note:**
>
> If you change the HSM credential on the HSM after initialization, then you must also update the HSM credential on the Oracle Key Vault server using the **Set Credential** command before the system restarts. Oracle does not recommend that you change the HSM credential after HSM initialization if there are primary-standby Oracle Key Vault deployments using the HSM, because the standby does not have its credential set by the **Set Credential** command on the primary.

2.2 HSM-Enabling in a Primary-Standby Oracle Key Vault Deployment

In an Oracle Key Vault primary-standby deployment, you must perform the HSM-enabling tasks separately on the Oracle Key Vault servers that will be become primary and standby servers.

You must perform this task before pairing these two servers in a primary-standby configuration. If you have already HSM-enabled either the primary or the standby server, or both, but do not follow these steps and then do a primary-standby pairing, then the configuration will fail. If the servers are already paired but neither are HSM-enabled, then you must unpair them, reinstall the standby server, and the follow these steps.

1. Install two separate Oracle Key Vault instances.
2. Choose one to be the primary and the other to be the standby server.
3. Install the HSM client software on both the servers that will be used as the primary and the standby servers.
4. Enroll the designated primary and standby servers as clients of the same HSM.
5. HSM-enable the designated primary server.

   If you are using Entrust, ensure that you have already executed `/opt/nfast/bin/rfs-sync --commit` on this server as user `oracle` before continuing.

6. Perform the following steps on the primary server:

a. 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_ip_address
support$ su root
root# su oracle
```

b. Securely copy the following files to the designated standby server:

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

7. Perform the following steps on the designated standby server:

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

```
$ ssh support@okv_standby_instance_ip_address
support$ su root
```

b. Set up the HSM-related files and in the `okv_security.conf` file, set the `HSM_ENABLED` and `HSM_PROVIDER` parameters.

Earlier versions of Oracle Key Vault may not contain certain parameters in `okv_security.conf` that are present in later versions.

```
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 *
root# vi /usr/local/okv/etc/okv_security.conf
   Set HSM_ENABLED="1"
   Set HSM_PROVIDER="provider_value"
   Set HSM_KEY_EXTRACTABLE="extractable_value"
   Set HSM_TOKEN_LABEL="token_label_value"
```

In this specification:

- `HSM_ENABLED` is set in this example to `1` to prepare the designated standby to use the HSM should a switchover or failover occur.
- `HSM_PROVIDER` refers to the HSM provider. For Thales Luna, set this value to `1`. For Entrust, set it to `2`. For Utimaco, set it to `3`.
  The `HSM_PROVIDER` may not be present in the `okv_security.conf` file. If this setting is present, then change it to the setting that is appropriate for the HSM provider. If it is not present, then add the following line. Ensure that the *`provider_value`* setting is in quotation marks.

```
HSM_PROVIDER="provider_value"
```

- `HSM_KEY_EXTRACTABLE` and `HSM_TOKEN_LABEL` should be set to the same value on the standby that is set on the primary server.

c. Save and quit by entering the following sequence of characters in the `vi` file:

```
:wq!
```

d. If you are using Entrust, then execute the following commands:

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

8. Without restarting the Oracle Key Vault instances, navigate to the primary and standby Oracle Key Vault management consoles and configure primary-standby environment.

Related Topics

- *Oracle Key Vault Administrator's Guide*

2.3 HSMs in a Multi-Master Cluster

You can configure HSMs in a multi-master cluster with a single node or multiple nodes.

- About HSMs in a Multi-Master Cluster
  You can configure each node in the cluster to use an HSM to store each node's Root of Trust (RoT) key.
- Configuring an HSM for a Multi-Master Cluster Starting with Single Node (Recommended)
  Oracle recommends that to use an HSM with a multi-master cluster, you start with a single HSM-enabled node and add additional HSM-enabled nodes using the node induction process.
- Configuring an HSM for a Multi-Master Cluster with Multiple Nodes
  You can configure HSM for multiple nodes by copying a bundle from the first HSM-enabled node to the other nodes in the cluster before configuring HSM for the other nodes.

2.3.1 About HSMs in a Multi-Master Cluster

You can configure each node in the cluster to use an HSM to store each node's Root of Trust (RoT) key.

This RoT protects master encryption keys that Oracle Key Vault uses. HSMs are built with specialized tamper-resistant hardware which is harder to access than normal servers. This protects the RoT and makes it difficult to extract encrypted data, lowering the risk of compromise. In addition, you can use HSMs in FIPS 140-2 level 3 mode, which enables you to meet certain compliance requirements.

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 will use different TDE wallet passwords and RoT keys and may or may not use different HSM credentials, depending on how you choose to configure each cluster node.

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

2.3.2 Configuring an HSM for a Multi-Master Cluster Starting with Single Node (Recommended)

Oracle recommends that to use an HSM with a multi-master cluster, you start with a single HSM-enabled node and add additional HSM-enabled nodes using the node induction process.

Oracle recommends the following steps to configure an HSM for a multi-master cluster with a single node:

1. Convert an Oracle Key Vault server into the first node of the cluster.
2. HSM-enable the first node before adding any new 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.
   Note the following:
   - If any node in the cluster is already HSM-enabled, you cannot add a new node that is not HSM-enabled.
   - The Add Node to Cluster page on the controller node will require the controller node's HSM credential.

Related Topics

- Configuring an HSM for a Multi-Master Cluster with Multiple Nodes
  You can configure HSM for multiple nodes by copying a bundle from the first HSM-enabled node to the other nodes in the cluster before configuring HSM for the other nodes.
- *Oracle Key Vault Administrator's Guide*

2.3.3 Configuring an HSM for a Multi-Master Cluster with Multiple Nodes

You can configure HSM for multiple nodes by copying a bundle from the first HSM-enabled node to the other nodes in the cluster before configuring HSM for the other nodes.

- About Configuring an HSM for a Multi-Master Cluster with Multiple Nodes
  The general procedure is to perform steps on first on one node of the cluster, then on the other nodes in the cluster.
- Step 1: Create and Copy the Bundle after HSM-Enabling the First Node
  After HSM-enabling the first node in the multi-master cluster, you must create a bundle and copy it to the other nodes in the cluster.

- Step 2: Configure the Remaining Nodes
  After you configure the first node, you are ready to install the bundle on the remaining nodes.

2.3.3.1 About Configuring an HSM for a Multi-Master Cluster with Multiple Nodes

The general procedure is to perform steps on first on one node of the cluster, then on the other nodes in the cluster.

The instructions for configuring an HSM for a multi-master cluster starting with a single node explain how to configure an HSM for a multi-master cluster, starting with a single node of the cluster and is the recommended way to configure a cluster to use HSM(s). However, if you have already configured a multi-master cluster, you can still configure the cluster to use HSMs. However, there are extra steps needed, involving manually copying a bundle from the first HSM-enabled node to all of the other nodes in the cluster and applying it before proceeding to HSM-enable any other node. Note that if the first node that is HSM-enabled has a read-write peer node, then the read-write peer will not be able to decrypt the replicated information from the HSM-enabled node until the bundle is copied and applied successfully to the read-write peer. This could result in data loss if the bundle is not immediately successfully created and applied to the read-write peer, even if the first node that is HSM-enabled is reverse-migrated afterwards.

After you HSM-enable the first node in the cluster, use the following steps to create the bundle on the HSM-enabled node and copy and apply it on all other nodes in the cluster before you proceed to HSM-enable any other node.

Related Topics

- Configuring an HSM for a Multi-Master Cluster Starting with Single Node (Recommended)
  Oracle recommends that to use an HSM with a multi-master cluster, you start with a single HSM-enabled node and add additional HSM-enabled nodes using the node induction process.

2.3.3.2 Step 1: Create and Copy the Bundle after HSM-Enabling the First Node

After HSM-enabling the first node in the multi-master cluster, you must create a bundle and copy it to the other nodes in the cluster.

You must HSM-enable the first node in the cluster similar to how you would HSM-enable a standalone Oracle Key Vault deployment, but with the additional steps in this section.

1. Log in to the Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **System** tab, then **Settings** in the left navigation bar.
3. In Network Services, click **HSM**.
4. On the HSM-enabled node, click **Create Bundle** on the HSM page.
5. In the Create Bundle dialog box, do the following:
   a. In the **HSM Credential** field, enter the HSM credential.
   b. In the **Recovery Passphrase** field, enter the recovery passphrase.
   c. Click the **Create Bundle** button.

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

   ```
   $ ssh support@hsm_enabled_node
   ```

7. Switch to the `root` user.

   ```
   support$ su root
   ```

8. To copy the bundle to the `/usr/local/okv/hsm` location on each of the other nodes using the IP address, use `SCP`.

   Ensure that you perform this step using the IP address of all other nodes in the cluster.

   ```
   root# scp /usr/local/okv/hsm/hsmbundle support@ip_address:/tmp
   ```

Related Topics

- HSM-Enabling in a Standalone Oracle Key Vault Deployment
  You can use the Oracle Key Vault management console to HSM-enable Oracle Key Vault, which configures additional protection for the TDE master encryption key.

2.3.3.3 Step 2: Configure the Remaining Nodes

After you configure the first node, you are ready to install the bundle on the remaining nodes.

Complete this procedure as soon as possible after you have HSM-enabled the first node and copied the bundle to all other nodes.

1. Log in to each node in the cluster using the IP address (except the original HSM-enabled node):.

   ```
   $ ssh support@ip_address
   ```

2. On each node, switch to the `root` user.

   ```
   support$ su root
   ```

3. Perform the following steps on each node:

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

4. On each node except the original HSM-enabled node, click **Apply Bundle** on the **HSM** page, and then follow these steps:

   a. In the **Recovery Passphrase** field, enter the recovery passphrase.

   b. Click the **Apply Bundle** button.

   You must apply the bundle immediately on all nodes before you reverse-migrate the original HSM-enabled node.

5. Proceed to HSM-enable each of the remaining nodes in the cluster.

6. After you have HSM-enabled all nodes and verified the replication between all nodes, remove the `hsmbundle` file from all of the nodes.

2.4 Backup and Restore Operations in an HSM-Enabled Oracle Key Vault Instance

You can back up and restore an HSM-enabled Oracle Key Vault instance.

- Backup Operations in an HSM-Enabled Oracle Key Vault Instance
  The steps to back up Oracle Key Vault data in an HSM-enabled instance are the same as the steps used to back up an instance that has not been HSM-enabled.
- Restore Operations in an HSM-Enabled Oracle Key Vault Instance
  Backups taken from an HSM-enabled Oracle Key Vault instance can only be restored onto a standalone Oracle Key Vault server with access to the same Root of Trust key that was in use when the backup was taken.

2.4.1 Backup Operations in an HSM-Enabled Oracle Key Vault Instance

The steps to back up Oracle Key Vault data in an HSM-enabled instance are the same as the steps used to back up an instance that has not been HSM-enabled.

You can use the Oracle Key Vault management console to perform a backup operation.

Related Topics

- *Oracle Key Vault Administrator's Guide*

2.4.2 Restore Operations in an HSM-Enabled Oracle Key Vault Instance

Backups taken from an HSM-enabled Oracle Key Vault instance can only be restored onto a standalone Oracle Key Vault server with access to the same Root of Trust key that was in use when the backup was taken.

Before you restore a backup onto a system, you must ensure that the system can access both the HSM and the Root of Trust (RoT) that was used to make the backup. You must therefore have installed the HSM client software on the Oracle Key Vault server and enrolled the Oracle Key Vault as a client of the HSM before proceeding with this step. If the backup was taken on an HSM-enabled cluster node, then when you restore the backup to a standalone server, you must ensure that the server has access to the same HSM and RoT as the node on which the backup was taken.

1. Log into the Oracle Key Vault management console as a user with the System Administrator role.

   The Oracle Key Vault Home page appears.
2. Select the **System** tab, then **Settings** in the left navigation bar.
3. In Network Services, click **HSM**.

   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.

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

   Consult the documentation that came with your HSM for this credential. The HSM credential for Thales Luna is the Thales Luna partition password. For Entrust, the credential is the password that is associated with the Operator Card Set or Softcard. The HSM-credential if you use an Operator Card Set is the Operator Card Set password. If you use a Softcard, then the password is the Softcard password.

6. If the backup you are restoring was taken while Oracle Key Vault was HSM-enabled and given a specific token to use, select **Use Token Label** and enter the token label of the token that Oracle Key Vault was using when the backup was taken.

7. Click **Set Credential**.

   > **Caution:**
   >
   > If a credential has already successfully been set either via the Set Credential or Initialize operations, if you set an incorrect credential for the HSM, the previous credential, token label, and vendor will continue to be stored and used. If a credential has not been set previously and the Set Credential operation fails, the incorrect credential, token label, and vendor are not stored.

   The HSM credential will be stored in the system. It must be stored on the system so that it can be used to perform a backup restore operation because it is not stored in backup itself.

8. If you are using Entrust, then run the following command as user `oracle`:

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

   This command is needed for an Entrust backup restore to complete successfully.

9. In the Oracle Key Vault management console, go to the **Restore** page and then restore the backup.

Related Topics

- *Oracle Key Vault Administrator's Guide*

2.5 Reverse Migration Operation

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 you no longer want to use the HSM to protect the TDE wallet password (for example, if the HSM must be decommissioned).

- Reverse Migrating a Standalone Deployment
  You can reverse migrate a standalone deployment by using the Oracle Key Vault management console.

- Reverse Migrating a Primary-Standby Deployment
  To reverse migrate a primary-standby deployment, use both the Oracle Key Vault management console and the command line.
- Reverse Migrating a Multi-Master Cluster
  You can reverse migrate a multi-master cluster by using the Oracle Key Vault management console.

2.5.1 Reverse Migrating a Standalone Deployment

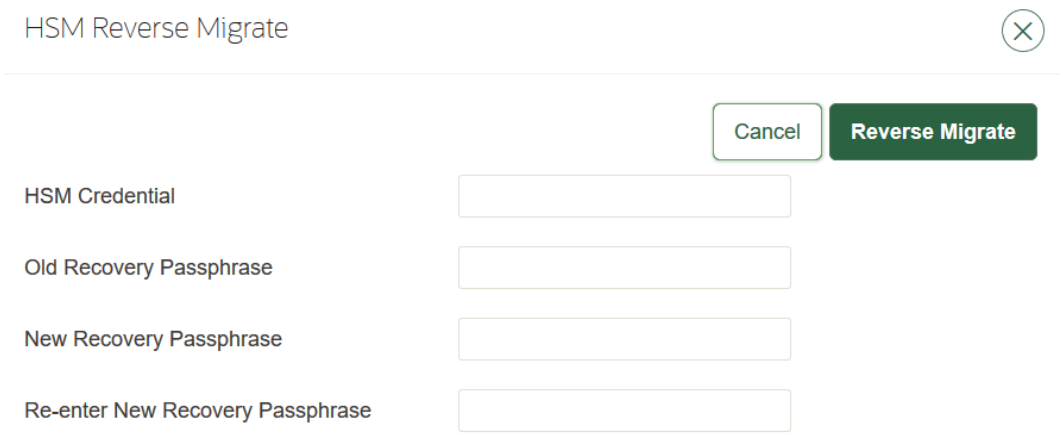
You can reverse migrate a standalone deployment by using the Oracle Key Vault management console.

1. Log into the Oracle Key Vault management console as a user with the System Administrator role.

   The Oracle Key Vault Home page appears.
2. Select the **System** tab, then **Settings** in the left navigation bar.
3. In Network Services, click **HSM**.

   The Hardware Security Module page appears.
4. Click **Reverse Migrate**.

   The HSM Reverse Migrate window is displayed.



Enter the following details:

- Enter the HSM credential in the **HSM Credential** field. Consult the HSM documentation for this credential. The HSM credential for Thales Luna is the Thales Luna partition password. For Entrust, the credential is the password that is associated with the Operator Card Set or Softcard. For Utimaco, the credential is the PIN that was initialized when the token was configured.
- Enter the old recovery passphrase in the **Old Recovery Passphrase** field.
- Enter the new recovery passphrase in the **New Recovery Passphrase** and **Re-enter New Recovery Passphrase** fields. If you do not want to change the recovery passphrase, then enter the same recovery passphrase in the **New Recovery Passphrase** and **Re-enter New Recovery Passphrase** fields as the one you entered in **Old Recovery Passphrase**.

5. Click **Reverse Migrate**

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

2.5.2 Reverse Migrating a Primary-Standby Deployment

To reverse migrate a primary-standby deployment, use both the Oracle Key Vault management console and the command line.

1. Log into the Oracle Key Vault management console as a user with the System Administrator role.

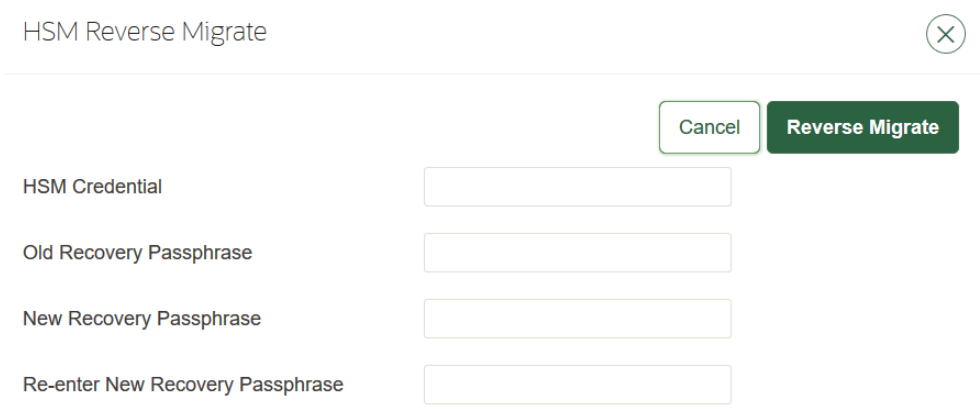
   The Oracle Key Vault **Home** page appears.

2. Select the **System** tab, then **Settings** in the left navigation bar.
3. In Network Services, click **HSM**.

   The Hardware Security Module page appears.

4. Click **Reverse Migrate**.

   The HSM Reverse Migrate screen is displayed.



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

   - Enter the HSM credential in the **HSM Credential** field. Consult the documentation that came with your HSM for this credential. The HSM credential for Thales Luna is the Thales Luna partition password. For Entrust, the credential is the password that is associated with the Operator Card Set or Softcard. For Utimaco, the credential is the PIN that was initialized when the token was configured.
   - Enter the old recovery passphrase in the **Old Recovery Passphrase** field.
   - Enter the new recovery passphrase in the **New Recovery Passphrase** and **Re-enter New Recovery Passphrase** fields. If you do not want to change the recovery passphrase, then enter the same recovery passphrase in the **New Recovery Passphrase** and re-enter the **New Recovery Passphrase** fields as the one you entered in **Old Recovery Passphrase**.

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
   support$ su root
   ```

7. Modify the `okv_security.conf` file.

   ```
   root# 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:

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

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

   ```
   root# su oracle
   ```

10. Run the following command:

    ```
    oracle$ /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 4.

The primary-standby deployment is successfully reverse migrated.

2.5.3 Reverse Migrating a Multi-Master Cluster

You can reverse migrate a multi-master cluster by using the Oracle Key Vault management console.

1. Log into the Oracle Key Vault management console as a user with the System Administrator role.

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

2. Select the **System** tab, then **Settings** in the left navigation bar.
3. In Network Services, click **HSM**.

   The Hardware Security Module page appears.

4. Click **Reverse Migrate**.

   The HSM Reverse Migrate window is displayed.

   Enter the following details:

   - In the **HSM Credential** field, enter the HSM credential. Consult the documentation that came with your HSM for this credential. The HSM credential for Thales Luna is the Thales Luna partition password. For Entrust, the credential is the password that is associated with the Operator Card Set or Softcard. For Utimaco, the credential is the PIN that was initialized when the token was configured.
   - In the **Recovery Passphrase** field, enter the recovery passphrase.

5. Click **Reverse Migrate**

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

3

Upgrade Considerations When Using HSMs in Oracle Key Vault

When you upgrade an Oracle Key Vault deployment that has HSMs, you should consider factors such as the release being upgraded from, Entrust, and tokens.

- Upgrades from Oracle Key Vault Release 12.2
  Upgrading from a standalone or a primary-standby Oracle Key Vault release 12.2 environment has special considerations for HSMs.
- Upgrade Considerations for Entrust
  When you upgrade while HSM-enabled and using an Entrust HSM, you must remake hardserver changes and consider changes to overriding security assurances.
- Using a Token Label After Upgrading Oracle Key Vault without Reverse-Migrating
  Starting in release 18.4, Oracle Key Vault can use a token label when you choose a slot while connecting to an HSM.

3.1 Upgrades from Oracle Key Vault Release 12.2

Upgrading from a standalone or a primary-standby Oracle Key Vault release 12.2 environment has special considerations for HSMs.

- Upgrading from an Oracle Key Vault Release 12.2 Standalone Deployment
  You can upgrade from an Oracle Key Vault standalone deployment by reverse-migrating before the upgrade, and after the upgrade completes, re-HSM-enabling Oracle Key Vault.
- Upgrading from an Oracle Key Vault Release 12.2 Primary-Standby Deployment
  You can upgrade from an Oracle Key Vault release 12.2 primary-standby deployment by reverse-migrating before the upgrade, and after the upgrade completes, re-HSM-enabling Oracle Key Vault.

3.1.1 Upgrading from an Oracle Key Vault Release 12.2 Standalone Deployment

You can upgrade from an Oracle Key Vault standalone deployment by reverse-migrating before the upgrade, and after the upgrade completes, re-HSM-enabling Oracle Key Vault.

1. Make a one-time backup of the Oracle Key Vault server to a remote backup destination.
2. Make a copy of the `/mnt/okvram/cwallet.sso` wallet file.

   ```
   $ ssh support@Oracle_Key_Vault_server_IP_address
   support$ su - root
   root# /bin/cp /mnt/okvram/cwallet.sso /var/lib/oracle/cwallet_hsm_upgrade.sso
   ```
3. Reverse migrate so that Oracle Key Vault is no longer using an HSM as the RoT.

   You can verify the success of this operation by checking the audit record in the audit trail.

If this step fails despite using the correct HSM credential and recovery passphrase, then do not continue with the rest of these steps. Contact Oracle Support.

4. Make another one-time backup of the Oracle Key Vault server to a remote backup destination.
5. Proceed with the rest of the upgrade steps as described in *Oracle Key Vault Administrator's Guide* for a standalone Oracle Key Vault server, including taking a one-time backup after the upgrade completes.
6. After the upgrade successfully completes, optionally HSM-enable your Oracle Key Vault server.
7. Remove the copied wallet in Step 2.

```
$ ssh support@Oracle_Key_Vault_server_IP_address
support$ su - root
root# /bin/rm /var/lib/oracle/cwallet_hsm_upgrade.sso
```

Related Topics

- HSM-Enabling in a Standalone Oracle Key Vault Deployment
  You can use the Oracle Key Vault management console to HSM-enable Oracle Key Vault, which configures additional protection for the TDE master encryption key.
- Backup Operations in an HSM-Enabled Oracle Key Vault Instance
  The steps to back up Oracle Key Vault data in an HSM-enabled instance are the same as the steps used to back up an instance that has not been HSM-enabled.
- Reverse Migrating a Standalone Deployment
  You can reverse migrate a standalone deployment by using the Oracle Key Vault management console.
- *Oracle Key Vault Administrator's Guide*

3.1.2 Upgrading from an Oracle Key Vault Release 12.2 Primary-Standby Deployment

You can upgrade from an Oracle Key Vault release 12.2 primary-standby deployment by reverse-migrating before the upgrade, and after the upgrade completes, re-HSM-enabling Oracle Key Vault.

1. On the current primary, make a one-time backup of the Oracle Key Vault server to a remote backup destination.
2. On the current primary, make a copy of the `/mnt/okvram/cwallet.sso wallet` file.

```
$ ssh support@Oracle_Key_Vault_server_IP_address
support$ su - root
root# /bin/cp /mnt/okvram/cwallet.sso /var/lib/oracle/cwallet_hsm_upgrade.sso
```

3. Unpair the primary and the standby servers.
   a. Log in to the primary server's management console as a user with the System Administrator role.
   b. Select the **System** tab, then **Settings** in the left navigation bar.
   c. In the System Configuration area, click **Primary-Standby**.
   d. Click **Unpair**.

The unpair operation takes about 10 minutes to complete. After the unpair operation is complete, the standby will no longer be usable.

4. Reverse migrate so that Oracle Key Vault is no longer using an HSM as the RoT.

   You can verify the success of this operation by checking the audit record in the audit trail.

   If this step fails despite using the correct HSM credential and recovery passphrase, then do not continue with the rest of these steps. Contact Oracle Support.

5. Make another one-time backup of the Oracle Key Vault server to a remote backup destination.

6. Proceed with the rest of the upgrade steps as described in *Oracle Key Vault Administrator's Guide* for a primary-standby Oracle Key Vault server, including taking a one-time backup after the upgrade completes.

7. After the upgrade successfully completes, optionally HSM-enable a new primary-standby configuration using the upgraded Oracle Key Vault server as the primary and a fresh installation of the same version as the standby.

8. Remove the copied wallet in Step 2.

```
$ ssh support@Oracle_Key_Vault_server_IP_address
support$ su - root
root# /bin/rm /var/lib/oracle/cwallet_hsm_upgrade.sso
```

Related Topics

- Backup Operations in an HSM-Enabled Oracle Key Vault Instance
  The steps to back up Oracle Key Vault data in an HSM-enabled instance are the same as the steps used to back up an instance that has not been HSM-enabled.
- Reverse Migrating a Standalone Deployment
  You can reverse migrate a standalone deployment by using the Oracle Key Vault management console.
- *Oracle Key Vault Administrator's Guide*
- HSM-Enabling in a Primary-Standby Oracle Key Vault Deployment
  In an Oracle Key Vault primary-standby deployment, you must perform the HSM-enabling tasks separately on the Oracle Key Vault servers that will be become primary and standby servers.

3.2 Upgrade Considerations for Entrust

When you upgrade while HSM-enabled and using an Entrust HSM, you must remake hardserver changes and consider changes to overriding security assurances.

- Remaking Hardserver Changes While Upgrading Oracle Key Vault
  Because HSM configurations can vary, it is your responsibility to run test upgrades on non-production environments to ensure that the upgrade will work with your HSM configuration.
- Overriding Security Assurances for the Oracle Key Vault Upgrade
  You can configure how the Oracle Key Vault security assurance attributes are set for future initialization operations.
- Re-installation of Entrust Software While Upgrading Oracle Key Vault from 18.x to 21.x
  During the upgrade from Oracle Key Vault 18.*x* to 21.*x*, the operating system is upgraded from Oracle Linux 6 to Oracle Linux 7.

3.2.1 Remaking Hardserver Changes While Upgrading Oracle Key Vault

Because HSM configurations can vary, it is your responsibility to run test upgrades on non-production environments to ensure that the upgrade will work with your HSM configuration.

Upgrading Oracle Key Vault can undo changes that had been made before the upgrade to accommodate the installed HSM software. After the upgrade, perform the following steps and then ensure that Oracle Key Vault has all of the necessary changes to support integration with your HSM.

1. After executing the steps to upgrade as described in *Oracle Key Vault Administrator's Guide*, and if the upgrade completed successfully, connect to the Oracle Key Vault server as the `root` user.

   ```
   $ ssh support@Oracle_Key_Vault_server_IP_address
   support$ su - root
   ```

2. Execute the following command as the `root` user:

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

Related Topics

- *Oracle Key Vault Administrator's Guide*

3.2.2 Overriding Security Assurances for the Oracle Key Vault Upgrade

You can configure how the Oracle Key Vault security assurance attributes are set for future initialization operations.

As of Oracle Key Vault release 18.4.0.0.0, the Root of Trust (RoT) key that is created will have its `CKA_EXTRACTABLE` attribute set to `CK_FALSE` by default. However, in release 18.3.0.0.0 and earlier, `CKA_EXTRACTABLE` was set to `CK_TRUE`. This meant that the file `/opt/nfast/cknfastrc` used to require the following additional line:

```
CKNFAST_OVERRIDE_SECURITY_ASSURANCES=explicitness;tokenkeys;longterm
```

If you are upgrading Oracle Key Vault from release 18.3.0.0.0 or earlier, then these parameters will continue to be needed until you have reverse-migrated and re-initialized, which will create a new RoT key with the `CKA_EXTRACTABLE` attribute set to `CK_FALSE`. If you prefer that future initialize operations continue have `CKA_EXTRACTABLE` set to `CK_TRUE` in release 18.4.0.0.0 and later, then perform the following steps before initializing.

1. Connect to the Oracle Key Vault server as the `root` user.

   ```
   $ ssh support@Oracle_Key_Vault_server_IP_address
   support$ su - root
   ```

2. Open the file `okv_security.conf`:

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

3. Set the `HSM_KEY_EXTRACTABLE` parameter in `okv_security.conf` as follows:

```
HSM_KEY_EXTRACTABLE="1"
```

4. Save and exit the `okv_security.conf` file.

   ```
   :wq
   ```

5. Open the `/opt/nfast/cknfastrc` file.

   ```
   root# vi /opt/nfast/cknfastrc
   ```

6. Add the following line to `/opt/nfast/cknfastrc`:

   ```
   CKNFAST_OVERRIDE_SECURITY_ASSURANCES=explicitness;tokenkeys;longterm
   ```

Future initialize operations on this Oracle Key Vault server will now create Root of Trust keys with the `CKA_EXTRACTABLE` attribute set to `CK_TRUE`.

3.2.3 Re-installation of Entrust Software While Upgrading Oracle Key Vault from 18.x to 21.x

During the upgrade from Oracle Key Vault 18.*x* to 21.*x*, the operating system is upgraded from Oracle Linux 6 to Oracle Linux 7.

In order to preserve HSM integration functionality after the upgrade, the following steps are performed automatically during the upgrade:

```
/opt/nfast/sbin/install
/sbin/usermod -a -G nfast oracle
Edits to the /etc/systemd/system/nc_hardserver.service file
```

No action is required unless these steps are insufficient to set up the Entrust HSM in your environment. Oracle recommends that you run a test upgrade on a non-production environment to ensure that the upgrade will work with your HSM configuration.

3.3 Using a Token Label After Upgrading Oracle Key Vault without Reverse-Migrating

Starting in release 18.4, Oracle Key Vault can use a token label when you choose a slot while connecting to an HSM.

If you are upgrading an HSM-enabled Oracle Key Vault from a previous version, then you can begin using a token label after successfully upgrading without reverse-migrating.

1. Perform the upgrade to Oracle Key Vault release 18.4 or later.
2. Locate the token label of the token that Oracle Key Vault is currently using.
   a. Log into the Oracle Key Vault management console as a user who has the System Administrator role.
   b. Select the **System** tab, then **Settings** in the left navigation bar.
   c. In the Network Services area, click **HSM** to display the Hardware Security Module page.

      If the HSM is configured and running, then the **Type** label indicates the token label. For example:
      Token label: myPartition1

Manufacturer ID: Safenet, Inc.

Firmware version: 6.109

d. Select the **Set Credential** button.

e. In the Prepare for HSM Restore window, select from the **Vendor** menu, enter the HSM credential in the **HSM Credential** and **Re-enter HSM Credential** fields. Check the **Use Token Label** check box and enter the token label that you found (for example, `myPartition1`) in the **Token Label** field. Then click **Set Credential**.

If this operation is successful, then Oracle Key Vault will begin to use the given token label when choosing a slot. If the operation fails, then your settings (token label, vendor, and HSM credential) are returned to what they were previously.

After this operation completes, if the green status arrow changes to a red arrow, this means that you have entered the wrong HSM credential, token label, vendor, or some combination of the three, and Oracle Key Vault was unable to revert the values to what they were previously. Oracle recommends that you try another set credential operation using the former HSM credential, token label, and vendor settings and do not restart Oracle Key Vault until the status arrow is again green. For more information about why the status arrow is red, check the most recent log files under the `/var/okv/log/hsm` directory.

4
Oracle Key Vault HSM Support Guidance

The Oracle Key Vault HSM support guidance provides information about troubleshooting and vendor specific notes.

- General Troubleshooting
  Oracle Key Vault provides general troubleshooting help. Vendor-specific notes cover vendor-specific troubleshooting.
- Vendor Specific Notes for Thales Luna
  Oracle Key Vault supports integration with Thales Luna (formerly Safenet Luna) NetworkHSM version 7000, but does not support Host Trust Link (HTL) for Thales Luna HSM.
- Vendor Specific Notes for Entrust
  You can integrate Oracle Key Vault release 12.2 BP 3 and later with the HSM from Entrust nShield Connect + and XC models.
- Vendor Specific Notes for Utimaco
  Oracle Key Vault supports Oracle Key Vault integration with Utimaco SecurityServer 4.31.1.

4.1 General Troubleshooting

Oracle Key Vault provides general troubleshooting help. Vendor-specific notes cover vendor-specific troubleshooting.

- Trace Files for Diagnosing Issues
  Oracle Key Vault provides trace files so that you can better diagnose issues that may arise.
- 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.
- Could Not Get Slot for HSM Error
  The `Could Not Get Slot for HSM` error indicates that Oracle Key Vault could not get a slot from the HSM.
- Could Not Load PKCS#11 Library Error
  The `Could Not Load PKCS#11 Library` error indicates that Oracle Key Vault could not load the PKCS#11 library.
- Oracle Key Vault Management Console Does Not Start After Restarting HSM-Enabled Oracle Key Vault Server
  The Oracle Key Vault management console may not appear after you restart the HSM-enabled Oracle Key Vault server.
- Primary-Standby Errors
  The `okv_security.conf` file contains settings that can help you diagnose primary-standby errors.

- Errors from HSM-Enabled Oracle Key Vault Backups
  You can use the `cwallet.sso` file to diagnose HSM-enabled Oracle Key Vault backup errors.
- Restoring an HSM-Enabled Backup
  Before you restore a backup that was taken on an HSM-enabled Oracle Key Vault, ensure that you have set the same HSM credential and token label that were used when the backup was taken.

4.1.1 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.

4.1.2 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 on the Configure Alerts page) to ensure that the Root of Trust key is available and the TDE wallet password can be decrypted. When a problem in the HSM configuration arises (for example, the HSM cannot be reached or if there are conflicting keys in the HSM with the same ID), then the up arrow on the **Hardware Security Module** page (accessed by selecting the **System** tab, then **Settings**, then in the Network Services area, clicking **HSM**) switches to a down arrow and an alert is raised. The down arrow signifies that the HSM is not configured or the HSM configuration has a problem. When an alert has been raised, the following error message appears: `HSM configuration error. Please refer to the HSM Alert section in the Oracle Key Vault Root of Trust HSM Configuration Guide`.

If this alert appears, then follow these steps:

1. Log in as root as follows:

   ```
   $ ssh support@okv_instance_ip_address
   support$ su - root
   ```

2. Back up the SSO wallet. For example:

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

3. Diagnose the source of the alert.
   The following `verify` command should show why the alert was raised. The `ls -ltrh` command shows the most recent log file at the bottom of the output.

   ```
   root# su - oracle
   oracle$ /usr/local/okv/hsm/bin/hsmclient verify
   oracle$ cd /var/okv/log/hsm
   oracle$ ls -ltrh
   ```

4. If you cannot resolve this problem, then contact Oracle Support.

4.1.3 Could Not Get Slot for HSM Error

The `Could Not Get Slot for HSM` error indicates that Oracle Key Vault could not get a slot from the HSM.

Consult the most recent trace files for more details. Possible causes include providing an invalid or nonexistent token label and the HSM failing to return a list of slots.

4.1.4 Could Not Load PKCS#11 Library Error

The `Could Not Load PKCS#11 Library` error indicates that Oracle Key Vault could not load the PKCS#11 library.

Possible reasons for this error could be due to file permission issues or failing to properly deploy the HSM client software on Oracle Key Vault. More details can be found in recent trace files. Oracle looks for the PKCS#11 library at the following locations, depending on the vendor:

- For Thales Luna, `/usr/safenet/lunaclient/lib/libCryptoki2_64.so`
- For Entrust, `/opt/nfast/toolkits/pkcs11/libcknfast.so`
- For Utimaco, `/opt/utimaco/lib/libcs_pkcs11_R2.so`

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

The Oracle Key Vault management console may not appear after you restart the HSM-enabled Oracle Key Vault server.

If this happens, then log into the Oracle Key Vault server using SSH as user `support` and try manually opening the wallet as follows:

```
$ ssh support@okv_instance_ip_address
support$ su root
root# su oracle
oracle$ cd /usr/local/okv/hsm/bin
oracle$ ./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, then check the recent log files under `/var/okv/log/hsm` and check for vendor-specific instructions.

4.1.6 Primary-Standby Errors

The `okv_security.conf` file contains settings that can help you diagnose primary-standby errors.

- Check that the files have been transported to the standby server.

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

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

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

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

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

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

  You should see the number within double quotes.

- Check the vendor-specific instructions.

4.1.7 Errors from HSM-Enabled Oracle Key Vault Backups

You can use the `cwallet.sso` file to diagnose HSM-enabled Oracle Key Vault backup errors.

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

Execute the command `ls -l` as `root` on the Oracle Key Vault server to which you are restoring the backup:

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

You should see the wallet file `cwallet.sso`, which indicates that the credential has successfully been set and stored on Oracle Key Vault.

You should also check that you have followed the instructions from the HSM vendor. In addition, check the most recent log files generated by the recent backup restore, which are in the `/var/okv/log/db` directory.

4.1.8 Restoring an HSM-Enabled Backup

Before you restore a backup that was taken on an HSM-enabled Oracle Key Vault, ensure that you have set the same HSM credential and token label that were used when the backup was taken.

The HSM credential for Thales Luna is the Thales Luna partition password. For Entrust, the credential is the password that is associated with the Operator Card Set or Softcard. For Utimaco, the credential is the PIN that was initialized when the token was configured.

When using the Set Credential operation, if you enter an incorrect credential or token label, or if Oracle Key Vault is unable to connect to the HSM, then the operation will not succeed and the credential, token label, and vendor provided will not be stored. Ensure that Oracle Key Vault has been enrolled as a client of the HSM and then ensure that the correct credential and token label are entered such that Oracle Key Vault will be able to access the same Root of Trust key that was in use when the backup was taken.

For more information about enrolling Oracle Key Vault as a client of the HSM, see Enrolling Oracle Key Vault as a Client of the HSM.

4.2 Vendor Specific Notes for Thales Luna

Oracle Key Vault supports integration with Thales Luna (formerly Safenet Luna) NetworkHSM version 7000, but does not support Host Trust Link (HTL) for Thales Luna HSM.

- Installing the HSM Client Software on the Oracle Key Vault Server for Thales Luna
  You must use the latest Luna Universal Client version for Linux x64 for the installation.
- HSM Credential for Thales Luna
  The HSM credential is the Thales Luna partition password.
- Token Label for Thales Luna
  The token label for Thales Luna is the name of the partition.
- Enrolling Oracle Key Vault as a Client of a Thales Luna HSM
  To perform the enrollment, you use the Oracle Key Vault management console and the command-line interface.
- HSM Provider Value for Thales Luna
  For Thales Luna, the provider value is 1.
- HSM Vendor Specific Checks for Thales Luna
  You should check the Thales Luna vendor-specific settings.

4.2.1 Installing the HSM Client Software on the Oracle Key Vault Server for Thales Luna

You must use the latest Luna Universal Client version for Linux x64 for the installation.

1. Obtain the latest Luna Universal client software package for Linux x64.
2. Transport the Thales Luna client software package to the Oracle Key Vault machine. Oracle recommends using SCP. For example, assuming the Thales Luna client software packages is called `safenet.tar`:

   ```
   $ scp safenet.tar support@okv_instance_ip_address:/tmp
   ```

3. Install the Thales Luna client software on Oracle 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_ip_address
   support$ su root
   root# cd /usr/local/okv/hsm
   ```

5. Find the Linux 64-bit packages:

```
root# tar -tvf /tmp/safenet.tar | grep 'linux/64/'; Output:
"SafeNet_package_version/linux/64/"
```

6. Only extract the Linux 64-bit packages:

```
root# tar -xvf /tmp/safenet.tar SafeNet_package_version/linux/64/
root# cd SafeNet_package_version/linux/64
root# ./install.sh
```

7. Accept the Thales Luna license by typing `y` at the prompt.
8. Install the Luna SA by entering `1`, `n`, `i` at the successive prompts.

   This installs the Thales Luna software in the directory `/usr/safenet/lunaclient`.
9. Delete the `safenet.tar` file from `/tmp` directory.

```
root# rm -f /tmp/safenet.tar
```

4.2.2 HSM Credential for Thales Luna

The HSM credential is the Thales Luna partition password.

If you are using Thales Luna as your HSM, then you can use the Thales Luna `assignPassword` command to assign a password for a partition. However, do not do this when a partition is currently in use by an Oracle Key Vault server, because Oracle Key Vault will no longer be able to access the Root of Trust key as its stored credential will no longer be correct.

4.2.3 Token Label for Thales Luna

The token label for Thales Luna is the name of the partition.

4.2.4 Enrolling Oracle Key Vault as a Client of a Thales Luna HSM

To perform the enrollment, you use the Oracle Key Vault management console and the command-line interface.

1. Log in to the Oracle Key Vault management console as a user who has the System Administrator role.
2. Set up the DNS servers.

   When enrolling Oracle Key Vault as a client of a Thales Luna HSM, if the HSM will be registered using a host name, you should first set up DNS using the Oracle Key Vault the management console. To access the DNS settings, select the **System** tab, and then from the left navigation bar, select **Settings**. In Network Services, click **DNS**.
   You must configure the DNS servers on each Oracle Key Vault server that you plan to register as a client of the HSM. In a primary-standby environment, configure the DNS servers on both primary and standby server before pairing. For a multi-master cluster, configure DNS on each node in the cluster that will be registered as a client of the HSM.

3. Exchange certificates between Oracle Key Vault and the Thales Luna SA HSM.

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

   ```
   $ ssh support@okv_instance_ip_address
   support$ su root
   root# cd /usr/safenet/lunaclient/bin
   root# scp admin@hsm_hostname:server.pem .
   root# ./vtl addServer -n hsm_hostname -c server.pem
   root# ./vtl createCert -n okv_hostname
   root# scp /usr/safenet/lunaclient/cert/client/okv_hostname.pem
   admin@hsm_hostname:
   ```

   You must enter the HSM administrative password when using SCP with the HSM.

4. Register Oracle Key Vault as a client of the Thales Luna SA.

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

   Access the HSM administrative console by using SSH to `admin@hsm_hostname` and providing the administrative password:

   ```
   $ client register -client client_name -hostname okv_hostname
   $ client hostip map -c client_name -i okv_ip_address
   $ client assignPartition -client client_name -partition partition_name
   ```

5. Verify the enrollment as follows:

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

   ```
   $ ssh support@okv_instance_ip_address
   support$ 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_number   partition_name
   ```

4.2.5 HSM Provider Value for Thales Luna

For Thales Luna, the provider value is 1.

If you are setting this value manually for a primary-standby configuration, then set `HSM_PROVIDER="1"` in the `okv_security.conf` file. For more information about enabling HSM in a primary-standby deployment, see Enabling HSM in a High Availability Deployment.

4.2.6 HSM Vendor Specific Checks for Thales Luna

You should check the Thales Luna vendor-specific settings.

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

   ```
   $ ssh support@okv_instance_ip_address
   support$ su root
   ```

2. Execute the `vtl verify` command.

   ```
   root# cd /usr/safenet/lunaclient/bin
   root# ./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, then it means that the HSM is not set up properly. You can diagnose further by completing the remaining steps in this procedure.

3. Log into the Thales Luna SA administrative console.
4. Enter the following command:

   ```
   client show -client client_name
   ```

5. Verify that the expected client exists and is assigned a partition.
6. If it does not exist, then register the client with the command:

   ```
   client register -client client_name-hostname host_name
   ```

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

   ```
   client assignPartition -client client_name -partition partition_name
   ```

8. Verify that all client IP addresses are mapped correctly. If entries are missing, run the command:

   ```
   client hostip map -c client_name -i ip_address
   ```

9. Verify that Oracle Key Vault can reach the HSM using the `vtl verify` command:

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

The output should look similar to the following output:

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

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

If the command fails, then it means that the Oracle 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 Oracle Key Vault's access to the HSM has not been revoked. If you are unable to resolve the problem, then contact Oracle Support.

4.3 Vendor Specific Notes for Entrust

You can integrate Oracle Key Vault release 12.2 BP 3 and later with the HSM from Entrust nShield Connect + and XC models.

- Installing the HSM Client Software on the Oracle Key Vault Server for Entrust
  The Entrust HSM requires a separate non-HSM computer on the network to use as the remote file system.
- HSM Credential for Entrust
  The HSM credential for Entrust is the password that is associated with the Operator Card Set or Softcard.
- Token Label for Entrust
  The token label for Entrust is the name of the Operator Card Set or Softcard.
- Enrolling Oracle Key Vault as a Client of an Entrust HSM
  You use both the Entrust user interface and the command line to enroll Oracle Key Vault as a client of an Entrust HSM.
- HSM Provider Value for Entrust
  For Entrust, the provider value is 2.

4.3.1 Installing the HSM Client Software on the Oracle Key Vault Server for Entrust

The Entrust HSM requires a separate non-HSM computer on the network to use as the remote file system.

1. Set up the remote file system.
2. Log in to the Oracle Key Vault server as support user using SSH:

   ```
   $ ssh support@okv_instance_ip_address
   ```

3. Switch to `root`:

   ```
   support$ su root
   ```

4. 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
```

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

   For example:

   ```
   root# scp user@remote_file_system_computer:/source_directory/ncipher/nfast/
   ctls/agg.tar ctls
   root# scp user@remote_file_system_computer:/source_directory/ncipher/nfast/
   hwsp/agg.tar hwsp
   root# scp user@remote_file_system_computer:/source_directory/ncipher/nfast/
   pkcs11/user.tar pkcs11
   ```

6. 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
   ```

7. As `root`, perform additional edits to the groups and service file on the Oracle Key Vault server:

   ```
   root# usermod -a -G nfast oracle
   root# vi /etc/systemd/system/nc_hardserver.service
   ```

   Inside the `vi` editor, edit the file so that the line `Before=dbfwdb.service` is added after the line that starts with `After=`.

   For example, before the change:

   ```
   [Unit]
   Description=nFast hardserver service
   Wants=remote-fs.target rsyslog.service nc_drivers.service
   nc_exard.service
   After=remote-fs.target rsyslog.service nc_drivers.service
   nc_exard.service
   ```

   After the change, it looks like this:

   ```
   [Unit]
   Description=nFast hardserver service
   Wants=remote-fs.target rsyslog.service nc_drivers.service
   nc_exard.service
   After=remote-fs.target rsyslog.service nc_drivers.service
   nc_exard.service
   Before=dbfwdb.service
   ```

8. 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.

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

In the Oracle Key Vault management console, log in as a user with the System Administrator role. Select the **System** tab, and then select **Status** in the left navigation bar. Then click the **Reboot** button.

4.3.2 HSM Credential for Entrust

The HSM credential for Entrust is the password that is associated with the Operator Card Set or Softcard.

The HSM-credential if they use an Operator Card Set is the Operator Card Set password. If they use a Softcard, then the password is the Softcard password.

4.3.3 Token Label for Entrust

The token label for Entrust is the name of the Operator Card Set or Softcard.

4.3.4 Enrolling Oracle Key Vault as a Client of an Entrust HSM

You use both the Entrust user interface and the command line to enroll Oracle Key Vault as a client of an Entrust HSM.

1. Add the Oracle Key Vault server IP address to the client list on the HSM using the front panel. Select privileged on any port.
   - In a primary-standby environment, register both the primary server and the standby server to use the Entrust HSM.
   - In a multi-master cluster environment, register each Oracle Key Vault node that will use the Entrust HSM.
2. Switch to user `oracle`:

   ```
   root# su oracle
   oracle$ PATH=/opt/nfast/bin:$PATH
   oracle$ export PATH
   ```
3. On the Oracle Key Vault server, enroll with the HSM:

   ```
   oracle$ nethsmenroll hsm_ip_address hsm_esn hsm_keyhash
   ```
4. Configure the TCP sockets:

   ```
   oracle$ config-serverstartup --enable-tcp --enable-privileged-tcp
   ```
5. Switch back to `root` from `oracle` by entering `exit`.

   ```
   oracle$ exit
   ```
6. In `root`, restart the hardserver (Entrust client process that communicates with the HSM):

   ```
   root# /opt/nfast/sbin/init.d-ncipher restart
   ```
7. On the remote file system computer, run the following command:

   ```
   $ /opt/nfast/bin/rfs-setup --gang-client --write-noauth okv_server_ip_address
   ```
8. On the Oracle Key Vault server as user `oracle`, run the following commands:

   ```
   oracle$ /opt/nfast/bin/rfs-sync --setup --no-authenticate
   remote_file_system_ip_address
   oracle$ /opt/nfast/bin/rfs-sync --update
   ```

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

9. Test PKCS#11 access as follows:

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

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

```
CKNFAST_NO_ACCELERATOR_SLOTS=1
CKNFAST_OVERRIDE_SECURITY_ASSURANCES=none
```

11. Perform the steps described in HSM-Enabling in a Standalone Oracle Key Vault Deployment.

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

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

If you do not run this command after each HSM initialize operation, then the Root of Trust key may not be available for other operations such as restoring backups and setting up a primary-standby configuration.

4.3.5 HSM Provider Value for Entrust

For Entrust, the provider value is 2.

If you are setting this value manually for the primary-standby, then set `HSM_PROVIDER="2"`. For more information about enabling HSM in a primary-standby deployment, see Enabling HSM in a High Availability Deployment.

4.4 Vendor Specific Notes for Utimaco

Oracle Key Vault supports Oracle Key Vault integration with Utimaco SecurityServer 4.31.1.

- Installing the HSM Client Software on the Oracle Key Vault Server for Utimaco
  The setup files for Utimaco are provided in the `SecurityServerEvaluation-V4.31.1.0.zip` file from Utimaco.
- HSM Credential for Utimaco
  The HSM credential for Utimaco is the PIN that was initialized when the token was configured.
- Token Label for Utimaco
  The token label for Utimaco is the name of the token that was set up for the HSM.
- HSM Provider Value for Utimaco
  For Utimaco, the provider value is 3.
- HSM Vendor Specific Checks for Utimaco
  You should check the Utimaco vendor-specific settings.

4.4.1 Installing the HSM Client Software on the Oracle Key Vault Server for Utimaco

The setup files for Utimaco are provided in the `SecurityServerEvaluation-V4.31.1.0.zip` file from Utimaco.

1. Locate the necessary setup files provided in the `SecurityServerEvaluation-V4.31.1.0.zip` file from Utimaco.
2. After unzipping the Utimaco zip file, transport the necessary files to the Oracle Key Vault machine. Oracle recommends using SCP. For example:

   ```
   $ scp unzip_directory/Software/Linux/x86-64/Crypto_APIs/PKCS11_R2/sample/
   cs_pkcs11_R2.cfg support@okv_instance_ip_address:/tmp
   $ scp unzip_directory/Software/Linux/x86-64/Crypto_APIs/PKCS11_R2/lib/
   libcs_pkcs11_R2.so support@okv_instance_ip_address:/tmp
   $ scp unzip_directory/Software/Linux/x86-64/Administration/p11tool2
   support@okv_instance_ip_address:/tmp
   $ scp unzip_directory/Software/Linux/x86-64/Administration/cxitool
   support@okv_instance_ip_address:/tmp
   $ scp unzip_directory/Software/Linux/x86-64/Administration/csadm
   support@okv_instance_ip_address:/tmp
   ```

3. Log in to the Oracle Key Vault server as user `support`, and switch user (`su`) to `root`:

   ```
   $ ssh support@okv_instance_ip_address
   support$ su - root
   ```

4. Create the appropriate directories for the Utimaco files:

   ```
   root# mkdir -p /opt/utimaco/lib
   root# mkdir /opt/utimaco/bin
   root# mkdir /etc/utimaco
   ```

5. Move the Utimaco files to the correct directories:

   ```
   root# mv /tmp/cs_pkcs11_R2.cfg /etc/utimaco
   root# mv /tmp/p11tool2 /opt/utimaco/bin
   root# mv /tmp/cxitool /opt/utimaco/bin
   root# mv /tmp/csadm /opt/utimaco/bin
   root# mv /tmp/libcs_pkcs11_R2.so /opt/utimaco/lib
   ```

6. Change the configuration file permissions:

   ```
   root# /bin/chmod 640 /etc/utimaco/cs_pkcs11_R2.cfg
   root# /bin/chown oracle:oinstall /etc/utimaco/cs_pkcs11_R2.cfg
   ```

7. Change the executable file permissions:

   ```
   root# /bin/chmod 550 /opt/utimaco/bin/*
   root# /bin/chown oracle:oinstall /opt/utimaco/bin/*
   ```

8. Change the library file permissions:

   ```
   root# /bin/chmod 440 /opt/utimaco/lib/libcs_pkcs11_R2.so
   root# /bin/chown oracle:oinstall /opt/utimaco/lib/libcs_pkcs11_R2.so
   ```

9. Modify the configuration file `/etc/utimaco/cs_pkcs11_R2.cfg` so that **Device** is set to the Utimaco HSM's IP address.

   ```
   Device = utimaco_ip_address
   ```

   If you are testing with an Utimaco HSM simulator, the line should be in the format:

   ```
   Device = 3001@utimaco_ip_address
   ```

   Oracle does not recommend that you use the simulator in a production environment.

10. To verify that you have set up your Utimaco HSM and the client files correctly, you can use `p11tool2`. The `p11tool2` command call can be used to verify that the PKCS11 token has been configured:

    ```
    root# /opt/utimaco/bin/p11tool2 GetSlotInfo
    ```

    The output should look similar to the following output:

    ```
    CK_SLOT_INFO (slot ID: 0x00000000):

      slotDescription     33303031 4031302e  3234302e 3131382e |
    3001@10.240.118.|
                          32333120 2d20534c  4f545f30 30303020 |231 -
    SLOT_0000 |
                          20202020 20202020  20202020 20202020
    |                |
                          20202020 20202020  20202020 20202020
    |                |

      manufacturerID      5574696d 61636f20  49532047 6d624820 |Utimaco
    IS GmbH |
                          20202020 20202020  20202020 20202020
    |                |

      flags: 0x00000005
        CKF_TOKEN_PRESENT    : CK_TRUE
        CKF_REMOVABLE_DEVICE : CK_FALSE
        CKF_HW_SLOT          : CK_TRUE

      hardwareVersion        : 5.01
      firmwareVersion        : 2.03
    ```

11. The `csadm` command call can be used to confirm that the PKCS11 users have been defined:

    ```
    root# /opt/utimaco/bin/csadm Dev=utimaco_ip_address ListUsers
    ```

The output should look similar to the following output:

```
Name       Permission   Mechanism      Attributes
ADMIN       22000000    RSA sign       Z[0]
SO_0000     00000200    HMAC passwd    Z[0]A[CXI_GROUP=SLOT_0000]
USR_0000    00000002    HMAC passwd    Z[0]A[CXI_GROUP=SLOT_0000]
```

4.4.2 HSM Credential for Utimaco

The HSM credential for Utimaco is the PIN that was initialized when the token was configured.

See the Utimaco documentation for more details.

4.4.3 Token Label for Utimaco

The token label for Utimaco is the name of the token that was set up for the HSM.

4.4.4 HSM Provider Value for Utimaco

For Utimaco, the provider value is 3.

If you are setting this value manually for primary-standby, set `HSM_PROVIDER="3"` in the `okv_security.conf` file. For more information about enabling HSM in a primary-standby deployment, see Enabling HSM in a High Availability Deployment.

4.4.5 HSM Vendor Specific Checks for Utimaco

You should check the Utimaco vendor-specific settings.

In addition to the `p11tool2 GetSlotInfo` and `csadm ListUsers` commands, you can also check to see that a key was created after completing the HSM Initialize operation. Note that more keys may be created after subsequent HSM initialize commands.

```
root# /opt/utimaco/bin/p11tool2 LoginUser=HSM_Credential ListObjects
```

The output should look similar to the following output:

```
CKO_DATA:

+ 1.1
  CKA_LABEL                     = OKV 18.1 HSM Key Number

CKO_SECRET_KEY:

+ 2.1
  CKA_KEY_TYPE                  = CKK_AES
  CKA_SENSITIVE                 = CK_TRUE
  CKA_EXTRACTABLE               = CK_FALSE
```

```
CKA_LABEL                       = OKV 18.1 HSM Root Key
CKA_ID                          = 0x00000001 ()
```