# Oracle® Key Vault

## Administrator's Guide

Release 12.2 BP12
F22626-09
October 2020

Oracle Key Vault Administrator's Guide, Release 12.2 BP12

F22626-09

Primary Author: Siddharth Naidu

Contributing Authors: Usha Krishnamurthy

Contributors: Dongwon Park, Eric Paapanen, Hui Li, Khushal Melana, Michael Leong, Min-Hank Ho, Radhika Siravara, Rahil Mir, Saikat Saha, Shirley Kumamoto, Sindhu Ravichandran, Srivatsan Kannan, Sunil Pulla, Swapna Jawarikapisha, Vipin Samar

# Contents

## Preface

## What's New in Oracle Key Vault 12.2

## 1 Introduction to Oracle Key Vault

## 2 Oracle Key Vault Concepts

## 3 Oracle Key Vault Installation and Configuration

## 4 High Availability, Backup and Restore Operations

## 5 Managing Oracle Key Vault Users

## 6 Managing Oracle Key Vault Virtual Wallets and Security Objects

## 7 Managing Oracle Key Vault Endpoints

## 11 Oracle Key Vault Use Case Scenarios

## 12 General Oracle Key Vault Management

## 13 Managing Certificates

## A Installing and Configuring Oracle Key Vault 12.2.0.4.0 and Earlier

## B Troubleshooting Oracle Key Vault

# List of Examples

# List of Figures

# List of Tables

# Preface

Welcome to *Oracle Key Vault Administrator's Guide*. This guide explains how to install, configure, and use 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, indeed any information security personnel, 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.

## Related Documents

For more information, see these Oracle resources:

- *Oracle Database Advanced Security Guide*
- *Oracle Database Security Guide*
- *Oracle Database Administrator's Guide*
- *Oracle Data Guard Concepts and Administration*
- *Oracle Real Application Clusters Administration and Deployment Guide*

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

`http://www.oracle.com/technetwork/database/options/key-management/overview/index.html?ssSourceSiteId=ocomen`

If you already have a user name and password for OTN, then you can go directly to the documentation section of the OTN website at

`http://www.oracle.com/technetwork/database/options/key-management/documentation/index.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. |

# What's New in Oracle Key Vault 12.2

This section lists the new features and enhancements of the following Oracle Key Vault Release 12.2 bundle patches.

## New Features in Bundle Patch 11

The following are the new features and enhancements in Bundle Patch 11 (also referred to as Oracle Key Vault Release 12.2.0.11.0).

### Re-Enroll All Endpoints With a Single RESTful Command

If a customer has many endpoints, it is time consuming to re-enroll all endpoints one by one, even with RESTful API. With this enhancement, Oracle Key Vault provides a single RESTful command to re-enroll all endpoints. The new RESTful command is `re_enroll_all`.

## New Features in Bundle Patch 10

The following are the new features and enhancements in Bundle Patch 7 (also referred to as Oracle Key Vault Release 12.2.0.7.0).

## Oracle Key Vault Server Certificate Rotation

Starting with this release, you can rotate certificates in the Oracle Key Vault server and for all endpoints in one operation. You can use this feature in both standalone and primary-standby environments. You also can configure an alert to warn you when it is time to rotate the certificates. This feature helps to prevent the scenario of inadvertently allowing certificates to expire. If this happens, then the endpoints cannot connect to the Oracle Key Vault server, and you would need to re-enroll the endpoints.

**Related Topics**

- Rotating Certificates (page 13-1)

## Re-Enroll All Endpoints With a Single RESTful Command

If a customer has many endpoints, it is time consuming to re-enroll all endpoints one by one, even with RESTful API. With this enhancement, Oracle Key Vault provides a single RESTful command to re-enroll all endpoints. The new RESTful command is `re_enroll_all`.

**Related Topics**

- re_enroll_all Command (page 10-16)
  The `re_enroll_al` command re-enrolls all previously enrolled endpoints in order to upgrade the endpoint software.

## Periodically Verify Oracle Key Vault Access to RoT in HSM in Alert Job

In Oracle Key Vault release 12.2.0.9 and earlier with Root-of-Trust provided by a hardware security module (HSM), Oracle Key Vault needed to access the HSM only once during startup.

Starting with Oracle Key Vault release 12.2.0.10.0, periodic checks were added to the internal alert framework that validate connectivity to the HSM, presence of the Root-of-Trust key in the HSM, as well as a sanity check of the HSM client software installation inside Oracle Key Vault. If any of these checks fail, then an alert is raised. The time between tests can be configured to multiples of 5 minutes.

In Oracle Key Vault 12.2.0.10.0, as a part of the regular alert job, the server will attempt to validate the HSM configuration by contacting the HSM and using the Root of Trust key, as well as doing sanity checks of the HSM-related files on the server. In the case where any of the aforementioned checks are failed, a new alert is raised so that you can immediately take action to protect your server. By default, it will contact the HSM every five minutes, but you can configure the amount of time between checks or disable the alert altogether.

**Related Topics**

- *Oracle Key Vault Integration with Hardware Security Module (HSM)*

# New Features in Bundle Patch 9

Bundle Patch 9 (also referred to as Oracle Key Vault Release 12.2.0.9.0) does not include any new features and enhancements.

# New Features in Bundle Patch 8

Bundle Patch 8 (also referred to as Oracle Key Vault Release 12.2.0.8.0) does not include any new features and enhancements.

# New Features in Bundle Patch 7

The following are the new features and enhancements in Bundle Patch 7 (also referred to as Oracle Key Vault Release 12.2.0.7.0).

- Automatic Endpoint Configuration tuning using OKV Management Console (page xxv)
- UEFI BOOT MODE Support (page xxv)
- Support for IBM AIX 5.3 Endpoint Platform (page xxvi)

## Automatic Endpoint Configuration tuning using OKV Management Console

Users with system administrator role can centrally update certain endpoint configuration parameters in the Oracle Key Vault Management Console. This feature enables system administrators to set certain endpoint configuration parameters globally, i.e. for all endpoints, or on a per-endpoint basis. It simplifies the process of managing multiple endpoints for system administrators.

Endpoint specific parameters if set take precedence over global parameters. Global parameters, if set will take effect when endpoint-specific parameters are cleared. OKV will use the default system parameters if both global and endpoint specific parameters are cleared or not set from OKV management console.

The configuration parameter values set in the OKV management console are pushed to endpoints dynamically. After configuration parameters have been set in the OKV Management Console, the next time the endpoint contacts the OKV server, it will get the configuration parameters update. Endpoint configuration parameter update is best-effort. In case of error, the update is not applied. Both `okvutil` and `PKCS11` library can get and apply the endpoint configuration updates.

For more information, see Configuring Global Endpoint Configuration Parameters (page 7-24) and Configuring Endpoint Configuration Parameters (page 7-7).

## UEFI BOOT MODE Support

Oracle Key Vault 12.2.0.7.0 supports both Legacy BIOS and UEFI BIOS boot modes. The support for UEFI BIOS mode allows the installation of Oracle Key Vault on servers that exclusively support UEFI BIOS only, such as Oracle X7-2 Server. OKV can be installed on Oracle X7–2 servers as a standalone server or in a High Availability (HA) configuration. For more information, see *Oracle Key Vault Administrator's Guide*.

## Support for IBM AIX 5.3 Endpoint Platform

IBM AIX 5.3 is one of the supported operating systems for Oracle Database 11g Release 2. Oracle Key Vault 12.2.0.7.0 has added AIX 5.3 as a new endpoint platform in limited capacity. This is in addition to the AIX platforms 6.2 and 7.1 that are already supported since OKV 12.2. Customers can enroll endpoints with AIX 5.3 as their endpoint platform. You can download the endpoint software for AIX 5.3 platform from the OKV server during endpoint enrollment and provisioning process. However, before deploying the endpoint software the customer should enable TLS 1.0 on the OKV server. For more information, see *Oracle Key Vault Administrator's Guide*.

# New Features in Bundle Patch 6

The following are the new features and enhancements in Bundle Patch 6 (also referred to as Oracle Key Vault Release 12.2.0.6.0).

- Quick Discovery of Unreachable Key Vault Server (page xxvi)
- Extend Persistent Master Key Cache for Improved Resiliency with Unavailable Key Vault Servers (page xxvii)
- Support for Bring Your Own TDE Master Encryption Keys (page xxvii)
- Update SNMP Settings on Standby Server (page xxviii)
- New Alerts for High Availability Operations (page xxviii)
- New install option to update the Oracle database-specific okvclient.ora symlink (page xxix)
- Support for Oracle Database 12.2.0.1.0 on Windows Server 2008 and 2012 (page xxix)

## Quick Discovery of Unreachable Key Vault Server

In Oracle Key Vault 12.2.0.5.0 and earlier, clients attempt to connect to Oracle Key Vault by checking each of the two Oracle Key Vault servers in HA deployment. If the Oracle Key Vault server is unavailable, the client currently encounters an OS-defined delay which could be 20 seconds or longer, depending upon the OS.

In the HA deployment, the endpoint first attempts to connect to primary and then to the standby as specified in the endpoint configuration file. A switchover or a failover does not update the configuration files on the endpoint. So the endpoint continues to try and setup a connection with the server that was configured as the primary before switchover. This endpoint encounters an OS-defined delay and then moves on to make a connection to the server configured as the standby and succeeds. An endpoint thus encounters an OS-defined delay of several seconds before it can get a response from Oracle Key Vault. Every new process that attempts to setup a connection to Oracle Key Vault where a switchover or failover has taken place encounters this delay. This significantly slows down database operations like database startup, where many processes, one after the other, attempt a connection to Oracle Key Vault.

In Oracle Key Vault 12.2.0.6.0, clients first establish a non-blocking TCP connection to Oracle Key Vault to quickly detect unreachable servers. Oracle Key Vault 12.2.0.6.0 introduces the `SERVER_POLL_TIMEOUT` parameter in the `okvclient.ora` file, after which

Oracle Key Vault would attempt to connect to the next server. The default value is 300 (milliseconds).

After the first attempt, the client makes a second and final attempt to connect to the server but this time waits for twice as long as the duration specified by the SERVER_POLL_TIMEOUT parameter. This is done to overcome possible network congestion or delays.

For more information about the SERVER_POLL_TIMEOUT parameter, see Endpoint okvclient.ora Configuration File (page 8-11).

## Extend Persistent Master Key Cache for Improved Resiliency with Unavailable Key Vault Servers

In Oracle Key Vault 12.2.0.5.0 and earlier, if the Oracle Key Vault server is not available, the PKCS#11 library retrieves master key from the Persistent Master Key Cache if set. However, in the unlikely scenario that the Key Vault Servers are still not available, and if the persistent master key cache time limit specified by the PKCS11_PERSISTENT_CACHE_TIMEOUT has expired, the PKCS#11 library attempt to refresh the master key fails and the endpoint database operations are affected.

The Refresh Window feature of the Persistent Master Key Cache enables the database endpoint to make multiple attempts to refresh the expired master key from the OKV server. In that sense, the endpoint waits for the OKV server to be back online for the master key refresh to complete. Meanwhile, if the master key refresh attempt fails, the keys are retrieved from the persistent cache for the duration of the refresh window.

The Refresh Window feature of the Persistent Master Key Cache thus extends the duration for which the master key is available after it is cached in the persistent master key cache. At the same time the endpoints can refresh the key during the refresh window instead of once at the end of the cache time. This addresses the possibility that persistent cache expires in the window when the Oracle Key Vault is unavailable such as when HA switchover is in progress. The refresh window terminates and the cache period begins as soon as the key is refreshed.

In the okvclient.ora file, the parameter PKCS11_PERSISTENT_CACHE_REFRESH_WINDOW is used to extend the duration for which the master key is available after it is cached in the persistent master key cache. This value reflects the amount of time it takes for the Oracle Key Vault server to recover and come back online. The value is specified in minutes. The default value for PKCS11_PERSISTENT_CACHE_REFRESH_WINDOW is 30 (minutes).

For more information about the Persistent Cache Refresh Window, see Persistent Master Key Cache - Refresh Window (page 11-21).

## Support for Bring Your Own TDE Master Encryption Keys

You can now import your generated key to be used as the Transparent Data Encryption (TDE) master encryption key in Oracle Key Vault.

Key Administrators can upload this user-defined key to the groups that they have write access to. This feature provides Key Administrators with more control on creation of the master key used to encrypt TDE data encryption keys.

The type parameter of the okvutil upload command features a new option TDE_KEY_BYTES that allows you to upload a user-defined key to Oracle Key Vault. The key is then registered as a TDE master encryption key by running the ADMINISTER KEY MANAGEMENT command on the database. For more information about activating a TDE master encryption key, see Activating TDE Master Encryption Keys.

For more information about importing a user-defined key to use as the Transparent Data Encryption (TDE) master key in Oracle Key Vault, see Using an User-Defined Key as the TDE Master Encryption Key (page 11-25).

## Update SNMP Settings on Standby Server

In a High Availability deployment of Oracle Key Vault 12.2.0.5.0 and earlier, SNMP Settings on the Standby server cannot be updated, as the Oracle Key Vault management console is unavailable on the Standby server.

Oracle Key Vault 12.2.0.6.0 introduces the stdby_snmp_enable script to enable the root user to modify SNMP settings on the standby server.

For more information about using the stdby_snmp_enable script, see Changing SNMP Settings on the Standby Server (page 12-3).

## New Alerts for High Availability Operations

Oracle Key Vault generates alerts to inform Administrators about certain conditions that may affect the functioning of Oracle Key Vault.

- Generate alert when FSFO failure causes HA configuration process to fail (page xxviii)
- Generate alert when HA nodes are not successfully synchronized (page xxviii)

## Generate alert when FSFO failure causes HA configuration process to fail

When FSFO (Fast Start Failover) is unavailable due to a failure, the High Availability configuration process fails. This issue is not displayed in the Oracle Key Vault management console.

In Oracle Key Vault 12.2.0.6.0 and later, the following alert is generated to inform the Administrator that the High Availability configuration process failed due to a Fast Start Failover failure:

```
HA FSFO is not synchronized. FSFO status is <HA status>
```

For more information about configuring alerts, see Oracle Key Vault Alert Configuration (page 12-23).

## Generate alert when HA nodes are not successfully synchronized

If the Oracle Key Vault primary server is unable to function in High Availability mode because of an Active Data Guard or other unknown failure, the issue is not displayed in the Oracle Key Vault management console.

In Oracle Key Vault 12.2.0.6.0 and later, the following alert is generated to inform the Administrator that the Oracle Key Vault primary server is unable to function in High Availability mode due to an Active Data Guard or other unknown failure:

```
Dataguard Broker is disabled
```

For more information about configuring alerts, see Oracle Key Vault Alert Configuration (page 12-23).

## New install option to update the Oracle database-specific okvclient.ora symlink

In Oracle Key Vault 12.2.0.5.0 and earlier, the symlink reference to `okvclient.ora` is not updated during re-enrollment.

In Oracle Key Vault 12.2.0.6.0, new `okvclient.jar` option `-o` allows you to overwrite the symlink reference pointing to `okvclient.ora` in the new directory.

## Support for Oracle Database 12.2.0.1.0 on Windows Server 2008 and 2012

Oracle Key Vault 12.2.0.6.0 supports Oracle Database 12.2.0.1.0 on Windows Server 2008 and Windows Server 2012.

Oracle Database 11.2.0.4 and 12.1.0.2 on Windows Server 2008 and Windows Server 2012 are also supported, as in previous versions of Oracle Key Vault.

# New Features in Bundle Patch 5

The following are the new features and enhancements in Bundle Patch 5 (also referred to as Oracle Key Vault Release 12.2.0.5.0).

- Support for Read- Only Restricted Mode (page xxix)
- Additional Attributes on the All Items page to improve Key Searchability (page xxx)
- Support for Sending Audit Records to Remote Syslog (page xxx)
- Persistent Cache - New Mode and Lookup (page xxx)
- Upgraded to Oracle Linux 6.9 (page xxx)
- Two Disc Installation (page xxx)

## Support for Read- Only Restricted Mode

Oracle Key Vault supports Read-Only Restricted mode in High Availability deployments. Read-Only Restricted mode ensures operational continuity when the primary or standby servers encounter a failure that disrupts communication between the servers. Read-Only Restricted mode ensures that keys are accessible in the event of a primary or standby server failure.

When Read-Only Restricted mode is enabled, the primary or standby server will ensure that operations such as key retrieval are not affected if the peer server encounters a failure. Operations that create or modify critical data are disabled.

In earlier releases, when the primary or standby servers encountered a failure, operations were disabled in order to prevent the risk of data loss. You can disable Read-Only Restricted mode to continue applying the previous mode of operation.

For more information, see High Availability Read-Only Restricted Mode (page 4-11).

## Additional Attributes on the All Items page to improve Key Searchability

Additional attributes like **Name**, **Deactivation Date** and **Protect Stop Date** added to the **All Items** page. This feature enables a user to search for the keys that are deactivated or will be deactivated soon. Keys uploaded using okvutil have multiple identifiers. This feature improves the lookup of such keys.

## Support for Sending Audit Records to Remote Syslog

Oracle Key Vault supports sending of audit records to a remote Syslog. Oracle Key Vault audit managers can enable this option, if remote Syslog has been configured by the System administrator.

For more information about configuring Syslog to store audit records, see Configuring Syslog to Store Audit Records (page 12-28).

## Persistent Cache - New Mode and Lookup

Persistent cache features a new mode of operation - Persistent Master Key Cache First. When a key is required by the database, a lookup is performed on the persistent cache before fetching the key from the Oracle Key Vault. This improves performance because the `PKCS#11` library will connect to Oracle Key Vault only if the key is not found in the persistent master key cache. This is the default persistent cache mode.

A new type `okv_persistent_cache`, is added to the okvutil list command. `Okv_persistent_cache` allows customers to view the persistent cache and check if the keys are available or expired.

For more information about the Persistent Master Key Cache, see Persistent Master Key Cache Modes of Operation (page 11-20).

## Upgraded to Oracle Linux 6.9

Oracle Key Vault 12.2.0.5.0 installs a stripped-down version of Oracle Linux 6.9 during a fresh installation.

## Two Disc Installation

Oracle Key Vault is now installed using two discs (created from two ISO files). For a fresh installation, Oracle Key Vault can be downloaded from Software Delivery Cloud. Note that this package cannot be used for an upgrade.

For more information, see Oracle Key Vault Installation and Configuration (page 3-1).

> **Note:**
>
> The upgrade package is installed using a single ISO file. For an upgrade, Oracle Key Vault can be downloaded from the Oracle Automated Release Updates (ARU) website.

# New Features in Bundle Patch 4

New in Release 12.2 BP 4 is support for Oracle Databse 11.2.0.4 (BP 9 and later) and 12.1.0.2 on Windows Server 2008 and 2012.

- Support for Oracle Database 11.2.0.4 (BP 9 and later) and 12.1.0.2 on Windows Server 2008 and 2012 (page xxxi)

## Support for Oracle Database 11.2.0.4 (BP 9 and later) and 12.1.0.2 on Windows Server 2008 and 2012

Oracle Key Vault supports Oracle Database 11.2.0.4 (BP 9 and later) and 12.1.0.2 on Windows Server 2008 and Windows Server 2012.

# New Features in Bundle Patch 3

The following are the new features and enhancements in Bundle Patch 3 (also referred to as Oracle Key Vault Release 12.2.0.3.0).

- Persistent Master Key Cache (page xxxi)
- Integration with nCipher Hardware Security Module (HSM) (page xxxii)
- Support for NIST CNSA Suite (page xxxii)
- SMTP Business Service for Email Notification (page xxxii)
- New REST API to Manage Virtual Wallets (page xxxii)

## Persistent Master Key Cache

The persistent master key cache feature enables databases to be operational when the Key Vault server is unavailable for any reason. The TDE master key is cached in the persistent cache in addition to the in-memory cache, so it works across database processes. It eliminates the need for databases to contact the Key Vault server for every new process, redo log switch, or database startup. Implemented in the Oracle Key Vault PKCS#11 library it additionally eliminates the need for database patching for previous database releases.

> **See Also:**
>
> - About the Persistent Master Key Cache (page 11-19)

## Integration with nCipher Hardware Security Module (HSM)

This release supports integration with the Hardware Security Module nCipher nShield Connect .

**See Also:**

- Vendor Specific Notes — nCipher

## Support for NIST CNSA Suite

Oracle Key Vault supports the Commercial National Security Algorithm Suite (CNSA), a list of strong encryption algorithms and key lengths that offer greater security and relevance into the future.

**See Also:**

- CNSA Suite Support

## SMTP Business Service for Email Notification

Supports Google and Office365 SMTP business service for email notifications and alerts. This feature is beneficial for customers who are dependent on external email services such as Google and Office365.

**See Also:**

- Configure Email Settings (page 12-9)

## New REST API to Manage Virtual Wallets

Release 12.2 BP 3 offers two new virtual wallet commands to better manage virtual wallets in Oracle Key Vault. Endpoint administrators can now retrieve the default wallet and all virtual wallets associated with an endpoint. This makes it easier to manage keys and credentials stored in virtual wallets.

**See Also:**

- get_default_wallet Command (page 10-29)
- get_wallets Command (page 10-30)

# New Features in Bundle Patch 2

The following are the new features and enhancements in Bundle Patch 2 (also referred to as Oracle Key Vault Release 12.2.0.2.0).

- Support for New Endpoint Platforms (page xxxiii)
- Support for Modern Hardware (page xxxiii)
- Preconfigured Management Reports (page xxxiv)
- Compliance (page xxxiv)
- Third-party CA support for the Management Console (page xxxiv)
- Automatic Email Notifications (page xxxv)
- Remote Monitoring via SNMP v3 (page xxxv)
- Automation of Endpoint Enrollment Using the RESTful Software Utility (page xxxv)
- Audit consolidation using Oracle Audit Vault and Database Firewall (page xxxvi)
- Diagnostics (page xxxvi)

## Support for New Endpoint Platforms

Support for two new platforms:

- AIX
- HP-UX (IA)

**See Also:**

"Supported Endpoint Platforms (page 3-3)"

## Support for Modern Hardware

The 12.2 Release is based on Oracle Linux Release 6 Update 6 operating system and Oracle Database 12.1.0.2 which are compatible with most modern hardware.

**See Also:**

"System Requirements (page 3-1)"

## Preconfigured Management Reports

Oracle Key Vault audits all endpoint and user activity and outputs the collected data in the form of preconfigured reports for endpoints, users, security objects and system. These reports provide a comprehensive and in-depth view of system activity that administrators can use for planning purposes.

**See Also:**

- "View Endpoint Reports (page 12-32)"
- "View User Reports (page 12-34)"
- "View Keys and Wallets Reports (page 12-35)"
- "View System Reports (page 12-35)"

## Compliance

A Security Technical Implementation Guide (STIG) is a methodology followed by the U.S. Department of Defense (DOD) to reduce the attack surface of computer systems and networks, thereby ensuring a lockdown of highly confidential information stored within the DOD network.

**See Also:**

"About Security Technical Implementation Guides (page C-1)"

## Third-party CA support for the Management Console

Key Vault administrators can use their own certificate or upload certificates signed by a third-party Certificate Authority (CA) to replace the self-signed certificate used by Key Vault's browser-based graphical user interface and users for administrative operations.

**See Also:**

"Third Party Certificates (page 13-4)"

## Automatic Email Notifications

This feature enables Key Vault system administrators to:

- Send the endpoint enrollment token directly to the endpoint administrator instead of using an out-of-band method.
- Reset a user's password in case of a security breach.
- Elect to be notified about system and status changes to respond quickly to security threats and risks.

**See Also:**

"Email Notification (page 12-8)"

## Remote Monitoring via SNMP v3

With SNMP enabled, system administrators can remotely monitor the Key Vault appliance for resource usage like memory, CPU utilization, processes, network bandwidth, and the key management server (KMIP daemon). The collected information can be used to monitor system performance and to recover quickly from any failures. Oracle Key Vault uses SNMP Version 3 for its user authentication and data encryption features.

**See Also:**

"Remote Monitoring Using SNMP (page 12-1)"

## Automation of Endpoint Enrollment Using the RESTful Software Utility

The RESTful Services utility is a scripting tool that enables you automate endpoint enrollment at scale. Automation reduces the multiple steps of enrolling and provisioning endpoints to a single command or script that you can execute at the command line. This is useful for administrators who might need to enroll and provision hundreds of endpoints simultaneously with minimum human intervention.

> **See Also:**
>
> "Endpoint Enrollment Automation with RESTful Services (page 10-1)"

## Audit consolidation using Oracle Audit Vault and Database Firewall

Key Vault can send its audit records to Audit Vault and Database Firewall (AVDF) allowing enterprise administrators to view Key Vault audit data from the AVDF management console. Enabling this feature frees up storage in Key Vault as the audit data no longer resides in Key Vault .

> **See Also:**
>
> "Audit Consolidation with Audit Vault and Database Firewall (page 12-30)"

## Diagnostics

Diagnostics may be gathered for the:

- Oracle Key Vault server and provided to Oracle support for further analysis.
- Endpoint with the diagnostics function in the endpoint software.

> **See Also:**
>
> - "Download System Diagnostics (page 12-19)"
> - "okvutil diagnostics Command (page 8-21)"

# New Features in Bundle Patch 1

The following are the new features and enhancements in Bundle Patch 1 (also referred to as Oracle Key Vault Release 12.2.0.1.0).

- Support for Oracle Cloud Database as a Service Endpoints (page xxxvi)
- Oracle Key Vault HSM Integration (page xxxvii)

## Support for Oracle Cloud Database as a Service Endpoints

An Oracle Key Vault on-premises appliance can manage Transparent Data Encryption (TDE) master keys for Oracle Cloud Database as a Service instances.

> **See Also:**
>
> "Oracle Cloud Database as a Service Endpoints (page 9-1)"

## Oracle Key Vault HSM Integration

Oracle Key Vault can use HSMs to generate and store a top-level encryption key, thereby acting as a Root of Trust (RoT) that protects encryption keys used by Key Vault. 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, lowering the risk of compromise. In addition HSMs can be used in FIPS 140-2 Level 3 mode which can help meet certain compliance requirements.

Note that an existing Oracle Key Vault deployment cannot be migrated to use an HSM as a Root of Trust. To use Oracle Key Vault with an HSM, a new Oracle Key Vault deployment is required.

> **See Also:**
>
> Oracle Key Vault HSM Integration Configuration Guide

# 1 Introduction to Oracle Key Vault

Oracle Key Vault is a full-stack, security-hardened software appliance built to centralize the management of security objects within the enterprise.

## 1.1 About Oracle Key Vault and Key Management

Oracle Key Vault is a robust, secure, and standards-compliant key management platform, where you can store, manage, and share your security objects like encryption keys, Oracle wallets, Java keystores (JKS), Java Cryptography Extension keystores (JCEKS), and credential files.

Oracle Key Vault will help you deploy encryption across your enterprise quickly and efficiently. Built upon Oracle Linux and Oracle Database technology, Oracle Key Vault's centralized, available and scalable security solution will help you overcome the biggest key-management challenges facing organizations today. With Key Vault you can retain, backup, and restore your security objects, prevent their accidental loss, and manage their lifecycle in a protected environment.

Oracle Key Vault is optimized for the Oracle Stack (Database, Middleware, Systems), and Advanced Security Transparent Data Encryption (TDE).

> **See Also:**
>
> - Support for OASIS Key Management Interoperability Protocol (KMIP) (page 1-7)
> - *Oracle Database Advanced Security Guide* for a complete discussion of Transparent Data Encryption (TDE)

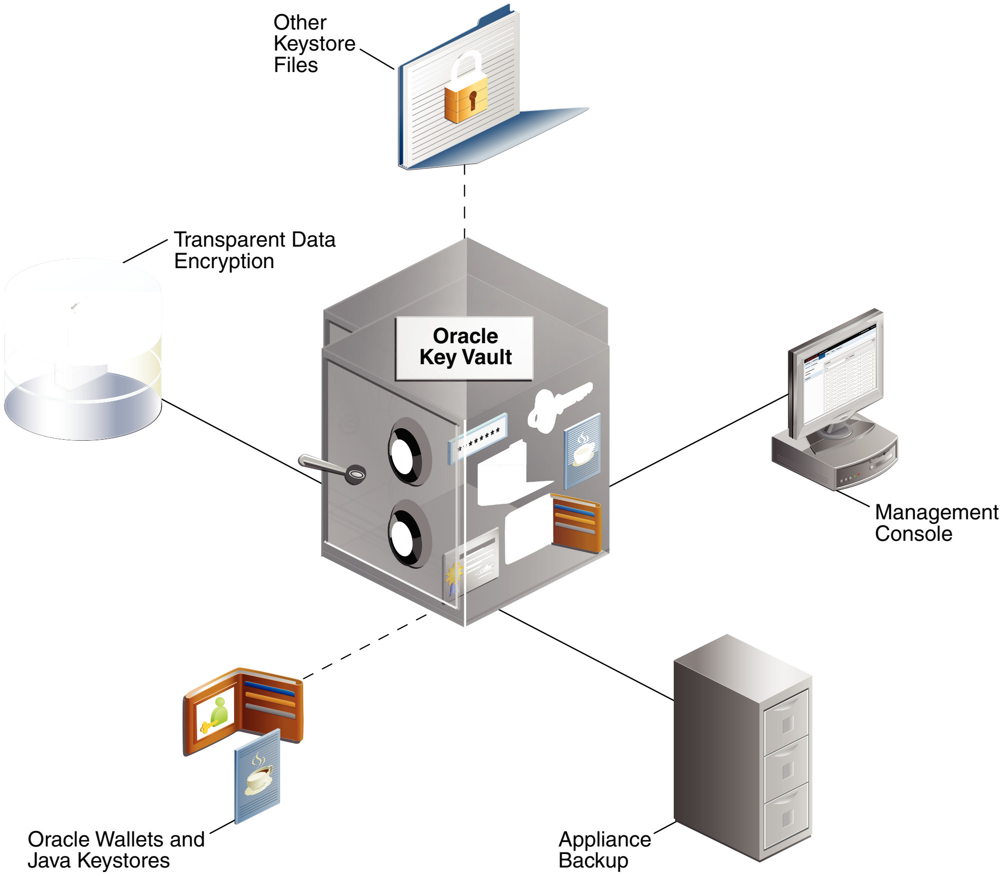
## 1.2 Benefits of Using Oracle Key Vault

Deploying Oracle Key Vault in your organization will help you:

- Manage the key lifecycle for endpoints, which includes key creation, rotation, deactivation, and removal.

- Prevent the loss of keys and wallets due to forgotten passwords or accidental deletion.
- Share keys securely across authorized endpoints in the enterprise.
- Enroll and provision endpoints easily using a single software package that contains all the necessary binaries, configuration files, and endpoint certificates for mutually authenticated connections between endpoints and Oracle Key Vault.
- Work with other Oracle products and features in addition to TDE like: Oracle Real Application Clusters (Oracle RAC), Oracle Active Data Guard, and Oracle GoldenGate. Oracle Key Vault facilitates the movement of encrypted data using Oracle Data Pump and transportable tablespaces, a key feature of Oracle Database.

**Figure 1-1 The Centralized Key-Management Platform of Oracle Key Vault**



This figure illustrates a typical deployment of Oracle Key Vault from a location central to the enterprise.

It interacts with the following components:

- **Transparent Data Encryption** refers to Oracle databases protected with TDE.
- **Oracle wallets and Java keystores** are containers for security objects that you upload and download between Oracle Key Vault and endpoints.

- **Other Keystore Files** are security objects like certificates, and credential files like Kerberos key tab files, SSH key files, and server password files, that you upload to Oracle Key Vault from endpoints.
- **Management Console** refers to the Oracle Key Vault graphical user interface, where you can log in to manage your security objects and administer the Key Vault system.
- **Appliance Backup** refers to a backup device, where security objects in Oracle Key Vault can be backed up on-demand or on-schedule.

## 1.3 Who Should Use Oracle Key Vault?

Oracle Key Vault is meant primarily for users who are responsible for deploying, maintaining, and managing security within the enterprise. These users can be database, system, or security administrators, indeed any information security personnel responsible for protecting enterprise data in database servers, application servers, operating systems, and other information systems. They manage encryption keys, Oracle wallets, Java keystores, and other security objects on a regular basis.

Other users can include personnel responsible for Oracle databases, and servers that interact with Oracle database, because Oracle Key Vault provides inherently tighter integration with Oracle database. These systems often deploy encryption on a large scale and may have a need to simplify key and wallet management.

## 1.4 Major Features of Oracle Key Vault

Oracle Key Vault enhances security in key management with a set of robust features, such as centralizing the storage and management of security objects.

### 1.4.1 Centralized Storage and Management of Security Objects

You can store and manage the following types of security objects using Key Vault:

- TDE master keys

For Oracle databases that use Transparent Data Encryption (TDE), Oracle Key Vault manages TDE master keys over a direct network connection as an alternative to using local wallet files. The keys stored in Oracle Key Vault can be shared across databases according to endpoint access control settings. This method of sharing keys without local wallet copies is useful when TDE is running on database clusters such as Oracle RAC, Oracle Active Data Guard, or Oracle GoldenGate. You can easily migrate master keys from Oracle wallets to Oracle Key Vault. Direct connections between TDE and Oracle Key Vault are supported for Oracle Database 11*g* Release 2 and Oracle Database 12*c*.

- Oracle wallets and Java keystores

  Oracle wallets and Java keystores are often widely distributed across servers and server clusters, with backup and distribution of these files performed manually. Oracle Key Vault itemizes and stores contents of these files in a master repository, yet allows server endpoints to continue operating with their local copies, while disconnected from Oracle Key Vault. After you have archived wallets and keystores, you can recover them to their servers if their local copies are mistakenly deleted or their passwords are forgotten. Oracle Key Vault streamlines the sharing of wallets across database clusters such as Oracle RAC, Oracle Active Data Guard, and Oracle GoldenGate. Sharing of wallets also facilitates the movement of encrypted data using Oracle Data Pump and the transportable tablespaces feature of Oracle Database. You can use Oracle Key Vault with Oracle wallets from all supported releases of Oracle middleware products and Oracle Database.

- Credential files

  Applications store keys, passwords, and other types of sensitive information in credential files, that are often widely distributed without appropriate protective mechanisms. Secure Shell (SSH) key files and Kerberos keytabs are examples of credential files. Oracle Key Vault backs up credential files for long-term retention and recovery, audits access to them, and shares them across trusted server endpoints.

- Certificate files

  X.509 certificate files (common file extensions include `.pem`, `.cer`, `.crt`, `.der`, `.p12`) used to authenticate and validate user identities, and encrypt data on communication channels may also be stored, shared, and managed in Oracle Key Vault.

## 1.4.2 Management of Key Lifecycle

The management of the key lifecycle is critical for maintaining security and regulatory compliance, and consists of four main functions: creation, backup, rotation, and expiration.

Oracle Key Vault provides mechanisms for facilitating periodic key rotations, backup, and recovery, which ensure that customers stay in regulatory compliance, unlike most systems that create keys and passwords, including TDE. You can create policies to track the key lifecycle, and configure Oracle Key Vault to report key lifecycle changes as they happen. In this way you will know when keys are due to expire, and can ensure that they are properly rotated and backed up.

Key lifecycle tracking is very important to maintain compliance with industry and governmental standards, such as the Payment Card Industry Data Security Standard (PCI DSS) which deal with highly sensitive data, and therefore have stringent requirements regarding the maximum lifetime of encryption keys and passwords.

## 1.4.3 Reporting and Alerts

Oracle Key Vault provides a comprehensive, and in-depth view of system activity in the form of reports and alerts.

- Reports

  Oracle Key Vault provides a set of audit and management reports with detailed statistics on system, user, and endpoint activity, certificate, key and password expiry, entitlement and metadata of security objects. Audit reports capture all user and endpoint actions, the objects of the actions, and their final result.
- Alerts

  You can configure the types of alerts you want to receive. These include alerts for the expiration of keys, endpoint certificates, and user passwords, disk utilization, system backup, and high availability events. You can choose to send alerts to syslog to allow for external monitoring.

## 1.4.4 Separation of Duties for Oracle Key Vault Users

Oracle Key Vault provides for a separation of duties in the form of three user roles: Key Administrator, System Administrator, and Audit Manager.

Each user role possesses privileges for a type of task and may be assigned singly to one user (for a strict separation of duties) or combined so a single user performs multiple user roles according to the needs of the organization. The user who is responsible for uploading and downloading security objects between Oracle Key Vault and the endpoint is referred to as the endpoint administrator. Only endpoint administrators can directly access security objects provided they have been granted access and only through installing the endpoint software. Security objects cannot be retrieved via the web-based GUI.

> **See Also:**
>
> "Overview of Administrative Roles (page 2-9)"

## 1.4.5 Support for a High Availability Environment

To ensure that Oracle Key Vault can always access security objects, Oracle Key Vault can be deployed in a highly available configuration. This configuration also supports disaster recovery scenarios.

You can deploy two Oracle Key Vault appliances in a high availability configuration. The primary appliance services the requests that come from endpoints. If the primary appliance fails, then the standby appliance takes over after a configurable preset delay. This configurable delay ensures that the standby server does not take over prematurely in case of short communication gaps.

Oracle Key Vault uses Oracle Data Guard to synchronize data between the primary and standby nodes in a high availability deployment.

**See Also:**

"About High Availability for Oracle Key Vault (page 4-1)"

## 1.4.6 Persistent Master Key Cache

The persistent master key cache feature enables databases to be operational when the Oracle Key Vault server is unavailable for any reason. The TDE master key is cached in the persistent master key cache in addition to the in-memory cache, to make the master key available across database processes. It eliminates the need for databases to contact the Oracle Key Vault server for every new process, redo log switch, or database startup.

**See Also:**

"Persistent Master Key Cache (page 11-19)"

## 1.4.7 Backup and Restore Functionality for Security Objects

Oracle Key Vault enables you to back up all security objects including keys, certificates, and passwords. It encrypts backups for better protection of the sensitive keys and security objects and supports storing them securely at a remote destination.

This feature prevents loss of your sensitive data in the case of appliance failure, because you can restore a new Oracle Key Vault appliance to a previous state from a backup.

Oracle Key Vault can transfer backup files to any remote location that implements the Secure Copy Protocol (SCP).

Users with the System Administrator role can perform the following backup and restore tasks in Oracle Key Vault:

- Create, delete, and modify remote backup locations.
- Set up, modify, or disable the current backup schedule.
- Initiate an immediate one-time backup.
- Schedule a future one-time backup.

Oracle Key Vault performs hot backup which means that the system is not interrupted while the backup is being created.

**See Also:**

"Backup and Restore Oracle Key Vault Data (page 4-17)"

## 1.4.8 Automation of Endpoint Enrollment Using Protected RESTful Services

The RESTful Services utility is an automation tool that enables you to quickly enroll and provision endpoints and endpoint groups at scale. Automation reduces the multiple steps of enrolling and provisioning endpoints to a single function call at the command line. This is useful for administrators of large distributed enterprise systems, who might need to enroll and provision many hundreds of endpoints simultaneously using the protective security measures of RESTful services.

**See Also:**

See "Endpoint Enrollment Automation with RESTful Services (page 10-1)"

## 1.4.9 Support for OASIS Key Management Interoperability Protocol (KMIP)

OASIS Key Management Interoperability Protocol (KMIP) standardizes key management operations between key management servers and endpoints provided by different vendors.

Oracle Key Vault implements the following OASIS KMIP Version 1.1 profiles:

- **Basic Discover Versions Server Profile:** Provides the server version to endpoints.
- **Basic Baseline Server KMIP Profile:** Provides core functionality to retrieve objects from the server.
- **Basic Secret Data Server KMIP Profile:** Provides endpoints the ability to create, store, and retrieve secret data (typically passwords) on the server.
- **Basic Symmetric Key Store and Server KMIP Profile:** Provides endpoints the ability to store and retrieve symmetric encryption keys on the server.
- **Basic Symmetric Key Foundry and Server KMIP Profile:** Provides endpoints the ability to create new symmetric encryption keys on the server.

**See Also:**

`http://docs.oasis-open.org/kmip/spec/v1.1/os/kmip-spec-v1.1-os.html` for information about the OASIS KMIP specification

## 1.4.10 Database Version and Platform Support

Oracle Key Vault supports Oracle Database versions 11g Release 2 and later on Oracle Linux x86-64, Solaris, AIX, and HP-UX (IA) as endpoints without patching. Oracle Key Vault also supports Oracle Database versions 11g Release 2 (BP 9 and later) and 12c Release 1 (12.1.0.2) on Windows Server 2008 and Windows Server 2012.

**See Also:**

"Oracle Key Vault Installation Requirements (page 3-1)"

## 1.4.11 Integration with External Audit and Monitoring Services

Oracle Key Vault supports integration with Oracle Audit Vault and Database Firewall for central storage of audit records generated. Oracle Key Vault also supports use of SNMP v3 to monitor the health and availability of the system.

## 1.4.12 MySQL Integration with Oracle Key Vault

Oracle Key Vault can manage MySQL TDE encryption keys.

**Note:**

MySQL Windows databases are not supported.

**Related Topics**

- MySQL Integration with Oracle Key Vault (page 11-37)

# 1.5 Oracle Key Vault Interfaces

Oracle Key Vault provides two interfaces: a management console and an endpoint command-line utility for uploading and retrieving security objects.

- Oracle Key Vault management console

  The Oracle Key Vault management console is a browser-based graphical user interface, that enables Oracle Key Vault administrators to manage security objects, wallets, endpoints, and users, and to configure system settings like high availability, backup, and recovery.

- Oracle Key Vault `okvutil` endpoint utility

  The `okvutil` command-line utility enables endpoint administrators to upload and download security objects between Oracle Key Vault and endpoints. The `okvutil` utility communicates with Oracle Key Vault over a mutually authenticated secure connection.

> **See Also:**
>
> - "Logging In to the Oracle Key Vault Management Console (page 3-14)"
> - "Oracle Key Vault okvutil Endpoint Utility Reference (page 8-12)"

## 1.6 Overview of a Successful Oracle Key Vault Deployment

You can use the following steps as a guideline to deploying Oracle Key Vault successfully within your organization:

1. Understand key concepts described in **Chapter 2**, **Concepts**.
2. Install and configure Oracle Key Vault as outlined in **Chapter 3**, **Oracle Key Vault Installation and Configuration**.
3. Create a high availability configuration by adding a second Key Vault appliance. Enable High Availability Read-Only Restricted mode to ensure operational continuity of the endpoints. This is described in **Chapter 4**, **High Availability, Backup and Restore**.

   > **Note:**
   >
   > You must have a separate license for each Oracle Key Vault server installation in a high availability environment.

4. Create users to manage the day-to-day tasks for Oracle Key Vault as described in **Chapter 5**, **Managing Oracle Key Vault Users**.
5. Upload or add virtual wallets to Oracle Key Vault described in **Chapter 6**, **Managing Oracle Key Vault Virtual Wallets and Security Objects**.
6. Add endpoints so that they can use Oracle Key Vault to store and manage their security objects described in **Chapter 7**, **Managing Oracle Key Vault Endpoints**.
7. Add endpoints in the cloud described in **Chapter 9**, **Oracle Cloud Database as a Service Endpoints**.
8. Enroll endpoints so that you can upload or download security objects between the endpoints and Oracle Key Vault described in **Chapter 8**, **Enrolling Endpoints for Oracle Key Vault**.
9. Read about automating endpoint enrollment and provisioning for large-scale enterprise deployments in **Chapter 10**, **Endpoint Enrollment Automation with RESTful Services**.

10. Read about typical use cases described in **Chapter 11**, **Oracle Key Vault Use Case Scenarios**.
11. Learn how to perform periodic maintenance tasks like administering and monitoring the system described in **Chapter 12**, **General Oracle Key Vault Management**.

# 2 Oracle Key Vault Concepts

You can deploy Oracle Key Vault successfully with a conceptual understanding of the deployment architecture, use cases, access control, administrative roles, and endpoints.

## 2.1 Overview of Oracle Key Vault Concepts

Oracle Key Vault endpoints are computer systems like database servers, application servers, and other information systems, where keys and credentials are used to access encrypted data and other systems. These systems have a need to store and manage their encryption keys efficiently, so that data is secure, accessible, and available to meet the day-to-day activities of the enterprise. Endpoints with pre-existing keys, or the capability to generate them, can use Oracle Key Vault as secure, external, long-term storage.

Endpoints must be registered and enrolled to communicate with Oracle Key Vault. Enrolled endpoints can upload their keys, share them with other endpoints, and download them to access their data. Oracle Key Vault keeps track of all enrolled endpoints.

Security objects can be grouped into a virtual wallet in Oracle Key Vault. The main purpose of a virtual wallet is to group related security objects so that they can be collectively shared with peers in an easy way. Any user can create a virtual wallet, add keys to the empty wallet, and then grant other users, endpoints, user and endpoints groups various levels of access to the wallet. A user must himself have access to security objects, before he can grant access on those same security objects to other users. The access level they grant can be equal to or less that their own. This flexibility is designed to meet the multiple and varying needs of any organization.

The owner of a security object is the entity who created the security object with full read, write, and modify access to the security object. The owner can add the security object to any number of wallets to be shared with other users at various access levels.

When an endpoint is registered with Key Vault, you can specify a default wallet for the endpoint. The main purpose of a default wallet is to ensure that keys have a virtual wallet to upload to when none are specified. All keys generated and uploaded by the endpoint will be automatically added to the default wallet associated with the endpoint.

Multiple endpoints can have a common default wallet. The contents of this default wallet are shared across all the endpoints, without the need to put these endpoints into an endpoint group. This feature enables multiple endpoints to create keys, or upload an Oracle wallet directly to the default wallet.

Oracle Key Vault audits all actions performed by users and endpoints.

## 2.2 Oracle Key Vault Deployment Architecture

Oracle Key Vault is deployed on a server, with connections to endpoints that have security objects to store and manage.

Oracle Key Vault is packaged as a software appliance. It is hardened for security according to operating system and database hardening best practices. Unnecessary packages and software have been removed, and unused services and ports are disabled. It is preconfigured with an operating system, database, and the Oracle Key Vault application itself, so that you do not have to install and configure individual components.

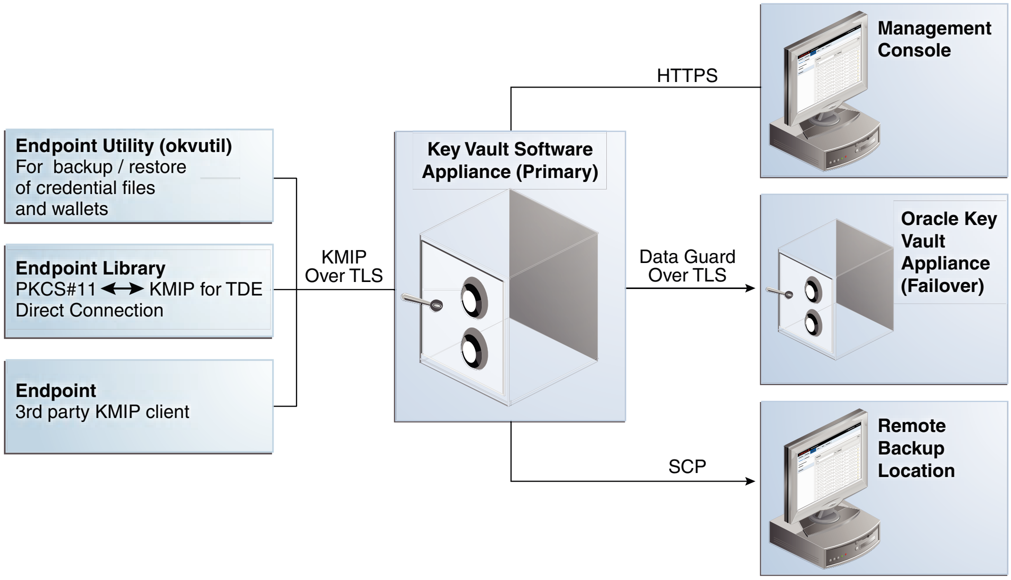
Endpoints communicate with Oracle Key Vault over a mutually authenticated Transport Layer Security (TLS) connection using the OASIS Key Management Interoperability Protocol (KMIP).

Administrators log in to the browser-based GUI of the Key Vault management console with a user ID and password.

The Oracle Key Vault high availability configuration defines one primary appliance and one standby appliance. The primary appliance is active and services requests from endpoints. The standby appliance takes over as primary, if the primary fails to communicate with the standby for a time exceeding a configured time threshold. Communication related to data replication between the primary and standby appliances is mutually authenticated TLS.

The following figure illustrates the deployment architecture of Oracle Key Vault.

**Figure 2-1 Oracle Key Vault Deployment Architecture**



## 2.3 Oracle Key Vault Use Cases

The most typical use cases for Oracle Key Vault are centralized storage and management of security objects.

- Centralized Storage of Oracle Wallet Files and Java Keystores (page 2-3)
- Centralized Management of TDE Master Keys Using Online Master Key (page 2-4)
- Storage of Credential Files (page 2-6)

### 2.3.1 Centralized Storage of Oracle Wallet Files and Java Keystores

You can store all your security objects like Oracle wallets and Java keystores centrally in Oracle Key Vault, and manage them with automatic mechanisms that Key Vault provides for tracking, backup, and recovery. This will help you overcome many operational and security challenges posed by the manual tracking and management of security objects dispersed widely across multiple servers.

Oracle Key Vault stores copies of Oracle wallet files, Java keystores, and other security objects in a centralized location for long-term retention and recovery. These security objects can later be downloaded to a new wallet or keystore file and shared with trusted server peer endpoints.

The Oracle Key Vault endpoint software can read the format of Oracle wallet files and Java keystores to store their contents at the granularity of individual security objects. You can upload both password-protected and auto-login wallets, and then download the wallet contents to a new wallet of either type. This enables users to manage security objects individually, and add them to virtual wallets for sharing.

Oracle Key Vault can individually store and manage the security objects contained in:

- Oracle wallet files

  Symmetric keys used for encryption (including TDE master keys), passwords (Secure External Password Store), and X.509 certificates (network encryption).

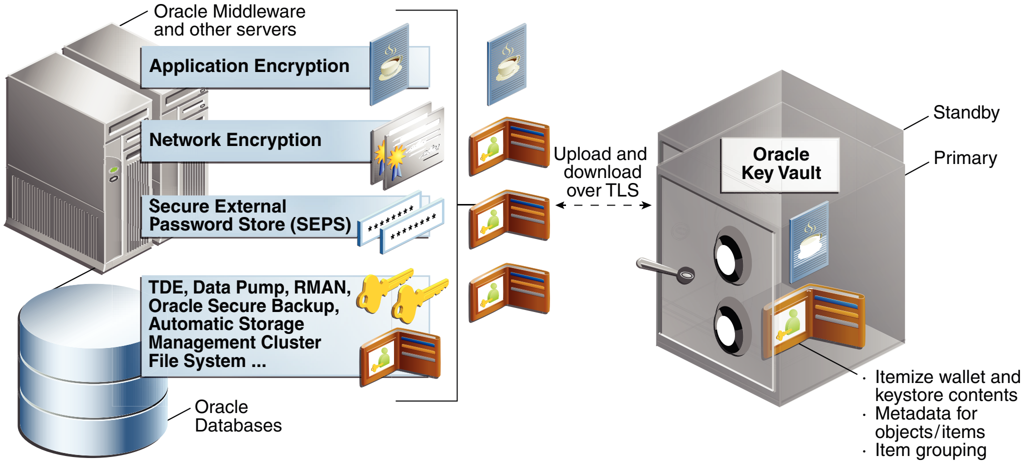
  Oracle wallet files from all supported releases of Oracle Database are supported.
- Java keystores

  Symmetric keys, asymmetric keys such as private keys, and X.509 certificates.

  Oracle Key Vault supports both JKS and JCEKS types of Java keystores.

The following figure illustrates the centralized storage of Oracle wallet files and Java keystores.

**Figure 2-2 Centralized Storage of Oracle Wallet Files and Java Keystores**


Oracle Middleware and other servers
Application Encryption
Network Encryption
Secure External Password Store (SEPS)
TDE, Data Pump, RMAN, Oracle Secure Backup, Automatic Storage Management Cluster File System ...
Oracle Databases
Upload and download over TLS
Oracle Key Vault
Standby
Primary
· Itemize wallet and keystore contents
· Metadata for objects/items
· Item grouping

**See Also:**

Managing Secure External Password Store

## 2.3.2 Centralized Management of TDE Master Keys Using Online Master Key

For Oracle databases using TDE, Oracle Key Vault provides centralized management of TDE master keys over a direct network connection, known as Online Master Key. In this use case TDE generates the master key and stores it in Oracle Key Vault.

Note that Online Master Key replaces the term TDE Direct Connection.

This is a convenient alternative to copying local wallet files to multiple endpoints manually. Sharing TDE master keys rather than maintaining local wallet copies is especially useful when TDE is running on database clusters such as Oracle RAC. The following comparison illustrates the difference:

- Local wallet copy

  When a key rotation operation is performed on the master key in the primary node of an Oracle RAC, the local wallet copy is updated and must then be manually copied to the other nodes to propagate the new TDE master key.

- Shared TDE key in a virtual wallet in Key Vault

  After a key rotation operation the new TDE master key is immediately shared with other nodes in the cluster. There is no need to copy the wallet manually to the other nodes.

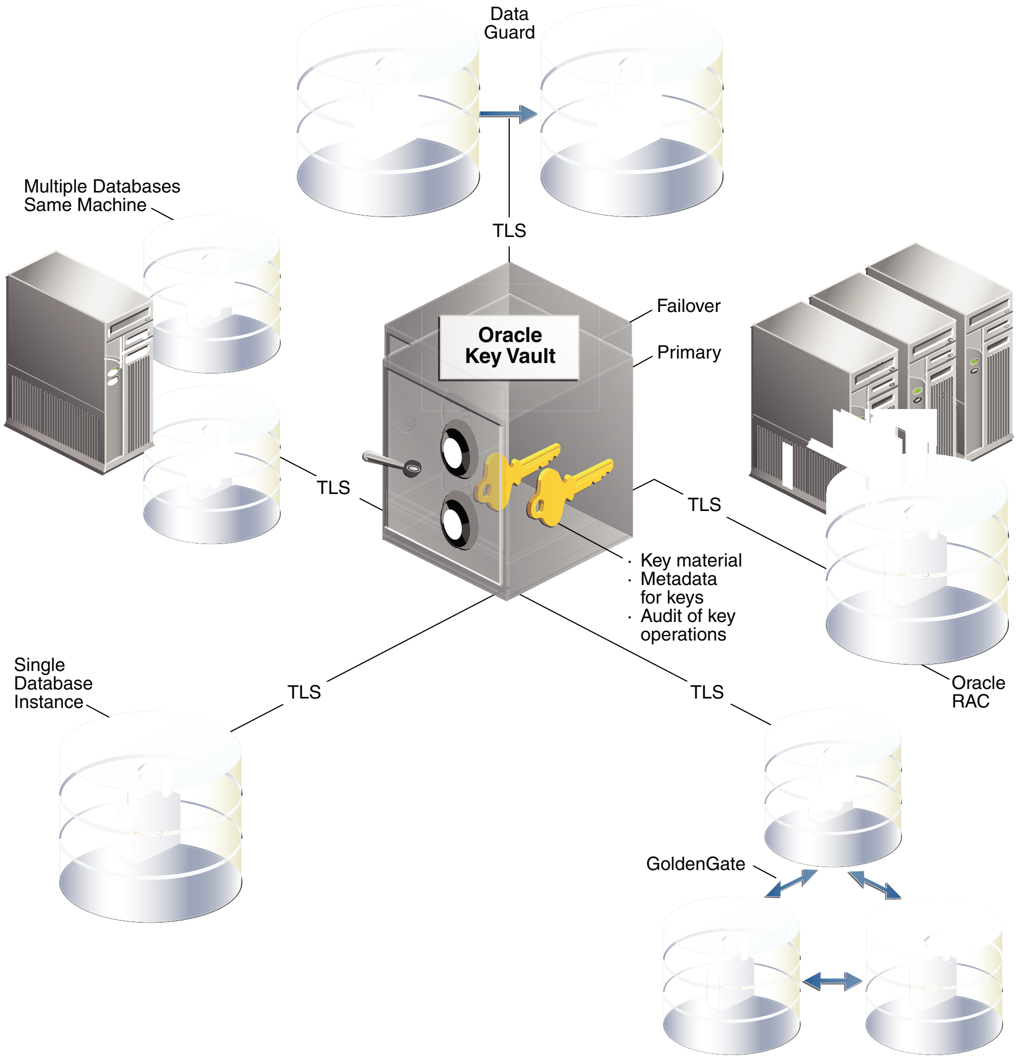
Centralized management facilitates the copying of encrypted data between databases using Oracle Data Pump Export, Import, or the transportable tablespaces feature of Oracle Database when master keys are stored in the wallet:

- In non-centralized management the wallet must be manually copied from source to target databases.
- In centralized management these master keys are easily shared by placing them in a virtual wallet in Oracle Key Vault, and then granting each endpoint access to the virtual wallet.

Online Master Keys between TDE and Oracle Key Vault are supported on Oracle Database 11*g* Release 2 and Oracle Database 12*c*.

The following figure illustrates the centralized management of Online Master Keys (formerly known as TDE Direct Connect).

**Figure 2-3 Centralized Management of Online Master Keys**



**See Also:**

Using an Online Master Key with Oracle Key Vault (page 11-10)

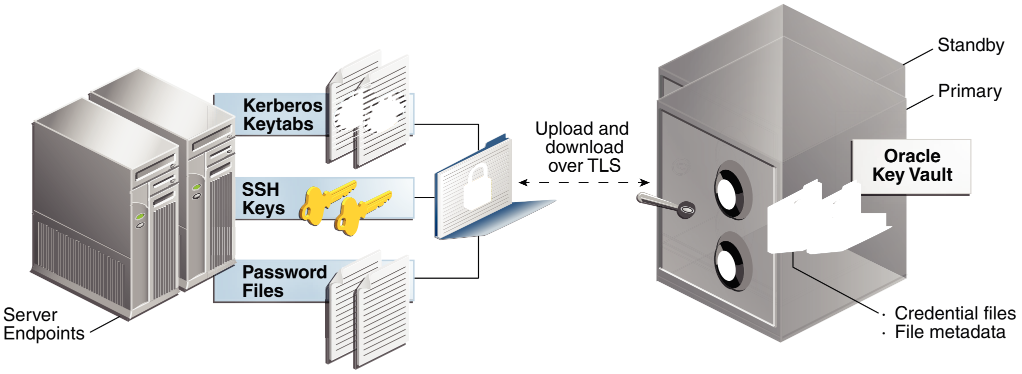
### 2.3.3 Storage of Credential Files

Oracle Key Vault can back up credential files other than Oracle wallets and Java keystores for long-term retention and recovery. Oracle Key Vault does not interpret the actual content of a credential file, it stores the entire file as an opaque object and provides a handle to the endpoint for retrieval at a later time. A credential file contains security objects such as keys, passwords, SSH keys, Kerberos keytabs, and X.509 certificates.

You can directly upload credential files into Oracle Key Vault, consolidate them in a central repository, and share them across endpoints in a trusted group. Key Vault backs up all credential files for continued and secure access at any time. Access control to credential files is managed by Key Vault administrators.

The following figure illustrates how credential files are backed up in Oracle Key Vault.

**Figure 2-4    Backing Up Credential Files**



**See Also:**

"Uploading and Downloading Credential Files (page 11-7)"

## 2.4 Access Control Configuration

Oracle Key Vault allows you control access to security objects at various access levels and time intervals. Any user may be granted access to security objects in Key Vault at a level appropriate to their function in the organization.

## 2.4.1 About Access Control Configuration

You can grant users access to security objects in Oracle Key Vault at a level appropriate to their function in the organization.

Access control can be set on security objects individually, or collectively when they are grouped into a virtual wallet. Oracle Key Vault uses the mechanism of a virtual wallet to share a set of security objects with others. A virtual wallet is a container for security objects like public and private encryption keys, TDE master encryption keys, passwords, credentials, and certificates in Oracle Key Vault. You can set access levels on a virtual wallet for an endpoint or user, thus granting simultaneous access to all the security objects contained within the virtual wallet.

In addition to being able to grant access to individual users or endpoints individually, access can be granted collectively by using endpoint groups or user groups. If multiple endpoints need access to a virtual wallet, it is simpler to add these endpoints to an endpoint group, and grant the endpoint group access to the virtual wallet. The alternative is to grant access to each endpoint individually. When you grant an endpoint group access to a virtual wallet, you are granting access to all the member endpoints in the endpoint group.

## 2.4.2 Access Grants

You can grant access to virtual wallets directly or indirectly:

- Grant users and endpoints access directly.
- Grant users and endpoints groups access indirectly via group membership. When you grant a user or endpoint group access, you are granting all members of the group access. This is a convenient alternative to individually granting each user or endpoint access.

**Note:**

You can set access mappings on a virtual wallet in two ways:

- From the **user**, **endpoint**, or their respective groups. You can start at the user, endpoint, or respective group and add the wallet and access mappings for this user.
- From the **virtual wallet**. You can start from the virtual wallet and add users, endpoints, and their respective groups that can access it at access mappings that you set.

**See Also:**

"Grant Access to Endpoint Groups, Endpoints, User Groups, and Users (page 6-5)"

"Grant an Endpoint Access to a Virtual Wallet (page 7-11)"

"Grant a User Group Access to a Virtual Wallet (page 5-16)"

### 2.4.3 Access Control Options

You can control access to virtual wallets by setting different access levels for users, user groups, endpoints, and endpoint groups corresponding to their role and function in the organization.

There are three access levels:

- **Read Only**: Grants read privileges on the security object.
- **Read and Modify**: Grants read and modify privileges on the security object.
- **Manage Wallet**. Grants the following privileges:
  - Add or remove security objects from the virtual wallet. The user must have **Read and Modify** access on the security object to be added to the virtual wallet.
  - Grant others access to the wallet.
  - Modify wallet settings, like its description.
  - Delete the wallet.

## 2.5 Administrative Roles within Oracle Key Vault

Oracle Key Vault provides three administrative roles: system administrator, key administrator, and audit manager, that may be combined in any way to meet enterprise needs.

- About Administrative Roles in Oracle Key Vault (page 2-8)
- Separation of Duties (page 2-9)
- Overview of Administrative Roles (page 2-9)
- Emergency System Recovery Process (page 2-10)

### 2.5.1 About Administrative Roles in Oracle Key Vault

The three administrative roles are designed to be flexible to support various organizational needs and structures. The roles may be used to separate duties, or combined in any way, or not used at all. Users without a specific administrative role may be responsible for a specific area, like a virtual wallet. These users can be granted access to the virtual wallet appropriate to their function within the organization, thus limiting access to security objects to just those users who need it. In this way access to security objects is controlled, yet flexible to meet the changing needs of the enterprise.

## 2.5.2 Separation of Duties

Oracle Key Vault users can assign the three administrative roles by function, so there a a clear separation of duties between the system and key administrator, and the audit manager.

You can achieve a separation of duties in two ways:

- Grant an administrative user (defined as a user with one of the roles of System Administrator, Key Administrator, or Audit Manager) privileges to one functional area: system, key, or audit management. Create three separate administrative users for each of the functional areas.
- Grant a user access to one object or function independently of all others using a fine-grained division of access control and operational privileges based on the responsibility of the user. These users need not have any of the administrative roles to perform their function.

You should ensure that every user who interacts with Oracle Key Vault has their own user account and password.

## 2.5.3 Overview of Administrative Roles

Oracle Key Vault has three administrative roles: Key Administrator, System Administrator, and Audit Manager for the administrative users who perform key, system, and audit management functions respectively.

Administrative users can grant their roles to other users, but not others. If one administrative user is performing two administrative functions, that user will have two roles in Key Vault. This user can grant other users one or both the roles as needed. For example if a user has both the System Administrator and Key Administrator role, he can grant another user both those roles or just one, depending on the needs of the organization.

In a strict separation of duties, a user with one administrative role must perform one part of the operation, and a user with a different administrative role performs a different but related part of the operation: for example, only System Administrators can enroll endpoints and only Key Administrators can create endpoint groups.

One of the post-installation tasks is to create three administrative users for the three roles. In a situation where there is no administrative user present, you can recover the system with the recovery passphrase. You will use the recovery passphrase to repeat the post-installation configuration, and create three administrative users in order to ensure continued operation and management of Key Vault.

When you use the management console, your access to the various tabs, menus, and actions depends on your role and the objects that you have access to. The following list outlines the duties of each administrative user:

- The Oracle Key Vault System Administrator
  - Creates and manages users
  - Adds and manages endpoints
  - Sets up high availability

  - Configures alerts and key rotation reminders
  - Schedules backups
  - Starts and stops Oracle Key Vault
  - Configures SMTP server settings for email notification
  - Configures SNMP for remote monitoring
  - Enables automated endpoint enrollment via RESTful Services
  - Enables audit consolidation with Audit Vault Database Firewall
  - Creates SSH tunnels for Oracle Cloud Database as a Service endpoints
  - Grants the System Administrator role to other users
- The Oracle Key Vault Key Administrator
  - Manages the lifecycle of security objects
  - Has full access on all virtual wallets and security objects
  - Controls access to virtual wallets for users, endpoints, user and endpoint groups
  - Creates and manages user groups
  - Creates and manages endpoint groups
  - Grants the Key Administrator role to other users
- The Oracle Key Vault Audit Manager
  - Manages the audit trail as the only user who has privileges to export or delete Oracle Key Vault audit records
  - Has Read access on all security objects
  - Grants the Audit Manager role to other users

## 2.5.4 Emergency System Recovery Process

During installation you will be required to create a special recovery passphrase that Oracle Key Vault uses to recover from emergency situations. These situations can arise due to administrative users not being immediately available, or something more commonplace like forgotten passwords.

The recovery passphrase is needed in the following situations:

1. If there are no administrative users available to log into Key Vault, you can use the recovery passphrase to repeat the post-installation tasks and create new administrative users for system, key, and audit management.
2. If you want to restore Oracle Key Vault from a previous backup you will need the recovery passphrase associated with that backup.
3. If you want to reset the recovery passphrase periodically.

For these reasons it is quite important to store the recovery passphrase in a a safe and accessible place and keep track of older recovery passphrases. The only way to recover from a lost recovery passphrase is to reinstall Key Vault.

**See Also:**

Recovery passphrase used in "System Recovery (page 12-16)"

Maintain the correct recovery passphrase with a backup in "Restoring Oracle Key Vault Data (page 4-27)"

Create or change the recovery passphrase in Performing Post-Installation Tasks (page 3-10)

## 2.6 Endpoint Administrators

Endpoint administrators own and manage endpoints. They are typically system, security, or database administrators, but they can be any personnel charged with deploying, managing and maintaining security within an enterprise. Endpoint administrators are responsible for enrolling endpoints.

The endpoint administrator for an Oracle database endpoint is the database administrator, who is responsible for managing the database.

# 3 Oracle Key Vault Installation and Configuration

Oracle Key Vault is a software appliance that is delivered as an ISO image. Key Vault should be installed onto its own dedicated physical server.

The software appliance consists of a pre-configured operating system, an Oracle database, and the Oracle Key Vault application.

You can install the Oracle Key Vault appliance by meeting specific system requirements and completing a set of post-installation tasks.

- Oracle Key Vault Installation Requirements (page 3-1)
- Installing and Configuring Oracle Key Vault 12.2.0.5.0 and Later (page 3-4)
- Logging In to the Oracle Key Vault Management Console (page 3-14)
- Overview of the Management Console (page 3-15)
- Performing Actions and Searches (page 3-15)

## 3.1 Oracle Key Vault Installation Requirements

The Oracle Key Vault installation requirements cover system requirements like CPU, RAM, disk space, network interfaces, and supported endpoint platforms.

- System Requirements (page 3-1)
- Network Ports (page 3-2)
- Supported Endpoint Platforms (page 3-3)
- Endpoint Database Requirements (page 3-3)

### 3.1.1 System Requirements

The Oracle Key Vault installation removes existing software on a server.

Deployment on virtual machines is not recommended for production systems. However, virtual machines are useful for testing and proof of concept purposes.

The minimum hardware requirements for deploying the Oracle Key Vault software appliance are:

- **CPU:** Minimum: x86–64 2 cores, Recommended: 8–16 cores with cryptographic acceleration support (Intel AESNI)
- **Memory:** Minimum 8 GB of RAM, Recommended: 32–64 GB
- **Disk:** Minimum 500 GB, Recommended: 1 TB
- **Network interface:** One network interface

- **Hardware Compatibility:** Refer to the hardware compatibility list (HCL) for Oracle Linux Release 6 Update 9 at the link in the **See Also** section.

  > **Note:**
  >
  > Ensure that the hardware supports booting in legacy BIOS mode. Hardware that supports Unified Extensible Firmware Interface (UEFI) only is currently unable to recognize the Oracle Key Vault ISO image. However, be aware that Oracle Key Vault does not support the QLogic QL4* family of network cards.

- **RESTful Services Client:** If RESTful Services are enabled, then each endpoint that connects to the Oracle Key Vault management console must have at least Java 1.7.0_21 installed.

  The REST API requires the cURL utility. Ensure that cURL 7.43 or higher is installed on the endpoint system before using the REST API to provision endpoints.

> **Note:**
>
> For deployment with a large number of endpoints the hardware requirement may need to scale to meet the workload.

> **See Also:**
>
> The hardware certification list for Oracle Linux and Oracle VM may be found at the Oracle Linux website at:
>
> `http://linux.oracle.com/pls/apex/f?p=117:1`
>
> You can find the supported hardware by filtering results through **All Operating Systems** and choosing **Oracle Linux 6.9**.

## 3.1.2 Network Ports

Oracle Key Vault and its endpoints use a set of special ports for communication. Network administrators must ensure that these ports are open in the network firewall.

Table 3-1 lists the required network ports for Oracle Key Vault:

**Table 3-1 Ports Required for Oracle Key Vault**

| Port Number | Protocol | Descriptions |
|---|---|---|
| 22 | SSH/SCP Port | Used by Oracle Key Vault administrators and support personnel to remotely administer Oracle Key Vault. |

**Table 3-1 (Cont.) Ports Required for Oracle Key Vault**

| Port Number | Protocol | Descriptions |
|---|---|---|
| 161 | SNMP Port | Used by monitoring software to poll Oracle Key Vault for system information. |
| 443 | HTTPS Port | Used by web clients such as browsers and RESTful Services to communicate with Oracle Key Vault. |
| 1522 | Database TCPS Listener Port | Listener port used in a high availability configuration by Oracle Data Guard to communicate between the primary and standby server. |
| 7443 | Database TCPS Listener Port | Listener port used in a high availability configuration to run OS commands like synchronizing wallets and configuration files via HTTPS. |
| 5696 | KMIP Port | Used by Oracle Key Vault endpoints and third party KMIP clients to communicate with the Oracle Key Vault KMIP Server. |

## 3.1.3 Supported Endpoint Platforms

Oracle supports both 32-bit and 64-bit Linux endpoints. However, only 64-bit endpoints are supported for Oracle databases that use Online Master Key, previously called TDE direct connections.

The supported endpoint platforms in this release are:

- Oracle Linux (5.*x*, 6.*x*, and 7.*x*)
- Oracle Solaris (10.*x* and 11.*x*)
- Oracle Solaris Sparc (10.*x* and 11.*x*)
- RHEL 5, 6, and 7
- IBM AIX (5.3, 6.1 and 7.1)
- HP-UX (IA) (11.31)
- Windows Server 2008
- Windows Server 2012

## 3.1.4 Endpoint Database Requirements

Endpoints that are Oracle Database 10 *g* Release 2 and later can use the `okvutil upload` command to upload Oracle wallets to Oracle Key Vault. Endpoints that are Oracle Database 11 *g* Release 2 and later can use the Online Master Key to manage TDE master keys.

Note, that the term Online Master Key replaces the term TDE direct connection.

Endpoints that are Oracle Database might need to set the `COMPATIBLE` initialization parameter.

For an endpoint that is Oracle Database 11.2 or 12.1, set the `COMPATIBLE` initialization parameter to 11.2.0.0 or higher. For example:

```
SQL> ALTER SYSTEM SET COMPATIBLE = '11.2.0.0' SCOPE=SPFILE;
```

This applies to an Oracle Database endpoint that is connected with Oracle Key Vault using an Online Master Key (formerly known as TDE direct connection). This compatibility mode setting is not required for Oracle wallet upload or download operations.

Also note that after setting the `COMPATIBLE` parameter to 11.2.0.0, you cannot set it to a lower value like 10.2. After setting the `COMPATIBLE` parameter you must restart the database.

> **See Also:**
>
> *Oracle Database Administrator's Guide* for more information about setting the `COMPATIBLE` parameter

# 3.2 Installing and Configuring Oracle Key Vault 12.2.0.5.0 and Later

This section explains how to install and configure Oracle Key Vault 12.2.0.5.0 and later. To install and configure Oracle Key Vault 12.2.0.4.0 and earlier, see Installing and Configuring Oracle Key Vault 12.2.0.4.0 and Earlier (page A-1).

- Downloading the Oracle Key Vault Appliance Software (page 3-4)
- Installing the Oracle Key Vault Appliance Software (page 3-5)
- Performing Post-Installation Tasks (page 3-10)

## 3.2.1 Downloading the Oracle Key Vault Appliance Software

For a fresh installation, the Oracle Key Vault appliance software can be downloaded from Software Delivery Cloud. Note that this package cannot be used to upgrade Oracle Key Vault.

For an upgrade, Oracle Key Vault can be downloaded from the Oracle Automated Release Updates (ARU) website.

To download the Oracle Key Vault Appliance Software:

1. Use a web browser to access the Oracle Software Delivery Cloud portal:

   https://edelivery.oracle.com

2. Click **Sign In**. Enter your **User ID** and **Password**, if required.
3. In the first field, select **Release**. In the next field, type **Key Vault** and click **Search**.
4. From the list that is displayed, select **Oracle Key Vault 12.2.0.12.0**.

   The download is added to your Cart.
5. Click **Selected Software**.
6. On the next page, verify the details of the installation package, and click **Continue**.

7. The **Oracle Standard Terms and Restrictions** dialog box is displayed.
8. Select **I have reviewed and accept the terms of the Commercial License, Special Programs License, and/or Trial License**, and click **Continue**.
9. The **File Download** dialog box is displayed. Click **View Digest Details**.

   Oracle Key Vault 12.2.0.12.0 consists of the following ISO files:

   - `Vxxxxxx-01.iso` (Oracle Key Vault 12.2.0.12.0 (12.2 Bundle Patch 12) - Disc 1)
   - `Vxxxxxx-02.iso` (Oracle Key Vault 12.2.0.12.0 (12.2 Bundle Patch 12) - Disc 2)
10. Copy both checksum values displayed beside **SHA256** and store them for later reference.
11. Click **Download** and select a location to save the ISO files.
12. Click **Save**.

    The combined size of both ISO files exceeds 4 GB, and will take time to download, depending on the network speed. The estimated download time and speed are displayed in the **File Download** dialog box.
13. The ISO files are downloaded to the specified location. Verify the sha256 checksums of the downloaded files:

    ```
    sha256sum Vxxxxxx-01.iso
    ```

    Ensure that the checksums match the values that you copied from the **File Download** dialog box in *Step 10*.
14. Burn the ISO files to two DVD-ROM discs and label the discs:
    - `OKV BP12 Disc 1`
    - `OKV BP12 Disc 2`

You can now install Oracle Key Vault on the server.

## 3.2.2 Installing the Oracle Key Vault Appliance Software

The installation process installs all required software components onto a dedicated server. The installation process may take from 30 minutes to an hour to complete, depending on the server resources where you are installing Oracle Key Vault.

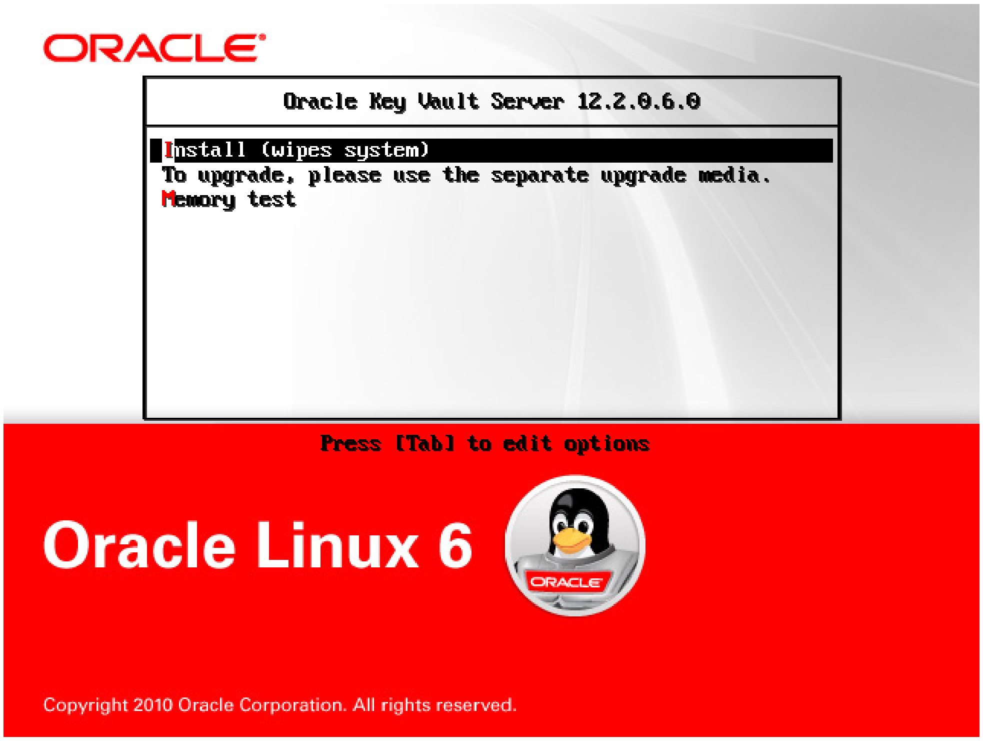
> **Caution:**
>
> The Oracle Key Vault installation wipes the server and installs a stripped-down version of Oracle Linux 6.9, thus erasing existing software and data on the server.

- Ensure that the server meets the recommended requirements.
- Request a fixed IP address, network mask, and gateway address from your network administrator for the dedicated server. You will need this information to configure the network in *Step 13*.

To install the Oracle Key Vault appliance:

1. Insert `OKV BP12 Disc 1` into the CD/DVD drive and restart the computer.
2. The installation starts, and the initial splash screen is displayed.

**Figure 3-1 Oracle Key Vault Install Screen**



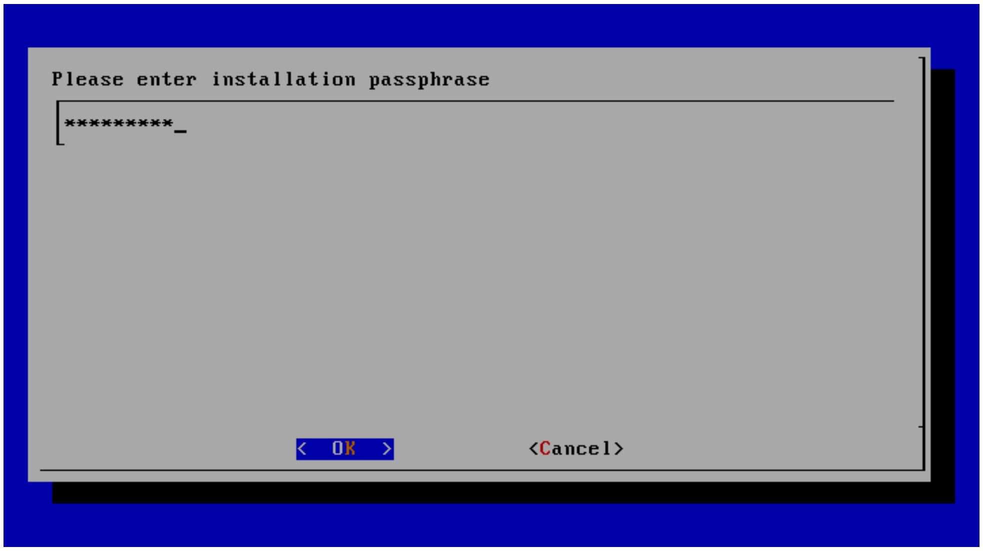
```
Oracle Key Vault Server 12.2.0.6.0

Install (wipes system)
To upgrade, please use the separate upgrade media.
Memory test

Press [Tab] to edit options
```

3. Using the up and down arrow keys, select **Install (wipes system)**, and press **Enter** .

   The installation begins and after several minutes, the message **Please insert disc 2** is displayed.
4. Insert `OKV BP12 Disc 2` into the CD/DVD drive, and press **Enter**.

   The installation proceeds and after several minutes, the message **Please insert disc 1** is displayed.
5. Insert `OKV BP12 Disc 1` into the CD/DVD drive, and press **Enter**.
6. The installation proceeds and after several minutes, the message **Please enter installation passphrase** is displayed.

**Figure 3-2    Installation Passphrase Screen**



Please enter installation passphrase

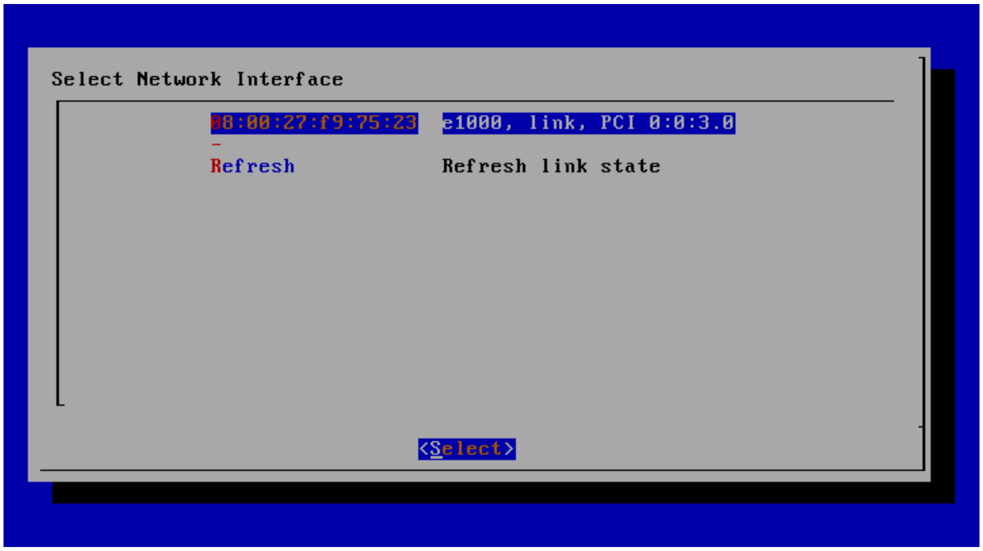
*********_

The installation passphrase must have 8 or more characters and contain at least one of each of the following: an uppercase letter, a lowercase letter, number, and special character from the set: period (`.`), comma (`,`), underscore (_), plus sign (+), colon (`:`), space.

It is important to store the installation passphrase securely. You will need it later to authenticate yourself at the Key Vault management console and complete the post-installation tasks.

7. Enter the installation passphrase, and press **Enter**.
8. Confirm the installation passphrase, and press **Enter**.
9. The message **Installation passphrase was successfully configured** is displayed. Press **Enter**. The **Select Network Interface** screen is displayed.

**Figure 3-3    Select Network Interface Screen**



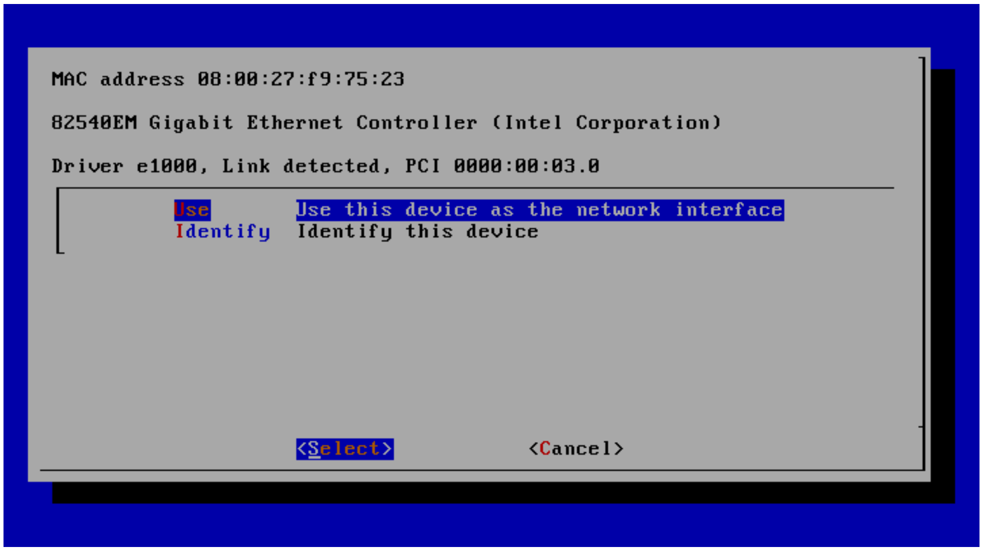
Select Network Interface

```
08:00:27:f9:75:23   e1000, link, PCI 0:0:3.0
Refresh             Refresh link state
```

10. Select the interface and press **Enter**. If more than one network interface is available, select the interface that you want to serve as the management interface, and to communicate with endpoints.

11. The **Identify Management Interface** screen is displayed.

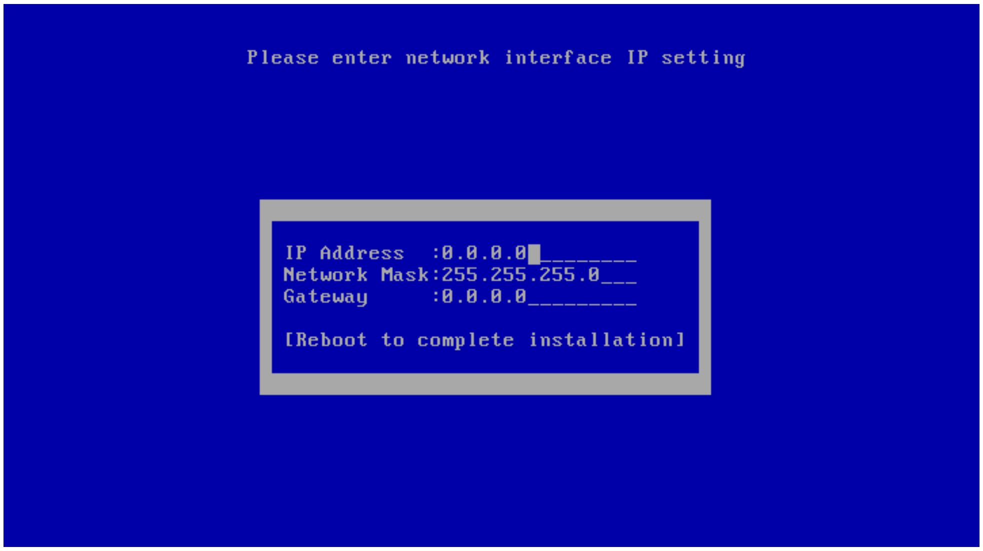
**Figure 3-4    Identify Management Interface Screen**



```
MAC address 08:00:27:f9:75:23

82540EM Gigabit Ethernet Controller (Intel Corporation)

Driver e1000, Link detected, PCI 0000:00:03.0

          Use         Use this device as the network interface
          Identify    Identify this device

                     <Select>            <Cancel>
```

12. Press **Enter**. The **IP Address Setting for Management Interface Screen** is displayed.

**Figure 3-5    IP Address Setting for Management Interface Screen**



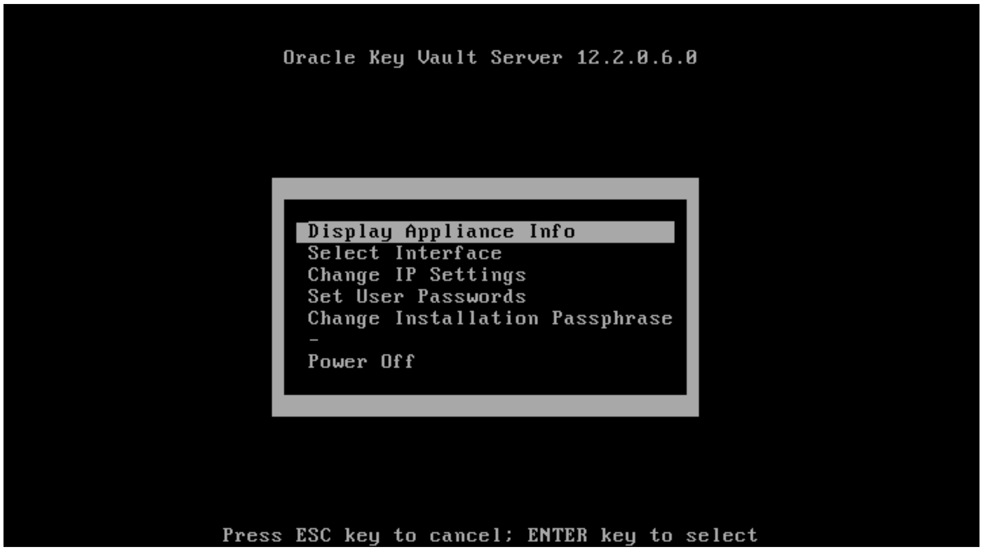
```
Please enter network interface IP setting

IP Address  :0.0.0.0_________
Network Mask:255.255.255.0___
Gateway     :0.0.0.0_________

[Reboot to complete installation]
```

13. Enter the fixed IP address, network mask, and gateway address you received from your network administrator. Select **Reboot to complete installation** and press **Enter**.

    The installer installs and configures the operating system, database, and Oracle Key Vault on the server to make it a self-contained hardened appliance. The installation and configuration process can take between 30 minutes to an hour. Press the **Shift** key to check installation status.

14. If the installation completed successfully, the **Oracle Key Vault Server <Release Number>** screen appears.

**Figure 3-6 Oracle Key Vault Server <Release Number> Screen**



```
Oracle Key Vault Server 12.2.0.6.0

Display Appliance Info
Select Interface
Change IP Settings
Set User Passwords
Change Installation Passphrase
-
Power Off

Press ESC key to cancel; ENTER key to select
```

Select **Display Appliance Info** and press **Enter** to see the IP address settings for the appliance. Make a note of the IP address of the appliance. You will need it to log into the browser-based management console of Oracle Key Vault.

If you need to correct the IP Address, network mask, or the IP gateway for any reason, you can select **Change IP Settings** and enter the new IP settings.

Select **Set User Passwords** to set the Root and Support User passwords. You can also set the Root and Support User passwords when performing Post-Installation Tasks (page 3-10).

You have the option to change the installation passphrase by selecting **Change Installation Passphrase**. For more information about changing the installation passphrase, see Change the Installation Passphrase (page 12-17).

> **Note:**
>
> You will need to enter the old installation passphrase in order to update the installation passphrase.

Make a note of the installation passphrase. You will need it to log into the management console for the first time, in order to complete the post-installation tasks.

## 3.2.3 Performing Post-Installation Tasks

After you install Oracle Key Vault, you must complete the following post-installation tasks: setting up the administrative user accounts, and passwords for recovery, root, and support.

To perform the post-installation tasks:

1. Use a web browser to connect to the Oracle Key Vault server.

   To connect in to an Oracle Key Vault server whose IP address is 192.0.2.254, enter the following in the Address Bar:

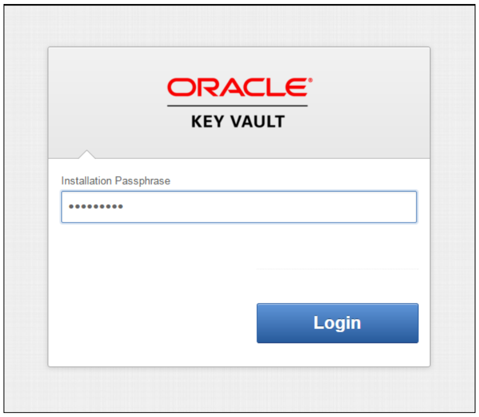
   ```
   https://192.0.2.254
   ```

2. If the web browser displays a security warning message stating that you are connecting to a website with an untrusted or self-signed security certificate, accept the security warning message and proceed to connect to the Oracle Key Vault server.

   > **Note:**
   >
   > After completing the post-installation tasks, you can upload a custom certificate or certificate chain that is trusted by the browser, so that you can connect to the Oracle Key Vault server without encountering the security warning message. For more information about uploading a custom certificate, see Third Party Certificates (page 13-4).

3. The **Installation Passphrase** screen is displayed.

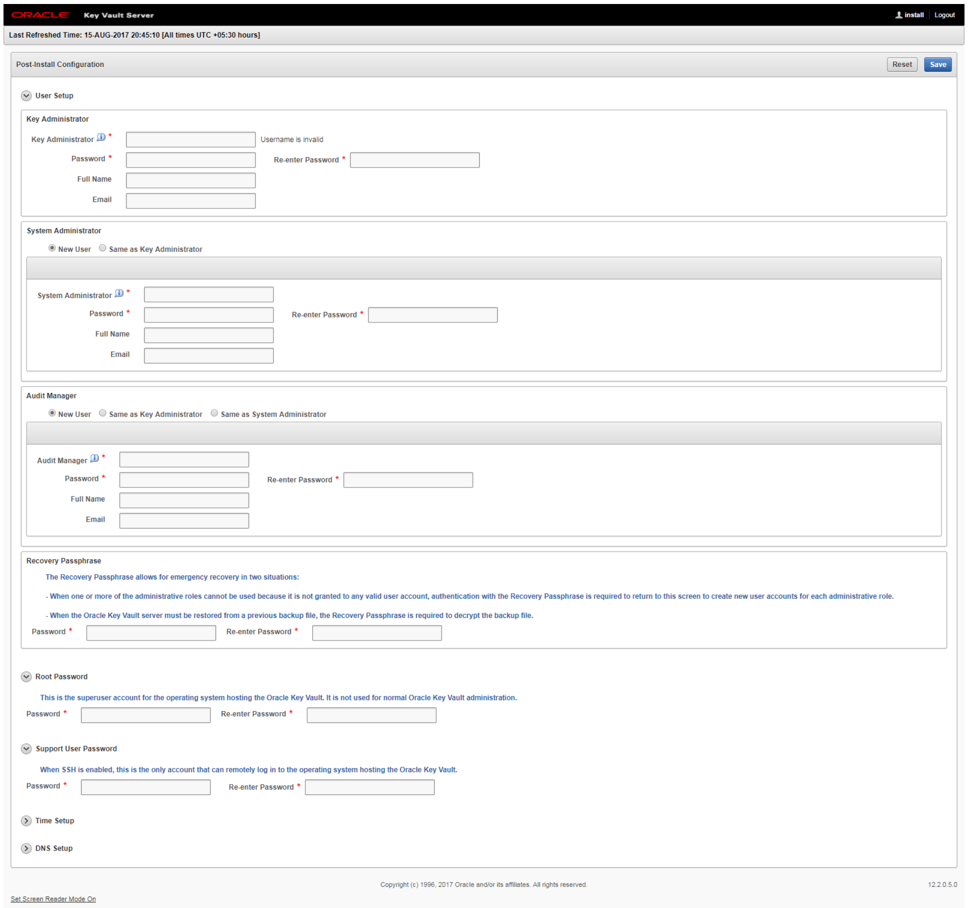
**Figure 3-7 Installation Passphrase Screen**



> **Note:**
>
> The **Installation Passphrase** screen is displayed when you connect to the Oracle Key Vault server for the first time, in order to complete the post-installation tasks. After you complete the post-installation tasks, the Oracle Key Vault login screen is displayed when you access the Oracle Key Vault management console through the web browser.

4. Enter the installation passphrase. The **Post-Install Configuration** screen is displayed.

**Figure 3-8 Post-Install Configuration Screen**



Last Refreshed Time: 15-AUG-2017 20:45:10 [All times UTC +05:30 hours]

Post-Install Configuration

User Setup

Key Administrator

Key Administrator * Username is invalid
Password * Re-enter Password *
Full Name
Email

System Administrator

● New User ○ Same as Key Administrator

System Administrator *
Password * Re-enter Password *
Full Name
Email

Audit Manager

● New User ○ Same as Key Administrator ○ Same as System Administrator

Audit Manager *
Password * Re-enter Password *
Full Name
Email

Recovery Passphrase

The Recovery Passphrase allows for emergency recovery in two situations:

- When one or more of the administrative roles cannot be used because it is not granted to any valid user account, authentication with the Recovery Passphrase is required to return to this screen to create new user accounts for each administrative role.

- When the Oracle Key Vault server must be restored from a previous backup file, the Recovery Passphrase is required to decrypt the backup file.

Password * Re-enter Password *

Root Password

This is the superuser account for the operating system hosting the Oracle Key Vault. It is not used for normal Oracle Key Vault administration.

Password * Re-enter Password *

Support User Password

When SSH is enabled, this is the only account that can remotely log in to the operating system hosting the Oracle Key Vault.

Password * Re-enter Password *

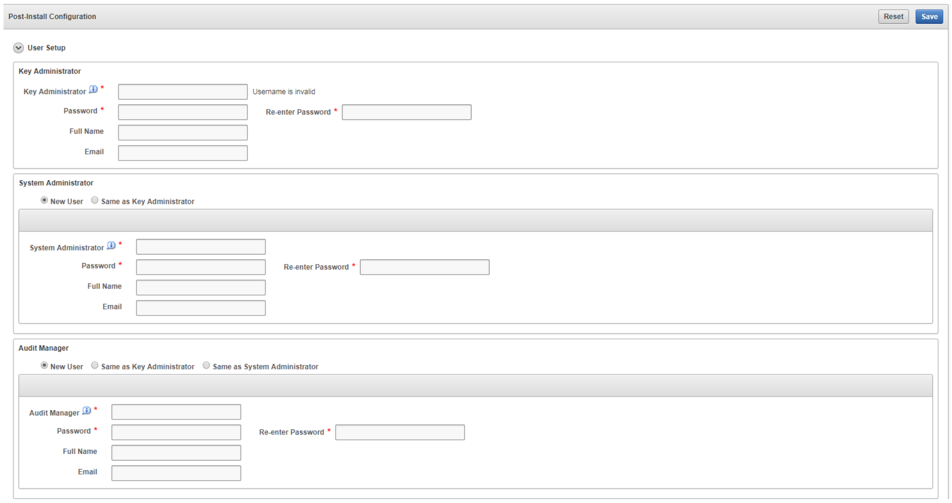
Time Setup

DNS Setup

Copyright (c) 1996, 2017 Oracle and/or its affiliates. All rights reserved. 12.2.0.5.0

5. In the **User Setup** section, create three administrative user accounts for the Key Administrator, System Administrator, and Audit Manager.

### Figure 3-9 Post-Install Configuration — User Setup



Username is invalid

System Administrator: ● New User ○ Same as Key Administrator

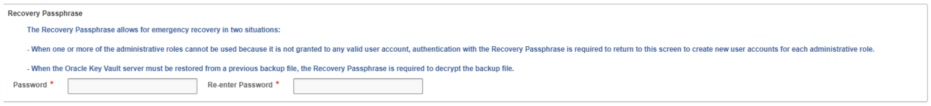
Audit Manager: ● New User ○ Same as Key Administrator ○ Same as System Administrator

In the **User Setup** section:

- Enter the user name and password, the full name (optional), and email (optional) for each administrative user account.
- You can create a different user account for each of these administrative roles for a strict separation of duties, or combine roles as needed.
- Passwords must have 8 or more characters and contain at least one of each of the following: an uppercase letter, a lowercase letter, number, and one special character from the set: period (.), comma (,), underscore (_), plus sign (+), colon (:), space.

6. In the **Recovery Passphrase** section, set the recovery passphrase.

### Figure 3-10 Post-Install Configuration — Recovery Passphrase



The Recovery Passphrase allows for emergency recovery in two situations:

- When one or more of the administrative roles cannot be used because it is not granted to any valid user account, authentication with the Recovery Passphrase is required to return to this screen to create new user accounts for each administrative role.

- When the Oracle Key Vault server must be restored from a previous backup file, the Recovery Passphrase is required to decrypt the backup file.

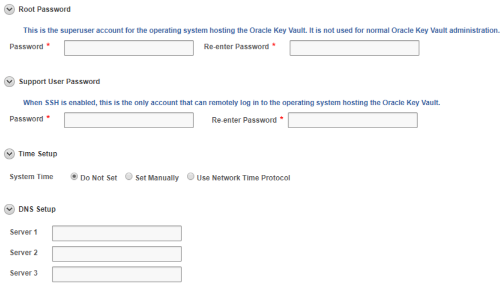
The recovery passphrase has the same minimum requirements as user passwords. For greater security, it is recommended that you make the recovery passphrase longer and more complex. You must keep the recovery passphrase safe and retrievable because it is required in the following situations:

- In an emergency, when there are no administrative users available to access Key Vault.
- To restore Key Vault data from a backup.
- To reset the recovery passphrase.

> **Caution:**
>
> **It is important to establish a secure process for the storage and retrieval of the recovery passphrase, including older recovery passphrases. The only way to recover from a lost recovery passphrase is to re-install Key Vault.**

7. In the next section, set the Root and Support User passwords, if you did not set the passwords using the **Set User Passwords** option on the **Oracle Key Vault Server <Release Number>** screen in the previous procedure, Installing the Oracle Key Vault Appliance Software (page 3-5).

**Figure 3-11 Post-Install Configuration — Root and Support User Passwords**



Root Password

This is the superuser account for the operating system hosting the Oracle Key Vault. It is not used for normal Oracle Key Vault administration.

Password * Re-enter Password *

Support User Password

When SSH is enabled, this is the only account that can remotely log in to the operating system hosting the Oracle Key Vault.

Password * Re-enter Password *

Time Setup

System Time ● Do Not Set ○ Set Manually ○ Use Network Time Protocol

DNS Setup

Server 1

Server 2

Server 3

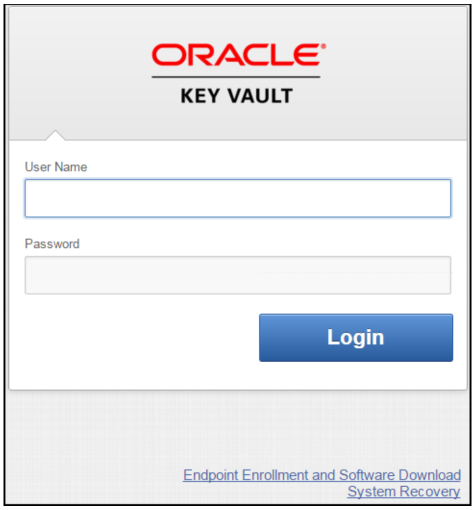
The root password is the super user account for the operating system hosting Key Vault. You will need the support password to log into Key Vault remotely using the SSH protocol.

> **Caution:**
>
> **Keep the root and support user passwords safe because these passwords are set during post-installation only. After post-installation you cannot change them from the Oracle Key Vault management console.**

The **Time Setup** and **DNS Setup** settings are optional at this stage, and can be set up later by a System Administrator.

8. Click **Save** in the upper right corner of the **Post-Install Configuration** screen. The Oracle Key Vault Management Console login screen is displayed.

**Figure 3-12    Oracle Key Vault Management Console Login Screen**



You can now login to the Oracle Key Vault management console with the credentials of any of the user accounts created during the post-installation process.

## 3.3 Logging In to the Oracle Key Vault Management Console

To use Oracle Key Vault, users can log in to the Oracle Key Vault management console.

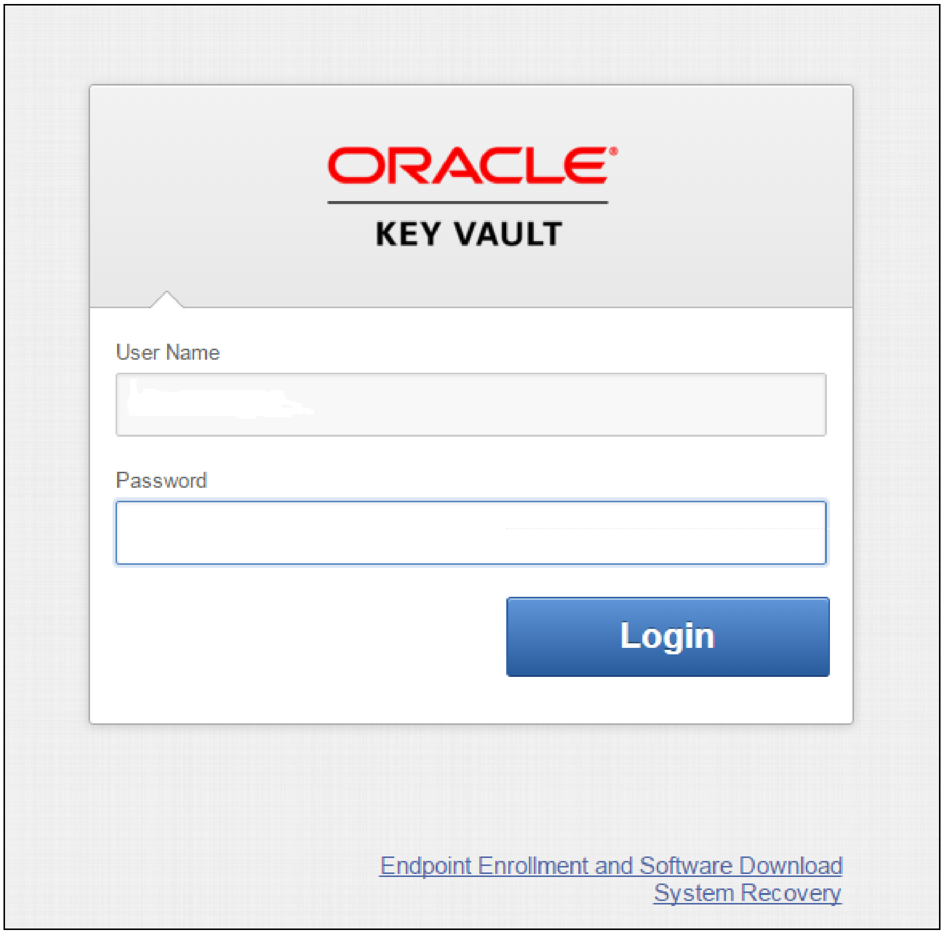
1. Open a web browser.
2. Connect using an HTTPS connection and the IP address of Oracle Key Vault.

   For example, to log in to a server whose IP address is 192.0.2.254, enter:

   ```
   https://192.0.2.254
   ```

   The login screen appears.

**Figure 3-13 Oracle Key Vault Screen with Username and Password**



3. Enter your user name and password.
4. Click **Login**.

## 3.4 Overview of the Management Console

The Oracle Key Vault management console is a browser-based console that connects to the appliance using the `https` secure communication channel. It provides the graphical user interface for Oracle Key Vault, where users can perform tasks like:

- Creating and managing users, endpoints, and their respective groups
- Creating and managing virtual wallets and security objects
- Setting system settings, like network and other services
- Setting up high availability and backup

## 3.5 Performing Actions and Searches

Many of the tab and menu pages contain an Actions menu or Search bars that allow you to search and perform actions on lists and the results of searches.

> **Note:**
>
> Detailed help for the Actions menus and Search bars is provided in the Help selection of the Actions drop-down list.

- Actions Menus (page 3-16)
- Search Bars (page 3-16)

## 3.5.1 Actions Menus

The actions available from an **Actions** drop-down menu can vary but typically include a set of standard menu items.

These items are as follows:

- **Select Columns:** Select which column should be displayed.
- **Filter:** Filter by column or row and a user-defined expression.
- **Rows Per Page:** Choose how many rows you want to view .
- **Format:** Choose formatting such as **Sort**, **Control Break**, **Highlight**, **Compute**, **Aggregate**, **Chart**, and **Group By**.
- **Save Report:** Save reports.
- **Reset:** Reset the report settings, removing any customizations.
- **Help:** Get information about these actions.
- **Download:** Download the result set in CSV or HTML.

## 3.5.2 Search Bars

Along with Actions menus, many tabs contain search bars.

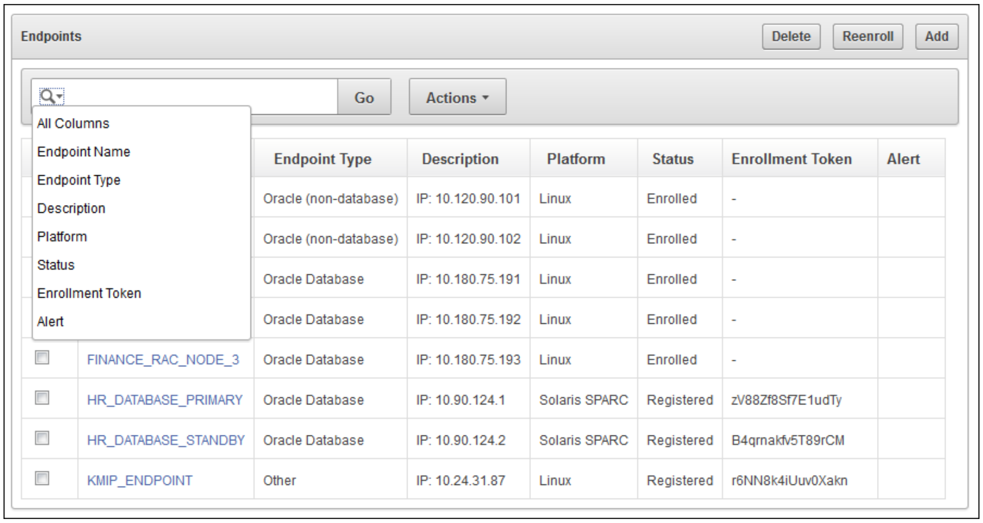
This demonstration searches for endpoints, but the process is the same for other searches, except that the column headings are different.

Wildcard characters are not supported, but the search does match any letter or phrase that you enter. You can use the **Filter** menu item under **Actions** to further fine-tune the search.

To perform a search:

1. Enter a name or other identifier in the search field or (optionally) place your cursor on the magnifying icon in the Search bar to select one of the table headings (in this case, **All Columns**, **Endpoint Name**, **Endpoint Type**, **Description**, **Platform**, **Status**, **Enrollment Token**, and **Alert**) and then enter a search term.

**Figure 3-14    Endpoints Page**



| | | Endpoint Type | Description | Platform | Status | Enrollment Token | Alert |
|---|---|---|---|---|---|---|---|
| | | Oracle (non-database) | IP: 10.120.90.101 | Linux | Enrolled | - | |
| | | Oracle (non-database) | IP: 10.120.90.102 | Linux | Enrolled | - | |
| | | Oracle Database | IP: 10.180.75.191 | Linux | Enrolled | - | |
| | | Oracle Database | IP: 10.180.75.192 | Linux | Enrolled | - | |
| □ | FINANCE_RAC_NODE_3 | Oracle Database | IP: 10.180.75.193 | Linux | Enrolled | - | |
| □ | HR_DATABASE_PRIMARY | Oracle Database | IP: 10.90.124.1 | Solaris SPARC | Registered | zV88Zf8Sf7E1udTy | |
| □ | HR_DATABASE_STANDBY | Oracle Database | IP: 10.90.124.2 | Solaris SPARC | Registered | B4qrnakfv5T89rCM | |
| □ | KMIP_ENDPOINT | Other | IP: 10.24.31.87 | Linux | Registered | r6NN8k4iUuv0Xakn | |

2. Click **Go**.

   A new endpoint list appears, displaying the endpoints that meet the search criteria. A filter icon (a funnel) indicates that a search has been performed and displays the search criteria.

3. You can select or deselect the filter icon to disable search and view the entire list.

4

# High Availability, Backup and Restore Operations

Oracle Key Vault may be configured for high availability and automatic backups for continuous, reliable, and protected access to security objects with minimum downtime.

## 4.1 Why High Availability?

With data centers geographically dispersed around the globe there is an increased need for data to be reliably accessible, on-demand, at any time. Users carrying out business-critical operations need data to be accessible and recoverable with minimum downtime. These requirements are met in a high availability deployment.

You achieve high availability by adding redundancy in the form of a standby server, that can take over the functions of the primary server in case of failure. The standby server helps you eliminate single points of failure and reduce server downtime, two main reasons to deploy Oracle Key Vault in a high availability configuration.

## 4.2 About High Availability for Oracle Key Vault

A high availability Oracle Key Vault deployment consists of two Key Vault peer servers called primary and standby. The primary is the active server servicing endpoint requests. The standby takes over if the primary fails.

### 4.2.1 How High Availability for Oracle Key Vault Works

You configure high availability by providing the primary and standby servers with each other's IP address and certificate, then pairing them. While pairing the primary and standby servers, you can select one as the primary server, and the other as the standby. A failover timeout that you set, determines when the standby starts to take over as primary.

**Prerequisites:**

It is highly recommended to keep the primary and standby systems as identical as possible, as their roles can be reversed in maintenance periods and failure situations. These include the following:

- Key Vault software versions
- Disk size
- RAM size
- System clocks on both systems must be synchronized

If your deployment requires high availability, we recommend configuring it *before* adding endpoints to Key Vault so that endpoints know about both primary and standby servers. An endpoint added before the standby server will not know about the standby server unless you re-enroll the endpoint. If you configure high availability after adding endpoints, you must re-enroll the endpoints that were previously enrolled with the primary and standby servers in standalone mode.

If you want to add SNMP support in a high availability environment, you must configure SNMP on both primary and standby servers before pairing them. This is because the standby server is no longer accessible from the management console as all requests are forwarded to the primary server.

If you want to use a third-party certificate in a high availability configuration, you must install it on the primary and standby servers first, and then pair them.

With persistent cache enabled, both the primary and the standby will cache the master keys from Oracle Key Vault independently. Ensure that a TDE operation is executed on primary and standby after they are started. The persistent cache feature also enables endpoints to be operational during high availability configuration, and switchover and failover operations.

If enabled, the Read-Only Restricted mode of Oracle Key Vault 12.2.0.5.0 and higher ensures endpoint operational continuity if either primary or standby server is not available. Read-Only Restricted mode also ensures uniform behavior of the surviving Oracle Key Vault server in the event that either the primary or standby servers go down.

A High Availability configuration is characterized by continuous synchronization between the primary and standby server. When synchronization is lost between the primary and standby servers, it is possible to encounter a split-brain scenario where two primary servers might be active simultaneously. In such a scenario, both servers record new data that diverges from the last synchronized state. When connectivity is restored between the primary and standby servers, it may not be possible to reconcile the changes on the two servers, and data loss may occur.

You can enable or disable restricted mode when configuring High Availability by setting the **Allow Read-Only Restricted Mode** radio button to **Yes** or **No** on the **Configure High Availability** page.

When Read-Only Restricted mode is enabled, the primary server enters Read-Only Restricted mode if the standby server is unavailable. In Read-Only Restricted mode, the primary server will allow keys to be retrieved, but will not allow keys to be modified or new keys to be added. This ensures that endpoints still have access to their keys, and key data or metadata is not lost due to a split-brain scenario. However, the primary server still writes audit records, which may be lost if a split-brain scenario occurs with the standby server.

When Read-Only Restricted mode is disabled, the primary server becomes unavailable, and stops accepting new requests if the standby server is unavailable. Endpoints connected to Oracle Key Vault will be unable to retrieve keys from the server until connectivity is restored between primary and standby servers. The Persistent Master Key Cache feature can be used to avoid endpoint downtime. Data integrity is ensured by allowing endpoints to communicate with one primary server at any given time. Split-brain situations, and the risk of data loss associated with such situations, are avoided.

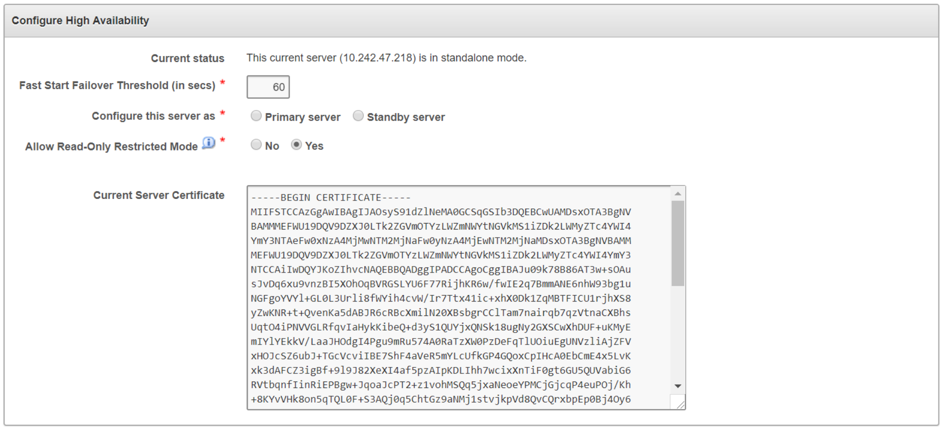
## 4.2.2 Configuring High Availability

To configure high availability you must be a user with System Administrator privileges. You must access the primary and standby appliance separately from two browser instances (separate tabs in the same browser or different browsers), and copy the IP address and certificate from one appliance to the other, and vice versa. With persistent cache enabled, endpoints will continue to be operational while high availability is configured.

To configure High Availability:

1. Open a web browser and enter the IP address of the designated primary server. The **Oracle Key Vault Management Console login screen** is displayed.
2. Log in as the System Administrator.
3. Click the **System** tab, then click **High Availability** in the left pane. The **Configure High Availability** page is displayed.

**Figure 4-1 Configure High Availability Page**



Configure High Availability

Current status: This current server (10.242.47.218) is in standalone mode.

Fast Start Failover Threshold (in secs) *: 60

Configure this server as *: ○ Primary server ○ Standby server

Allow Read-Only Restricted Mode *: ○ No ● Yes

Current Server Certificate:

```
-----BEGIN CERTIFICATE-----
MIIFSTCCAzGgAwIBAgIJAOsyS91dZlNeMA0GCSqGSIb3DQEBCwUAMDsxOTA3BgNV
BAMMMEFWU19DQV9DZXJ0LTk2ZGVmOTYzLWZmNWYtNGVkMS1iZDk2LWMyZTc4YWI4
YmY3NTAeFw0xNzA4MjMwNTM2MjNaFw0yNzA4MjEwNTM2MjNaMDsxOTA3BgNVBAMM
MEFWU19DQV9DZXJ0LTk2ZGVmOTYzLWZmNWYtNGVkMS1iZDk2LWMyZTc4YWI4YmY3
NTCCAiIwDQYJKoZIhvcNAQEBBQADggIPADCCAgoCggIBAJu09k78B86AT3w+sOAu
sJvDq6xu9vnzBI5XOhOqBVRGSLYU6F77RijhKR6w/fwIE2q7BmmANE6nhW93bg1u
NGFgoYVYl+GL0L3Urli8fWYih4cvW/Ir7Ttx41ic+xhX0Dk1ZqMBTFICU1rjhXS8
yZwKNR+t+QvenKa5dABJR6cRBcXmilN20XBsbgrCClTam7nairqb7qzVtnaCXBhs
UqtO4iPNVVGLRfqvIaHykKibeQ+d3yS1QUYjxQNSk18ugNy2GXSCwXhDUF+uKMyE
mIYlYEkkV/LaaJHOdgI4Pgu9mRu574A0RaTzXW0PzDeFqTlUOiuEgUNVzliAjZFV
xHOJcSZ6ubJ+TGcVcviIBE7ShF4aVeR5mYLcUfkGP4GQoxCpIHcA0EbCmE4x5LvK
xk3dAFCZ3igBf+9l9J82XeXI4af5pzAIpKDLIhh7wcixXnTiF0gt6GU5QUVabiG6
RVtbqnfIinRiEPBgw+JqoaJcPT2+z1vohMSQq5jxaNeoeYPMCjGjcqP4euPOj/Kh
+8KYvVHk8on5qTQL0F+S3AQj0q5ChtGz9aNMj1stvjkpVd8QvCQrxbpEp0Bj4Oy6
```

The following are the fields on the **Configure High Availability** page:

- **Current status**: Displays the IP address and status of the current server.
- **Fast Start Failover Threshold (in secs)**: Displays the duration (in seconds) that will elapse before the server takes over from a failed peer server. The default is 60 seconds.

To avoid failover during brief or intermittent failures, increase the duration.

- **Configure this server as**: Displays whether the server is configured as a **Primary server** or **Standby server**.
- **Allow Read-Only Restricted Mode**: Displays the status of Read-Only Restricted mode. The default is **Yes**.

  When enabled, Read-Only Restricted mode ensures operational continuity of the endpoints if the primary or standby Oracle Key Vault server is affected by server, hardware, or network failures
- **Current Server Certificate**: Displays the server certificate.

4. Copy the following information, and store it in a text file named `primary.txt`. You will require this information when you configure the standby server:

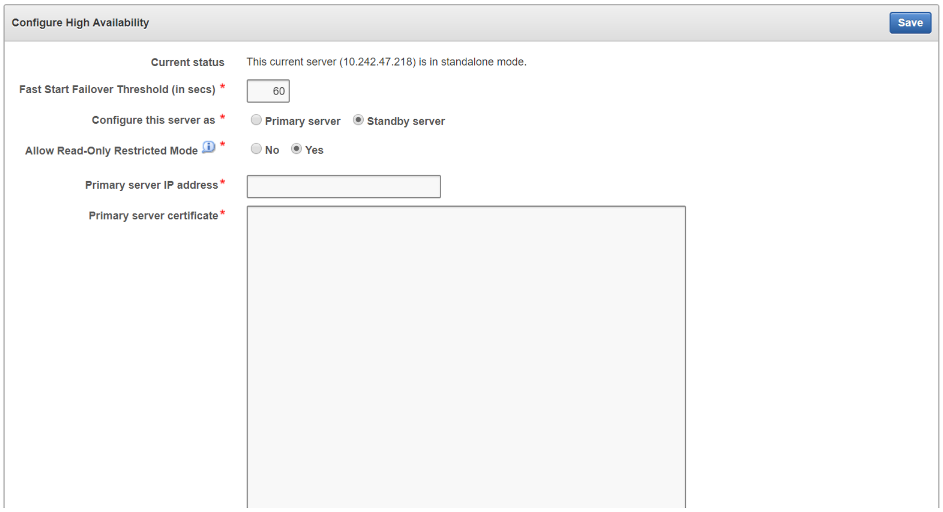
   - From the **Current status** field, copy the IP address and paste it in `primary.txt`.
   - From the **Current Server Certificate** field, copy the server certificate and paste it in `primary.txt`.

   Save `primary.txt`.
5. Open a web browser and enter the IP address of the designated standby server. The **Oracle Key Vault Management Console login screen** is displayed.
6. Log in as the System Administrator.
7. Click the **System** tab, then click **High Availability** in the left pane. The **Configure High Availability** page is displayed.
8. Copy the following information, and store it in a text file named `standby.txt`. You will require this information when you configure the primary server:

   - From the **Current status** field, copy the IP address and paste it in `standby.txt`.
   - From the **Current Server Certificate** field, copy the server certificate and paste it in `standby.txt`.

   Save `standby.txt`.
9. In the **Configure this server as** field, select **Standby server**. The **Primary server IP address** and **Primary server certificate** fields are displayed.

### Figure 4-2 Configure High Availability Page (Standby Server)



Configure High Availability

Current status: This current server (10.242.47.218) is in standalone mode.

Fast Start Failover Threshold (in secs) *: 60

Configure this server as *: ○ Primary server ● Standby server

Allow Read-Only Restricted Mode *: ○ No ● Yes

Primary server IP address *:

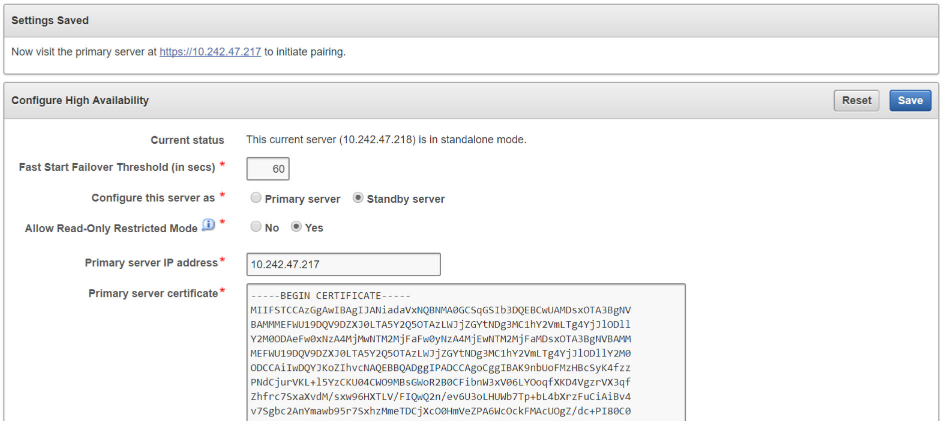
Primary server certificate *:

Ensure that **Yes** is selected in the **Allow Read-Only Restricted Mode** field.

> **Note:**
>
> Do not disable Read Only Restricted mode unless necessary. If High Availability is set up with Read Only Restricted mode disabled, you must enable it by reinstalling and configuring Oracle Key Vault again.

10. Copy the following information from `primary.txt`, and paste it in the **Configure High Availability** page of the standby server:
    - Copy the IP address and paste it in the **Primary server IP address** field.
    - Copy the server certificate and paste it in the **Primary server certificate** field.
11. Click **Save**. The **Settings Saved** page is displayed.

### Figure 4-3 Settings Saved Page



Settings Saved

Now visit the primary server at https://10.242.47.217 to initiate pairing.

Configure High Availability

Current status: This current server (10.242.47.218) is in standalone mode.

Fast Start Failover Threshold (in secs) *: 60

Configure this server as *: ○ Primary server ● Standby server

Allow Read-Only Restricted Mode *: ○ No ● Yes

Primary server IP address *: 10.242.47.217

Primary server certificate *:

```
-----BEGIN CERTIFICATE-----
MIIFSTCCAzGgAwIBAgIJANiadaVxNQBNMA0GCSqGSIb3DQEBCwUAMDsxOTA3BgNV
BAMMMEFWU19DQV9DZXJ0LTA5Y2Q5OTAzLWJjZGYtNDg3MC1hY2VmLTg4YjJlODll
Y2M0ODAeFw0xNzA4MjMwNTM2MjFaFw0yNzA4MjEwNTM2MjFaMDsxOTA3BgNVBAMM
MEFWU19DQV9DZXJ0LTA5Y2Q5OTAzLWJjZGYtNDg3MC1hY2VmLTg4YjJlODllY2M0
ODCCAiIwDQYJKoZIhvcNAQEBBQADggIPADCCAgoCggIBAK9nbUoFMzHBcSyK4fzz
PNdCjurVKL+l5YzCKU04CWO9MBsGWoR2B0CFibnW3xV06LYOoqfXKD4VgzrVX3qf
Zhfrc7SxaXvdM/sxw96HXTLV/FIQwQ2n/ev6U3oLHUWb7Tp+bL4bXrzFuCiAiBv4
v7Sgbc2AnYmawb95r7SxhzMmeTDCjXcO0HmVeZPA6WcOckFMAcUOgZ/dc+PI80C0
```

The **Reset** button allows you to delete the High Availability configuration, if required.

High Availability is enabled on the designated standby server. The next step is to enable High Availability on the designated primary server.

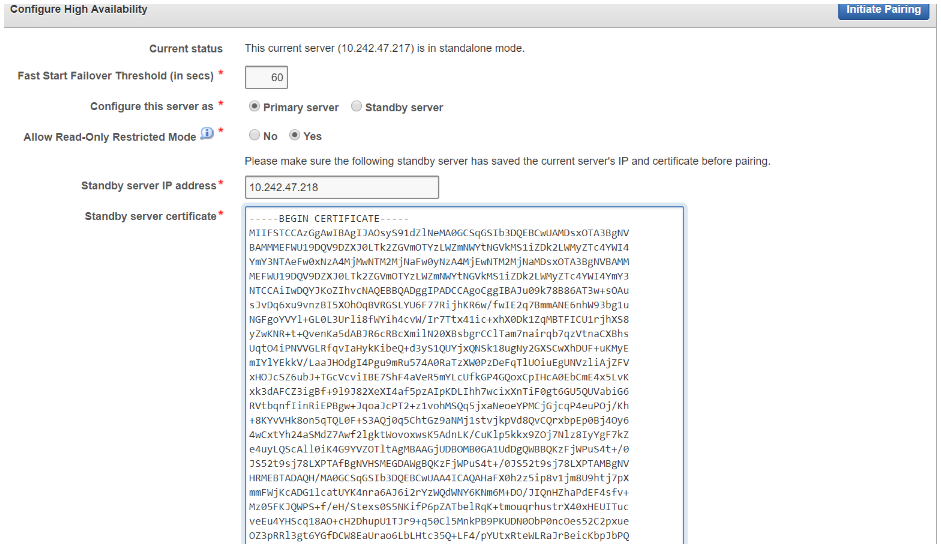
12. On the **Settings Saved** page, click the IP address of the primary server displayed at the top of the page. The **Oracle Key Vault Management Console login screen** of the primary server is displayed.
13. Log in as the System Administrator.
14. Click the **System** tab, then click **High Availability** in the left pane. The **Configure High Availability** page is displayed.
15. In the **Configure this server as** field, select **Primary server**. The **Standby server IP address** and **Standby server certificate** fields are displayed.

    Ensure that **Yes** is selected in the **Allow Read-Only Restricted Mode** field.

> **Note:**
>
> Do not disable Read Only Restricted mode unless necessary. If High Availability is set up with Read Only Restricted mode disabled, you must enable it by reinstalling and configuring Oracle Key Vault again.

16. Copy the following information from `standby.txt`, and paste it in the **Configure High Availability** page of the primary server:
    - Copy the IP address and paste it in the **Standby server IP address** field.
    - Copy the server certificate and paste it in the **Standby server certificate** field.

**Figure 4-4 Configure High Availability Page (Primary Server)**



Current status: This current server (10.242.47.217) is in standalone mode.

Fast Start Failover Threshold (in secs) *: 60

Configure this server as *: ● Primary server ○ Standby server

Allow Read-Only Restricted Mode *: ○ No ● Yes

Please make sure the following standby server has saved the current server's IP and certificate before pairing.

Standby server IP address *: 10.242.47.218

Standby server certificate *:

```
-----BEGIN CERTIFICATE-----
MIIFSTCCAzGgAwIBAgIJAOsyS91dZlNeMA0GCSqGSIb3DQEBCwUAMDsxOTA3BgNV
BAMMMEFWU19DQV9DZXJ0LTk2ZGVmOTYzLWZmNWYtNGVkMS1iZDk2LWMyZTc4YWI4
YmY3NTAeFw0xNzA4MjMwNTM2MjNaFw0yNzA4MjEwNTM2MjNaMDsxOTA3BgNVBAMM
MEFWU19DQV9DZXJ0LTk2ZGVmOTYzLWZmNWYtNGVkMS1iZDk2LWMyZTc4YWI4YmY3
NTCCAiIwDQYJKoZIhvcNAQEBBQADggIPADCCAgoCggIBAJu09k78B86AT3w+sOAu
sJvDq6xu9vnzBI5XOhOqBVRGSLYU6F77RijhKR6w/fwIE2q7BmmANE6nhW93bg1u
NGFgoYVYl+GL0L3Urli8fWYih4cvW/Ir7Ttx41ic+xhX0Dk1ZqMBTFICU1rjhXS8
yZwKNR+t+QvenKa5dABJR6cRBcXmilN20XBsbgrCClTam7nairqb7qzVtnaCXBhs
UqtO4iPNVVGLRfqvIaHykKibeQ+d3yS1QUYjxQNSk18ugNy2GXSCwXhDUF+uKMyE
mIYlYEkkV/LaaJHOdgI4Pgu9mRu574A0RaTzXW0PzDeFqTlUOiuEgUNVzliAjZFV
xHOJcSZ6ubJ+TGcVcviIBE7ShF4aVeR5mYLcUfkGP4GQoxCpIHcA0EbCmE4x5LvK
xk3dAFCZ3igBf+9l9J82XeXI4af5pzAIpKDLIhh7wcixXnTiF0gt6GU5QUVabiG6
RVtbqnfIinRiEPBgw+JqoaJcPT2+z1vohMSQq5jxaNeoeYPMCjGjcqP4euPOj/Kh
+8KYvVHk8on5qTQL0F+S3AQj0q5ChtGz9aNMj1stvjkpVd8QvCQrxbpEp0Bj4Oy6
4wCxtYh24aSMdZ7Awf2lgktWovoxwsK5AdnLK/CuKlp5kkx9ZOj7Nlz8IyYgF7kZ
e4uyLQScAll0iK4G9YVZOTltAgMBAAGjUDBOMB0GA1UdDgQWBBQKzFjWPuS4t+/0
JS52t9sj78LXPTAfBgNVHSMEGDAWgBQKzFjWPuS4t+/0JS52t9sj78LXPTAMBgNV
HRMEBTADAQH/MA0GCSqGSIb3DQEBCwUAA4ICAQAHaFX0h2z5ip8v1jm8U9htj7pX
mmFWjKcADG1lcatUYK4nra6AJ6i2rYzWQdWNY6KNm6M+DO/JIQnHZhaPdEF4sfv+
Mz05FKJQWPS+f/eH/Stexs0S5NKifP6pZATbelRqK+tmouqrhustrX40xHEUITuc
veEu4YHScq18AO+cH2DhupU1TJr9+q50Cl5MnkPB9PKUDN0ObP0ncOes52C2pxue
OZ3pRRl3gt6YGfDCW8EaUrao6LbLHtc35Q+LF4/pYUtxRteWLRaJrBeicKbpJbPQ
```

17. Click **Initiate Pairing**.

    High Availability is enabled on the designated primary server.
18. In the confirmation message that is displayed, click **OK**. The **Operation in Progress** page is displayed.

> **Caution:**
>
> Allow at least 10 minutes to elapse before performing the next operation.

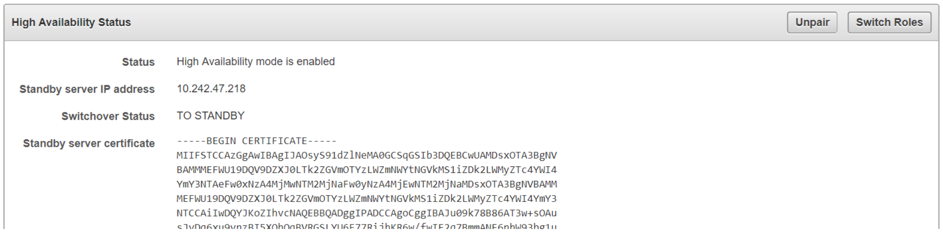
19. After at least 10 minutes have elapsed, click **Refresh**.

    If the pairing of primary and standby servers is successful, the current session is terminated. The **Oracle Key Vault Management Console login screen** of the primary server is displayed.

20. Log in as the System Administrator.

21. Click the **System** tab, then click **High Availability** in the left pane. The **High Availability Status** page is displayed.

    **Figure 4-5 High Availability Status Page**

    High Availability Status

    | | |
    |---|---|
    | Status | High Availability mode is enabled |
    | Standby server IP address | 10.242.47.218 |
    | Switchover Status | TO STANDBY |
    | Standby server certificate | |

    ```
    -----BEGIN CERTIFICATE-----
    MIIFSTCCAzGgAwIBAgIJAOsyS91dZlNeMA0GCSqGSIb3DQEBCwUAMDsxOTA3BgNV
    BAMMMEFWU19DQV9DZXJ0LTk2ZGVmOTYzLWZmNWYtNGVkMS1iZDk2LWMyZTc4YWI4
    YmY3NTAeFw0xNzA4MjMwNTM2MjNaFw0yNzA4MjEwNTM2MjNaMDsxOTA3BgNVBAMM
    MEFWU19DQV9DZXJ0LTk2ZGVmOTYzLWZmNWYtNGVkMS1iZDk2LWMyZTc4YWI4YmY3
    NTCCAiIwDQYJKoZIhvcNAQEBBQADggIPADCCAgoCggIBAJu09k78B86AT3w+sOAu
    sJvDq6xu9vnzBI5XQhOqBVRGSLYU6F77RijhKR6w/fwIF2q7BmmANF6nhW93hg1u
    ```

    The **Unpair** button allows you to disconnect the primary server from the standby server, if required.

    The **Switch Roles** button allows you to switch the roles of the primary server and the standby server, if required. The primary server then assumes the role of the standby server, while the standby server assumes the role of the new primary server.

High Availability is successfully configured.

> **Note:**
>
> When High Availability is configured, you cannot log in to the standby server using a web browser.
>
> To manage High Availability, log in to the primary server using a web browser.

> **Caution:**
>
> Ensure that you leave Read-Only Restricted Mode enabled while configuring High Availability. Enabling it later requires a reinstall of the Oracle Key Vault appliance software on the standby server.
>
> After configuring High Availability, do not change the system time on the primary server. The changed system time causes the standby server to go down, thus disrupting the functioning of the High Availability configuration.

### 4.2.3 Switching Primary and Standby Servers

The high availability configuration allows you to switch roles of the primary and standby server. This is useful during maintenance periods, when you might want to bring a server down to upgrade software or install patches.

If you have persistent cache enabled and the persistent cache timeout is sufficiently tuned, then the endpoints will continue to be operational during the switchover, minimizing endpoint downtime.

You can switch roles as follows:

1. Log in to the Oracle Key Vault management console of the primary node as a user with the System Administrator role.

   Before switching the primary and standby servers, ensure that there are no HA-related alerts on the **Alerts** page. To access the **Alerts** page, click the **Reports** tab, and then click **Alerts** in the left pane.

   Ensure that all HA-related alerts on the **Alerts** page are addressed before switching the primary and standby servers.

2. Click the **System** tab, then **High Availability** from the left side bar.

   The **High Availability Status** page appears.

3. Click **Switch Roles** on the top right.

   The **Switch Roles** button allows you to switch the roles of the primary server and the standby server. The primary server then assumes the role of the standby server, while the standby server assumes the role of the new primary server.

   Select **OK** to the confirmation message.

   An operation initiated message is followed by the **Operation in Progress** page indicating that the switchover operation will take 10 minutes to complete successfully.

   > **Caution:**
   >
   > You must wait for a minimum period of 10 minutes for the switchover operation to complete successfully. If you refresh the UI before the switchover operation is complete, an error message is displayed. The error message is displayed until the switchover is completed successfully.

4. Ensure that at least 10 minutes have elapsed, and only then, click **Refresh**.

   This will log you out of the current session, and open a login page to the switched primary server.

   Both the primary and standby are restarted, however, note that you will only be able to log in to the primary node's web console. The primary server is the active server, and all requests to the standby will be forwarded to the primary.

5. Log in to the primary server to see the IP address of the switched standby node.
6. Click the **System** tab, then **High Availability** from the left side bar.

   The **High Availability Status** page appears.

   The **Standby server IP address** field displays the IP address.

## 4.2.4 Restoring High Availability After a Failover

A failover takes place if the primary server fails. If the primary is not available, the standby server takes over the primary role. If the standby does not hear from the primary for a time exceeding the **Fast Start Failover Threshold** value, it will assume that the primary is down and start the failover process. You can configure the value in the **Fast Start Failover Threshold** field from the Key Vault user interface from the default of 60 seconds. Should the failed server (old primary) come back up, it will automatically become the new standby server, in most cases.

If the primary server fails permanently, the standby server will take over as primary. In this case high availability will have to be restored with manual intervention as follows:

1. Replace the failed server with a newly installed Oracle Key Vault appliance. Be sure to use the original IP address for the failed server.
2. Log on to the newly installed appliance and follow the steps to configure high availability. You may designate the new appliance as the standby (since the cluster has a functional primary) and pair it with the functioning primary.
3. If you want to restore the original configuration and set the new appliance as primary, you can use the **Switch Roles** option, after you successfully pair the two nodes and enable high availability.

   The **Switch Roles** button allows you to switch the roles of the primary server and the standby server. The primary server then assumes the role of the standby server, while the standby server assumes the role of the new primary server.

> **Note:**
>
> When Read-Only Restricted mode is disabled, the primary server's failover status goes into suspended state causing the standby server to wait indefinitely for the primary server to come back up. This is expected behavior to avoid a split-brain scenario where two primary servers are simultaneously active.
>
> When Read-Only Restricted mode is enabled, a primary or standby server failure causes the operational peer to enter Read-Only Restricted mode, thus ensuring endpoint operational continuity.

> **See Also:**
>
> For more information about recovering from High Availability failover situations, see Failover Situations in High Availability Mode (page B-7).

## 4.2.5 Disabling High Availability

You can disable high availability by un-pairing the primary and standby servers. After un-pairing, the primary and standby will operate in standalone mode. To prevent endpoints from connecting to the old standby (new standalone Oracle Key Vault server) you need to take the old standby off the network.

To disable high availability follow the steps below:

1. Log in to the primary server's management console as a user with System Administrator privileges.
2. Select the **System** tab on top, then select **High Availability** from the left side bar.

   The **High Availability Status** page appears with **Unpair** and **Switch Roles** on the top right. The **Unpair** and **Switch Roles** options do the following:

   - The **Unpair** button allows you to disconnect the primary server from the standby server, if required.
   - The **Switch Roles** button allows you to switch the roles of the primary server and the standby server, if required. The primary server then assumes the role of the standby server, while the standby server assumes the role of the new primary server.
3. Click **Unpair**.

   A brief message with a green check appears indicating that the operation has been successfully initiated.

   The **Operation in Progress** page appears indicating a wait time of at least 10 minutes for the un-pairing to complete.

   Wait 10 minutes.
4. Click the **Refresh** button to get logged out of the current session.
5. Log back in to the management console of the primary server. Select **System**, then **High Availability** from the left side bar.

   The **Configure High Availability** page appears.

   The **Current status** field shows the server in standalone mode.

   > **Caution:**
   >
   > If you want to use the (now standalone) standby Key Vault server as a standby in a new HA deployment, you must reinstall the Oracle Key Vault software on the standby server.

## 4.2.6 High Availability Read-Only Restricted Mode

You can configure Oracle Key Vault for high availability restricted mode.

### 4.2.6.1 About High Availability Read-Only Restricted Mode

High Availability Read-Only Restricted mode ensures endpoint operational continuity when primary or standby Oracle Key Vault servers are affected by server, hardware, or network failures.

#### 4.2.6.1.1 How High Availability Read-Only Restricted Mode Works

High Availability Read-Only Restricted mode is supported in Oracle Key Vault 12.2.0.5.0 and later.

In Oracle Key Vault 12.2.0.4.0, when an unplanned shutdown caused the standby server to go offline, the primary server was also unavailable to the endpoints. However, when a planned shutdown caused the standby server to go offline, the primary server was still available to the endpoints.

Oracle Key Vault 12.2.0.5.0 and later support High Availability Read-Only Restricted mode. High Availability Read-Only Restricted mode ensures endpoint operational continuity when the primary or standby Oracle Key Vault servers are affected by server, hardware, or network failures.

When an unplanned shutdown causes the primary or standby server to go offline, the endpoints can still connect to the surviving peer server in order to perform critical operations. High Availability Read-Only Restricted mode ensures that operations that replicate data are blocked. Operations that replicate data are allowed when both primary and standby servers are back online, thus ensuring that no critical data is lost.

In High Availability Oracle Key Vault deployments, the single point of failure is eliminated by replicating the primary server’s data to the standby server. In Read-Only Restricted mode, generation of non-critical data such as audit records is enabled. However, generation of critical data such as keys is disabled. When the primary server is down, operations that generate new critical data on the standby are disabled. The reverse is also true. When the standby server is down, operations that attempt to modify or create any data on the primary server are disabled.

In a High Availability deployment without Read-Only Restricted mode, most endpoint operations are blocked because endpoint operations generate audit records, which is data that needs replication, thus disrupting operational continuity.

The following are the benefits of using Read-Only Restricted mode:

- Allows endpoint operational continuity when the primary or standby server is offline
- Ensures symmetrical behavior when the primary or standby server is offline

### 4.2.6.1.2 High Availability without Read-Only Restricted Mode

When High Availability is configured without Read-Only Restricted mode, the impact on endpoint operations differs, depending on the type of failure encountered: primary failure, standby failure, or a network failure that prevents communication between the primary and standby servers. The following are the possible scenarios:

- **Primary server failure:** The standby server will failover and take over from the affected primary server. This allows the Oracle Key Vault service to remain operational. Data modifications are stored on the primary server until they can be replicated to the standby server. This ensures endpoint operational continuity when the primary server goes offline due to an unplanned shutdown.
- **Standby server failure:** The primary server is unavailable to the endpoints, as it is not possible to distinguish a standby server failure from a network failure that prevents communication between the primary and standby servers.
- **Power loss or network connectivity failure:** The primary and standby servers are unable to communicate. The standby server will failover and take over from the primary server. To avoid a split-brain scenario, only one of the servers is allowed to service the endpoints.

> **Note:**
>
> A split-brain scenario in Oracle Key Vault occurs when the primary server fails, causing the standby server to failover and take over from the primary server. This causes a situation where the primary and standby servers are available to service the endpoints, and create new data. A split-brain scenario causes data on the primary and standby servers to go out of sync. This can lead to data loss and corruption, as well as loss of operational continuity. To avoid a split-brain scenario, only one of the servers is allowed to service the endpoints after a failover occurs.

In High Availability without Read-Only Restricted mode, one of the following situations is triggered when a failure occurs:

- Endpoints suffer a temporary operational disruption to avoid a split-brain scenario.
- The standby server accepts new requests and generates new data without attempting to synchronize the data with the failed primary server. Replication

of data is temporarily disabled until the primary server is online, thus ensuring operational continuity.

#### 4.2.6.1.3 High Availability with Read-Only Restricted Mode

Read-Only Restricted mode is the default High Availability mode in Oracle Key Vault 12.2.0.5.0 and later, and can be disabled during High Availability configuration, if required.

> **Note:**
>
> It is recommended that you configure High Availability with Read-Only Restricted mode enabled, which is the default mode. While configuring High Availability, ensure that **Yes** is selected in the **Allow Read-Only Restricted Mode** field on the **Configure High Availability** page.

Read-Only Restricted mode ensures endpoint operational continuity as well as symmetrical behavior when the primary or standby server is offline. Symmetrical behavior ensures that the online server seamlessly takes over from its failed peer, and continues to service the endpoints without any disruption. For more information about High Availability failover situations with Read-Only Restricted mode, see Failover Situations with Read-Only Restricted Mode (page B-10).

In Read-Only Restricted mode, the surviving Oracle Key Vault server operates with limited functionality. Endpoint operations that add or modify critical data on the Oracle Key Vault server are blocked. However, endpoint operations that involve fetching of data are allowed. This ensures endpoint operational continuity and data integrity. For more information about blocked and allowed operations, see Read-Only Restricted State Functionality (page 4-14).

For more information about Read-Only Restricted state, see States of Read-Only Restricted Mode (page 4-13).

> **Note:**
>
> Read-Only Restricted Mode has no impact on a standalone server.

### 4.2.6.2 States of Read-Only Restricted Mode

A server using read-only restricted mode is affected by the failure in a primary server, a standby server, and the network.

### 4.2.6.3 Recovering from Read-Only Restricted Mode

To recover an instance from Read-Only Restricted mode after a network failure or standby server failure, manual intervention may be required. You will need to unpair and reset the surviving instance, reinstate a new Oracle Key Vault server, and pair it as the new standby to the surviving server. The following are the possible scenarios:

- **Primary server failure:** Depending on the operational state of the primary server at the time of failure, it could be restarted and some functionality may be available. However, due to possible corruption of the embedded key vault database, recovery may not be possible and the Oracle Key Vault instance would need to be reinstated because of the partial failure. If the failed server is unable to re-pair with the peer server within 20 minutes, the server needs to be re-instantiated.

  Even though the endpoint processes communicating with the Oracle Key Vault servers retain the IP address of the last known reachable server, they do have to determine the IP address of the new Key Vault server when spawned. The endpoint processes attempt to communicate with the Oracle Key Vault server configured as the primary server in the configuration scripts, and then waits for a response before trying to reach the server configured as the standby server in the configuration scripts. The wait for the response is an unnecessary delay that every new process incurs when communicating with the standby Oracle Key Vault server. To minimize downtime, it is recommended that you initiate a switchover after reinstating the failed primary server.
- **Standby server failure:** The primary server will run in the Read-Only Restricted mode if there is a standby server failure. Re-instate the standby server if it does not automatically re-pair with the primary server.
- **Power loss or network connectivity failure:** When a network failure occurs, the primary and standby servers are unable to communicate, and both servers enter Read-Only Restricted mode. The standby also attempts to failover to the primary server. Once communication is re-established between the primary and standby servers, the old primary server is automatically converted to the new standby. The data from the new primary server overwrites the old primary server's data, resulting in the loss of audit records from the old primary server. It is recommended that you enable SYSLOG auditing to preserve the audit records that were overwritten on the old primary. Similar to recovering from primary server failure, it is recommended to do a switchover after recovery. It is also recommended that you do not enroll any new endpoints before the switchover.

For more information about restoring High Availability after a failover, see Restoring High Availability After a Failover (page 4-9).

### 4.2.6.4 Read-Only Restricted State Functionality

Read-Only Restricted mode puts the Oracle Key Vault instance into the Read-Only Restricted mode state, but does not put the embedded Oracle Key Vault database into the Read-Only Restricted mode state. Read-Only Restricted mode introduces the following deviations from normal functionality:

- All operations that generate new data are blocked. Operations that fetch existing data are allowed. Audit records for endpoint operations are generated as in normal operation. Internal system operations of the Oracle Key Vault database are not impacted. Functionality such as alerts continue to work normally.
- Endpoints are allowed to fetch keys from the Oracle Key Vault server. Endpoints cannot create new keys or modify existing keys.
- Administrators can log in to the Oracle Key Vault Management Console. Creation of an endpoint or a wallet, deletion of keys, and operations that modify or delete data are blocked.
- Unpairing of primary and standby Oracle Key Vault servers running in Read-Only Restricted mode are allowed.

- Backup operations are blocked to avoid data mismatches between backups.

**Table 4-1 Allowed and Blocked Operations in Read-Only Restricted Mode**

| Operation | Allowed or Blocked |
|---|---|
| Log in to Oracle Key Vault | Allowed |
| Endpoint operations such as fetching keys from the cache | Allowed |
| Endpoint operations that add, modify, or delete data such as rotation of keys on the database | Blocked |
| System operations such as enabling SSH access | Allowed |
| System operations that write data such as setting up a REST server and creating virtual wallets | Blocked |
| Oracle Key Vault Management Console access | Allowed |
| All Administrator and endpoint operations that add new data or modify existing data | Blocked |
| Backup operations | Blocked |

In Read-Only Restricted mode, if you attempt to execute operations that generate new data or modify existing data on the Oracle Key Vault server, the **Key Vault Server in Read-Only Restricted Mode** error is displayed.

If you attempt to upload a wallet to the Java keystore, you are prompted for the source Java keystore password. After entering the password, the **Key Vault Server in Read-Only Restricted Mode** error is displayed.

### 4.2.6.5 Enabling Read-Only Restricted Mode

Read-Only Restricted mode is enabled by default when High Availability is configured.

> **Caution:**
>
> It is recommended that you configure High Availability with Read-Only Restricted mode enabled.

If Read-Only Restricted mode is disabled, you must perform the following steps to enable it:

1. Unpair the primary server from the standby server, and reinstall Oracle Key Vault on the standby server. For more information about installing the Oracle Key Vault appliance software, see Installing the Oracle Key Vault Appliance Software (page 3-5).
2. Perform post-installation tasks on the standby server. For more information about performing post-installation tasks, see Performing Post-Installation Tasks (page 3-10).
3. Configure High Availability on the standby server. On the **Configure High Availability** page, ensure that **Yes** is selected in the **Allow Read-Only Restricted Mode** field.

For more information about configuring High Availability, see Configuring High Availability (page 4-3).

4. Log in to the primary server as the System Administrator, and on the **Configure High Availability** page, ensure that **Yes** is selected in the **Allow Read-Only Restricted Mode** field.
5. Click **Initiate Pairing**.

   Read-Only Restricted mode is enabled on the High Availability deployment.

Read-Only Restricted mode takes effect if connectivity is lost between the primary and standby servers.

> **Note:**
>
> Read-Only Restricted mode has no effect on a standalone server.

## 4.2.6.6 Disabling Read-Only Restricted Mode

Read-Only Restricted mode is enabled by default when High Availability is configured.

> **Caution:**
>
> It is recommended that you configure High Availability with Read-Only Restricted mode enabled. Do not disable Read-Only Restricted mode while configuring High Availability, unless it is necessary.

To disable Read-Only Restricted mode, you must perform the following steps:

1. Unpair the primary server from the standby server, and reinstall Oracle Key Vault on the standby server. For more information about installing the Oracle Key Vault appliance software, see Installing the Oracle Key Vault Appliance Software (page 3-5).
2. Perform post-installation tasks on the standby server. For more information about performing post-installation tasks, see Performing Post-Installation Tasks (page 3-10).
3. Configure High Availability on the standby server. On the **Configure High Availability** page, ensure that **No** is selected in the **Allow Read-Only Restricted Mode** field.

   For more information about configuring High Availability, see Configuring High Availability (page 4-3).
4. Log in to the primary server as the System Administrator, and on the **Configure High Availability** page, ensure that **No** is selected in the **Allow Read-Only Restricted Mode** field.
5. Click **Initiate Pairing**.

   Read-Only Restricted mode is disabled on the High Availability deployment.

Read-Only Restricted mode is disabled, and will not take effect if connectivity is lost between the primary and standby servers.

**Note:**

Read-Only Restricted mode has no effect on a standalone server.

### 4.2.6.7 Best Practices

The following are best practices to ensure operational continuity and minimal downtime of Oracle Key Vault:

- Configure auto-login for Hardware Security Module (HSM) on TDE-enabled endpoint databases. For more information about configuring auto-login for Hardware Security Module (HSM), see “Configuring Auto-Login Hardware Security Modules” in the *Oracle Database Advanced Security Guide*.
- Apply the database patch for Bug 22734547 to tune the Oracle Key Vault heartbeat.
- Ensure the Read-Only Restricted mode is enabled in High Availability Oracle Key Vault deployments.
- Set the duration in the **Fast Start Failover Threshold** field on the **Configure High Availability** page to a value that avoids unnecessary failover due to transient network interruptions.
- Configure Syslog auditing to capture audit records in Read-Only Restricted mode.
- Switch over to the original primary server in case the primary server is reinstated.

## 4.3 Backup and Restore Oracle Key Vault Data

Oracle Key Vault provides the facility to backup and restore Key Vault data for disaster recovery purposes.

It is highly recommended to back up data periodically to reduce down time and recover from unexpected data losses and system failures. You can restore a new or existing Oracle Key Vault appliance from a backup.

### 4.3.1 About Backing Up and Restoring Data in Key Vault

Key Vault provides the facility to backup and restore security objects stored and managed in Key Vault for disaster recovery purposes.

Backup and restore operations may be performed from the Key Vault management console. You must be a user with system administrative privileges to back up and restore Key Vault data. Backups may be scheduled at periodic intervals to run automatically at designated times. They may also be run on-demand to save a current snapshot of the system.

It is highly recommended that you back up Key Vault data regularly on a schedule. This practice ensures that backups are current and hold the most recent data. The backup can be used to restore a new or existing Key Vault server and be fully operational with minimum downtime and data loss.

Key Vault encrypts all backed up data, which is copied to the backup destination using the secure copy protocol (SCP). You must therefore ensure that SCP is supported at the backup destination.

## 4.3.2 Backing Up Key Vault Data

The first step to backing up data is to create a backup destination, which is the location where Key Vault data will be copied to and stored.

The primary reason for adding a backup destination is to have the backup data available in a location other than the Key Vault server itself. This ensures that you have all the relevant data to recover in case of a catastrophic failure with the Key Vault server or hardware.

The backup destination is usually another server or computer system that you have access to. You can add, delete, and modify a backup destination.

- About Key Vault Backup Destinations (page 4-18)
- Create a Remote Backup Destination (page 4-19)
- Change Settings on a Remote Backup Destination (page 4-21)
- Delete a Remote Backup Destination (page 4-22)

### 4.3.2.1 About Key Vault Backup Destinations

The backup operation copies Oracle Key Vault data to a backup destination of your choice. The backup destination stores the data until it is needed.

Key Vault provides two types of backup destinations: local and remote. The local backup destination resides on the Key Vault server itself, the remote one resides externally in a different server or computer system. You can create more than one backup destination for greater availability.

Backup destinations may be:

- **Local**

  The local backup destination, `LOCAL`, is present out of the box and cannot be removed.

  Backups to `LOCAL` are useful to save a current state of Key Vault. Since these backups are stored in Key Vault, they will be lost in case of a failover or switchover in a high availability deployment. It is therefore highly recommended that you back up the data to a remote destination before you perform operations like failover and switchover.

  A `LOCAL` destination can store only the last full backup and the cumulative incremental backups after that full backup. After a new full backup of the periodic backup to `LOCAL` completes, the previous periodic full or cumulative incremental backups are deleted. For more information about backups, see Two Types of Key Vault Backups (page 4-23).

- **Remote**

  Remote backup destinations reside on external servers and can be dispersed geographically for disaster recovery purposes.

  Each backup destination on the external server is associated with a backup catalog file called `okvbackup.mgr` that Key Vault maintains at the backup destination. The file `okvbackup.mgr` catalogs the backups performed and is used to restore data.

  > **Note:**
  >
  > You cannot use another Key Vault appliance as a remote backup destination.

  > **Caution:**
  >
  > - Oracle Key Vault may not be able to find the backups if you delete or modify the backup catalog file. Therefore do not delete or modify this file.
  > - Do not configure the same remote backup destination directory for different Key Vault servers as backup destinations, because backups that happen concurrently from different Key Vault servers will overwrite each other's catalog file, with the result that Key Vault may not be able to locate the backups correctly.

### 4.3.2.2 Create a Remote Backup Destination

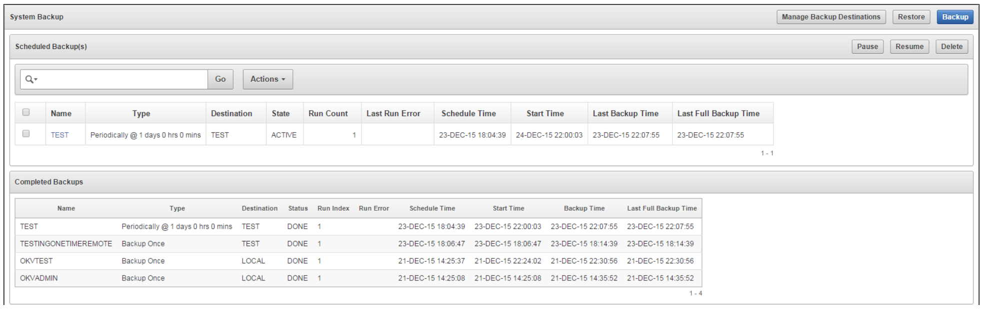
To create a remote backup destination, you must provide a user account, a unique existing directory location on an external server, and an authentication method (password or key-based). Oracle Key Vault needs this information to make a secure connection with the remote server.

To create a remote backup destination:

1. Log in to the Oracle Key Vault management console as a user with the System Administrator role.
2. Select the **System** tab, then click **System Backup** from the left sidebar.

   The **System Backup** page appears. It lists the scheduled backups and details of the last 10 backups performed.

**Figure 4-6    System Backup Page**



| □ | Name | Type | Destination | State | Run Count | Last Run Error | Schedule Time | Start Time | Last Backup Time | Last Full Backup Time |
|---|---|---|---|---|---|---|---|---|---|---|
| □ | TEST | Periodically @ 1 days 0 hrs 0 mins | TEST | ACTIVE | 1 | | 23-DEC-15 18:04:39 | 24-DEC-15 22:00:03 | 23-DEC-15 22:07:55 | 23-DEC-15 22:07:55 |

1 - 1

Completed Backups

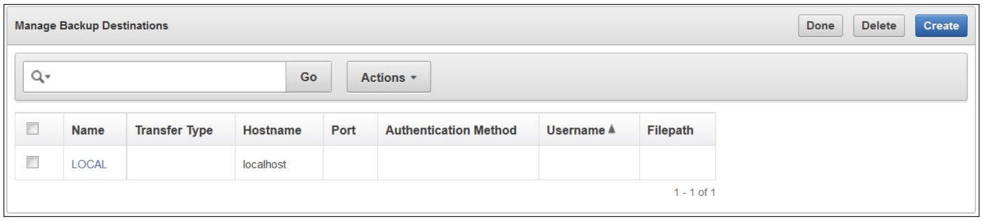
| Name | Type | Destination | Status | Run Index | Run Error | Schedule Time | Start Time | Backup Time | Last Full Backup Time |
|---|---|---|---|---|---|---|---|---|---|
| TEST | Periodically @ 1 days 0 hrs 0 mins | TEST | DONE | 1 | | 23-DEC-15 18:04:39 | 23-DEC-15 22:00:03 | 23-DEC-15 22:07:55 | 23-DEC-15 22:07:55 |
| TESTINGONETIMEREMOTE | Backup Once | TEST | DONE | 1 | | 23-DEC-15 18:06:47 | 23-DEC-15 18:06:47 | 23-DEC-15 18:14:39 | 23-DEC-15 18:14:39 |
| OKVTEST | Backup Once | LOCAL | DONE | 1 | | 21-DEC-15 14:25:37 | 21-DEC-15 22:24:02 | 21-DEC-15 22:30:56 | 21-DEC-15 22:30:56 |
| OKVADMIN | Backup Once | LOCAL | DONE | 1 | | 21-DEC-15 14:25:08 | 21-DEC-15 14:25:08 | 21-DEC-15 14:35:52 | 21-DEC-15 14:35:52 |

1 - 4

3. Click **Manage Backup Destinations**.

   The **Manage Backup Destinations** page displays the local backup destination that comes with Key Vault, and any remote destinations, if configured.

**Figure 4-7    Manage Backup Destinations**



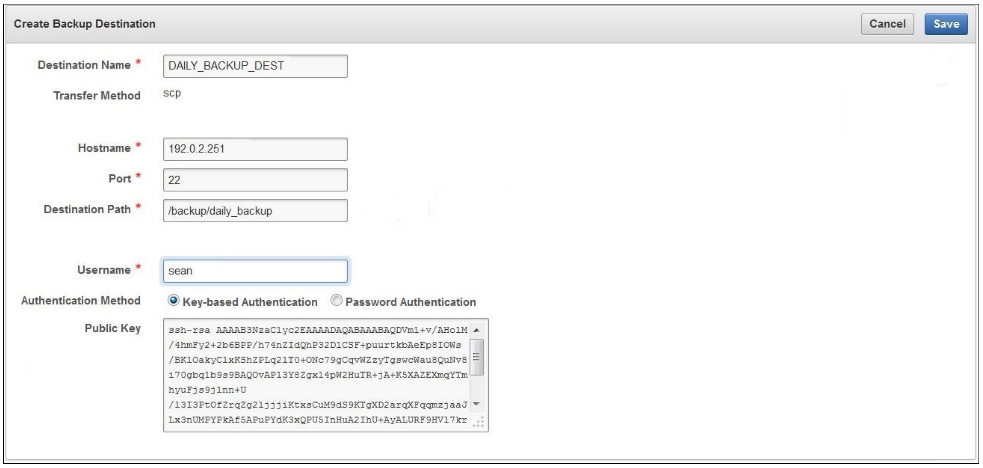
| □ | Name | Transfer Type | Hostname | Port | Authentication Method | Username ▲ | Filepath |
|---|---|---|---|---|---|---|---|
| □ | LOCAL | | localhost | | | | |

1 - 1 of 1

4. Click **Create**.

   The **Create Backup Destination** page appears.

**Figure 4-8    Create Backup Destination**



| | |
|---|---|
| Destination Name * | DAILY_BACKUP_DEST |
| Transfer Method | scp |
| Hostname * | 192.0.2.251 |
| Port * | 22 |
| Destination Path * | /backup/daily_backup |
| Username * | sean |
| Authentication Method | ● Key-based Authentication ○ Password Authentication |

Public Key

```
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQDVm1+v/AHo1M
/4hmFy2+2b6BPP/h74nZIdQhP32D1CSF+puurtkbAeEp8IOWs
/BK1OakyClxKShZPLq21T0+ONc79gCqvWZzyTgswcWau8QuNv8
i70gbq1b9s9BAQOvAP13Y8Zgx14pW2HuTR+jA+K5XAZEXmqYTm
hyuFjs9j1nn+U
/13I3PtOfZrqZg21jjjiKtxsCuM9dS9KTgXD2arqXFqqmzjaaJ
Lx3nUMPYPkAf5APuPYdK3xQPU5InHuA2IhU+AyALURF9HV17kr
```

5. Enter the following information for the backup location:

> ⚠ **Caution:**
>
> The username, hostname or the destination directory paths for remote backup destinations should not have a space, single quotes or double quotes in them.

- **Destination Name:** Enter a descriptive name to identify the backup destination.
- **Transfer Method:** This is automatically populated with the value `scp` for the SCP protocol that is used to copy files to the remote destination.
- **Hostname:** Enter the hostname or IP address of the remote server for the backup. If you enter the host name, ensure that DNS is configured to translate the host name to its corresponding IP address.
- **Port:** Enter the SCP port number on the external server. The default is 22.
- **Destination Path:** Enter the path to an existing directory on the external server, where the backup file will be copied. This directory location is cannot be modified after the backup destination is created.This path should not be the destination for backups from another Oracle Key Vault appliance.
- **Username:** Enter the username of the user account on the remote server. Ensure that write permissions are set on the directory specified in **Destination Path** for the user identity that establishes the SCP connection.
- **Authentication Method:** Choose one of the following:
  - **Password Authentication**

    The password of the user account entered in the **Username** field.
  - **Key-based Authentication**

    Copy the public key that appears and paste it in the appropriate configuration file, such as `authorized_keys`, on the destination server. Check that the permissions of the configuration file are set to allow access only to the backup account owner and no other group or user.

6. Click **Save**.

   Oracle Key Vault validates the input supplied to create the backup destination. If the validation fails, the backup destination is not created. If this happens, recheck values for the user account on the remote server (username and password or key) and ensure that the directory has write permissions for the user. Finally, ensure that the remote server is up and running.

## 4.3.2.3 Change Settings on a Remote Backup Destination

Once the backup destination is created, you can only change the SCP port number and details of the user account. You may not change any other setting.

To change allowable settings of a backup destination:

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 click **System Backup**.

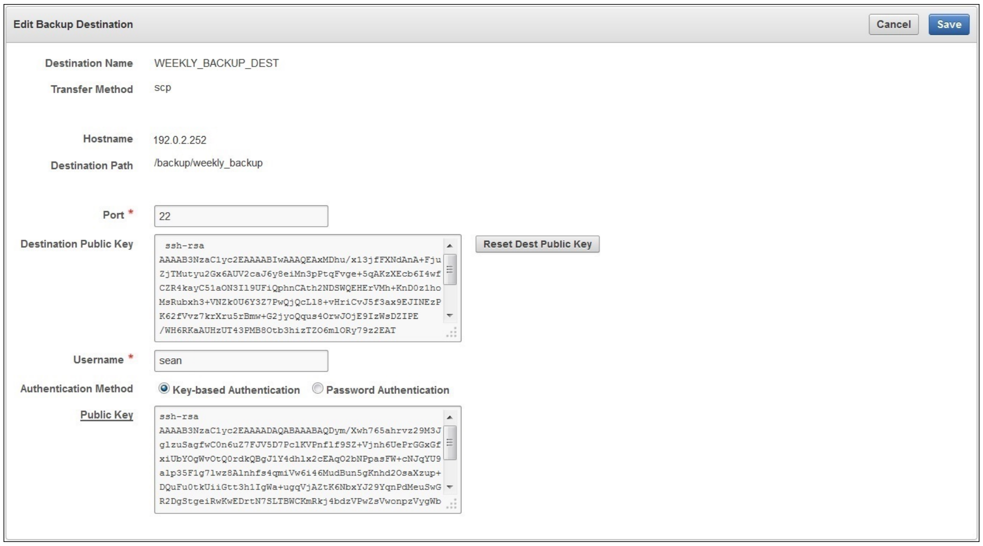
3. Select **Manage Backup Destinations**.

   The **Manage Backup Destinations** page appears displaying LOCAL and remote backup destinations.

4. Click the backup destination name to edit it. The **Edit Backup Destination** page appears.

5. Modify the following information:

   **Figure 4-9 Edit Backup Destination**

   Destination Name WEEKLY_BACKUP_DEST

   Transfer Method scp

   Hostname 192.0.2.252

   Destination Path /backup/weekly_backup

   Port * 22

   Destination Public Key

   ```
    ssh-rsa
   AAAAB3NzaC1yc2EAAAABIwAAAQEAxMDhu/x13jfFXNdAnA+Fju
   ZjTMutyu2Gx6AUV2caJ6y8eiMn3pPtqFvge+5qAKzXEcb6I4wf
   CZR4kayC51aON3I19UFiQphnCAth2NDSWQEHErVMh+KnD0z1ho
   MsRubxh3+VNZk0U6Y3Z7PwQjQcL18+vHriCvJ5f3ax9EJINEzP
   K62fVvz7krXru5rBmw+G2jyoQqus4OrwJOjE9IzWsDZIPE
   /WH6RKaAUHzUT43PMB8Otb3hizTZO6ml0Ry79z2EAT
   ```

   Username * sean

   Authentication Method ● Key-based Authentication ○ Password Authentication

   Public Key

   ```
   ssh-rsa
   AAAAB3NzaC1yc2EAAAADAQABAAABAQDym/Xwh765ahrvz29M3J
   glzuSagfwC0n6uZ7FJV5D7PclKVPnflf9SZ+Vjnh6UePrGGxGf
   xiUbYOgWvOtQ0rdkQBgJ1Y4dhlx2cEAqO2bNPpasFW+cNJqYU9
   alp35F1g7lwz8Alnhfs4qmiVw6i46MudBun5gKnhd2OsaXzup+
   DQuFu0tkUiiGtt3h1IgWa+ugqVjAZtK6NbxYJ29YqnPdMeuSwG
   R2DgStgeiRwKwEDrtN7SLTBWCKmRkj4bdzVPwZsVwonpzVygWb
   ```

   - **Port:** Change the default port number running SCP on the external server.
   - **Username:** Enter the username of the user account on the remote server. Ensure that the new user has write permissions on the directory specified in **Destination Path** since this path may not be changed.
   - **Authentication Method:** Choose one of the following:
     - **Password Authentication**

       The password of the user account entered in the **Username** field.
     - **Key-based Authentication**

       Copy the public key that appears and paste it in the appropriate configuration file, such as `authorized_keys`, on the destination server.

6. Click **Save**.

   Key Vault validates the input supplied to update the backup destination. If the validation fails, the backup destination is not created. If this happens, recheck values for the user account on the remote server (username and password) and ensure that the directory has write permissions for the user. Finally, ensure that the remote server is up and running.

## 4.3.2.4 Delete a Remote Backup Destination

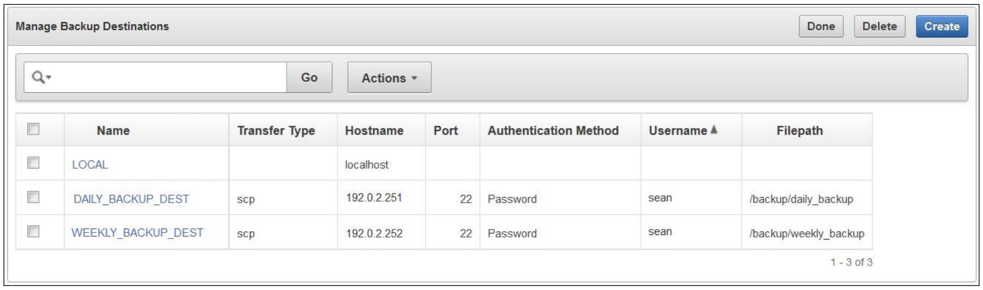
You can delete a remote backup destination to stop future backups to that destination server. Backups already on the destination server will remain there.

To delete a remote backup destination:

1. Log in to the Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **System** tab, and then click **System Backup**.
3. Select **Manage Backup Destinations**.

   The **Manage Backup Destinations** page appears displaying `LOCAL` and remote backup destinations.

**Figure 4-10 Manage Backup Destinations Page**



| □ | Name | Transfer Type | Hostname | Port | Authentication Method | Username ▲ | Filepath |
|---|---|---|---|---|---|---|---|
| □ | LOCAL | | localhost | | | | |
| □ | DAILY_BACKUP_DEST | scp | 192.0.2.251 | 22 | Password | sean | /backup/daily_backup |
| □ | WEEKLY_BACKUP_DEST | scp | 192.0.2.252 | 22 | Password | sean | /backup/weekly_backup |

1 - 3 of 3

4. Check the box(es) for the backup destinations you want to delete.
5. Click **Delete**.

## 4.3.3 About Backup Schedule Types and States

You can schedule backups in Key Vault for specific times and backup destinations. The backup process starts at the scheduled time and generates a system backup, which is a file that is stored on the backup destination. There is one backup file for each completed backup.

No backup can start if another backup is in progress. You can change the schedule of backups as needs change. You can continue working with Key Vault while the backup is in progress.

A system reboot will terminate any ongoing backup. If you must reboot the system, you can cancel a backup that is scheduled to happen at the same time, and backup the system after the reboot.

- Two Types of Key Vault Backups (page 4-23)
- Scheduled Backup States in Key Vault (page 4-24)

### 4.3.3.1 Two Types of Key Vault Backups

Oracle Key Vault provides two types of backups that can be scheduled:

1. One-time backup

   A one-time backup makes a full backup of the Key Vault system. More than one such one-time backup can be scheduled together.

   One-time local backups should be taken before making significant configuration changes to Key Vault, in case you need to recover from configuration failures.

`LOCAL` destinations can only store the last one-time backup. When a one-time backup to `LOCAL` completes, the previous backup is deleted.

2. Periodic backup

   A periodic backup makes a backup at regular intervals at a specified frequency. The process first makes a full backup of the Key Vault system and puts it in active state. For more information about backup states, see Scheduled Backup States in Key Vault (page 4-24). At the end of the subsequent periodic interval, a cumulative incremental backup starts. This cumulative incremental backup holds changes from the last full backup. Another full backup is made after 7 days have passed since the last full backup.

   For example, if the backup period is once a day, then every seventh one is a full backup. If the backup period is every 8 days, then all backups are full backups. If the backup period is 12 hours, then there are 13 cumulative backups before a full backup.

   Periodic backups should be scheduled with a period of at least one day to minimize data loss.

   A `LOCAL` destination can store only the last full backup and the cumulative incremental backups after that full backup. After a new full backup of the periodic backup to `LOCAL` completes, previous periodic full or cumulative incremental backups are deleted.

   Cumulative incremental backups are faster than full backups. Only one periodic backup can be scheduled at any time.

## 4.3.3.2 Scheduled Backup States in Key Vault

Scheduled backups have four states, which indicate whether the backup is scheduled, in progress, completed, or paused:

1. ACTIVE

   The backup is scheduled and will be processed at the specified start time or period.

2. ONGOING

   The backup is in progress.

3. DONE

   The backup is complete.

4. PAUSED

   All future backups are on hold and will not start even if the start time has passed. They will start when they are explicitly resumed.

   You can change the state from active to paused and back. Put a scheduled backup in the paused state for these situations:

   - When communication between Key Vault and the remote destination is broken
   - If the remote destination is down
   - If you want to defer the backup

   You can delete the scheduled backups that have not completed.

## 4.3.4 Scheduling and Managing Key Vault Backups

You can schedule Key Vault backup(s) to specific backup destination(s) and time(s). Note, that you must create the backup destinations that you will use prior to this step. Backup schedules may be modified or deleted to accommodate changes.

- Schedule a Backup on Key Vault (page 4-25)
- Change Backup Schedule on Oracle Key Vault (page 4-26)
- Delete a Backup Schedule from Key Vault (page 4-26)
- How High Availability Affects Key Vault Backups (page 4-27)
- Protecting the Backup Using the Recovery Passphrase (page 4-27)

### 4.3.4.1 Schedule a Backup on Key Vault

You can also schedule a one-time or periodic backup to a local or remote backup destination. You can start a one-time backup to start immediately without setting a time.

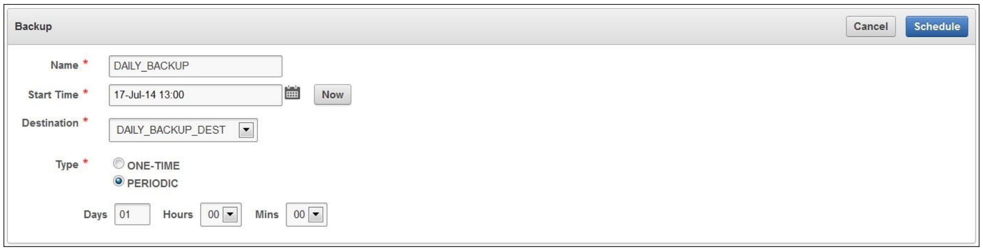
To schedule a backup:

1. Log in to the Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **System** tab, and then **System Backup** from the left sidebar.

   The **System Backup** page appears.
3. Click **Backup**.

   The **Backup** page appears.
4. In the **Name** field, enter a name for the backup.
5. Click the Calendar icon and select the **Start Time**.
6. Select the **Destination** for the backup from the list.
7. Select the **Type**: **ONE-TIME** or **PERIODIC**.

**Figure 4-11 Backup Page**



Name * DAILY_BACKUP

Start Time * 17-Jul-14 13:00

Destination * DAILY_BACKUP_DEST

Type * ○ ONE-TIME ● PERIODIC

Days 01 Hours 00 Mins 00

The **Backup** page is the same for one-time or periodic backups. In case of a periodic backup additional fields for **Days**, **Hours** and **Mins** appear. You must enter values for these fields.

8. Click **Schedule**.

The **System Backup** page appears listing the scheduled backup in **Scheduled Backup(s)**.

To start an immediate one-time backup:

1. Enter a name in the **Name** field for the backup on the **Backup** page.
2. Select the type of backup as **ONE-TIME**.
3. Select the **Destination** for the backup from the list.
4. Click **Now**.
5. Click **Schedule**.

   The **System Backup** page appears listing the scheduled backup in **Scheduled Backup(s)**.

### 4.3.4.2 Change Backup Schedule on Oracle Key Vault

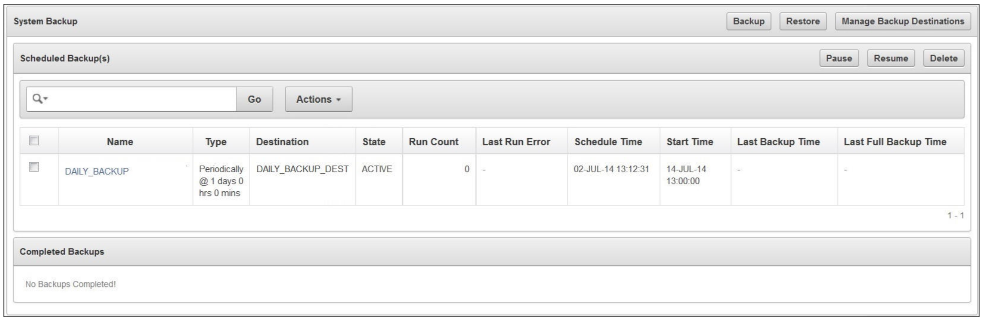
You can not change the schedule of a backup in progress. To change the backup schedule the state must be active or paused.

To edit a scheduled Key Vault backup:

1. Log in to the Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **System** tab, and then select **System Backup**.
3. Click the **Name** of the scheduled backup in **Scheduled Backup(s)**.

   The **Backup** page appears.

**Figure 4-12 System Backup Page**



| □ | Name | Type | Destination | State | Run Count | Last Run Error | Schedule Time | Start Time | Last Backup Time | Last Full Backup Time |
|---|---|---|---|---|---|---|---|---|---|---|
| □ | DAILY_BACKUP | Periodically @ 1 days 0 hrs 0 mins | DAILY_BACKUP_DEST | ACTIVE | 0 | - | 02-JUL-14 13:12:31 | 14-JUL-14 13:00:00 | - | - |

1 - 1

Completed Backups

No Backups Completed!

4. Enter the **Start Time** or click the calender icon for a one-time backup.

   Note, that for a one-time backup you can only change the start time if the backup has not already started. This means that the state cannot be ongoing or done. For a periodic backup you can change the start time if the scheduled start time has not passed.
5. Enter the **Days**, **Hours**, and **Mins** for a periodic backup.
6. Select **Save** to save the changes.

### 4.3.4.3 Delete a Backup Schedule from Key Vault

To delete an Oracle Key Vault scheduled backup:

1. Log in to the Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **System** tab, and then **System Backup** from the left sidebar.
3. Check the box(es) of scheduled backup(s) listed in **Scheduled Backup(s)**.
4. Click **Delete** to delete the scheduled backups.

### 4.3.4.4 How High Availability Affects Key Vault Backups

It is important to note that backups are performed on the primary server in a high availability deployment. Since the standby synchronizes its state with the primary, it is not necessary to backup the standby.

Points to consider during failover or switchover in a high availability deployment would be:

- Any backups in progress will terminate if there is a failover or a high availability switchover. Backups to `LOCAL` are private to the Oracle Key Vault appliance and therefore the local backup on the primary appliance is not available after a failover or switchover.
- Backups scheduled with password authentication start as usual after the failover or switchover.
- Remote backups using key-based authentication will need to update the public key on the destination to match the one shown on the new primary system.

### 4.3.4.5 Protecting the Backup Using the Recovery Passphrase

Oracle Key Vault uses the recovery passphrase to control who can restore user and system data.

To restore a backup, use the Oracle Key Vault recovery passphrase from the time when the backup was initiated. This is necessary even if the recovery passphrase was changed after the backup completed. Oracle recommends that you make a new backup every time the recovery passphrase is changed to ensure that there is always a copy of the backup that is protected by the most recent recovery passphrase.

> **See Also:**
>
> "Emergency System Recovery Process (page 2-10)" for information on the recovery passphrase and how it is used

## 4.3.5 Restoring Oracle Key Vault Data

You can restore Key Vault data from a remote backup destination onto a new or existing Key Vault server to minimize downtime and data loss. The restore process replaces all the data on the new server except the root and support user passwords. You will not be able to restore data to a server if there is a scheduled backup in process on the server.

> **Note:**
>
> You must restore Key Vault data to a server only after ensuring that all scheduled backups on the server are completed.

- About the Key Vault Restore Process (page 4-28)
- Restore Key Vault Data (page 4-28)
- High Availability and the Restore Operation (page 4-29)
- Third-Party Certificates and the Restore Operation (page 4-29)
- Changes Resulting from a System State Restore (page 4-30)

### 4.3.5.1 About the Key Vault Restore Process

Restoring data to a Key Vault server replaces the data in the server with that of the backup. Any changes made since the last backup will be lost.

The maximum life of a backup is 1 year. Any backup older than a year cannot be restored.

You must have the recovery passphrase that was in effect at the time of the backup in order to restore data from a backup. If you have not changed the recovery passphrase since installing Key Vault, then you must use the recovery passphrase that you created during the post-installation process.

There are two steps to restoring data in Key Vault: Setup and Restore.

1. Setup consists of:
   a. Installing the Oracle Key Vault appliance.
   b. Setting up the backup destinations.
2. Restore the Oracle Key Vault appliance by:
   a. Determining the backup to use from a remote or local backup destination.
   b. Providing the recovery passphrase to begin the restore process.

> **Note:**
>
> The recovery passphrase was created in Performing Post-Installation Tasks (page 3-10)

### 4.3.5.2 Restore Key Vault Data

Before you restore ensure that you have the correct recovery passphrase. You will need to enter it during the restore process.

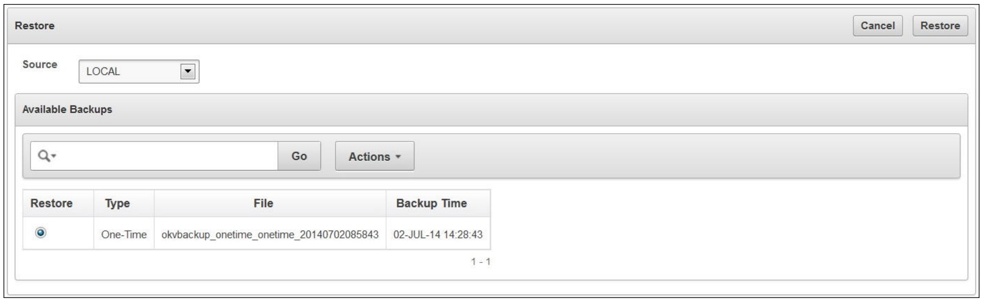
To restore data from a backup:

1. Log in to the Oracle Key Vault management console as a user who has the System Administrator role.

2. Select the **System** tab, and then **System Backup**.
3. Click **Restore**.

   The **Restore** page appears.
4. Select **Source** from the drop-down list. Values are either **LOCAL** or **Remote**.
5. Select **Restore** next to the backup you want to restore from.

**Figure 4-13 Restore Page**



| Restore | Type | File | Backup Time |
|---|---|---|---|
| ● | One-Time | okvbackup_onetime_onetime_20140702085843 | 02-JUL-14 14:28:43 |

6. Click **Restore** to initiate the restore or recovery process.

   You are prompted for the recovery passphrase.
7. Enter the recovery passphrase and then click **Restore** to begin.

   The system will restore from the backup and then reboot.

## 4.3.5.3 High Availability and the Restore Operation

In a high availability deployment you must consider the following points while restoring data to Key Vault:

1. Restore only if both primary and standby are lost.
2. You must restore the backup on a standalone Key Vault appliance only, even if the backup was taken from the primary.
3. The restore operation replaces the Key Vault appliance with the backup. This means that some data can be lost. You might need to restore the endpoint database.
4. If you restore a backup taken from the primary node, then you must discard (or reinstall) the standby server and configure a new standby.
5. If the standby server has taken over as primary, then there is no need to restore data from a backup to the new standby server. Just configure a new standby server and it automatically synchronizes with the functioning primary.

## 4.3.5.4 Third-Party Certificates and the Restore Operation

A third-party certificate installed at the time of a backup will not be copied when you restore another appliance from this backup. You will have to re-install the third-party certificate on the new appliance in order to use it.

**See Also:**

" Third-party certificates in Key Vault (page 13-4)"

### 4.3.5.5 Changes Resulting from a System State Restore

Restoring an Oracle Key Vault appliance brings the system state back to the time when the backup was created.

Therefore, any changes made after the backup was made do not exist on the restored system. For example, if a user's password was changed after the backup operation, the new password will not be available in the restored system. The restored system will have the password that was in effect when the backup was made.

**Note:**

Restoring also changes the recovery passphrase to the one that was in effect during the backup.

You should change the user passwords, enroll the endpoints created after backup, and make other similar changes, if required. You should confirm that everything is configured correctly after restoring.

If you are not certain that you restored the correct backup, then you can restore a different one. To restore another backup, first configure the remote destination of this backup on the restored Oracle Key Vault itself, and then start the restore process. You do not need to reinstall the Oracle Key Vault appliance.

When the appliance has been restored and functional, you can continue to backup Key Vault data to new or previous remote destinations.

Depending on the age of your backup, the restored server may be missing endpoints, security objects, and other changes made after the restored backup was taken. You may need to enroll missing endpoints and upload missing security objects, or choose a more recent backup to restore. It is also recommended that you change user passwords after a restore operation.

## 4.3.6 Backup and Restore Best Practices

The following best practices will help you to keep backups current so that you can recover from catastrophic failures with minimum down time and data loss:

1. Ensure that the recovery passphrase at the time of backup is accessible because you will need it to restore data from a backup.
2. Backup data any time you change the recovery passphrase.
3. Ensure that you create at least one remote backup destination in a high availability deployment. Since the local backup resides on the Key Vault server itself, it will be lost in a failover or switchover situation.

4. Do not delete the backup catalog file associated with a remote backup destination, even if you stop using the backup destination. If you ever need to restore from a backup on this server, you will need the backup catalog file.
5. If you use the same remote server for multiple backup destinations, ensure that the directories are unique so that you have distinct backup catalog files associated with each backup destination. If you fail to do this, the backup catalog file will get overwritten during subsequent backups and become unusable.
6. Before you restore data ensure that all scheduled backups are complete.
7. To create remote backup destinations successfully:
   a. Ensure that the servers used as remote backup destinations are up.
   b. Ensure connectivity between Key Vault and remote server you plan to use as a backup destination.
   c. Ensure that the remote server designated as a backup destination supports the secure copy protocol (SCP).
   d. Validate user account credentials on remote server before you create the backup destination on Key Vault.
   e. Ensure that the destination directory has write permissions.
   f. Create more than one remote backup destination on multiple servers for redundancy.
   g. Ensure that destination directories are unique if you are using the same remote server for multiple backup destinations. You must do this to prevent later backups from overwriting previous ones.
8. Schedule a periodic backup with a period of one day. This ensures that you have a full backup once in seven days.
9. Perform a one-time backup once every seven days.
10. Perform a local one-time backup before system changes. You can use this backup as a restore point.
11. Backup before and after upgrading Key Vault server software.
12. Change the backup destination after each upgrade. If at all possible do not reuse the backup destination.

5

# Managing Oracle Key Vault Users

Oracle Key Vault users administer the system, enroll endpoints, manage users and endpoints, control access to security objects, and grant other users administrative roles as needed.

## 5.1 About Oracle Key Vault Users

Key Vault users fulfill multiple functions. A key function is to register and enroll Key Vault endpoints, who can manage their security objects using Key Vault.

### 5.1.1 Types of Oracle Key Vault Users

There are two types of Oracle Key Vault users:

- Administrative users who have one or more of the three administrative roles: System Administrator, Key Administrator, Audit Manager.
- Ordinary users who have none of the administrative roles, but who have access to security objects.

Separation of duties in Key Vault means that users with an administrative role have access to functions pertaining to their role but not the others. For example, only the system administrator sees the **System** tab, not the key administrator or the audit manager. Likewise, the system administrator can add endpoints, but cannot create endpoint groups. The user interface elements needed to create endpoint groups are visible only to the key administrator.

Users who have no administrative role can be granted access to security objects specific to their function, thus restricting their privileges. For example, you can grant a user access to a specific virtual wallet. This user can log into the Key Vault management console and add, delete, and manage his own security objects, but he cannot see system menus, details of other users and endpoints, their wallets, or audit reports.

Although the separation of user duties is recommended, organizations may opt to have a single user perform all the administrative functions by granting that user all the administrative roles.

An Oracle Key Vault user name cannot be the same as an Oracle Key Vault endpoint name.

## 5.1.2 Create an Oracle Key Vault User

A user with the System Administrator role can create user accounts from the Oracle Key Vault management console.

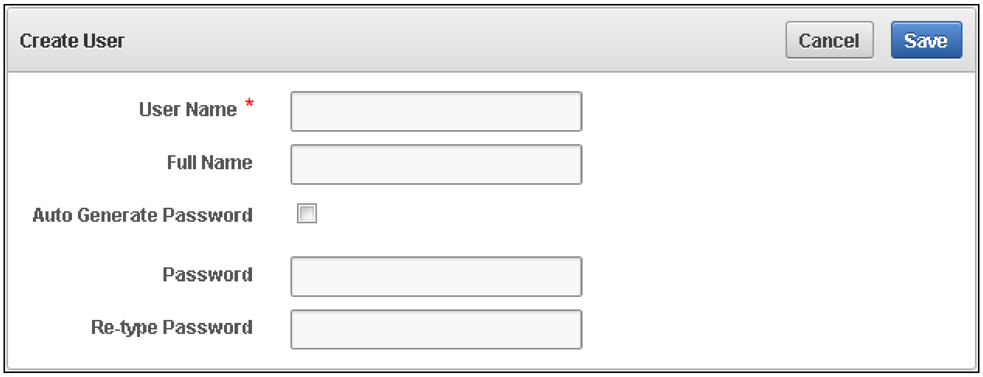
To add a user to Key Vault follow these steps:

1. Log in to the Oracle Key Vault management console as a user who has the System Administrator role.
2. Click the **Users** tab.

   The **Manage Users** page appears with a list of existing users.
3. Click **Create**.

   The **Create User** page appears.

   **Figure 5-1 Create User**

4. Enter a user name in **User Name**. You must ensure that the user name is not the same as an Oracle Key Vault endpoint name.
5. Optionally, add the user's full name in **Full Name**.
6. For the password, do *one* of the following:
   - **Auto Generate Password:** Select this option to have a password automatically generated and sent to the user. The user receives a message with `Oracle Key Vault: System Generated User Password` in the subject line. When the user logs in to the Oracle Key Vault management console for the first time, he will be asked to change the password.

     Note, that the SMTP server configuration must be set up to use this option.
   - **Password and Re-type password:** Enter a valid password. Passwords must have 8 or more characters and contain at least one of each of the following: an uppercase letter, lowercase letter, number, and special characters. The special

characters allowed are period (.), comma (,), underscore (_), plus sign (+), colon (:), and space.

7. Click **Save**.

   The **Manage Users** page appears and lists the new user.

> **See Also:**
>
> "Email Notification (page 12-8)"
>
> "Grant, Change or Revoke Administrative Roles (page 5-3)"

## 5.1.3 Grant, Change or Revoke Administrative Roles

You can grant or change an administrative role to a user you have added. You must be a user with the administrative role to grant it to other users. You can also revoke the administrative role when it is no longer needed.

- Grant or Change an Administrative Role of a User (page 5-3)
- Revoke an Administrative Role from a User (page 5-4)

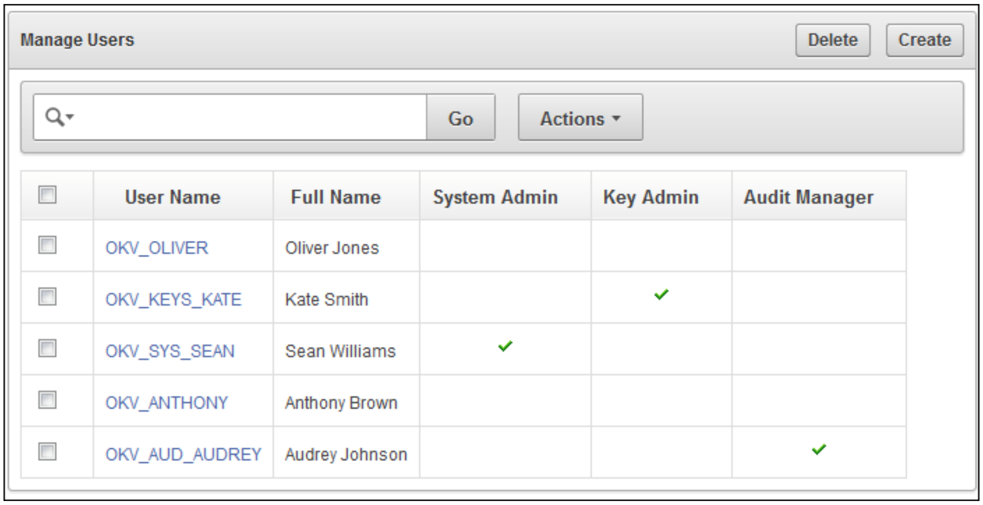
### 5.1.3.1 Grant or Change an Administrative Role of a User

To grant or change an administrative role:

1. Log in to the Oracle Key Vault management console as a user who has the role that the user is to be granted.
2. Click the **Users** tab.

   The **Manage Users** page appears displaying the list of users.

**Figure 5-2 Manage Users**

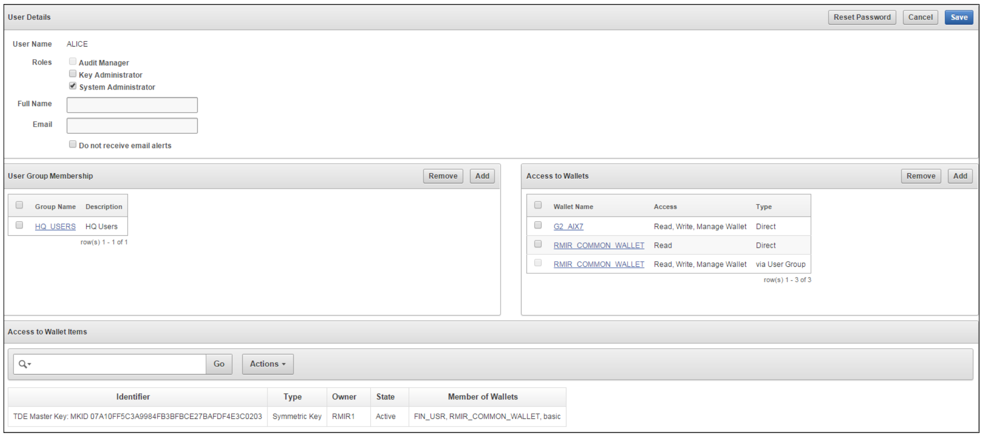


| □ | User Name | Full Name | System Admin | Key Admin | Audit Manager |
|---|---|---|---|---|---|
| □ | OKV_OLIVER | Oliver Jones | | | |
| □ | OKV_KEYS_KATE | Kate Smith | | ✓ | |
| □ | OKV_SYS_SEAN | Sean Williams | ✓ | | |
| □ | OKV_ANTHONY | Anthony Brown | | | |
| □ | OKV_AUD_AUDREY | Audrey Johnson | | | ✓ |

3. Click the name of the user in the **User Name** column.

   The **User Details** page appears. The **User Details** page provides a consolidated view of the Key Vault user. It displays the following user information: name, email, administrative role(s), membership in user group(s), and access to security object(s).

**Figure 5-3    User Details**



User Details

User Name ALICE

Roles
☐ Audit Manager
☐ Key Administrator
☑ System Administrator

Full Name

Email

☐ Do not receive email alerts

User Group Membership

| ☐ | Group Name | Description |
|---|---|---|
| ☐ | HQ_USERS | HQ Users |

row(s) 1 - 1 of 1

Access to Wallets

| ☐ | Wallet Name | Access | Type |
|---|---|---|---|
| ☐ | G2_AIX7 | Read, Write, Manage Wallet | Direct |
| ☐ | RMIR_COMMON_WALLET | Read | Direct |
| ☐ | RMIR_COMMON_WALLET | Read, Write, Manage Wallet | via User Group |

row(s) 1 - 3 of 3

Access to Wallet Items

| Identifier | Type | Owner | State | Member of Wallets |
|---|---|---|---|---|
| TDE Master Key: MKID 07A10FF5C3A9984FB3BFBCE27BAFDF4E3C0203 | Symmetric Key | RMIR1 | Active | FIN_USR, RMIR_COMMON_WALLET, basic |

4. To grant a role, check the box for the role you want to grant by **Roles**. These will be one of **Audit Manager**, **Key Administrator**, or **System Administrator**.

   To change a role un-check the box for the previous role and check the box(es) by the new role(s).

5. Click **Save**.

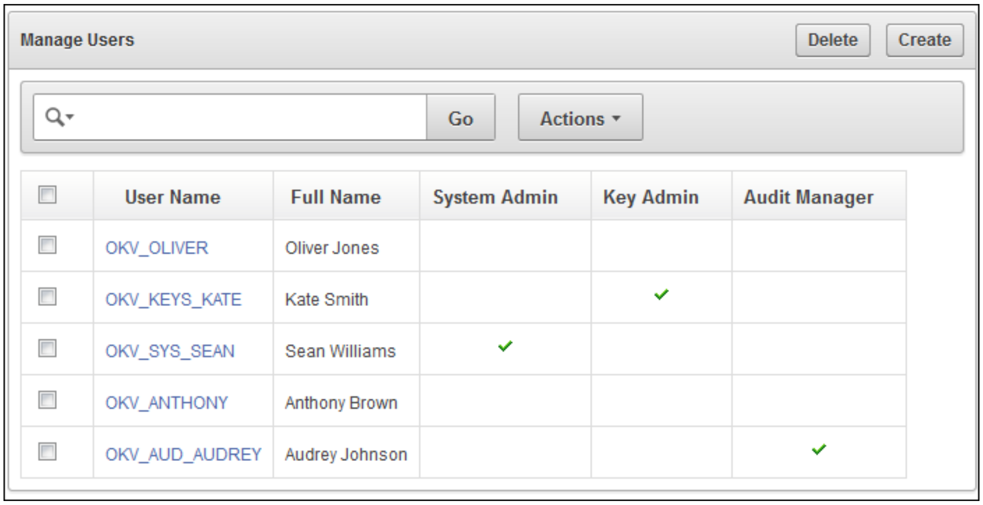
## 5.1.3.2 Revoke an Administrative Role from a User

To revoke a role from a user follow these steps:

1. Log in to the Oracle Key Vault management console as a user who has the role that the user is to be granted.
2. Click the **Users** tab.

   The **Manage Users** page appears displaying the list of users.

**Figure 5-4 Manage Users**



| ☐ | User Name | Full Name | System Admin | Key Admin | Audit Manager |
|---|---|---|---|---|---|
| ☐ | OKV_OLIVER | Oliver Jones | | | |
| ☐ | OKV_KEYS_KATE | Kate Smith | | ✓ | |
| ☐ | OKV_SYS_SEAN | Sean Williams | ✓ | | |
| ☐ | OKV_ANTHONY | Anthony Brown | | | |
| ☐ | OKV_AUD_AUDREY | Audrey Johnson | | | ✓ |

3. Click the user name, whose role you want to revoke.

   The **User Details** page appears.

4. Un-check the box for the role you want to revoke.
5. Click **Save**.

## 5.1.4 Delete Oracle Key Vault User(s)

You can delete a Key Vault user if the user's function in the organization changes. Deleting a user removes them from Key Vault and removes them from any user groups they were part of. The operation does not delete any security objects managed by the user.

To delete a user from Oracle Key Vault follow these steps:

1. Log in to the Oracle Key Vault management console as a user with the System Administrator role and the same role(s) as the user being deleted.
2. Select the **Users** tab.

   The **Manage Users** page appears displaying the list of users.

3. Check the box(es) by the user(s) you want to delete.
4. Click **Delete**.
5. In the confirmation dialog box, click **OK**.
6. Click **Save**.

## 5.1.5 View User Details

All administrative users can view the list of Oracle Key Vault users and their details. Users without any of the three administrative roles can only see their own user details.

The **User Details** page provides a consolidated view of the Key Vault user. This is the page where all user management tasks are performed.

To view user details for a given user:

1. Log in to the Oracle Key Vault management console.
2. Select **Users**.

   The **Manage Users** page appears displaying the list of users.

   You can sort and search the list by column: user name, full name or roles.
3. Click on a user name to get to the **User Details** page.

> **See Also:**
>
> "Administrative Roles within Oracle Key Vault (page 2-8)"
>
> "Performing Actions and Searches (page 3-15)"

# 5.2 About Changing User Passwords

Any valid Oracle Key Vault user can change his or her own password.

- How User Password Changes Work (page 5-6)
- Change Your Own Password (page 5-7)
- Reset Another User's Password (page 5-7)

## 5.2.1 How User Password Changes Work

You can reset another user's password as a Key Vault System Administrator. You can also reset the password of another user if you have at minimum the same administrative role as that user. For example, if you want to change the password of a user who has the Audit Manager role, then you also must have this role before you can change the password.

Consider the following users and roles:

| User | System Admin | Key Admin | Audit Manager |
|---|---|---|---|
| OKV_ALL_JANE | Yes | Yes | Yes |
| OKV_SYS_KEYS_JOE | Yes | Yes | - |
| OKV_SYS_SEAN | Yes | - | - |
| OKV_KEYS_KATE | - | Yes | - |
| OKV_AUD_AUDREY | - | - | Yes |
| OKV_OLIVER | - | - | - |

Suppose that user OKV_SYS_KEYS_JOE, who has the System Administrator and Key Administrator roles is logged in and wants to change the other users' passwords. This happens:

- OKV_KEYS_KATE: OKV_SYS_KEYS_JOE can change the password for OKV_KEYS_KATE because they have the Key Administrator role in common.

- `OKV_AUD_AUDREY`: `OKV_SYS_KEYS_JOE` cannot change `OKV_AUD_AUDREY`'s password because `OKV_SYS_KEYS_JOE` does not have the Audit Manager role.
- `OKV_ALL_JANE`: `OKV_SYS_KEYS_JOE` cannot change the password for user `OKV_ALL_JANE` because he does not have Audit Manager role.
- `OKV_OLIVER`: `OKV_SYS_KEYS_JOE` can change the password for user `OKV_OLIVER`, who has no roles at all.

## 5.2.2 Change Your Own Password

Any user can change his or her own Oracle Key Vault account password.

To change your own password:

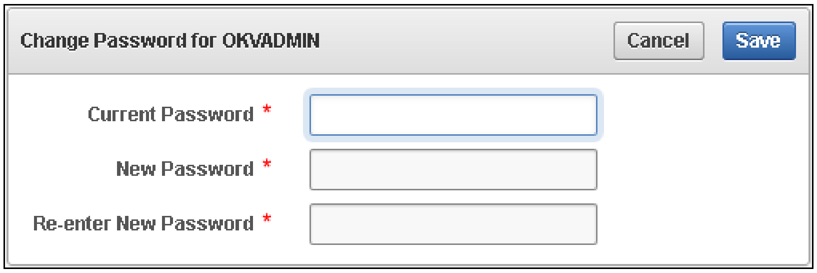
1. Log in to the Oracle Key Vault management console.

   See **References** below to learn how to log in to Key Vault.
2. Select the **Users** tab.

   The **Manage Users** page appears displaying the list of users.
3. Select **Change Password** from the left sidebar.

   The **Change Password for <your user name>** page appears.

**Figure 5-5 Change Your Own User Password**



Change Password for OKVADMIN

Current Password *

New Password *

Re-enter New Password *

4. Enter your current password in **Current Password**. Enter the new password in **New Password** and **Re-enter New Password**.
5. Click **Save**.

## 5.2.3 Reset Another User's Password

You can reset the password of another user if you are a Key Vault System Administrator or a user with the identical administrative role (at minimum) as the user, whose password you wish to reset. Key Vault provides two ways to reset a user's password.

- Reset Password Manually (page 5-7)
- Reset Password Automatically (page 5-8)
- Reset Operating System User Account Passwords (page 5-9)

### 5.2.3.1 Reset Password Manually

You can set the password manually for a user, and then use any out-of-band method to notify the user of the new password.

To reset another user's password follow these steps:

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

   See **References** below to learn how to log in to Key Vault.

2. Select the **Users** tab.

   The **Manage Users** page appears displaying the list of users.

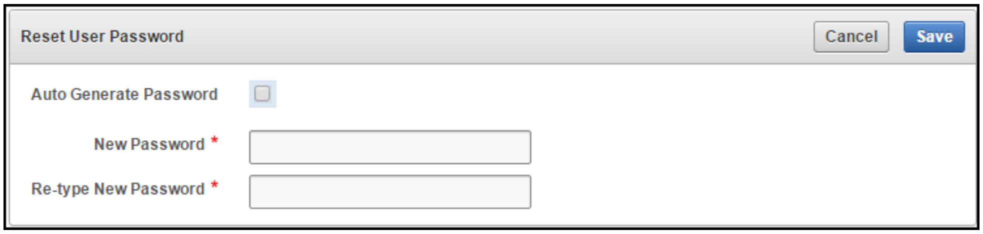
3. Click the user name, whose password you want to change.

   The **User Details** page appears.

4. Click **Reset Password**.

   The **Reset User Password** page appears.

**Figure 5-6    Reset User Password Manually**



5. Enter the new password in **New Password** and **Re-type New Password**.

6. Click **Save**.

## 5.2.3.2 Reset Password Automatically

Another way to reset a user's password is to have it generated automatically by Key Vault. This password can be sent directly from Key Vault to the user. You must configure SMTP in Email Settings in order to use this feature.

To automatically generate a password and have it sent to the user follow these steps:

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

2. Select the **Users** tab.

   The **Manage Users** page appears displaying the list of users.

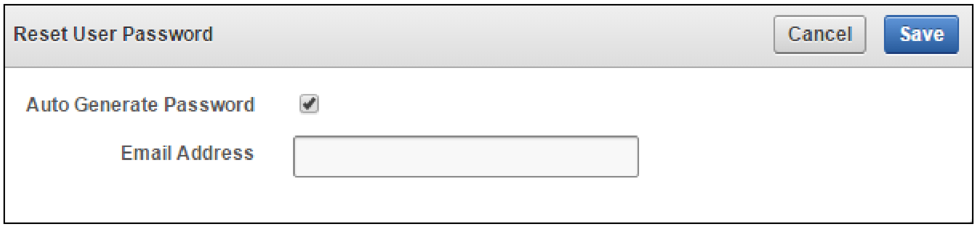
3. Click the user name, whose password you want to change.

   The **User Details** page appears.

4. Click **Reset Password**.

   The **Reset User Password** page appears.

**Figure 5-7    Reset User Password Automatically**



Auto Generate Password ☑

5. Check the box by **Auto Generate Password**.

   An email address field appears.

6. Enter the email address of the user.

7. Click **Save**.

   A confirmation message appears.

If you check **Auto Generate Password** without configuring SMTP, a link to **Email Settings** appears. Click the link to configure email settings and repeat **Steps 1-7**.

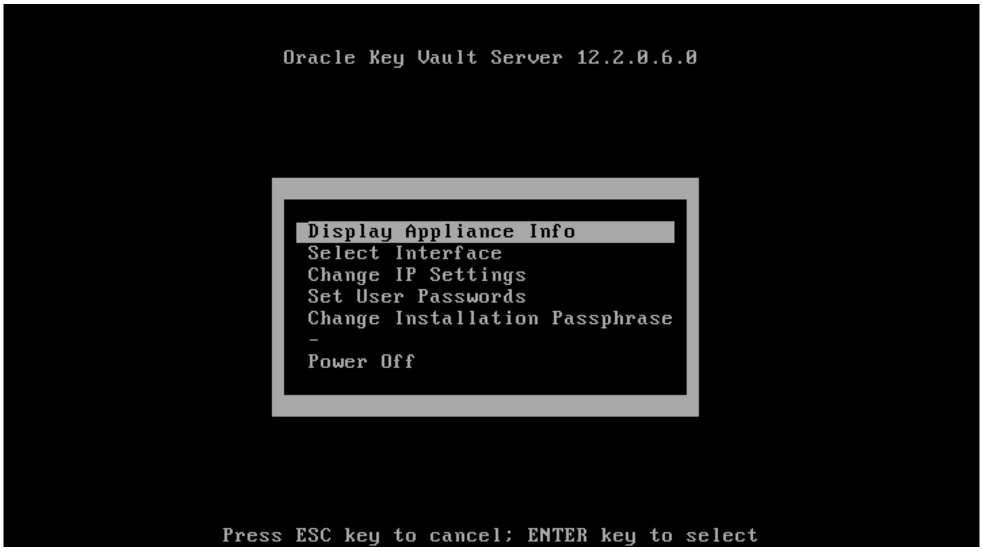
### 5.2.3.3 Reset Operating System User Account Passwords

You can reset the passwords for the Operating System user accounts, Root and Support. The Root and Support users will be prompted to change their password when the next time they log in is past the expiration time of their passwords. The expiration times are 365 days with a warning at 120 days, and with STIG it is 60 days with a warning at 60 days.

1. Using SSH, log in to the Oracle Key Vault server terminal as the System Administrator.

   The **Oracle Key Vault Server <Release Number>** screen appears.

**Figure 5-8    Oracle Key Vault Server <Release Number> Screen**

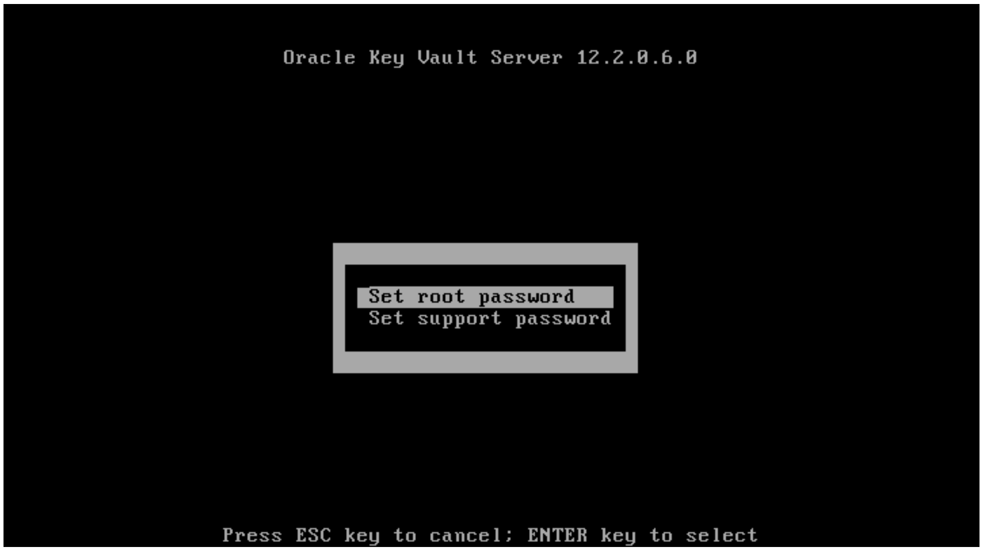


```
Oracle Key Vault Server 12.2.0.6.0

Display Appliance Info
Select Interface
Change IP Settings
Set User Passwords
Change Installation Passphrase
-
Power Off

Press ESC key to cancel; ENTER key to select
```

2. Select **Set User Passwords** to set the Root and Support User passwords. Press **Enter**.

The **Set User Passwords** screen appears.

**Figure 5-9 Set User Passwords Screen**

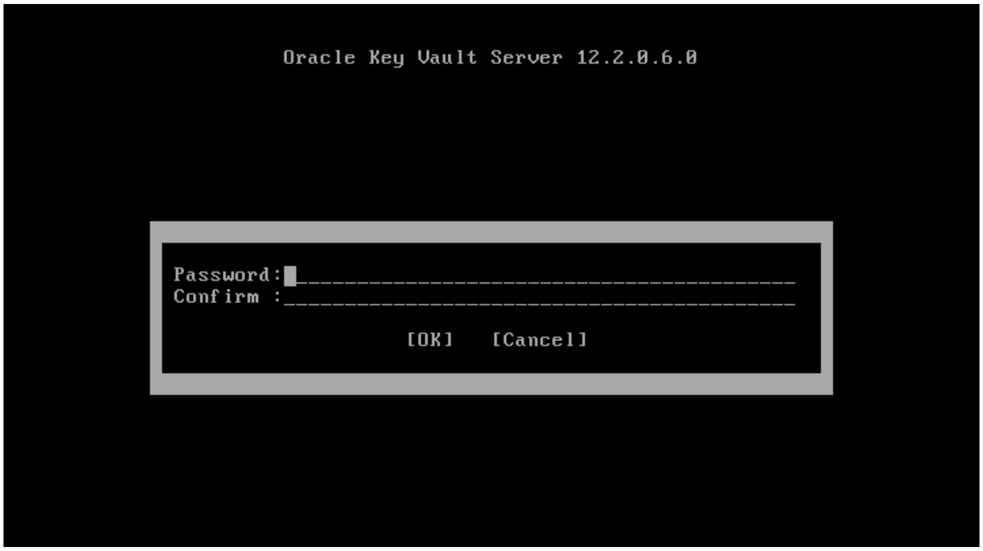


```
Oracle Key Vault Server 12.2.0.6.0

Set root password
Set support password

Press ESC key to cancel; ENTER key to select
```

3. Select **Set root password** or **Set support password** and press **Enter**.

   The **Set Password** screen appears.

**Figure 5-10 Set Password Screen**

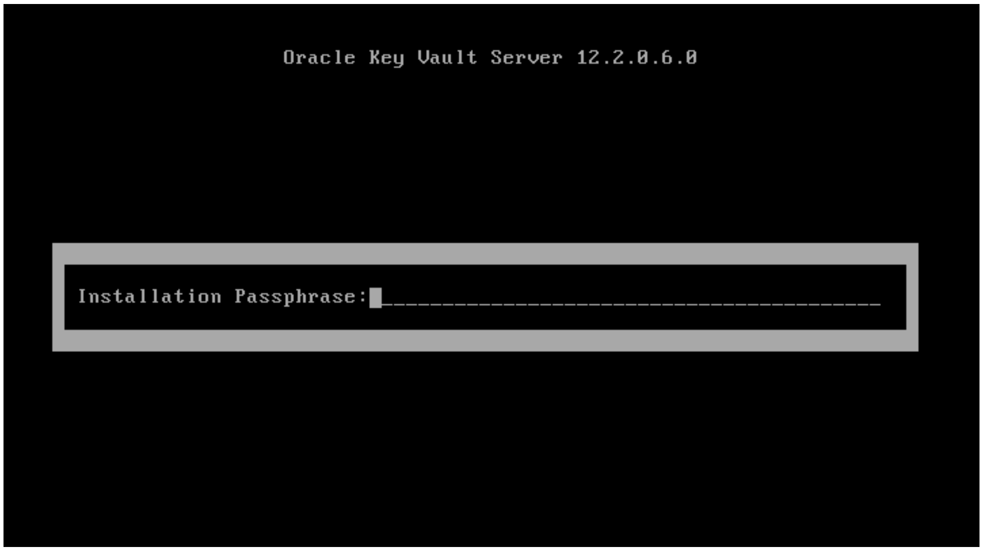


```
Oracle Key Vault Server 12.2.0.6.0

Password:_____________________________________________
Confirm :_____________________________________________

                 [OK]   [Cancel]
```

4. Type the new password in the **Password** and **Confirm** fields. Select **OK** and press **Enter**.

   The **Installation Passphrase** screen appears.

**Figure 5-11 Installation Passphrase Screen**



```
Oracle Key Vault Server 12.2.0.6.0

Installation Passphrase:_________________________________________
```

5. Enter the Installation Passphrase and press **Enter**.

The Operating System User password is changed.

## 5.3 Grant a User Access to a Virtual Wallet

A user with the Key Administrator role controls access to security objects for users, endpoints, and their respective groups. Any user may be granted access to security objects in Key Vault at a level appropriate to their function in the organization.

You can grant a user access to a virtual wallet as follows:

1. Log in as a user who has the Key Administrator role.
2. Select the **Users** tab, and then select **Manage Users**.

   The **Manage Users** page appears displaying the list of users.
3. Click the name of the user you want to grant access.

   The **User Details** page appears.
4. Click **Add** in the **Access to Wallets** section.

   The **Add Access to User Group** page appears.
5. Select the wallet under **Select Wallet**.
6. Set the access level to the selected wallet under **Select Access Level**. Select **Read Only**, **Read and Modify**, or **Manage Wallet**.

   Set access levels when you grant access to the wallet, if you know the level to grant. You can also set or modify access levels from the wallet menu.
7. Click **Save**.

> **See Also:**
>
> "Access Control Configuration (page 2-6)"
>
> "Manage Access to Virtual Wallets from Keys & Wallets Menu (page 6-4)"

# 5.4 About User Email

It is important that Key Vault users have their current email on file so that system changes like alerts and password changes may be communicated directly from Key Vault. User email can be updated in the **User Details** page. Users can also elect to opt out of email notifications.

- Disable Email Notifications for a User (page 5-12)
- Change User Email (page 5-13)

## 5.4.1 Disable Email Notifications for a User

You can disable email notifications for a user on the user details page.

To get to the **User Details** page:

1. Log in to the Oracle Key Vault management console as a user who has the System Administrator role:

   See **References** below to learn how to log in to Key Vault.

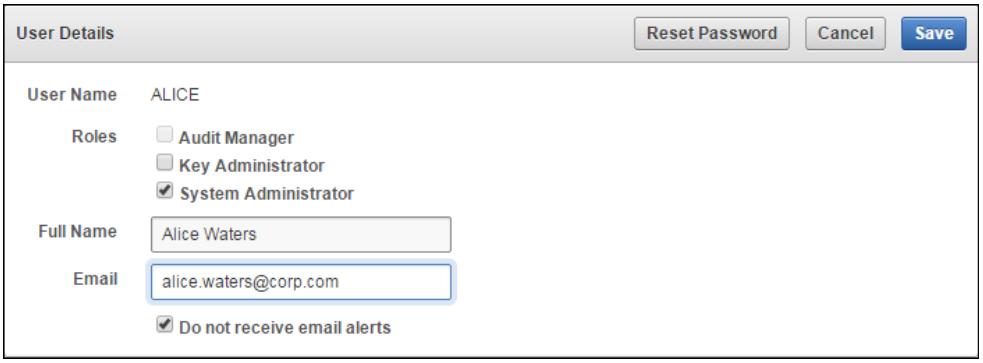
2. Select the **Users** tab.

   The **Manage Users** page appears displaying the list of users.

3. Click the user's name in the **User Name** column.

   The **User Details** page appears:

**Figure 5-12 User Details Page**



User Details

User Name ALICE

Roles
☐ Audit Manager
☐ Key Administrator
☑ System Administrator

Full Name Alice Waters

Email alice.waters@corp.com

☑ Do not receive email alerts

4. Check the box **Do not receive email alerts**.
5. Click **Save**.

### 5.4.2 Change User Email

After adding the user you can add or modify the user's email address as follows:

1. Log in to the Oracle Key Vault management console as a user who has the System Administrator role:

   See **References** below to learn how to log in to Key Vault.
2. Select the **Users** tab.

   The **Manage Users** page appears displaying the list of users.
3. Click the user's name in the **User Name** column.

   The **User Details** page appears:
4. Enter the email address in **Email**.
5. Click **Save**.

## 5.5 Manage User Groups

A user group is a named collection of users who have a specific purpose.

- How User Groups Work (page 5-13)
- Create a User Group (page 5-13)
- Add a User to a User Group (page 5-15)
- Remove a User from a User Group (page 5-15)
- Grant a User Group Access to a Virtual Wallet (page 5-16)
- Modify User Group Description (page 5-16)
- Delete a User Group (page 5-16)

### 5.5.1 How User Groups Work

Once a user group is created you can modify all its details except the group name, which may not be changed. Users who have the Key Administrator role can create, modify, and delete user groups, in order to manage their access to virtual wallets.

The main purpose of a user group is simplify access control to security objects. If a set of users need access to a common set of security objects, you can put these users in a group and grant the group access instead of granting access per user or per security object. When certain users do not need access to the security objects any more, they may be removed from the group. New users may be added to the group.

The group's access level to security objects may be modified at any time.

### 5.5.2 Create a User Group

Create a user group when a set of users needs to manage a set of common security objects. You can add users to the group when you create the group or later after creating the group.

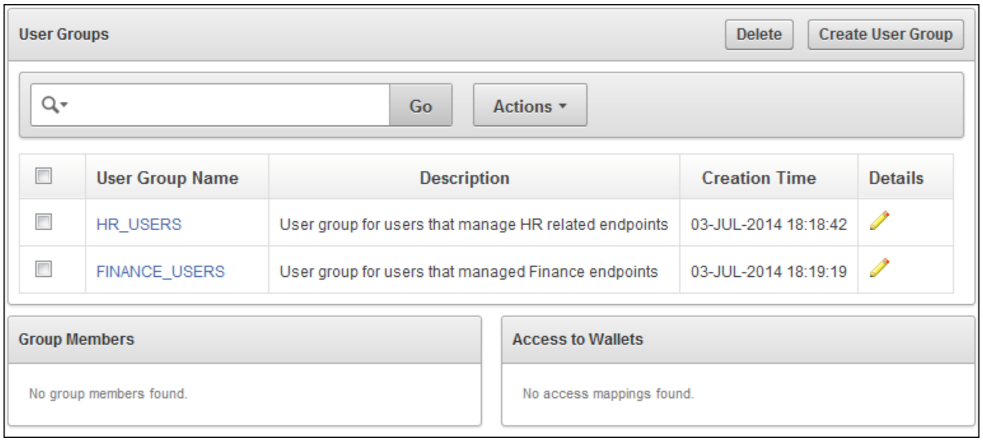
To create a user group and add users to the group at the same time:

1. Log in as a user who has the Key Administrator role.
2. Select the **Users** tab.

   The **Manage Users** page appears displaying the list of users.
3. Select **Manage Access** from the left sidebar.

   The **User Groups** page appears displaying existing user groups.

**Figure 5-13 User Groups Page**



| ☐ | User Group Name | Description | Creation Time | Details |
|---|---|---|---|---|
| ☐ | HR_USERS | User group for users that manage HR related endpoints | 03-JUL-2014 18:18:42 | ✎ |
| ☐ | FINANCE_USERS | User group for users that managed Finance endpoints | 03-JUL-2014 18:19:19 | ✎ |

Group Members

No group members found.

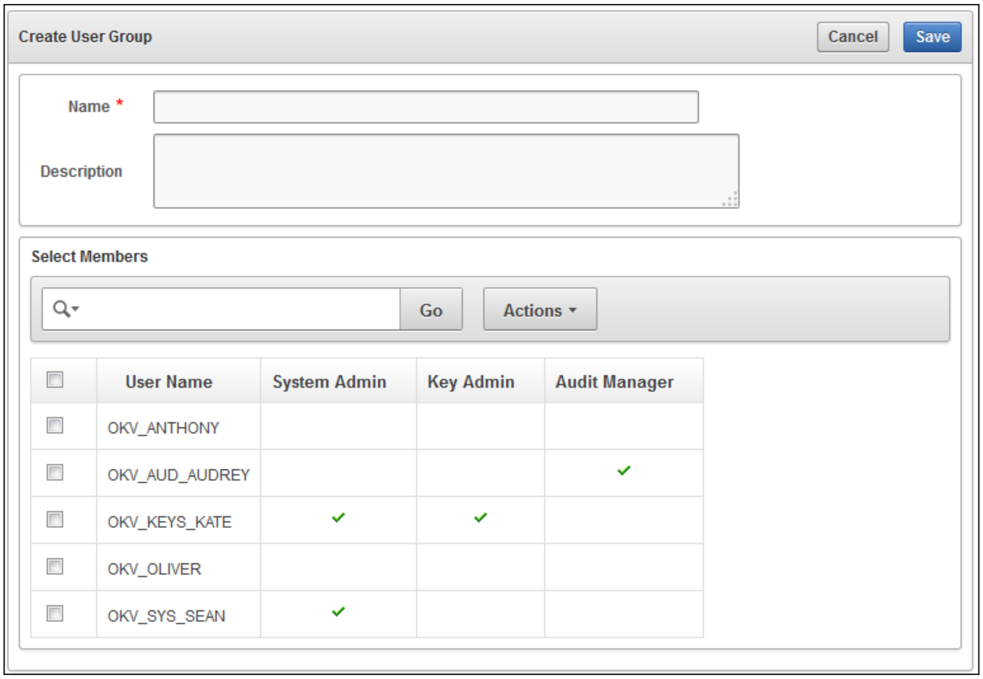
Access to Wallets

No access mappings found.

4. Click **Create User Group**.

   The **Create User Group** page appears with list of users in **Select Members**.

**Figure 5-14 Create User Group Page**



| | |
|---|---|
| Name * | |
| Description | |

Select Members

| ☐ | User Name | System Admin | Key Admin | Audit Manager |
|---|---|---|---|---|
| ☐ | OKV_ANTHONY | | | |
| ☐ | OKV_AUD_AUDREY | | | ✓ |
| ☐ | OKV_KEYS_KATE | ✓ | ✓ | |
| ☐ | OKV_OLIVER | | | |
| ☐ | OKV_SYS_SEAN | ✓ | | |

5. On the **Create User Group** page, do the following:
    - **Name:** Enter a name for the group.
    - **Description:** Optionally, enter a description for the group.
    - **Select Members:** Check the box(es) by the users you want to add to the group.
6. Click **Save**.

## 5.5.3 Add a User to a User Group

You can add an existing user to a user group if that user needs to manage the same security objects as the group. You can add users to a group when you create the group or later after creating the groups.

To add a user to a user group after the group is created:

1. Log in as a user who has the Key Administrator role.
2. Click the **Users** tab, then **Manage Access**.

   The **User Groups** page appears displaying a list of existing user groups.
3. Click the pencil icon in the **Details** for the user group.

   The **User Group Details** page appears displaying a list of existing user groups.
4. Click **Add** in the **User Group Members** pane. The **Add User Group Members** page appears displaying the list of existing users not in the user group.
5. Check the box(es) for the user(s) you want to add.
6. Click **Save**.

   A dialog box appears, indicating that the user has been successfully added.

## 5.5.4 Remove a User from a User Group

You can remove users from a user group when their function in the organization changes, and they no longer need to manage the same security objects as the group.

To remove a user from a user group:

1. Log in as a user who has the Key Administrator role.
2. Click the **Users** tab, then **Manage Access**.

   The **User Groups** page appears displaying a list of existing user groups.
3. Click the pencil icon in the **Details** for the user group.

   The **User Group Details** page appears.
4. In the **User Group Members** region, check the box(es) for the user(s) you want to remove.
5. Click **Remove**.
6. Click **OK** to confirm.

   A success message appears confirming the deletion.

## 5.5.5 Grant a User Group Access to a Virtual Wallet

The access level to a virtual wallet for a user group may be modified at any time as functional needs change.

To change the access level on a virtual wallet for a user group:

1. Log in as a user who has the Key Administrator role.
2. Select the **Users** tab, and then select **Manage Access**.

   The **User Groups** page appears displaying a list of existing user groups.
3. Click the pencil icon in the **Details** column, for the user group that you want to modify.

   The **User Group Details** page appears.
4. Click **Add** in the **Access to Wallets** section.

   The **Add Access to User Group** page appears.
5. Select the wallet in **Select Wallet**.
6. Set the access level to the selected wallet in **Select Access Level**. Select **Read Only**, **Read and Modify**, or **Manage Wallet**.
7. Click **Save**.

> **See Also:**
>
> "Access Control Configuration (page 2-6)"

## 5.5.6 Modify User Group Description

A group description is useful to identify the purpose of the group. This description may be modified at any time to match the purpose of the group.

You can change the description of a user group as follows:

1. Log in as a user who has the Key Administrator role.
2. Select the **Users** tab, and then select **Manage Access**. The **User Groups** page appears.
3. On the **User Groups** page, select the pencil icon in the **Details** column, for the user group that you want to modify. The **User Group Details** page appears.
4. Enter a new description in the **Description** field.
5. Click **Save**.

## 5.5.7 Delete a User Group

You can delete a user group when the users in the group do not need to access the same security objects. This will automatically delete the group's access to wallets and security objects.

To delete a user group, follow these steps:

1. Log in to Oracle Key Vault as a user who has been granted the Key Administrator role.
2. Select the **Users** tab, and then select **Manage Access**.

   The **User Groups** page appears.
3. Check the box(es) for the user group(s) that you want to delete.
4. Click **Delete.**
5. Click **OK** to confirm.

   A success message appears confirming the deletion.

# 6
# Managing Oracle Key Vault Virtual Wallets and Security Objects

Oracle Key Vault provides the mechanism of a virtual wallet to upload and store your security objects that you can then share with trusted peers at access levels appropriate to their organizational function.

- About Virtual Wallets (page 6-1)
- Manage Access to Virtual Wallets from Keys & Wallets Menu (page 6-4)
- Manage Access to Virtual Wallets from User’s Menu (page 6-7)
- Manage State of a Security Object (page 6-9)
- Manage Details of Security Objects (page 6-11)

## 6.1 About Virtual Wallets

A virtual wallet is a container for security objects like public and private encryption keys, including TDE, Oracle wallets, Java keystores, certificates, and credential files.

- How Virtual Wallets Work (page 6-1)
- Create a Virtual Wallet (page 6-2)
- Add Security Objects to a Virtual Wallet (page 6-3)
- Remove Security Objects from a Virtual Wallet (page 6-3)
- Delete a Virtual Wallet (page 6-4)

### 6.1.1 How Virtual Wallets Work

Oracle Key Vault provides the mechanism of a virtual wallet to group security objects for sharing with multiple users, who need them to access encrypted data.

Any user can create a virtual wallet. After you create a virtual wallet, you can add keys and other security objects to the wallet. You can then grant other users, endpoints, user and endpoint groups access to the virtual wallet at various levels of access. A virtual wallet can be modified at any time. You can modify wallet contents, the users that must have access, and the access level of users according to the needs of the moment.

Other than the Key Administrator, access to the virtual wallet must be granted explicitly to all users. Read, modify, and manage wallet permissions are required to add and remove objects from the wallet, and to grant or modify wallet access to other users and groups.

## 6.1.2 Create a Virtual Wallet

You can create a virtual wallet, and add security objects to it at the same time. However, you can also create an empty virtual wallet, and add security objects to it later. You can modify access-mappings on a virtual wallet at any time.

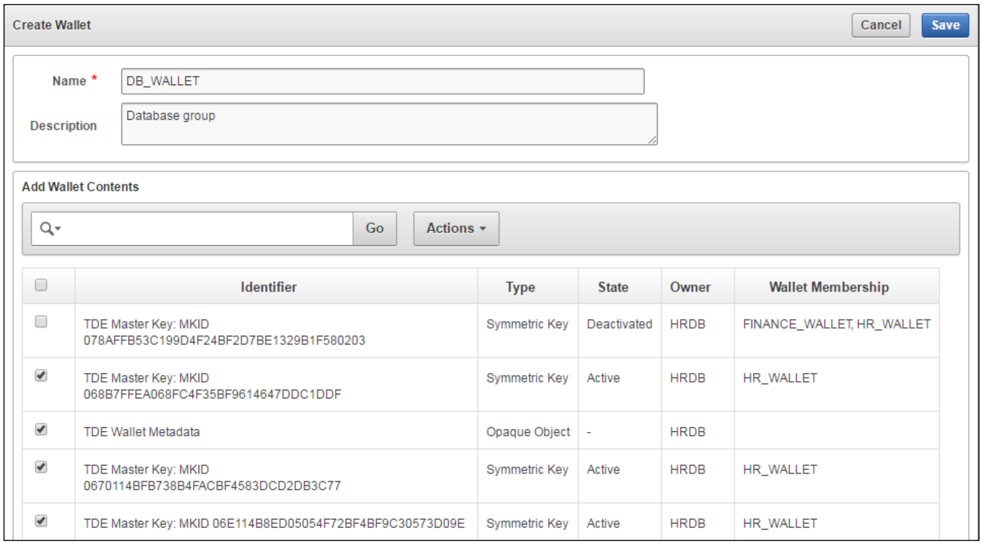
To create a virtual wallet and add security objects to it:

1. Log in to the Oracle Key Vault management console as a user with the Key Administrator role.
2. Select the **Keys & Wallets** tab.

   The **Wallets** page appears.
3. Click **Create**.

   The **Create Wallet** page appears.

   **Figure 6-1 Create Wallet**

   Name * DB_WALLET

   Description Database group

   | ☐ | Identifier | Type | State | Owner | Wallet Membership |
   |---|---|---|---|---|---|
   | ☐ | TDE Master Key: MKID 078AFFB53C199D4F24BF2D7BE1329B1F580203 | Symmetric Key | Deactivated | HRDB | FINANCE_WALLET, HR_WALLET |
   | ☑ | TDE Master Key: MKID 068B7FFEA068FC4F35BF9614647DDC1DDF | Symmetric Key | Active | HRDB | HR_WALLET |
   | ☑ | TDE Wallet Metadata | Opaque Object | - | HRDB | |
   | ☑ | TDE Master Key: MKID 0670114BFB738B4FACBF4583DCD2DB3C77 | Symmetric Key | Active | HRDB | HR_WALLET |
   | ☑ | TDE Master Key: MKID 06E114B8ED05054F72BF4BF9C30573D09E | Symmetric Key | Active | HRDB | HR_WALLET |

4. Enter a name for the wallet in **Name** and a identifying description in **Description**.

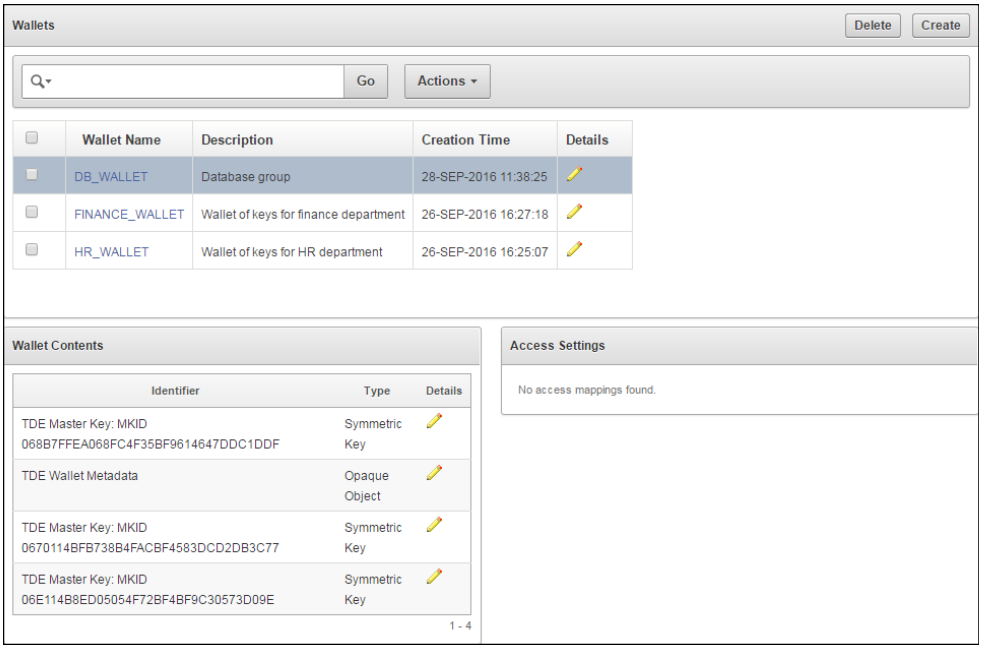
   Virtual wallet names are case-sensitive. For example, `wallet1` and `Wallet1` are two different wallets. It is recommended that you add a user friendly description to the wallet to identify it easily.
5. In the **Add Wallet Contents** pane, check the box(es) by the name(s) of the listed security objects that you want to add to the wallet.

   The **Add Wallet Contents** pane lists the security objects you have **Read and Modify** access to. If the list is empty, it means that you have no access to the security objects already in Key Vault. In this case, you would add security objects to the wallet after you upload them to Key Vault.
6. Click **Save** to create the new wallet with the security objects added to it.

   A **Wallet created successfully** message appears. The **Wallets** page appears and displays the new wallet in the list.

   To see the contents in the wallet click the wallet name as the following figure shows.

**Figure 6-2 Creation of a New Wallet**



| ☐ | Wallet Name | Description | Creation Time | Details |
|---|---|---|---|---|
| ☐ | DB_WALLET | Database group | 28-SEP-2016 11:38:25 | ✎ |
| ☐ | FINANCE_WALLET | Wallet of keys for finance department | 26-SEP-2016 16:27:18 | ✎ |
| ☐ | HR_WALLET | Wallet of keys for HR department | 26-SEP-2016 16:25:07 | ✎ |

Wallet Contents

| Identifier | Type | Details |
|---|---|---|
| TDE Master Key: MKID 068B7FFEA068FC4F35BF9614647DDC1DDF | Symmetric Key | ✎ |
| TDE Wallet Metadata | Opaque Object | ✎ |
| TDE Master Key: MKID 0670114BFB738B4FACBF4583DCD2DB3C77 | Symmetric Key | ✎ |
| TDE Master Key: MKID 06E114B8ED05054F72BF4BF9C30573D09E | Symmetric Key | ✎ |

1 - 4

Access Settings

No access mappings found.

## 6.1.3 Add Security Objects to a Virtual Wallet

You can add new security objects to a virtual wallet at any time as needed.

To add items to wallets:

1. Log in to the Oracle Key Vault management console as a user who has the **Manage Wallet** access on the virtual wallet or as a user with the Key Administrator role.
2. Select the **Keys & Wallets** tab.

   The **Wallets** page appears.
3. From the **Wallets** page, click the pencil icon in the **Details** column corresponding to the wallet you want to work with.

   The **Wallet Overview** page appears. The **Wallet Contents** pane lists the security objects already in the wallet.
4. Click **Add Items**. The **Add Wallet Contents** page appears.
5. Check the box(es) by the security objects you want to add to the wallet.
6. Click **Save**.

   A confirmation message appears.

   The **Wallet Overview** page appears and **Wallet Contents** lists the new security objects added.

## 6.1.4 Remove Security Objects from a Virtual Wallet

You can remove security objects from virtual wallets at any time as needed.

To remove security objects from wallets:

1. Log in to the Oracle Key Vault management console as a user who has the **Manage Wallet** access on the virtual wallet or as a user with the Key Administrator role.
2. Select the **Keys & Wallets** tab.

   The **Wallets** page appears.
3. From the **Wallets** page, click the pencil icon in the **Details** column corresponding to the wallet you want to work with.

   The **Wallet Overview** page appears. The **Wallet Contents** pane lists the security objects already in the wallet.
4. Check the box(es) by the security objects you want to remove from the wallet.
5. Click **Remove Items.** A confirmation message appears.

   The **Wallet Contents** pane in the **Wallet Overview** page displays the new list with the items deleted.

### 6.1.5 Delete a Virtual Wallet

Deleting a virtual wallet removes the wallet as a container, but does not delete the security objects that were contained in it. These security objects will continue to remain in Key Vault. Endpoints that have downloaded this virtual wallet will continue to retain their local copy.

To delete a virtual wallet:

1. Log in to the Oracle Key Vault management console as a user who has the Manage Wallet permission on the virtual wallet, or as a user with the Key Administrator role.
2. Select the **Keys & Wallets** tab.

   The **Wallets** page appears.
3. Check the box(es) next to the name of the wallet that you want to delete from the **Wallets** table. You may delete more than one virtual wallet at the same time.
4. Click **Delete**.
5. Click **OK** to confirm.
6. Select the **Keys and Wallets** tab to see the updated list of wallets in the **Wallets** page.

## 6.2 Manage Access to Virtual Wallets from Keys & Wallets Menu

Access control is about deciding which users and endpoints need to share virtual wallets and security objects, and what operations they can perform on those virtual wallets.

- How Access to Virtual Wallets from Keys & Wallets Menu Works (page 6-5)
- Grant Access to Endpoint Groups, Endpoints, User Groups, and Users (page 6-5)
- Modify Access to Endpoint Groups, Endpoints, User Groups, and Users (page 6-6)

## 6.2.1 How Access to Virtual Wallets from Keys & Wallets Menu Works

You must have access to a virtual wallet or be a key administrator to manage access control for users, endpoints, and their respective groups.

Oracle Key Vault provides two ways to manage access control on virtual wallets for users, endpoints, and their respective groups:

- From the **Keys & Wallets** menu, where you select the wallet, and then grant an endpoint, endpoint group, user, or user group access to the wallet.
- From the **Users** or **Endpoints** menu, where you select the user, user group, endpoint or endpoint group, and then grant one of these entities access to the wallet.

This section focusses on managing access to a virtual wallet for users and user groups from the **Keys & Wallets** menu.

## 6.2.2 Grant Access to Endpoint Groups, Endpoints, User Groups, and Users

You can choose a virtual wallet and grant endpoint groups, endpoints, user groups, or users **Read Only**, **Read and Modify**, and **Manage Wallet** access levels on the wallet. Once they have access to the wallet, they will have access to all the security objects in the wallet.

To grant access to a virtual wallet:

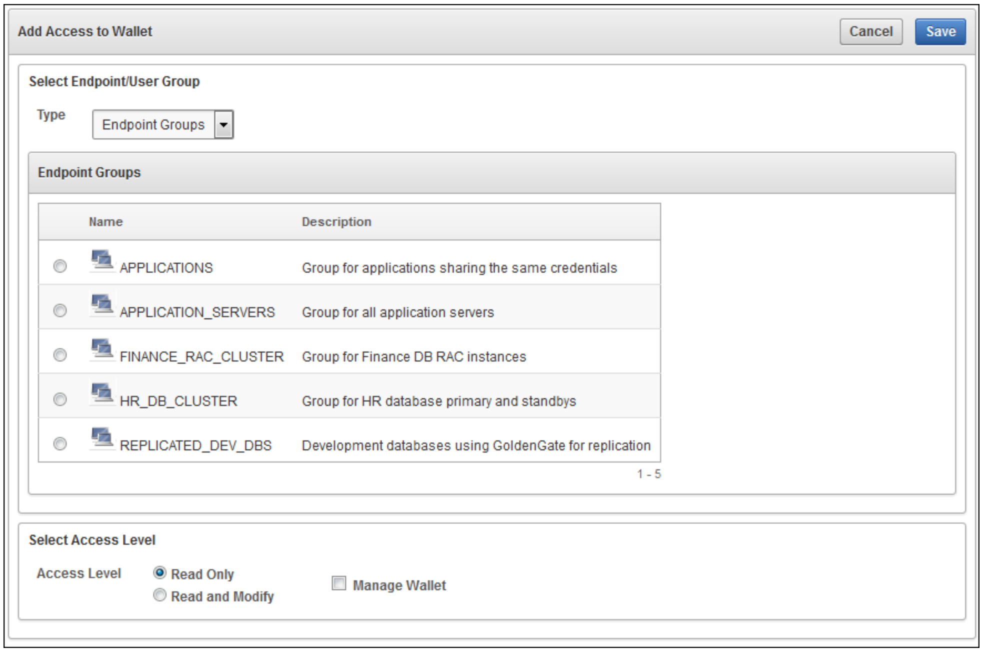
1. Log in to the Oracle Key Vault management console as a user who has the **Manage Wallet** access on the virtual wallet, or as a user with the Key Administrator role.
2. Select the **Keys & Wallets** tab.

   The **Wallets** page appears.
3. Click the pencil icon in the **Details** column corresponding to the wallet you want to grant access to.

   The **Wallet Overview** page appears.
4. In the **Wallet Access Settings** pane, click **Add**.

   The **Add Access to Wallet** page appears.

**Figure 6-3 Add Access to Wallet**



Select Endpoint/User Group

Type: Endpoint Groups

Endpoint Groups

| | Name | Description |
|---|---|---|
| ○ | APPLICATIONS | Group for applications sharing the same credentials |
| ○ | APPLICATION_SERVERS | Group for all application servers |
| ○ | FINANCE_RAC_CLUSTER | Group for Finance DB RAC instances |
| ○ | HR_DB_CLUSTER | Group for HR database primary and standbys |
| ○ | REPLICATED_DEV_DBS | Development databases using GoldenGate for replication |

1 - 5

Select Access Level

Access Level: ● Read Only ○ Read and Modify □ Manage Wallet

5. Select the entity type you want to grant access from the **Select Endpoint/User Group** drop down list next to **Type**.

   Possible values for **Type** are **Endpoint Groups**, **Endpoints**, **User Groups**, and **Users**.

   The type you select determines the list that is displayed. For example, if you select **Endpoint Groups** as the **Type**, the list of Key Vault endpoint groups is displayed under the heading **Endpoint Groups**. If you select **Users**, the list of Key Vault users are displayed under the heading **Users**.

6. Select the radio button in the **Name** table corresponding to the entity you want to grant access.
7. Select one of **Read Only** or **Read and Modify** in the **Select Access Level** pane.
8. Check the box to **Manage Wallet** if needed.
9. Click **Save**.

   A message appears: **Access mapping successfully added**.

   The **Wallet Access Settings** pane displays the new entity.

## 6.2.3 Modify Access to Endpoint Groups, Endpoints, User Groups, and Users

You can modify the access settings on a virtual wallet for endpoint groups, endpoints, user groups and users as follows:

1. Log in to the Oracle Key Vault management console as a user who has the Manage Wallet permission on the virtual wallet or as a user with the Key Administrator role.
2. Select the **Keys & Wallets** tab, and then select **Wallets** from the left sidebar. The **Wallets** page appears.

3. Click the pencil icon in the **Details** column corresponding to the wallet name.

   The **Wallet Overview** page appears, with **Wallet Access Settings** listing the entities that have access to the wallet and their access levels.

4. In **Wallet Access Settings**, click the pencil icon corresponding to the entity under **Subject Name**.

   A **Modify Access** window appears.

   Note, that **Wallet Access Settings** lists all the entities that have access to this wallet under **Subject Name**, and can include users, endpoints, user and endpoint groups.

5. Select the access settings you want to modify, then click **Save**.

   A message appears: **Successfully updated**.

   The **Wallet Overview** page appears and **Wallet Access Settings** displays the new access mapping for the entity.

6. Click Save in the **Wallet Overview** page.

## 6.3 Manage Access to Virtual Wallets from User’s Menu

Oracle Key Vault provides two ways to manage access control on virtual wallets for users, endpoints, and their respective groups.

- How Access to Virtual Wallets from User's Menu Works (page 6-7)
- Grant a User Access to a Virtual Wallet (page 6-8)
- Revoke User Access from a Virtual Wallet (page 6-8)
- Grant a User Group Access to a Virtual Wallet (page 6-9)
- Revoke User Group Access from a Virtual Wallet (page 6-9)

### 6.3.1 How Access to Virtual Wallets from User's Menu Works

The two menus that you can use are as follows:

- From the **Keys & Wallets** menu, where you select the wallet, and then grant an endpoint, endpoint group, user, or user group access to the wallet.
- From the **Users** or **Endpoints** menu, where you select the user, user group, endpoint or endpoint group, and then grant one of these entities access to the wallet.

This section focuses on managing access to a virtual wallet for users and user groups from the **Users** menu.

> **See Also:**
>
> "Manage Endpoint Access to a Virtual Wallet (page 7-11)"

## 6.3.2 Grant a User Access to a Virtual Wallet

To grant a user access to a virtual wallet:

1. Log in to the Oracle Key Vault management console as a user who has the Manage Wallet permission on the virtual wallet, or as a user with the Key Administrator role.
2. Select the **Users** tab.

   The **Manage Users** page appears.
3. Click the user's name **User Name** column.

   The **User Details** page appears.
4. In the **Access to Wallets** pane, click **Add**.

   The **Add Access to User** page appears.
5. Select a virtual wallet from the available list.
6. In the **Select Access Level** pane select the desired access levels.
7. Click **Save**.

   A message appears: **Access mapping successfully added**.

   Check **Access to Wallets** in **User Details** for the user to see the wallet added.

> **See Also:**
>
> "Access Control Options (page 2-8)"

## 6.3.3 Revoke User Access from a Virtual Wallet

To revoke access to a virtual wallet for a user:

1. Log in to the Oracle Key Vault management console as a user who has the Manage Wallet access on the virtual wallet, or as a user with the Key Administrator role.
2. Select the **Users** tab.

   Then **Manage Users** page appears.
3. Click the user's name under **User Name**.

   The **User Detail**s page appears.
4. In Access to Wallets check the box by the virtual wallet you want to revoke access to.
5. Click **Remove**.

   A confirmation dialog box appears.
6. Click **OK.**

   A message appears: **Access mapping(s) deleted successfully**.

Check **Access to Wallets** in **User Details** for the user to see the wallet deleted.

### 6.3.4 Grant a User Group Access to a Virtual Wallet

When you grant a user group access to a virtual wallet all members of the group will have access to the security objects within the wallet.

To grant a user group access to a virtual wallet:

1. Log in to the Oracle Key Vault management console as a user who has the Key Administrator role.
2. Select the **Users** tab, and then select **Manage Access** in the left sidebar.

   The **User Groups** page appears.
3. Click the pencil icon in the **Details** column corresponding to the user group. The **User Group Details** page appears.
4. Click **Add** in the **Access to Wallets** pane.

   The **Add Access to User Group** page appears.
5. Select a virtual wallet from the available list
6. In the **Select Access Level** pane select the desired access levels.
7. Click **Save**.

   A message appears: **Access mapping successfully added**.

   Check **Access to Wallets** in **User Groups** for the user to see the wallet added.

### 6.3.5 Revoke User Group Access from a Virtual Wallet

You can remove user group access to a virtual wallet as follows:

1. Log in to the Oracle Key Vault management console as a user who has the Key Administrator role.
2. Select the **Users** tab, and then select **Manage Access** in the left sidebar. The **User Groups** page appears.
3. Click the pencil icon in the **Details** column corresponding to the user group. The **User Group Details** page appears.
4. In the **Access to Wallets** pane, check the box by the virtual wallet you want to revoke access to.
5. Click **Remove**.
6. Click **OK** to confirm.

   A message appears: **Access mapping(s) deleted successfully.**

   Check **Access to Wallets** in **User Groups** to see the wallet removed from the list.

## 6.4 Manage State of a Security Object

You can set the start date for a security object to become active, or deactivate it. You can also change the state of some virtual wallet security objects as needed.

- Activate a Key or Security Object (page 6-10)

## 6.4.1 Activate a Key or Security Object

Currently, only keys uploaded via a third-party KMIP client can be in a **Pre-Active** state and have the **Activation** date set. For all other keys the **Activation Date** is system generated and cannot be set.

Most keys are in **Active** state when they are created. However you can set the **Process Start Date** for a key to be used for securing data, later than its creation date as follows:

1. Log in to the Oracle Key Vault management console as a user who has the Read and Modify access on this key.
2. Select the **Keys & Wallets** tab.
3. Select the **All Items** menu and then click the edit pencil icon corresponding to the item, whose start date you want to set.
4. On the **Item Details** page for the item, set the **Process Start Date** to the desired date.
5. Click **Save**.

## 6.4.2 Deactivate a Key or Security Object

A key deactivates or expires when it passes the date that has been set for deactivation.

1. Log in to the Oracle Key Vault management console as a user who has the Read and Modify access on this key.
2. Select the **Keys & Wallets** tab.
3. Select the **All Items** menu and then click the edit pencil icon corresponding to the item to be deactivated.
4. On the **Item Details** page for the item, set the **Date of Deactivation** to the date by which you want the key to be deactivated.
5. Click **Save**.

## 6.4.3 Revoke a Key or Security Object

When you revoke a key, its state transitions to **Deactivated** or **Compromised**, and the key should no longer be used to encrypt new data.

However, deactivated keys may be downloaded and used to decrypt old data.

1. Log in to the Oracle Key Vault management console as a user who has the Read and Modify access on this key.
2. Select the **Keys & Wallets** tab.
3. Select the **All Items** from the left side bar.

   The All Items page appears listing all the security objects.

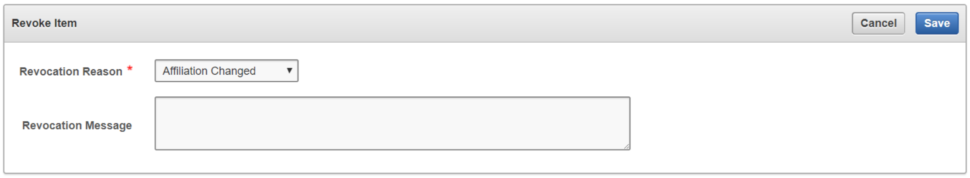
4. Click the pencil icon in the **Details** column corresponding to the item to be revoked.

   The **Item Details** page appears.

5. Click **Revoke**.

   The **Revoke Item** page appears.

**Figure 6-4 Revoke Item**



Revocation Reason * Affiliation Changed

Revocation Message

6. Select a **Revocation Reason** from the drop down list.
7. Optionally, add more details in **Revocation Message**
8. Click **Save**.

   A message appears, indicating that the revocation succeeded.

## 6.4.4 Destroy a Key or Security Object

When a key is no longer used or compromised in some way you might want to destroy it.

> **Note:**
>
> Meta data for destroyed keys and security objects are kept in Key Vault even after they have been destroyed.

To destroy a key:

1. Log in to the Oracle Key Vault management console as a user who has the Read and Modify access on this key.
2. Select the **Keys & Wallets** tab.
3. Select the **All Items** menu and then click the edit pencil icon corresponding to the item, whose start date you want to set.
4. On the **Item Details** page for the item, click **Destroy**.
5. Click **Save**.

# 6.5 Manage Details of Security Objects

After you create a virtual wallet and add security objects to it, you can search the virtual wallet for the security objects contained in it. You can add new security objects to the virtual wallet, or delete security objects from them. You can modify the contents of a virtual wallet at any time according to your specific needs.

Security objects are managed by Key Vault administrative users with a clear separation of duties. You must be an administrative user with the Key Administrator role to manage wallet privilege on the virtual wallet containing the security objects. A user with the Audit Manager role can view security objects, but cannot modify them, whereas security objects are not even viewable to a user with the System Administrator role.

- Search for Security Object Items (page 6-12)
- View Details of a Security Object (page 6-13)
- Add or Modify Details of a Security Object (page 6-16)

## 6.5.1 Search for Security Object Items

The term *item* refers to a single security object managed by Oracle Key Vault, such as an encryption key, keystore, certificate, password, or opaque object.

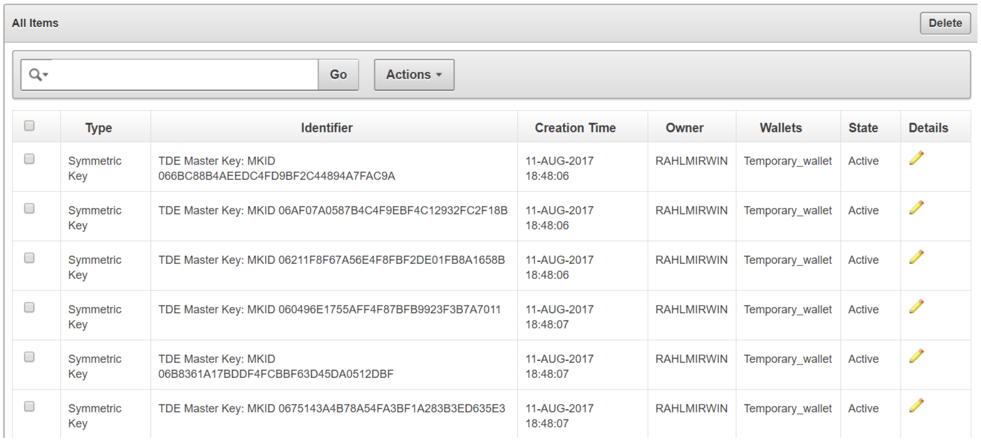
To search for items:

1. Log in to the Oracle Key Vault management console as a user with the Key Administrator role, an Audit Manager role, or as a user with access to a virtual wallet.
2. Click the tab **Keys & Wallets**.

   The **Wallets** page appears.
3. Click **All Items** in the left sidebar.

   The **All Items** page appears displaying all the security objects in a table.

**Figure 6-5 All Items Lists all Security Objects in Key Vault**



| □ | Type | Identifier | Creation Time | Owner | Wallets | State | Details |
|---|---|---|---|---|---|---|---|
| □ | Symmetric Key | TDE Master Key: MKID 066BC88B4AEEDC4FD9BF2C44894A7FAC9A | 11-AUG-2017 18:48:06 | RAHLMIRWIN | Temporary_wallet | Active | ✎ |
| □ | Symmetric Key | TDE Master Key: MKID 06AF07A0587B4C4F9EBF4C12932FC2F18B | 11-AUG-2017 18:48:06 | RAHLMIRWIN | Temporary_wallet | Active | ✎ |
| □ | Symmetric Key | TDE Master Key: MKID 06211F8F67A56E4F8FBF2DE01FB8A1658B | 11-AUG-2017 18:48:06 | RAHLMIRWIN | Temporary_wallet | Active | ✎ |
| □ | Symmetric Key | TDE Master Key: MKID 060496E1755AFF4F87BFB9923F3B7A7011 | 11-AUG-2017 18:48:07 | RAHLMIRWIN | Temporary_wallet | Active | ✎ |
| □ | Symmetric Key | TDE Master Key: MKID 06B8361A17BDDF4FCBBF63D45DA0512DBF | 11-AUG-2017 18:48:07 | RAHLMIRWIN | Temporary_wallet | Active | ✎ |
| □ | Symmetric Key | TDE Master Key: MKID 0675143A4B78A54FA3BF1A283B3ED635E3 | 11-AUG-2017 18:48:07 | RAHLMIRWIN | Temporary_wallet | Active | ✎ |

The table has the following columns for each security object:

- **Type:** Indicates the object type of the item. Valid values are Symmetric Key, Private Key, and Opaque Object.
- **Identifier:** Lists the identifier for the item and includes a prefix that helps identify a subtype for the item.
- **Creation Time:** Date and time that the item was added to Oracle Key Vault.

- **Owner:** The endpoint that owns the item.
- **Wallets:** The virtual wallet that contains the security object.
- **State:** Indicates the state of the object. Valid values are Active and N/A.
- **Details:** A pencil icon links to the **Item Details** for the security object.

4. Search for specific items using the Search bar or the Actions menu.

> **See Also:**
>
> "Performing Actions and Searches (page 3-15)"

## 6.5.2 View Details of a Security Object

An administrative user with the Key Administrator role can view, add, and modify the details of a security object from its corresponding **Item Details** page. Item details are attributes of a specific security object and depend on the type of security object.

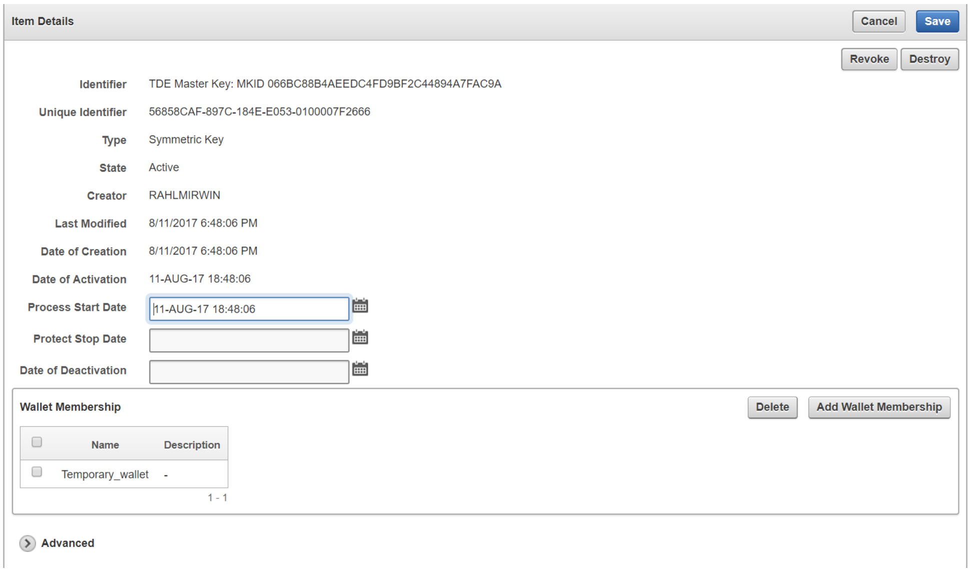
To view the attributes of a security object from the **Item Details** page:

1. Log in to the Oracle Key Vault management console as a user with the Key Administrator role or as a user with access to the virtual wallet.
2. Click the tab **Keys & Wallets**.

   The **Wallets** page appears.
3. Click **All Items** in the left sidebar.

   The **All Items** page appears displaying all the security objects in Key Vault.
4. Click the pencil icon in the **Details** column corresponding to the security object. The **Item Details** page appears displaying the attributes of the security object.

**Figure 6-6 Item Details**



| | |
|---|---|
| Identifier | TDE Master Key: MKID 066BC88B4AEEDC4FD9BF2C44894A7FAC9A |
| Unique Identifier | 56858CAF-897C-184E-E053-0100007F2666 |
| Type | Symmetric Key |
| State | Active |
| Creator | RAHLMIRWIN |
| Last Modified | 8/11/2017 6:48:06 PM |
| Date of Creation | 8/11/2017 6:48:06 PM |
| Date of Activation | 11-AUG-17 18:48:06 |
| Process Start Date | 11-AUG-17 18:48:06 |
| Protect Stop Date | |
| Date of Deactivation | |

Wallet Membership

| □ | Name | Description |
|---|---|---|
| □ | Temporary_wallet | - |

1 - 1

You can set the dates when the security object should be deactivated or not used on the **Item Details** page. The attributes shown in Item Details depends on the type of security object. The attributes for a **Symmetric Key** are different from those of **Private Key** or **Opaque Object**.

You can revoke or destroy a security object, and add or remove it to and from a wallet from the **Item Details** page.

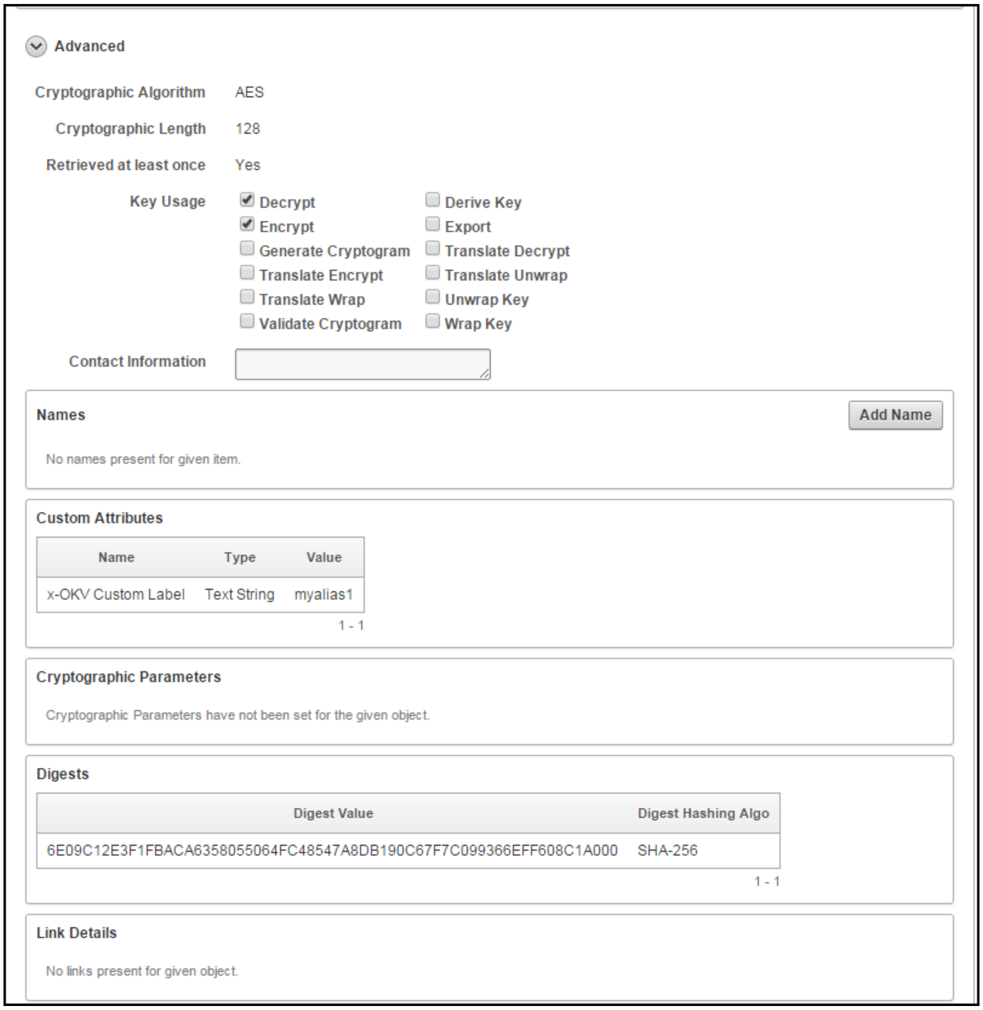
The **Wallet Membership** pane in the **Item Details** page allows you to add or delete the security object to or from a wallet.

The **Item Details** page contains the following attributes:

- **Identifier:** A summary description to help identify the item to the user. For example, if the item is a TDE master key, the **Identifier** shows the prefix TDE Master Key followed by the identifier used by the database to identify the key.
- **Unique Identifier:** This is a globally unique ID that identifies an item.
- **Type:** Indicates the object type of the item. Valid values are Symmetric Key, Private Key, Template, Opaque Object, Certificate, and Secret Data.
- **State:** Indicates the state of items. Values are as follows:
  - **Pre-active:** The object exists but is not yet usable for any cryptographic purpose.
  - **Active:** The object is available for use. Endpoints should examine the Cryptographic Usage Mask attribute to determine which uses are appropriate for this object.
  - **Deactivated:** The object is no longer active and should not be used to apply cryptographic protection (for example, encryption or signing). It may still be appropriate to use for decrypting or verifying previously protected data.
  - **Compromised:** The object is believed to be compromised and should not be used.
  - **Destroyed:** The object is no longer usable for any purpose.
  - **Destroyed Compromised:** The object was compromised and subsequently destroyed. It is no longer usable for any purpose.
- **Creator**: The endpoint that created the security object.
- **Last Modified:** The date last modified.
- **Date of Creation:** The date created.
- **Date of Activation:** The date of activation.
- **Process Start Date**: The date when the key may start to be used to encrypt data. It can be equal or later than the Activation Date but cannot precede it.
- **Protect Stop Date**: When this date is passed, the key should not be used to encrypt any more data. It cannot be later than the Deactivation Date.
- **Date of Deactivation:** The date of deactivation.

5. Click **Advanced** to view the cryptographic attributes of the security object.

**Figure 6-7    Item Details - Advanced Pane**



Advanced

| | |
|---|---|
| Cryptographic Algorithm | AES |
| Cryptographic Length | 128 |
| Retrieved at least once | Yes |

Key Usage

☑ Decrypt
☑ Encrypt
□ Generate Cryptogram
□ Translate Encrypt
□ Translate Wrap
□ Validate Cryptogram
□ Derive Key
□ Export
□ Translate Decrypt
□ Translate Unwrap
□ Unwrap Key
□ Wrap Key

Contact Information

Names

No names present for given item.

Custom Attributes

| Name | Type | Value |
|---|---|---|
| x-OKV Custom Label | Text String | myalias1 |

1 - 1

Cryptographic Parameters

Cryptographic Parameters have not been set for the given object.

Digests

| Digest Value | Digest Hashing Algo |
|---|---|
| 6E09C12E3F1FBACA6358055064FC48547A8DB190C67F7C099366EFF608C1A000 | SHA-256 |

1 - 1

Link Details

No links present for given object.

Attribute information and queries may vary depending on the item type. These are some attributes:

- **Cryptographic Algorithms:** The encryption algorithm used by the item
- **Key Usage:** Operations that the key can be used for
- **Names:** Labels attached by a user or endpoint to identify the key
- **Custom attributes:** Additional attributes defined by the endpoint and not interpreted by Oracle Key Vault
- **Cryptographic Parameters:** Optional parameters for the encryption algorithm used by the item, such as block cipher mode and padding method
- **Digests:** Digest values of the security object
- **Link Details:** Links to related objects

**See Also:**

*Key Management Interoperability Protocol Specification Version 1.1*

## 6.5.3 Add or Modify Details of a Security Object

To modify the attributes of a security object you must be a user with the Key Administrator role, or you must have **Read and Modify** access on the security object.

You can get **Read and Modify** access on a security object in two ways:

- You own the security object.
- You have access to a wallet that contains the security object.

To modify details of a security object:

1. Log in to the Oracle Key Vault management console as a user with the Key Administrator role, an Audit Manager role, or as a user with access to a virtual wallet.
2. Click the tab **Keys & Wallets**.

   The **Wallets** page appears.
3. Click **All Items** in the left sidebar.

   The **All Items** page appears displaying all the security objects in a table.
4. Click the pencil icon corresponding to the security object.

   The **Item Details** page appears.
5. Click **Advanced**.

   The **Advanced** pane appears.
6. Make the needed changes.
7. Click **Save** in the top right.

# 7 Managing Oracle Key Vault Endpoints

Oracle Key Vault endpoints are computer systems like database servers, application servers, and other information systems, where keys and credentials are used to access encrypted data and other systems. Endpoints must be registered and enrolled to communicate with Oracle Key Vault, after which they can upload their keys to Key Vault, share them with other endpoints, and download them to access their data.

## 7.1 About Managing Endpoints

Endpoints must be registered and enrolled to communicate with Oracle Key Vault. Only a user with the System Administrator role can add an endpoint to Key Vault. Once the endpoint is added, the endpoint administrator can enroll the endpoint by downloading and installing the endpoint software at the endpoint. The endpoint can then use the utilities packaged with the endpoint software to upload and download security objects to and from Key Vault.

All users can create virtual wallets but only a user with Key Administrator privileges can grant endpoints access to security objects contained in virtual wallets. The Key Administrator can also create endpoint groups to enable shared access to virtual wallets. When you grant an endpoint group access to a virtual wallet, all the member endpoints will have access to the virtual wallet. For example, you can grant all the nodes in an Oracle RAC access to a virtual wallet by putting them in an endpoint group. This saves you the step of granting each node access to the virtual wallet.

An Oracle Key Vault user name cannot be the same as an Oracle Key Vault endpoint name.

Below is a summary of the two administrative roles as they pertain to endpoints.

A user with the **System Administrator** role:

- Manages the endpoint meta-data like the name, type, platform, description, and email
- Manages the endpoint lifecycle which consists of enrolling, deleting, suspending, and reenrolling endpoints

A user with the **Key Administrator** role:

- Manages the endpoint group lifecycle which consists of creating, modifying, and deleting endpoint groups
- Manages the lifecycle of security objects, which consists of creating, modifying and deleting security objects
- Grants, modifies, and revokes access mappings on shared virtual wallets to endpoints and endpoint groups
- Associates an endpoint with a default wallet

# 7.2 Managing Endpoints

You can enroll new endpoints, reenroll existing endpoints, delete them when no longer integrated with Oracle Key Vault, and disable them temporarily for security reasons.

- Types of Endpoint Enrollment (page 7-2)
- Add an Endpoint as a Key Vault System Administrator (page 7-3)
- Add an Endpoint Using Self-Enrollment (page 7-6)
- Configuring Endpoint Configuration Parameters (page 7-7)
- Delete, Suspend or Reenroll Endpoints (page 7-8)

## 7.2.1 Types of Endpoint Enrollment

The first step to enrolling an endpoint is to add the endpoint to Key Vault. There are two methods for adding or registering an endpoint:

- Administrator-initiated

  An Oracle Key Vault user who has the System Administrator role initiates the enrollment from the Key Vault side by adding the endpoint to Key Vault. When the endpoint is added, a one-time enrollment token is generated. This token may be communicated to the endpoint administrator in two ways:

  1. Directly from Key Vault by email. To use email notification you must configure SMTP in email settings.
  2. Out-of-band method such as email or telephone.

  The endpoint administrator uses the enrollment token to download the endpoint software and complete the enrollment process on the endpoint side.

  Once the enrollment token is used to enroll an endpoint, it cannot be used again for another enrollment. If you need to reenroll an endpoint, the reenrollment process will generate a new one-time enrollment token for this purpose.

- Self-enrolled

  Endpoints may enroll themselves during specific times without human administrative intervention. Endpoint self-enrollment is useful when the endpoints do not share security objects, and use Oracle Key Vault primarily to store and restore their own security objects. Another use for endpoint self-enrollment is testing.

  A self-enrolled endpoint is created with a generic endpoint name in this format: `ENDPT_001`. Initially, a self-enrolled endpoint has access only to the security objects that it uploads or creates. It does not have access to any virtual wallets. You can later grant the endpoint access to virtual wallets after verifying its identity.

Endpoint self-enrollment is disabled by default, and must be enabled by a user with the System Administrator role. A best practice is to enable self-enrollment for short periods, when you expect endpoints to self enroll, and disable it when the self-enrollment period ends.

> **See Also:**
>
> "Email Notification (page 12-8)"

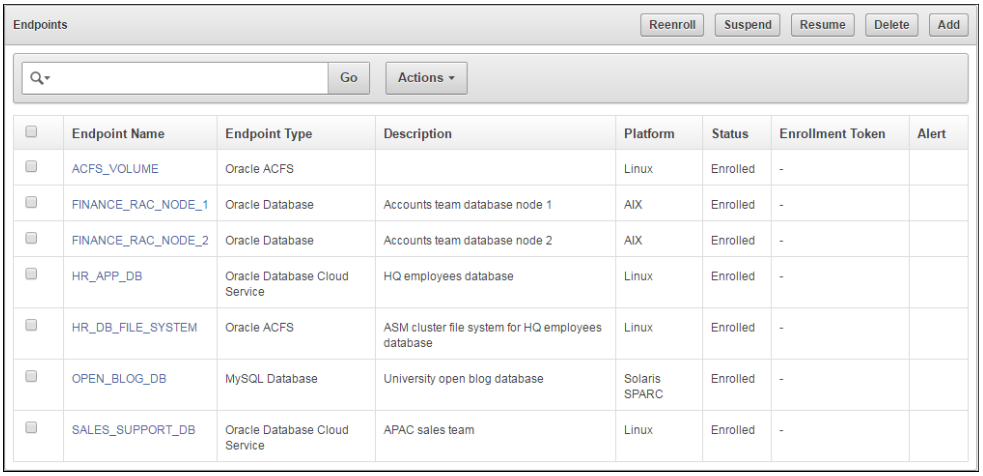
## 7.2.2 Add an Endpoint as a Key Vault System Administrator

To add an endpoint as a Key Vault System Administrator follow these steps:

1. Log in to the Oracle Key Vault management console as a user who has the System Administrator role.
2. Click the **Endpoints** tab.

   The **Endpoints** page appears listing all the Key Vault endpoints.

   **Figure 7-1 Endpoints Page**

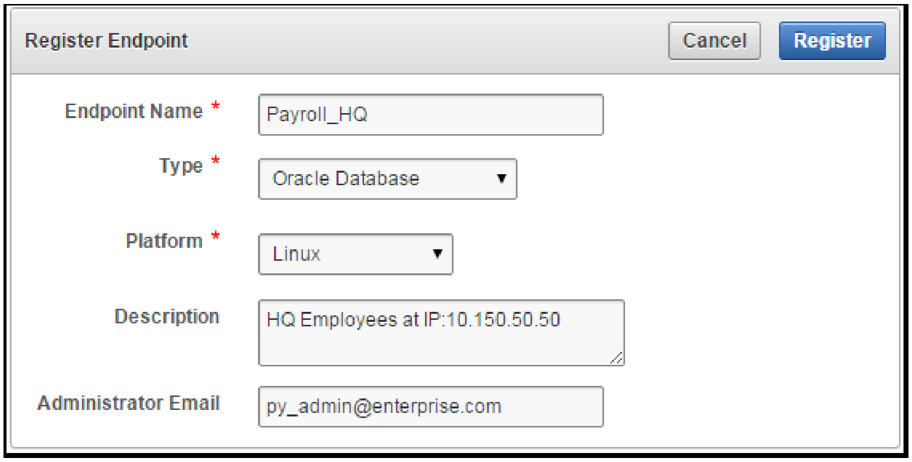
   | □ | Endpoint Name | Endpoint Type | Description | Platform | Status | Enrollment Token | Alert |
   |---|---|---|---|---|---|---|---|
   | □ | ACFS_VOLUME | Oracle ACFS | | Linux | Enrolled | - | |
   | □ | FINANCE_RAC_NODE_1 | Oracle Database | Accounts team database node 1 | AIX | Enrolled | - | |
   | □ | FINANCE_RAC_NODE_2 | Oracle Database | Accounts team database node 2 | AIX | Enrolled | - | |
   | □ | HR_APP_DB | Oracle Database Cloud Service | HQ employees database | Linux | Enrolled | - | |
   | □ | HR_DB_FILE_SYSTEM | Oracle ACFS | ASM cluster file system for HQ employees database | Linux | Enrolled | - | |
   | □ | OPEN_BLOG_DB | MySQL Database | University open blog database | Solaris SPARC | Enrolled | - | |
   | □ | SALES_SUPPORT_DB | Oracle Database Cloud Service | APAC sales team | Linux | Enrolled | - | |

   The **Endpoints** page displays the list of registered and enrolled endpoints with the following endpoint details: name, type, description, platform, status, enrollment token, and alert. The endpoint status can be either **Registered** or **Enrolled**:

   - **Registered Status:** The endpoint has been added and the one-time enrollment token has been generated. This token will be displayed in the corresponding **Enrollment Token** column.
   - **Enrolled Status:** The one-time enrollment token has been used to download the endpoint software. The **Enrollment Token** column displays a dash ('-') to indicate that the enrollment token has been used.
3. Click **Add** on the **Endpoints** page.

   The **Register Endpoint** page appears.

**Figure 7-2 Register Endpoint Page**



Register Endpoint

Endpoint Name * Payroll_HQ

Type * Oracle Database

Platform * Linux

Description HQ Employees at IP:10.150.50.50

Administrator Email py_admin@enterprise.com

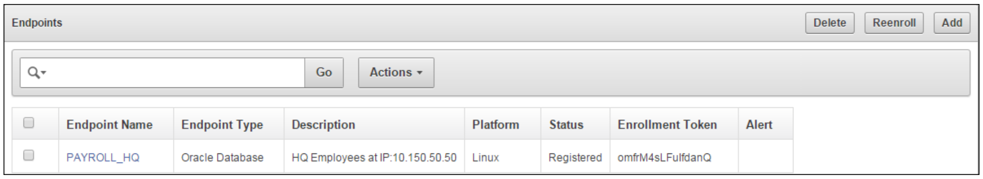
4. Enter information for the new endpoint as follows:
   - **Endpoint Name (required):** The name can have letters, numbers, and underscores. The endpoint name is not case-sensitive. For example, a name entered as "`app_server1`" will show up "`APP_SERVER1`" in the endpoints table. The endpoint will be referred to by this name throughout.
   - **Type (required):** Supported types are **Oracle Database**, **Oracle Database Cloud Service**, **Oracle (non-database)**, **Oracle ACFS**, **MySQL Database**, and **Other**. An example of **Other** is a third-party KMIP endpoint.

     > **Note:**
     >
     > If you are using Oracle Advanced Security Transparent Data Encryption (TDE) and want to use Oracle Key Vault to manage a TDE master key or wallet, then you must set **Type** to **Oracle Database**.

   - **Platform (required):** Supported platforms are **Linux**, **Solaris SPARC**, **Solaris x64**, **AIX**, **HP-UX**, **Windows**.
   - **Description (optional but recommended):** Enter useful identifying description like the host name, IP address, function, or location of the endpoint.
   - **Administrator Email (optional but recommended):** Enter the email address of the endpoint administrator to have the enrollment token and other endpoint related alerts sent directly from Key Vault. Note that you must have configured SMTP to use the email notification feature.
5. Click **Register**.

   The **Endpoints** page appears listing the new endpoint with a status of **Registered**. The **Enrollment Token** column displays the one-time enrollment token.

#### Figure 7-3 Endpoint in Registered Status

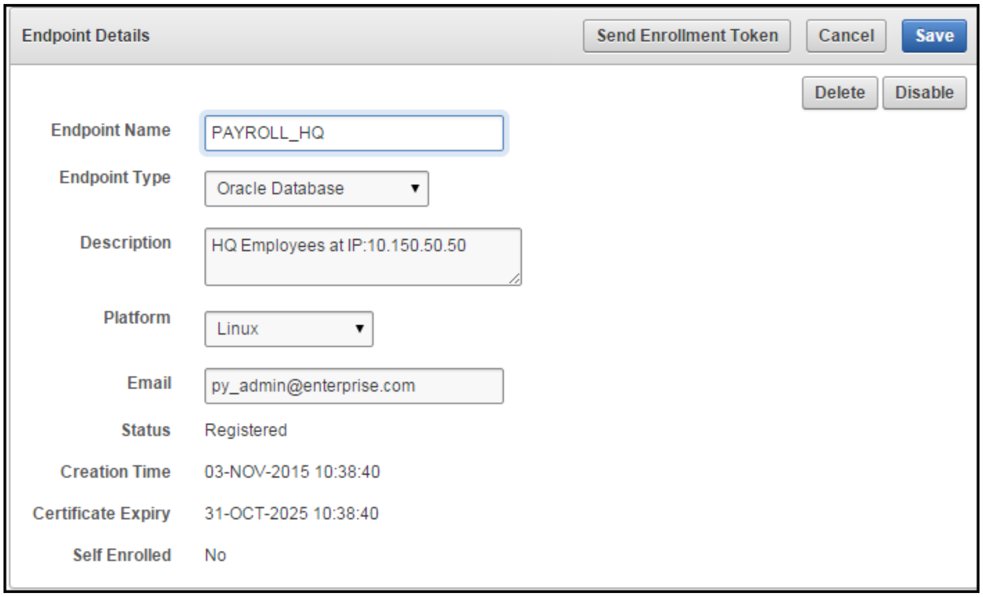


| □ | Endpoint Name | Endpoint Type | Description | Platform | Status | Enrollment Token | Alert |
|---|---|---|---|---|---|---|---|
| □ | PAYROLL_HQ | Oracle Database | HQ Employees at IP:10.150.50.50 | Linux | Registered | omfrM4sLFulfdanQ | |

6. Click the **Endpoint Name** to see details for the endpoint.

   The **Endpoint Details** page appears.

#### Figure 7-4 Endpoint Details



| | |
|---|---|
| Endpoint Name | PAYROLL_HQ |
| Endpoint Type | Oracle Database |
| Description | HQ Employees at IP:10.150.50.50 |
| Platform | Linux |
| Email | py_admin@enterprise.com |
| Status | Registered |
| Creation Time | 03-NOV-2015 10:38:40 |
| Certificate Expiry | 31-OCT-2025 10:38:40 |
| Self Enrolled | No |

> **Note:**
>
> The **Send Enrollment Token** button on the **Endpoint Details** page *only* appears for an endpoint whose **Status** is **Registered**.

There are two ways to send the one-time enrollment token to the endpoint administrator:

a. If you configured SMTP and entered the email address, you can have Key Vault send the enrollment token directly to the endpoint administrator, shown in **Step 7**.

b. If you did not configure SMTP or enter the email address, you must use an out-of-band method to send the enrollment token to the endpoint administrator.

7. Click **Send Enrollment Token**.

   A confirmation message appears, saying that the email was sent.

   Now it is up to that endpoint's administrator to complete the enrollment process for the endpoint.

When the enrollment token is used to download and install the endpoint software on the endpoint side, the endpoint status changes from **Registered** to **Enrolled**.

> **See Also:**
>
> - "Email Notification (page 12-8)"
> - Task 1: Enroll Endpoint and Download Software (page 8-3)

## 7.2.3 Add an Endpoint Using Self-Enrollment

Endpoint self-enrollment is disabled by default and must be enabled by a user who has the System Administrator role.

A best practice is to enable endpoint self-enrollment for limited periods when you expect endpoints to enroll. After the expected endpoints have enrolled, you should disable endpoint self-enrollment.

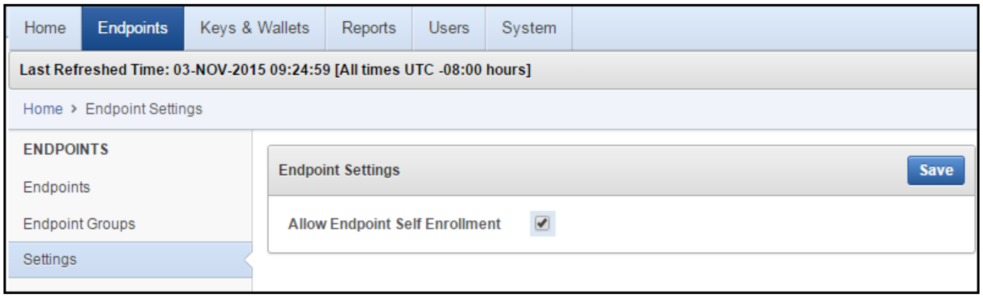
Oracle Key Vault associates a *self-enrolled* attribute with all endpoints that are enrolled through endpoint self-enrollment. Self-enrolled endpoints go directly to **Enrolled** status without the intermediate **Registered** status when they download the endpoint software. You can recognize self-enrolled endpoints by their system generated names in the form ENDPT_001.

To enable endpoint self-enrollment follow these steps:

1. Log in to the Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **Endpoints** tab, and then **Settings** from the left side bar.

   The **Endpoint Settings** page appears.

**Figure 7-5 Endpoint Settings for Self-Enrollment**

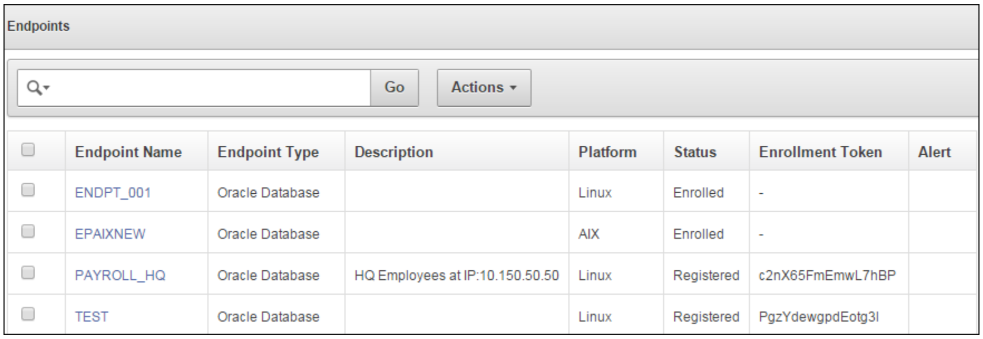


Last Refreshed Time: 03-NOV-2015 09:24:59 [All times UTC -08:00 hours]

Allow Endpoint Self Enrollment ☑

3. Check the box to the right of **Allow Endpoint Self-Enrollment**.
4. Click **Save**.

**Figure 7-6 Self-Enrolled Endpoint**



| □ | Endpoint Name | Endpoint Type | Description | Platform | Status | Enrollment Token | Alert |
|---|---|---|---|---|---|---|---|
| □ | ENDPT_001 | Oracle Database | | Linux | Enrolled | - | |
| □ | EPAIXNEW | Oracle Database | | AIX | Enrolled | - | |
| □ | PAYROLL_HQ | Oracle Database | HQ Employees at IP:10.150.50.50 | Linux | Registered | c2nX65FmEmwL7hBP | |
| □ | TEST | Oracle Database | | Linux | Registered | PgzYdewgpdEotg3I | |

> **See Also:**
>
> Task 1: Enroll Endpoint and Download Software (page 8-3)

## 7.2.4 Configuring Endpoint Configuration Parameters

Users with system administrator role can centrally update certain endpoint configuration parameters in the Oracle Key Vault Management Console. This feature enables system administrators to set certain endpoint configuration parameters globally, that is, for all endpoints, or on a per-endpoint basis. It simplifies the process of managing multiple endpoints for system administrators.

Endpoint specific parameters if set take precedence over global parameters. Global parameters, if set will take effect when endpoint-specific parameters are cleared. OKV will use the default system parameters if both global and endpoint specific parameters are cleared or not set from OKV management console.

The configuration parameter values set in the OKV management console are pushed to endpoints dynamically. After configuration parameters have been set in the OKV Management Console, the next time the endpoint contacts the OKV server, it will get the configuration parameters update. Endpoint configuration parameter update is best-effort. In case of error, the update is not applied. Both okvutil and PKCS11 library can get and apply the endpoint configuration updates.

To configure endpoint configuration parameters:

1. Log in to the Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **Endpoints** tab.

   The **Endpoints** page is displayed.
3. On the **Endpoints** page, click the **Endpoint Name**.

   The **Endpoint Details** page is displayed.
4. In the **Endpoint Configuration Parameters** section, configure the following settings:
   - **PKCS 11 In-Memory Cache Timeout**

Specify the duration for which the master key is available after it is cached in the in-memory cache. The value is specified in minutes. For more information about the PKCS 11 In-Memory Cache Timeout setting, see PKCS11_CACHE_TIMEOUT Parameter (page 11-21).

**PKCS 11 Persistent Cache Timeout**

Specify the duration for which the master key is available after it is cached in the persistent master key cache. The value is specified in minutes. For more information about the PKCS 11 Cache Persistent Timeout setting, see PKCS11_PERSISTENT_CACHE_TIMEOUT Parameter (page 11-21).

- **PKCS 11 Persistent Cache Refresh Window**

  Specify the duration to extend the period of timefor which the master key is available after it is cached in the persistent master key cache. The value is specified in minutes. For more information about the PKCS 11 Persistent Cache Refresh Window setting, see PKCS11_PERSISTENT_CACHE_REFRESH_WINDOW Parameter (page 11-22).

- **Server Poll Timeout**

  Specify a timeout for a client's attempt to connect to an Oracle Key Vault server, before trying the next server in the list. The default value is 300 (milliseconds).

- **PKCS 11 Trace Directory Path**

  Specify a directory to save the trace files.

5. Click **Save**.

The endpoint configuration settings are saved.

## 7.2.5 Delete, Suspend or Reenroll Endpoints

When endpoints are no longer using Oracle Key Vault to store security objects, System Administrators can delete them, and then reenroll them when they are needed again. Endpoints may also be temporarily suspended and later enabled.

- About Deleting Endpoints (page 7-8)
- Delete One or More Endpoint(s) (page 7-9)
- Delete one Endpoint (page 7-9)
- Suspend one Endpoint (page 7-10)
- Reenroll an Endpoint (page 7-10)

### 7.2.5.1 About Deleting Endpoints

Deleting an endpoint removes it permanently from Oracle Key Vault. However, security objects previously created or uploaded by that endpoint will remain in Oracle Key Vault. Likewise, security objects associated with that endpoint will also remain. To permanently delete or reassign these security objects, you will need to be a user with the Key Administrator role or authorize to merge these objects by managing wallet privileges. The endpoint software previously downloaded at the endpoint also remains on the endpoint until the endpoint administrator removes it.

### 7.2.5.2 Delete One or More Endpoint(s)

The **Endpoints** page provides the mechanism to delete a group of endpoints from Key Vault at one time. You can also delete a single endpoint from this page.

To delete one or more endpoints do the following:

1. Log in to the Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **Endpoints** tab to get to the **Endpoints** page.

   The **Endpoints** page lists all the endpoints currently registered or enrolled.
3. Select the check box(es) to the left of the endpoint(s) you want to delete. You may select more than one.
4. Click **Delete**.
5. Click **OK** in the confirmation dialog box that appears.

> **See Also:**
>
> "Performing Actions and Searches (page 3-15)"

### 7.2.5.3 Delete one Endpoint

The **Endpoint Details** page provides a consolidated view for the selected endpoint including a mechanism to delete the endpoint from Key Vault..

To delete an endpoint follow these steps:

1. Log in to the Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **Endpoints** tab to get to the **Endpoints** page.
3. The **Endpoints** page lists all the endpoints currently registered or enrolled.
4. Click on the endpoint name you want to delete. The **Endpoint Details** page appears.
5. Click **Delete**.
6. Click **OK** to confirm.

> **See Also:**
>
> "Performing Actions and Searches (page 3-15)"

### 7.2.5.4 Suspend one Endpoint

You can suspend an endpoint temporarily for security reasons, and reinstate the endpoint once the threat has passed. When you suspend an endpoint, its status will change from **Enrolled** to **Suspended**.

To suspend an endpoint do the following:

1. Log in to the Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **Endpoints** tab to get to the **Endpoints** page.
3. The **Endpoints** page lists all the endpoints currently registered or enrolled.
4. Click on the endpoint name you want to suspend. The **Endpoint Details** page appears.
5. Click **Suspend**.
6. A confirmation message appears asking if you are sure. Click **OK.**
7. When you suspend an endpoint, its **Status** on the **Endpoints** page will be **Suspended**.
8. To enable the endpoint, perform **Steps 1-4**. From the **Endpoint Details** pane click **Enable**. The endpoint **Status** on the **Endpoints** page will now read **Enrolled**.

> **See Also:**
>
> "Performing Actions and Searches (page 3-15)"

### 7.2.5.5 Reenroll an Endpoint

You must reenroll an endpoint to upgrade the endpoint software on the endpoint. You would also reenroll an endpoint to accommodate changes in an Oracle Key Vault deployment, for example, you need to pair a primary Oracle Key Vault server with a new secondary server in a high availability configuration.

The following procedure describes how to reenroll an endpoint:

1. Log in to the Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **Endpoints** tab to get to the **Endpoints** page.
3. The **Endpoints** page lists all of the endpoints in Key Vault.
4. Check the boxes to the left of the endpoints you want to reenroll.
5. Click **Reenroll**.

   A confirmation message appears, saying that the endpoints were reenrolled successfully.

> **Note:**
>
> In Oracle Key Vault 12.2.0.5.0 and earlier, the symlink reference to `okvclient.ora` is not updated during re-enrollment. In Oracle Key Vault 12.2.0.6.0, new `okvclient.jar` option `-o` allows you to overwrite the symlink reference pointing to `okvclient.ora` in the new directory.

A new enrollment token will be generated for each reenrolled and appear in the corresponding **Enrollment Token** column.

You can use this one-time token to reenroll the endpoint.

> **See Also:**
>
> "Task 1: Enroll Endpoint and Download Software (page 8-3)"

## 7.3 Manage Endpoint Access to a Virtual Wallet

You can grant an endpoint access to a virtual wallet, and revoke or modify access when it is no longer necessary. Note, that the endpoint must be granted **Read and Modify** and **Manage Wallet** access privileges on the wallet in order to upload and download security objects to and from Key Vault.

- Grant an Endpoint Access to a Virtual Wallet (page 7-11)
- Revoke Endpoint Access to a Virtual Wallet (page 7-13)
- View Wallet Items (page 7-13)

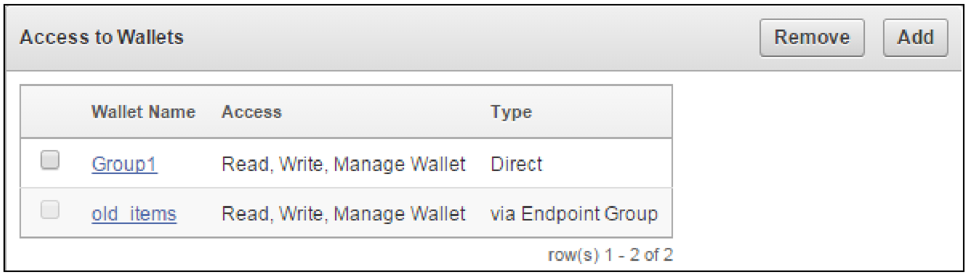
### 7.3.1 Grant an Endpoint Access to a Virtual Wallet

You can grant an endpoint access to a virtual wallet as soon as the endpoint has been added to Oracle Key Vault, when it is still in **registered** status.

To grant an endpoint access to wallets already added to Oracle Key Vault:

1. Log in to the Oracle Key Vault management console as an administrator who has the Key Administrator role.
2. Select the **Endpoints** tab to get to the **Endpoints** page.
3. On the **Endpoints** page, select the endpoint that must have access to the virtual wallet. The **Endpoint Details** page appears with the **Access to Wallets** pane.

**Figure 7-7    Endpoint Details - Access to Wallets**



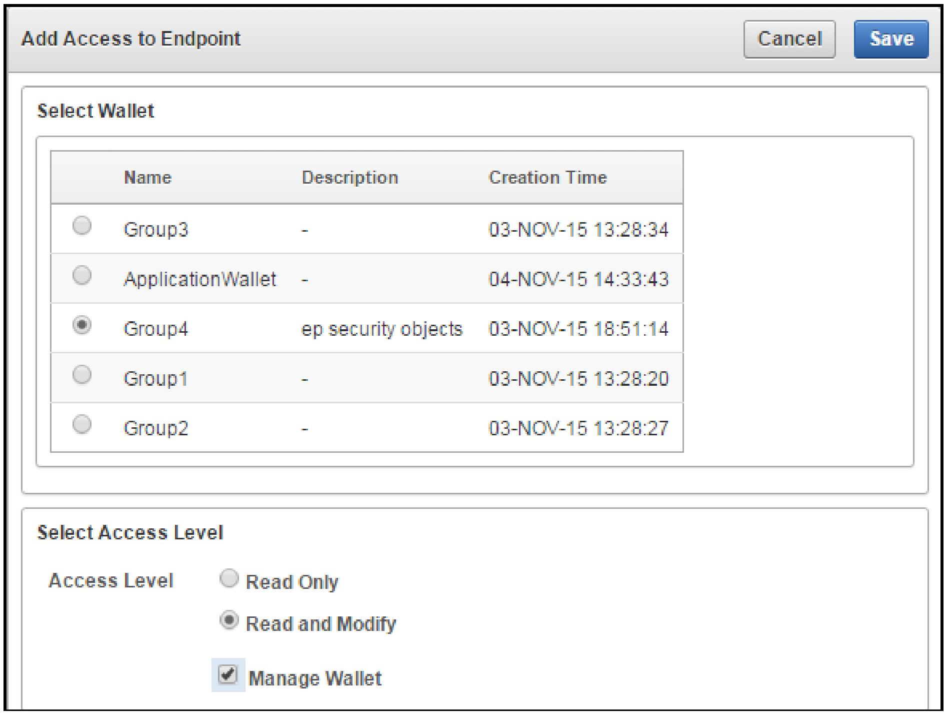
| | Wallet Name | Access | Type |
|---|---|---|---|
| □ | Group1 | Read, Write, Manage Wallet | Direct |
| □ | old_items | Read, Write, Manage Wallet | via Endpoint Group |

row(s) 1 - 2 of 2

4. The **Access to Wallets** pane lists the wallets the endpoint already has access to. Click **Add** to add another wallet to this list.

   The **Add Access to Endpoint** page appears.

**Figure 7-8    Add Access to Endpoint**



| | Name | Description | Creation Time |
|---|---|---|---|
| ○ | Group3 | - | 03-NOV-15 13:28:34 |
| ○ | ApplicationWallet | - | 04-NOV-15 14:33:43 |
| ● | Group4 | ep security objects | 03-NOV-15 18:51:14 |
| ○ | Group1 | - | 03-NOV-15 13:28:20 |
| ○ | Group2 | - | 03-NOV-15 13:28:27 |

Select Access Level

Access Level ○ Read Only
● Read and Modify
☑ Manage Wallet

5. Select a wallet from the available list of wallets shown on the **Add Access to Endpoint** page.
6. Select the desired **Access Level** in the **Select Access Level** pane.
7. Click **Save**.

   You will see a confirmation message indicating that the access mapping succeeded.

> ✎ **See Also:**
>
> "Access Control Options (page 2-8)"

## 7.3.2 Revoke Endpoint Access to a Virtual Wallet

Use the following procedure to revoke access to a virtual wallet for an endpoint:

1. Log in to the Oracle Key Vault management console as an administrator who has the Key Administrator role.
2. Select the **Endpoints** tab to get to the **Endpoints** page.
3. On the **Endpoints** page, select the endpoint name, which will bring you to the **Endpoint Details** page. Look for the **Access to Wallets** pane on this page.

   The **Access to Wallets** pane shows a list of wallets that the endpoint has access to.
4. Select the wallet, you want to revoke access to.
5. Click **Remove**.
6. When the confirmation dialog box asks if you want to remove this access, click **OK**.

   A confirmation message appears, indicating that the access mapping was removed.

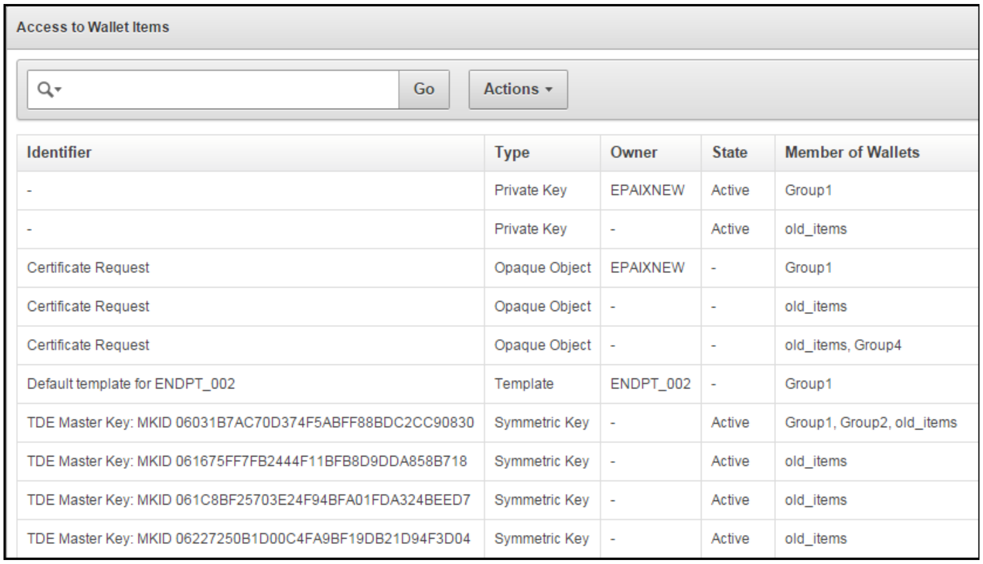
## 7.3.3 View Wallet Items

Wallet items refers to the security objects that the endpoint has access to.

To view these follow these steps:

1. Log in to the Oracle Key Vault management console as an administrator who has the Key Administrator role.
2. Select the **Endpoints** tab to get to the **Endpoints** page,
3. Click the **Endpoint Name** to get to **Endpoint Details**.
4. The **Access to Wallet Items** pane in **Endpoint Details** lists the wallet items that the endpoint has access to.

**Figure 7-9 Endpoint Details - Access to Wallet Items**



| Identifier | Type | Owner | State | Member of Wallets |
|---|---|---|---|---|
| - | Private Key | EPAIXNEW | Active | Group1 |
| - | Private Key | - | Active | old_items |
| Certificate Request | Opaque Object | EPAIXNEW | - | Group1 |
| Certificate Request | Opaque Object | - | - | old_items |
| Certificate Request | Opaque Object | - | - | old_items, Group4 |
| Default template for ENDPT_002 | Template | ENDPT_002 | - | Group1 |
| TDE Master Key: MKID 06031B7AC70D374F5ABFF88BDC2CC90830 | Symmetric Key | - | Active | Group1, Group2, old_items |
| TDE Master Key: MKID 061675FF7FB2444F11BFB8D9DDA858B718 | Symmetric Key | - | Active | old_items |
| TDE Master Key: MKID 061C8BF25703E24F94BFA01FDA324BEED7 | Symmetric Key | - | Active | old_items |
| TDE Master Key: MKID 06227250B1D00C4FA9BF19DB21D94F3D04 | Symmetric Key | - | Active | old_items |

> **See Also:**
>
> - Figure 7-1 (page 7-3)
> - Figure 7-4 (page 7-5)

## 7.4 Associate a Default Wallet with an Endpoint

A default wallet is a type of virtual wallet that security objects are uploaded to when a wallet is not explicitly specified. Default wallets are useful for sharing with other endpoints such as nodes in an Oracle RAC, or primary and standby nodes in Dataguard (DG) by having all endpoints use the same default wallet.

The default wallet must be set during the registration process to ensure that the downloaded endpoint software is configured to use the default wallet.

An enrollment status of **registered** means that the endpoint has been added to Oracle Key Vault, but the endpoint software has not yet been downloaded and installed. This is when you must associate the default wallet with the endpoint.

The endpoint's enrollment status becomes **enrolled** when you download and install the endpoint software to the endpoint. If you set the default wallet after you enroll the endpoint, then you must re-enroll the endpoint to ensure that all future security objects created by the endpoint are automatically associated with that wallet.

## 7.5 Set the Default Wallet for an Endpoint

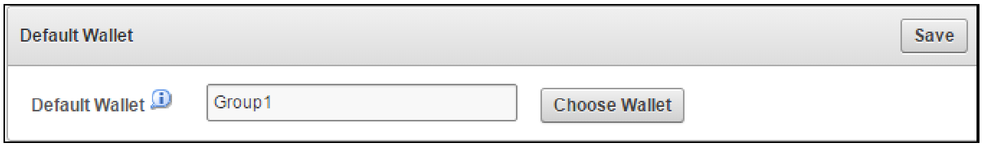
When you set the default wallet for an endpoint, all the endpoint's security objects will be automatically uploaded to this wallet if a wallet is not explicitly specified. Oracle requires that you set the default wallet right after registering the endpoint, and *before* downloading the endpoint software.

To set the default wallet follow the steps below:

1. Log in to the Oracle Key Vault management console as an administrator who has the Key Administrator role.
2. Select the **Endpoints** tab, then click on the endpoint name.

   The **Endpoint Details** page appears.
3. Select **Choose Wallet** in **Default Wallet**.

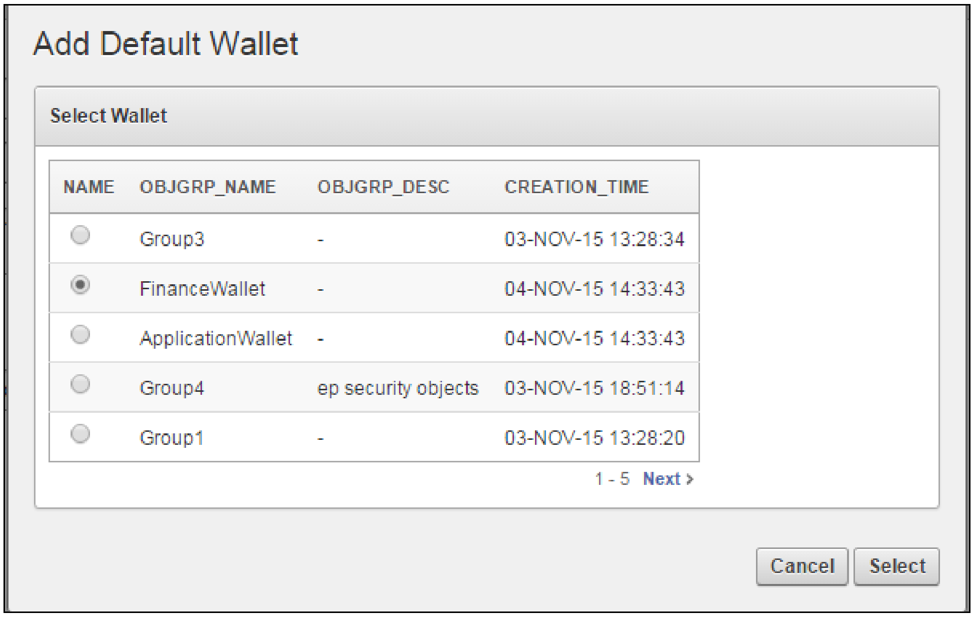
**Figure 7-10 Endpoint Details - Default Wallet**



Default Wallet: Group1

The **Add Default Wallet** page appears displaying a list of available wallets.

**Figure 7-11 Add Default Wallet**

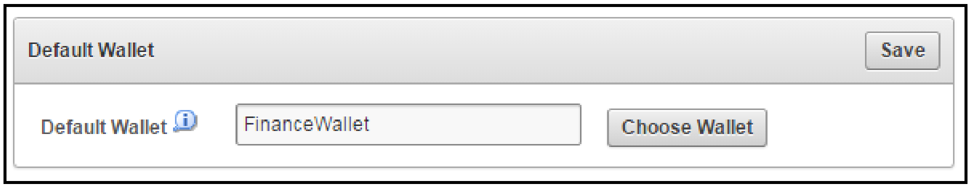


| NAME | OBJGRP_NAME | OBJGRP_DESC | CREATION_TIME |
|---|---|---|---|
| ○ | Group3 | - | 03-NOV-15 13:28:34 |
| ● | FinanceWallet | - | 04-NOV-15 14:33:43 |
| ○ | ApplicationWallet | - | 04-NOV-15 14:33:43 |
| ○ | Group4 | ep security objects | 03-NOV-15 18:51:14 |
| ○ | Group1 | - | 03-NOV-15 13:28:20 |

4. Select a wallet from the list to be the default wallet by clicking the radio button to the left of the wallet. Click **Select**.

   The selected wallet name appears in the **Default Wallet** pane.

**Figure 7-12 Post Default Wallet Selection**



Default Wallet: FinanceWallet

5. Click **Save**.

   A confirmation message appears saying that the update has been made.

# 7.6 Manage Endpoint Groups

An endpoint group is a group of endpoints that share a common set of wallets.

- Create an Endpoint Group (page 7-16)
- Modify Endpoint Group Details (page 7-17)
- Grant an Endpoint Group Access to a Virtual Wallet (page 7-18)
- Remove an Endpoint from an Endpoint Group (page 7-19)
- Delete Endpoint Groups (page 7-20)

## 7.6.1 Create an Endpoint Group

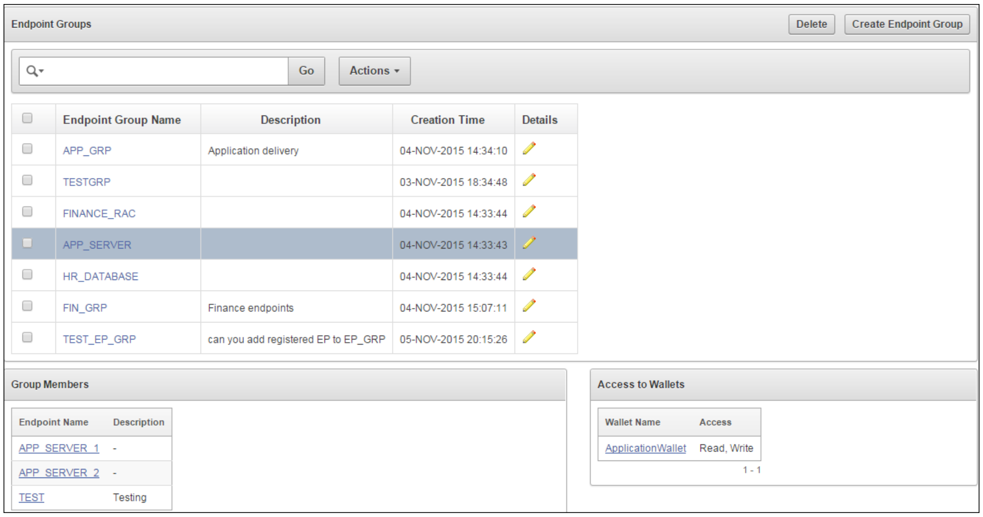
Endpoints that must share a common set of security objects stored in wallets can be grouped into an endpoint group. For example, endpoints using Oracle RAC, Oracle GoldenGate, or Oracle Active Data Guard may need to share keys for access to shared data.

To create an endpoint group:

1. Log in to the Oracle Key Vault management console as a user who has the Key Administrator role.
2. Select the **Endpoints** tab, then **Endpoint Groups**.

   The **Endpoint Groups** page appears.

**Figure 7-13 Endpoint Groups Page**



| □ | Endpoint Group Name | Description | Creation Time | Details |
|---|---|---|---|---|
| □ | APP_GRP | Application delivery | 04-NOV-2015 14:34:10 | ✎ |
| □ | TESTGRP | | 03-NOV-2015 18:34:48 | ✎ |
| □ | FINANCE_RAC | | 04-NOV-2015 14:33:44 | ✎ |
| □ | APP_SERVER | | 04-NOV-2015 14:33:43 | ✎ |
| □ | HR_DATABASE | | 04-NOV-2015 14:33:44 | ✎ |
| □ | FIN_GRP | Finance endpoints | 04-NOV-2015 15:07:11 | ✎ |
| □ | TEST_EP_GRP | can you add registered EP to EP_GRP | 05-NOV-2015 20:15:26 | ✎ |

Group Members

| Endpoint Name | Description |
|---|---|
| APP_SERVER_1 | - |
| APP_SERVER_2 | - |
| TEST | Testing |

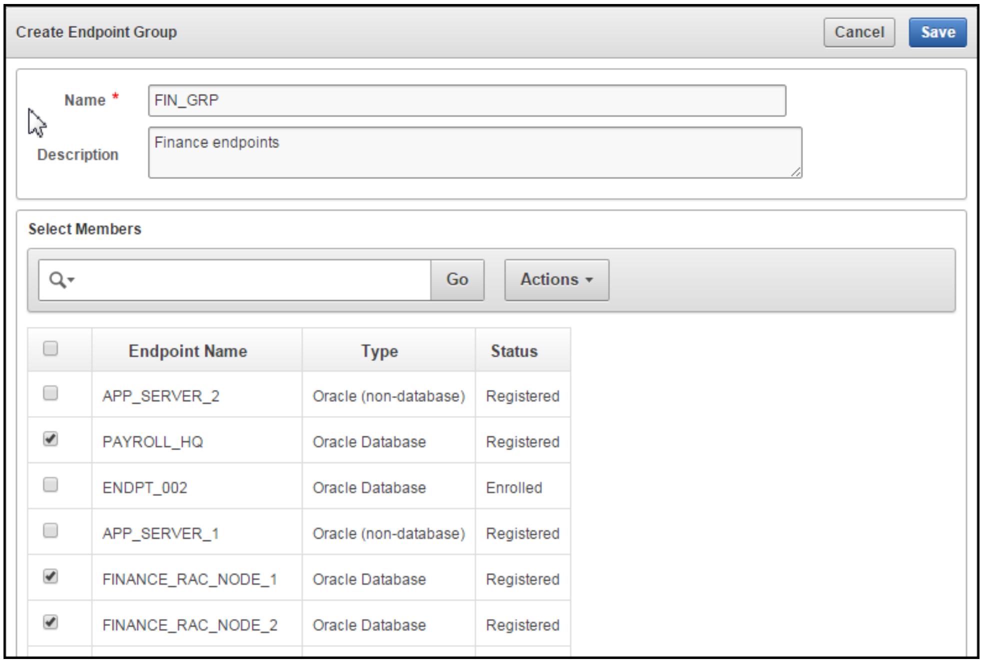
Access to Wallets

| Wallet Name | Access |
|---|---|
| ApplicationWallet | Read, Write |

1 - 1

3. Click **Create Endpoint Group**. The **Create Endpoint Group** page appears.

**Figure 7-14    Create Endpoint Group Page**



Name * FIN_GRP

Description Finance endpoints

| □ | Endpoint Name | Type | Status |
|---|---|---|---|
| □ | APP_SERVER_2 | Oracle (non-database) | Registered |
| ☑ | PAYROLL_HQ | Oracle Database | Registered |
| □ | ENDPT_002 | Oracle Database | Enrolled |
| □ | APP_SERVER_1 | Oracle (non-database) | Registered |
| ☑ | FINANCE_RAC_NODE_1 | Oracle Database | Registered |
| ☑ | FINANCE_RAC_NODE_2 | Oracle Database | Registered |

4. Enter the name of the new group and a brief description. You can add members to the group right away, from the list in the **Select Members** pane, just below **Create Endpoint Group**.
5. The **Select Members** pane lists all the endpoints. To add endpoints to the endpoint group, check the boxes to the left of each endpoint.
6. Click **Save** to complete creating the endpoint group.

   A message appears indicating that the endpoint group has been successfully saved. The new endpoint group now appears in the **Endpoint Groups** page.

**See Also:**

- Figure 7-13 (page 7-16)
- Figure 7-14 (page 7-17)
- "Modify Endpoint Group Details (page 7-17)"
- "Performing Actions and Searches (page 3-15)"

## 7.6.2 Modify Endpoint Group Details

You can add endpoints and access mappings to an endpoint group after creating the endpoint group. An endpoint can belong to more than one endpoint group. You cannot add one endpoint group to another endpoint group.

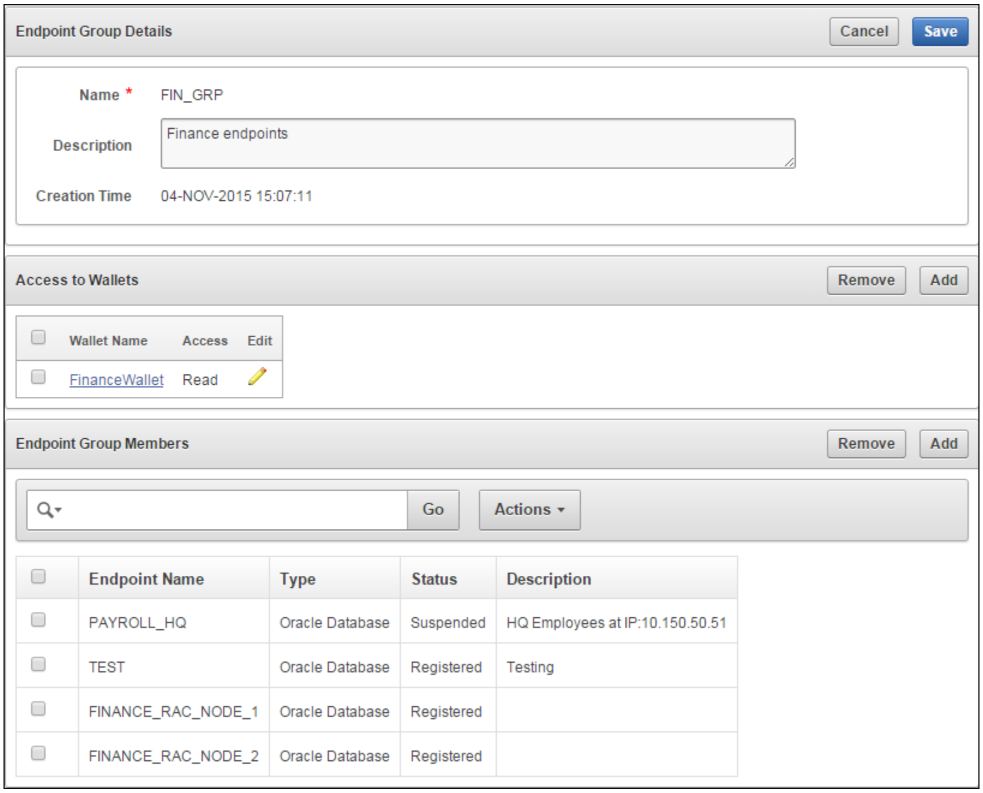
To modify an endpoint group after creating it, follow these steps:

1. Log in to the Oracle Key Vault management console as a user who has the Key Administrator role.
2. Select the **Endpoints** tab, and then select **Endpoint Groups**.

   The **Endpoint Groups** page appears.
3. Click the edit pencil icon in the **Details** column corresponding to the endpoint group.

   The **Endpoint Group Details** page appears.

**Figure 7-15 Endpoint Group Details Page**



Name * FIN_GRP

Description Finance endpoints

Creation Time 04-NOV-2015 15:07:11

| □ | Wallet Name | Access | Edit |
|---|---|---|---|
| □ | FinanceWallet | Read | ✎ |

| □ | Endpoint Name | Type | Status | Description |
|---|---|---|---|---|
| □ | PAYROLL_HQ | Oracle Database | Suspended | HQ Employees at IP:10.150.50.51 |
| □ | TEST | Oracle Database | Registered | Testing |
| □ | FINANCE_RAC_NODE_1 | Oracle Database | Registered | |
| □ | FINANCE_RAC_NODE_2 | Oracle Database | Registered | |

4. Modify the description as needed.

   Add or remove access to wallets endpoint group members by clicking **Add**.
5. Click **Save**.

> **See Also:**
>
> - Figure 7-13 (page 7-16)
> - Figure 7-15 (page 7-18)

## 7.6.3 Grant an Endpoint Group Access to a Virtual Wallet

The following procedure grants an endpoint group access to an existing virtual wallet:

1. Log in to the Oracle Key Vault management console as a user who has the Key Administrator role.
2. Select the **Endpoints** tab, and then **Endpoint Groups**.
3. Click the pencil icon in the **Details** column corresponding to the endpoint group. The **Endpoint Group Details** page appears.
4. In the **Access to Wallets** pane, click **Add**.
5. Select a virtual wallet from the available list.
6. Select an **Access Level**:
   - **Read Only**: This level grants the endpoint group read access to the virtual wallet and its items.
   - **Read and Modify**: This level grants the endpoint group read and write access to the virtual wallet and its items.
7. Select the **Manage Wallet** check box if you want endpoints to:
   - Add or remove objects from the virtual wallet.
   - Grant other endpoints or endpoint groups access to the virtual wallet.
8. Click **Save**.

   A message appears, indicating that the access mapping was successful.

## 7.6.4 Remove an Endpoint from an Endpoint Group

You can remove an endpoint from an endpoint group. This will remove all access to wallets associated with that endpoint group unless the endpoint has been separately granted access to the wallet(s) directly or through another endpoint group. You may remove more than one endpoint at the same time.

To remove an endpoint from an endpoint group, follow these steps:

1. Log in to the Oracle Key Vault management console as a user who has the Key Administrator role.
2. Select the **Endpoints** tab, and then select **Endpoint Groups**.

   The **Endpoint Groups** page appears.
3. Click the edit pencil icon next in the **Details** column corresponding to the endpoint group.

   The **Endpoint Group Details** page appears.
4. In the **Endpoint Group Members** pane, check the box(es) to the left of the endpoint names to be removed.
5. Click **Remove**.
6. When the confirmation dialog box asks if you want to remove the endpoint from the group, click **OK**.

   A dialog box appears, indicating that the endpoint has been successfully removed from the group.

## 7.6.5 Delete Endpoint Groups

You can delete endpoint groups, if their member endpoints no longer require access to the same virtual wallets. This action removes the shared access of member endpoints to wallets, not the endpoints themselves.

The following procedure describes how to delete an endpoint group from Key Vault:

1. Log in to the Oracle Key Vault management console as a user who has the Key Administrator role.
2. Select the **Endpoints** tab, and then select **Endpoint Groups**.

   This brings up the **Endpoint Group** page.
3. Check the box(es) to the left of the endpoint group name.
4. Click **Delete**.
5. When the confirmation dialog box asks if you want to delete the endpoint group(s), click **OK** to confirm.

# 7.7 Manage Endpoint Details

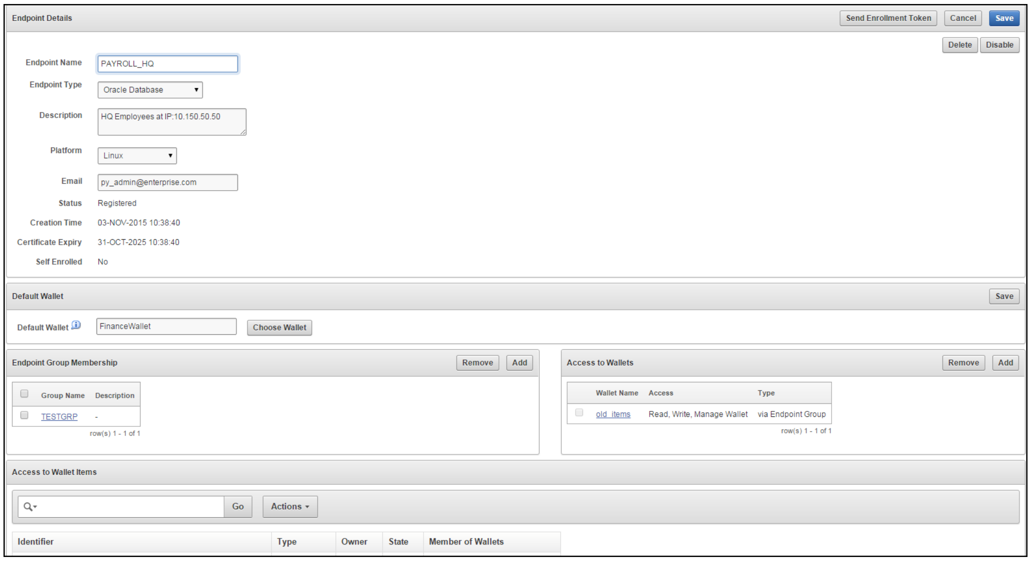
After registering or enrolling the endpoint you can modify the endpoint name, type, description, platform, and email as needed. You can add the endpoint to an endpoint group, and upgrade the software on the endpoint.

- About Endpoint Details (page 7-20)
- Modify Endpoint Details (page 7-21)
- Add an Endpoint to an Endpoint Group (page 7-22)
- Configuring Global Endpoint Configuration Parameters (page 7-24)
- Delete an Endpoint from an Endpoint Group (page 7-25)
- Upgrade Endpoint Software (page 7-26)

## 7.7.1 About Endpoint Details

The endpoint details page provides a consolidated view of the endpoint. From here you can modify endpoint details and complete endpoint management tasks.

**Figure 7-16    Endpoint Details Page**



Endpoint Details

| | |
|---|---|
| Endpoint Name | PAYROLL_HQ |
| Endpoint Type | Oracle Database |
| Description | HQ Employees at IP:10.150.50.50 |
| Platform | Linux |
| Email | py_admin@enterprise.com |
| Status | Registered |
| Creation Time | 03-NOV-2015 10:38:40 |
| Certificate Expiry | 31-OCT-2025 10:38:40 |
| Self Enrolled | No |

Default Wallet

Default Wallet: FinanceWallet

Endpoint Group Membership

| □ | Group Name | Description |
|---|---|---|
| □ | TESTGRP | - |

row(s) 1 - 1 of 1

Access to Wallets

| | Wallet Name | Access | Type |
|---|---|---|---|
| □ | old_items | Read, Write, Manage Wallet | via Endpoint Group |

row(s) 1 - 1 of 1

Access to Wallet Items

| Identifier | Type | Owner | State | Member of Wallets |
|---|---|---|---|---|

**See Also:**

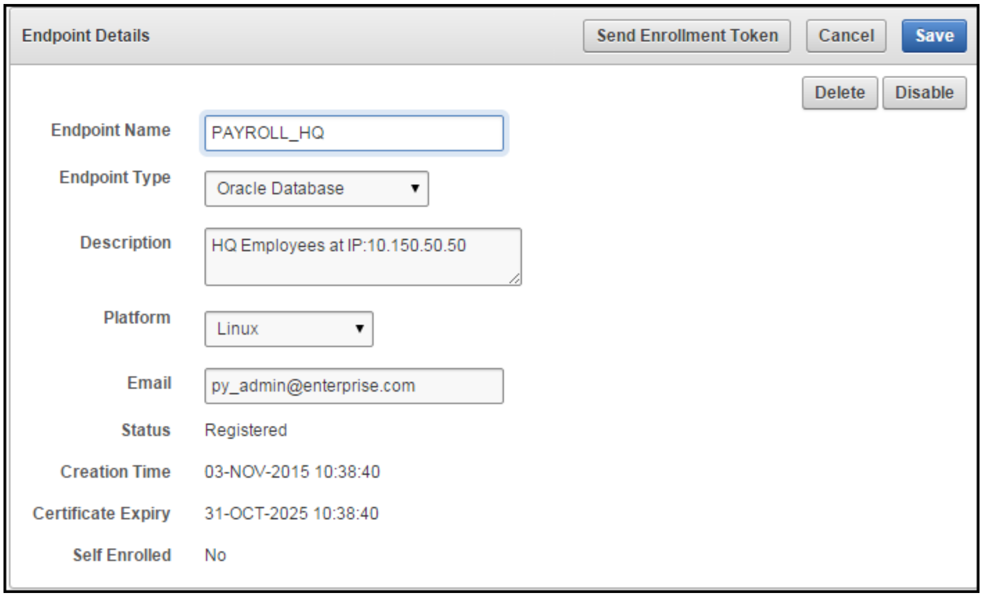
- **Default Wallet** Figure 7-10 (page 7-15)
- **Endpoint Group Membership** Figure 7-19 (page 7-23)
- **Access to Wallets Figure 7-7 (page 7-12)**
- **Access to Wallet Items** Figure 7-9 (page 7-14)

## 7.7.2 Modify Endpoint Details

You can modify the endpoint name, type, platform and email from the **Endpoint Details** page as follows:

1. Log in to the Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **Endpoints** tab to get to the **Endpoints** page.
3. Click the name of the endpoint to get to the **Endpoint Details** page.

**Figure 7-17 Endpoint Details Pane**



| | |
|---|---|
| Endpoint Name | PAYROLL_HQ |
| Endpoint Type | Oracle Database |
| Description | HQ Employees at IP:10.150.50.50 |
| Platform | Linux |
| Email | py_admin@enterprise.com |
| Status | Registered |
| Creation Time | 03-NOV-2015 10:38:40 |
| Certificate Expiry | 31-OCT-2025 10:38:40 |
| Self Enrolled | No |

4. Modify any of the following: endpoint name, endpoint type, description, platform, email as needed.
5. Click **Save**.

## 7.7.3 Add an Endpoint to an Endpoint Group

You can add an endpoint to an endpoint group if you want shared access to wallets as follows:

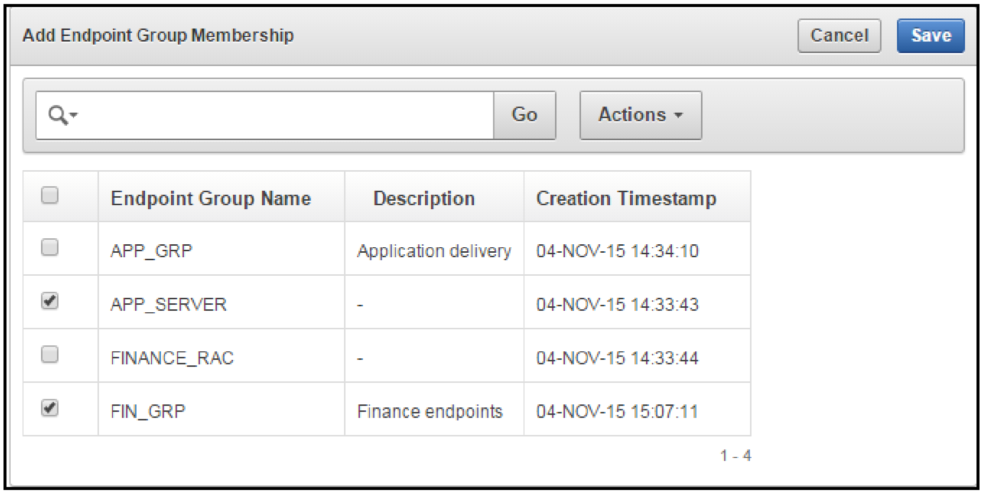
1. Log in to the Oracle Key Vault management console as an administrator who has the Key Administrator role.
2. Select the **Endpoints** tab.

   The **Endpoints** page appears.
3. Select the endpoint you want to add to a group.

   The **Endpoint Details** page appears.
4. Click **Add** in **Endpoint Group Membership**.

   The **Add Endpoint Group Membership** page appears.

**Figure 7-18 Adding Endpoint to Endpoint Group**



| ☐ | Endpoint Group Name | Description | Creation Timestamp |
|---|---|---|---|
| ☐ | APP_GRP | Application delivery | 04-NOV-15 14:34:10 |
| ☑ | APP_SERVER | - | 04-NOV-15 14:33:43 |
| ☐ | FINANCE_RAC | - | 04-NOV-15 14:33:44 |
| ☑ | FIN_GRP | Finance endpoints | 04-NOV-15 15:07:11 |

1 - 4

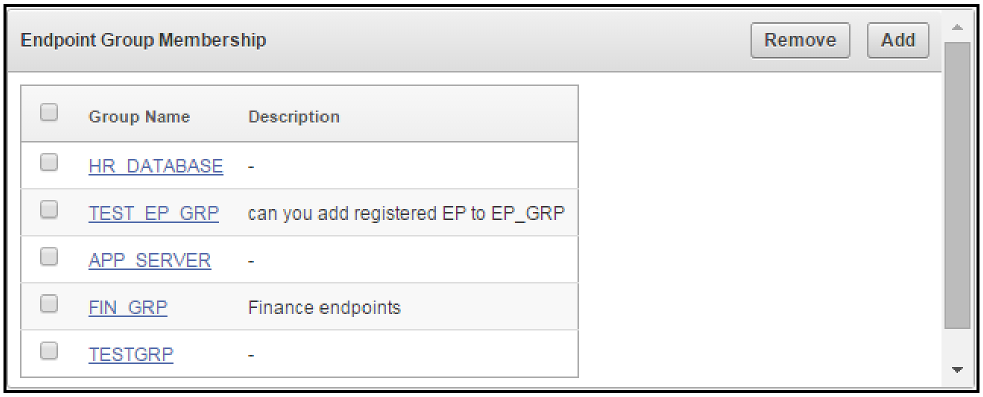
A list of endpoint groups is displayed under **Endpoint Group Name**.

5. Check the box(es) to the left of the endpoint group(s) you want to add the endpoint to.
6. Click **Save**.

   A message appears saying that the endpoint has been added to the group.

   You will see the checked endpoint groups in the Endpoint Group Membership pane.

**Figure 7-19 Endpoint Details - Endpoint Group Membership**



| ☐ | Group Name | Description |
|---|---|---|
| ☐ | HR_DATABASE | - |
| ☐ | TEST_EP_GRP | can you add registered EP to EP_GRP |
| ☐ | APP_SERVER | - |
| ☐ | FIN_GRP | Finance endpoints |
| ☐ | TESTGRP | - |

### See Also:

Add endpoint to group Figure 7-18 (page 7-23)

Create an endpoint group described in "Create an Endpoint Group (page 7-16)"

## 7.7.4 Configuring Global Endpoint Configuration Parameters

Users with system administrator role can centrally update certain endpoint configuration parameters in the Oracle Key Vault Management Console. This feature enables system administrators to set certain endpoint configuration parameters globally, i.e. for all endpoints, or on a per-endpoint basis. It simplifies the process of managing multiple endpoints for system administrators.

Endpoint specific parameters if set take precedence over global parameters. Global parameters, if set will take effect when endpoint-specific parameters are cleared. OKV will use the default system parameters if both global and endpoint specific parameters are cleared or not set from OKV management console.

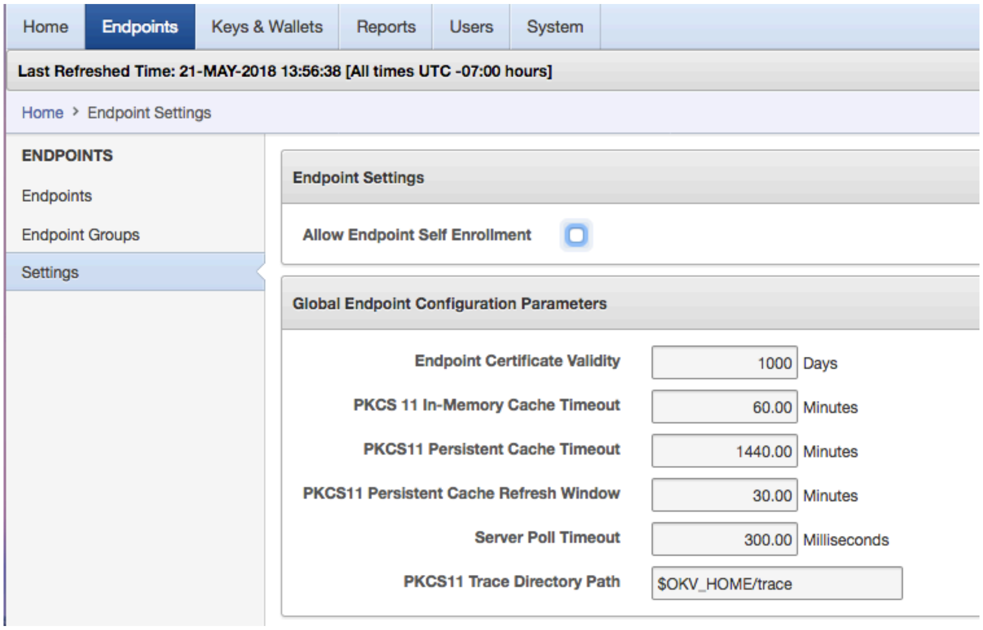
The configuration parameter values set in the OKV management console are pushed to endpoints dynamically. After configuration parameters have been set in the OKV Management Console, the next time the endpoint contacts the OKV server, it will get the configuration parameters update. Endpoint configuration parameter update is best-effort. In case of error, the update is not applied. Both okvutil and PKCS11 library can get and apply the endpoint configuration updates.

To configure global endpoint configuration parameters:

1. Log in to the Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **Endpoints** tab, and then **Settings** from the left side bar.

   The **Endpoint Settings** page is displayed.

**Figure 7-20 Endpoint Settings**



Last Refreshed Time: 21-MAY-2018 13:56:38 [All times UTC -07:00 hours]

Allow Endpoint Self Enrollment □

Endpoint Certificate Validity 1000 Days

PKCS 11 In-Memory Cache Timeout 60.00 Minutes

PKCS11 Persistent Cache Timeout 1440.00 Minutes

PKCS11 Persistent Cache Refresh Window 30.00 Minutes

Server Poll Timeout 300.00 Milliseconds

PKCS11 Trace Directory Path $OKV_HOME/trace

3. In the **Global Endpoint Configuration Parameters** section, configure the following settings:
   - **Endpoint Certificate Validity**

Specify the number of days for which the current endpoint certificate is valid.

- **PKCS 11 In-Memory Cache Timeout**

  Specify the duration for which the master key is available after it is cached in the in-memory cache. The value is specified in minutes. For more information about the PKCS 11 In-Memory Cache Timeout setting, see PKCS11_CACHE_TIMEOUT Parameter (page 11-21).

  **PKCS 11 Cache Persistent Timeout**

  Specify the duration for which the master key is available after it is cached in the persistent master key cache. The value is specified in minutes. For more information about the PKCS 11 Cache Persistent Timeout setting, see PKCS11_PERSISTENT_CACHE_TIMEOUT Parameter (page 11-21).

- **PKCS 11 Persistent Cache Refresh Window**

  Specify the duration to extend the period of timefor which the master key is available after it is cached in the persistent master key cache. The value is specified in minutes. For more information about the PKCS 11 Persistent Cache Refresh Window setting, see PKCS11_PERSISTENT_CACHE_REFRESH_WINDOW Parameter (page 11-22).

- **Server Poll Timeout**

  Specify a timeout for a client's attempt to connect to an Oracle Key Vault server, before trying the next server in the list. The default value is 300 (milliseconds).

- **PKCS 11 Trace Directory Path**

  Specify a directory to save the trace files.

4. Click **Save**.

The endpoint configuration settings are saved.

## 7.7.5 Delete an Endpoint from an Endpoint Group

You can delete an endpoint from an endpoint group if the endpoint no longer needs shared access to wallets as follows:

1. Log in to the Oracle Key Vault management console as an administrator who has the Key Administrator role.
2. Select the **Endpoints** tab.

   The **Endpoints** page appears.
3. Select the endpoint you want to delete from a group.

   The **Endpoint Details** page appears.
4. Check the box(es) in **Endpoint Group Membership** to the left of the endpoint group(s) you want to remove the endpoint from.
5. Click **Remove**.

   A confirmation message will ask if you want to delete the endpoint from the selected endpoint group(s). Click **OK**.

**See Also:**

- Add endpoint to group Figure 7-18 (page 7-23)
- Create an endpoint group described in "Create an Endpoint Group (page 7-16)"

## 7.7.6 Upgrade Endpoint Software

To upgrade to the latest endpoint software for an enrolled endpoint, you can download the endpoint software without having to reenroll the endpoint.

To download the latest version of the endpoint software follow **Steps 1-4** of "Task 1: Enroll Endpoint and Download Software (page 8-3)".

**Step 4** brings up the **Enroll Endpoint & Download Software** page.

On the **Enroll Endpoint & Download Software** page, do the following:

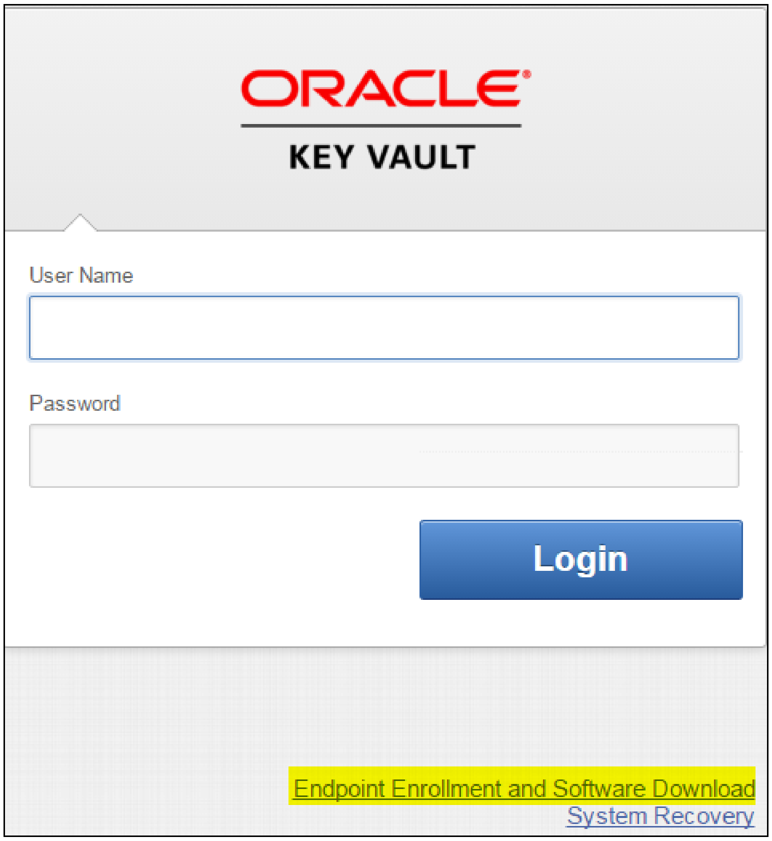
1. Log in to the endpoint server as the endpoint administrator.
2. Connect to the Oracle Key Vault management console.

   For example:

   ```
   https://192.0.2.254
   ```
3. The login page to the Oracle Key Vault management console appears.

   *Do not log in*.

Figure 7-21    Key Vault Management Console Login Screen

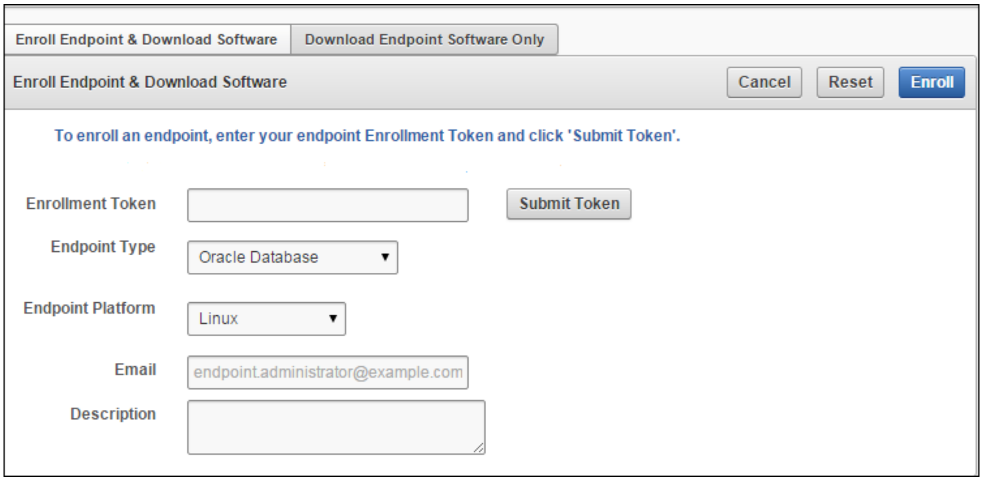


4. Click the highlighted link **Endpoint Enrollment and Software Download** below **Login**.

   The **Enroll Endpoint & Download Software** page appears with two tabs:

   - **Enroll Endpoint & Download Software**
   - **Download Endpoint Software Only**

Figure 7-22    Enroll Endpoint & Download Software Page


Enroll Endpoint & Download Software
Download Endpoint Software Only
Enroll Endpoint & Download Software
Cancel
Reset
Enroll
Submit Token

To enroll an endpoint, enter your endpoint Enrollment Token and click 'Submit Token'.

Enrollment Token

Endpoint Type: Oracle Database

Endpoint Platform: Linux

Email: endpoint.administrator@example.com

Description

Note that **Figure 8-2** has been trimmed and contains the following text between **Download Endpoint Software** and the **Cancel**, **Reset**, and **Enroll** buttons on the right:

“**To enroll an endpoint, enter your endpoint Enrollment Token and click 'Submit Token'. Update the endpoint details if necessary and click 'Enroll' to complete the enrollment. Download the endpoint package when prompted.**"

5. Click the **Download Endpoint Software Only** tab.

   The **Download Endpoint Software Only** page appears.

6. Select the endpoint platform from the drop down **Platform** menu and click **Download**.

7. Save the file:`okvclient.jar` to a desired location.

8. Ensure that you have the necessary administrative privileges to install software on the endpoint.

9. Ensure that you have JDK 1.5 or later installed, and that the `PATH` environment variable includes the `java` executable (in the `JAVA_HOME/bin` directory).

   Oracle Key Vault supports JDK versions 1.5, 1.6, 7, and 8.

10. Run the Shell utility `ORAENV` or `source ORAENV` command to set the correct environment variables on Oracle Database servers.

11. Check that the environment variables `ORACLE_BASE` and `ORACLE_HOME` are correctly set.

    If you used `ORAENV` to set these variables, you must verify that `ORACLE_BASE` points to the root directory for Oracle Databases, and that `ORACLE_HOME` points to a sub-directory under `ORACLE_BASE` where an Oracle Database is installed.

12. Navigate to the directory in which you saved the `okvclient.jar` file.

13. Run the `java` command to install the `okvclient.jar` file.

    ```
    java -jar okvclient.jar -d /home/oracle/okvutil -v
    ```

    In this specification:

    - The `-d` argument specifies the directory location for the endpoint software and configuration files, in this case `/home/oracle/okvutil`.

      The environment variable `$OKV_HOME` refers to the directory where the endpoint software is installed, in this case `/home/oracle/okvutil`.

    - The `-v` argument writes the installation logs to the `$OKV_HOME/log/okvutil.deploy.log` file at the server endpoint.

    > **Note:**
    >
    > `-o` is an optional argument that allows you to overwrite the symlink reference to `okvclient.ora`, when `okvclient.jar` is deployed in a directory other than the original directory. This argument is used only when re-enrolling an endpoint.

14. The installation process prompts for a password. You can enter a password to create a **password-protected** wallet or create an **auto-login** wallet without a password as described below:

- A password-protected wallet is an Oracle wallet file that store the endpoint's credentials to access Oracle Key Vault. This password will be required whenever the endpoint connects to Oracle Key Vault.

  Create a password-protected wallet by entering a password between 8 and 30 characters. Then press **Enter**.
- Create an auto-login wallet by simply clicking **Enter**.

  No password will be required when the endpoint connects to Oracle Key Vault. An auto-login wallet enables endpoint provisioning without human intervention.

```
Enter new Key Vault endpoint password (<enter> for auto-login):
Key_Vault_endpoint_password
Confirm new endpoint password: Key_Vault_endpoint_password
```

The installation proceeds and completes with the following message:

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

A successful installation of the endpoint software creates the following directories:

- `bin`: contains the `okvutil` program, the `root.sh` and `root.bat` scripts, and the binary files `okveps.x64` and `okveps.x86`
- `conf`: contains the configuration file `okvclient.ora`
- `jlib`: contains the Java library files
- `lib`: contains the file `liborapkcs.so`
- `log`: contains the log files
- `ssl`: contains the TLS-related files and wallet files. The wallet files contain the endpoint credentials to connect to Oracle Key Vault.

  The `ewallet.p12` file refers to a password-protected wallet. The `cwallet.sso` file refers to an auto-login wallet.

15. On UNIX platforms, the `liborapkcs.so` file contains the library that the Oracle database uses to communicate with Oracle Key Vault. On Windows platforms, the `liborapkcs.dll` file contains the library that the Oracle database uses to communicate with Oracle Key Vault.

    If you are planning to use a TDE direct connection, then run `root.sh` on Oracle Linux x86-64, Solaris, AIX, and HP-UX (IA) installations. The `liborapkcs.so` file is copied to the following directory: /opt/oracle/extapi/64/hsm/oracle/1.0.0

    On Windows installations, run `root.bat`. The `liborapkcs.dll` file is copied to C:\oracle\extapi\64\hsm\oracle\1.0.0

    Log in as the root user and run the `root.sh` script. On Windows installations, run `root.bat`.

    ```
    $ sudo bin/root.sh
    ```

    ```
    bin\root.bat
    ```

    Or:

    ```
    $ su -
    # bin/root.sh
    ```

On Windows platforms, you are prompted for the version of the RDBMS in use when you execute `root.bat`. Switch out of user root after completing this step.

16. Run the `okvutil list` command to verify that the endpoint software installed correctly, and that the endpoint can connect to the Oracle Key Vault server.

    If the endpoint is able to connect to Key Vault, a No objects found message appears:

    ```
    $ ./okvutil list
    No objects found
    ```

    If a **Server connect failed** message appears at any time, you must troubleshoot the installation for possible issues. First check that environment variables are correctly set.

17. You can get help on the endpoint software with the `-h` option:

    ```
    java -jar okvclient.jar -h
    ```

    The following output appears:

    ```
    Oracle Key Vault Release 12.2.0.12.0 (2020-03-15 15:36:49.839 PDT)
    Production on Fri Mar 15 19:55:31 PDT 2018
    Copyright (c) 1996, 2020 Oracle. All Rights Reserved.
    Usage: java -jar okvclient.jar [-h | -help] [[-v | -verbose] [-d
    <destination directory>] [-o]]
    ```

18. After installation Oracle recommends that you securely delete the endpoint software file `okvclient.jar`.

**See Also:**

- Special Notes About Endpoint Provisioning (page 8-9) to check environment and setup
- Using an Online Master Key with Oracle Key Vault (page 11-10) for TDE

8

# Enrolling Endpoints for Oracle Key Vault

After a Key Vault system administrator registers an endpoint, an endpoint administrator must enroll and provision the endpoint to manage security objects using Key Vault.

## 8.1 About Endpoints for Oracle Key Vault

Endpoints are clients of Oracle Key Vault that use the appliance to store and manage their security objects, share them with trusted peers, and retrieve them when needed. These clients can be systems like Oracle database servers, Oracle middleware servers, operating systems, and other information systems.

A Key Vault system administrator first adds (or registers) the endpoint to Key Vault, then sends the endpoint's enrollment token (generated during registration) to the endpoint administrator.The endpoint administrator verifies the enrollment token before enrolling and provisioning the endpoint. An enrolled endpoint can upload, download, and manage security objects using Key Vault.

**See Also:**

- Administrative Roles within Oracle Key Vault (page 2-8)
- Endpoint Administrators (page 2-11)

## 8.2 Overview of Endpoint Enrollment and Provisioning

Endpoint enrollment is a three step process performed by two kinds of administrative users summarized in the following table.

**Table 8-1 Summary of Endpoint Enrollment**

| Step # | Task(s) | Performed by | Endpoint Status (as seen on Key Vault Management Console) |
|---|---|---|---|
| 1. | 1. Add/register the endpoint to Key Vault. An enrollment token for the endpoint is generated.<br>2. Send the enrollment token to the endpoint administrator to complete the enrollment process. | Key Vault system administrator on Key Vault | Registered |
| 2. | 1. Verify enrollment token.<br>2. Submit enrollment token to download endpoint software `okvclient.jar` to the endpoint. | Endpoint administrator via Key Vault User Interface | Enrolled |
| 3. | Install `okvclient.jar` at the endpoint. | Endpoint administrator on endpoint | Enrolled |

Endpoint enrollment ensures that only authorized endpoints can communicate with Key Vault because the utilities needed to communicate are bundled with the endpoint software `okvclient.jar`.

`okvclient.jar` contains the following:

- A TLS certificate and private key that the endpoint uses to authenticate itself to Oracle Key Vault
- A TLS certificate for Oracle Key Vault that serves as the root CA
- Endpoint libraries and utilities
- Additional information like the Key Vault IP address that is used by `okvutil` to create the `okvclient.ora` configuration file

In an Oracle Real Application Clusters (RAC) environment, you must enroll and provision each Oracle RAC node as an endpoint.

**See Also:**

- "Types of Endpoint Enrollment (page 7-2)" for more information about adding an endpoint to Key Vault
- "Endpoint okvclient.ora Configuration File (page 8-11)"

# 8.3 Finalizing Enrollment and Provisioning

To enroll and provision a registered endpoint an endpoint administrator must complete two tasks.

- Task 1: Enroll Endpoint and Download Software (page 8-3)
- Task 2: Install Oracle Key Vault Software on the Endpoint (page 8-6)

## 8.3.1 Task 1: Enroll Endpoint and Download Software

You will need the endpoint's enrollment token to download the endpoint software `okvclient.jar`.

After registering the endpoint the Key Vault system administrator sends this endpoint's enrollment token to the endpoint administrator by email or other out-of-band method.

To learn more about endpoint registration please see the **See Also** section following **Task 1**.

To download the endpoint software:

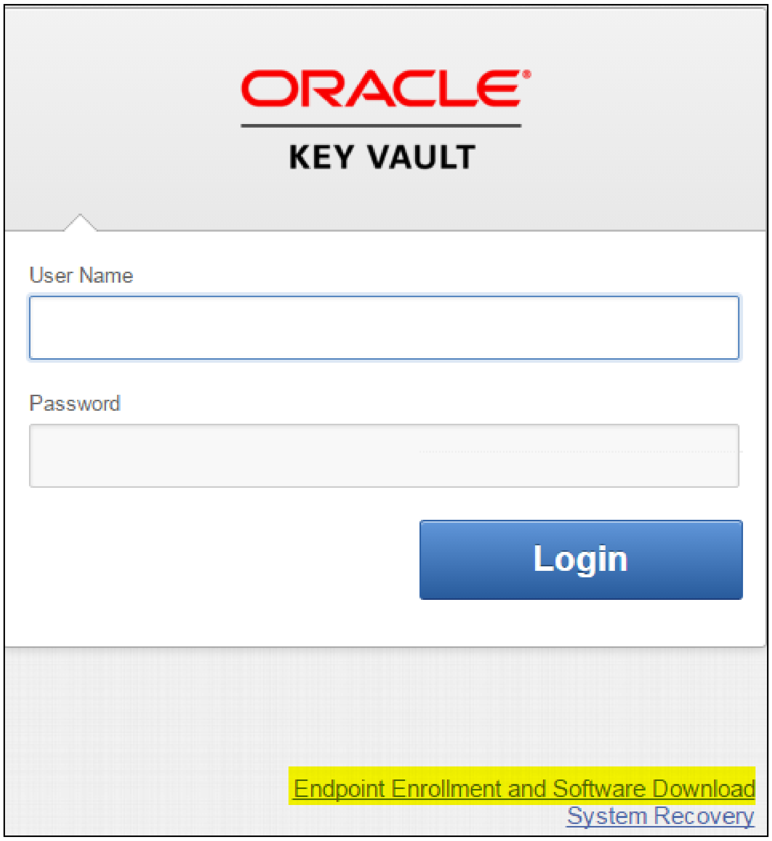
1. Log in to the endpoint server as the endpoint administrator.
2. Connect to the Oracle Key Vault management console.

   For example:

   ```
   https://192.0.2.254
   ```
3. The login page to the Oracle Key Vault management console appears.

   *Do not log in*.

**Figure 8-1    Key Vault Management Console Login Screen**

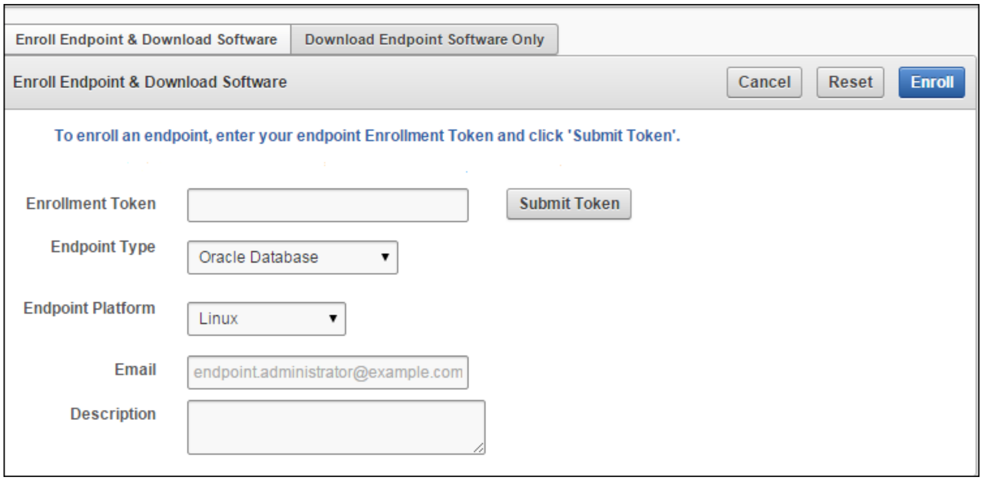


4. Click the highlighted link **Endpoint Enrollment and Software Download** below **Login**.

   The **Enroll Endpoint & Download Software** page appears with two tabs:

   - **Enroll Endpoint & Download Software**
   - **Download Endpoint Software Only**

**Figure 8-2    Enroll Endpoint & Download Software Page**


Enroll Endpoint & Download Software
Download Endpoint Software Only
Enroll Endpoint & Download Software
Cancel
Reset
Enroll
Submit Token

To enroll an endpoint, enter your endpoint Enrollment Token and click 'Submit Token'.

Enrollment Token

Endpoint Type: Oracle Database

Endpoint Platform: Linux

Email: endpoint.administrator@example.com

Description

Note that **Figure 8-2** has been trimmed and contains the following text between **Download Endpoint Software** and the **Cancel**, **Reset**, and **Enroll** buttons on the right:

“**To enroll an endpoint, enter your endpoint Enrollment Token and click 'Submit Token'. Update the endpoint details if necessary and click 'Enroll' to complete the enrollment. Download the endpoint package when prompted.**"

5. Click **Enroll Endpoint & Download Software**.

   The next step depends on how the endpoint was added (or registered with) to Key Vault:

   - If the endpoint was registered by a Key Vault System Administrator do the following:
     - Enter the endpoint's enrollment token in **Enrollment Token**, and click **Submit Token**.

       If the token is valid, a valid token message appears to the right of **Submit Token.**

       The fields **Endpoint Type**, **Endpoint Platform**, **Email** and **Description** get automatically populated with the values entered during endpoint registration.

       If the token is invalid, an invalid token message appears. Check the token and retry.
   - If the endpoint was registered by self-enrollment do the following:
     - Self-enrolled endpoints have no enrollment token, so skip the step of validating the token.
     - Enter values for the following fields:

       a. **Endpoint Type:** Can be Oracle Database, Oracle (non-database), or Other. If you are using TDE, you must enter Oracle Database.

       b. **Endpoint Platform:** One of **Linux**, **Solaris SPARC**, **Solaris x64**, **AIX**, **HP-UX**, **Windows**.

       c. **Email:** Email address of the endpoint administrator for notification purposes. This is optional but recommended.

       d. **Description**: This is optional but strongly recommended for ease of identification in reports. Enter meaningful and identifying information for the endpoint.

6. Click **Enroll** on top right.

   A directory window appears and prompts you to save the endpoint software file: `okvclient.jar`.

   Navigate to the folder where you want to save the file.

7. Save the file in a secure directory with appropriate permissions in place so it cannot be read or copied by others.

8. Verify that the file has been downloaded. If the download fails for any reason, you must obtain a new enrollment token from the key administrator for the endpoint and repeat **Steps 6** and **7.** Note that if you did not download the file to the endpoint system, you must use an out-of-band method to copy the file to that system and install it there.

9. Now you are ready to install okvclient.jar file on the endpoint as described in Task 2: Install Oracle Key Vault Software on the Endpoint (page 8-6).

> **See Also:**
>
> - "Add an Endpoint as a Key Vault System Administrator (page 7-3)"
> - "Add an Endpoint Using Self-Enrollment (page 7-6)"

## 8.3.2 Task 2: Install Oracle Key Vault Software on the Endpoint

To install the software okvclient.jar on the endpoint:

1. Ensure that you have the necessary administrative privileges to install software on the endpoint.
2. Ensure that you have JDK 1.5 or later installed, and that the PATH environment variable includes the java executable (in the JAVA_HOME/bin directory).

   Oracle Key Vault supports JDK versions 1.5, 1.6, 7, and 8.
3. Run the Shell utility ORAENV or source ORAENV command to set the correct environment variables on Oracle Database servers.
4. Check that the environment variables ORACLE_BASE and ORACLE_HOME are correctly set.

   If you used ORAENV to set these variables, you must verify that ORACLE_BASE points to the root directory for Oracle Databases, and that ORACLE_HOME points to a sub-directory under ORACLE_BASE where an Oracle Database is installed.
5. Navigate to the directory in which you saved the okvclient.jar file.
6. Run the java command to install the okvclient.jar file.

   ```
   java -jar okvclient.jar -d /home/oracle/okvutil -v
   ```

   > **Note:**
   >
   > For a new installation, ensure that the directory location specified in the -d argument does not contain a subdirectory named ssl. If you are upgrading an existing deployment, the original ssl subdirectory is used.

   If you are re-enrolling the endpoint, add the -o argument to the command.

   ```
   java -jar okvclient.jar -d /home/oracle/okvutil -v -o
   ```

   In the above commands:

   - The -d argument specifies the directory location for the endpoint software and configuration files, in this case /home/oracle/okvutil.

     The environment variable $OKV_HOME refers to the directory where the endpoint software is installed, in this case /home/oracle/okvutil.

- The `-v` argument writes the installation logs to the `$OKV_HOME/log/okvutil.deploy.log` file at the server endpoint.
- The `-o` argument overwrites the symlink reference to `okvclient.ora`.

> **Note:**
>
> `-o` is an optional argument that allows you to overwrite the symlink reference to `okvclient.ora`, when `okvclient.jar` is deployed in a directory other than the original directory. This argument is used only when re-enrolling an endpoint.

7. The installation process prompts for a password. You can enter a password to create a **password-protected** wallet or create an **auto-login** wallet without a password as described below:

   - A password-protected wallet is an Oracle wallet file that store the endpoint's credentials to access Oracle Key Vault. This password will be required whenever the endpoint connects to Oracle Key Vault.

     Create a password-protected wallet by entering a password between 8 and 30 characters. Then press **Enter**.

   - Create an auto-login wallet by simply clicking **Enter**.

     No password will be required when the endpoint connects to Oracle Key Vault. An auto-login wallet enables endpoint provisioning without human intervention.

   ```
   Enter new Key Vault endpoint password (<enter> for auto-login):
   Key_Vault_endpoint_password
   Confirm new endpoint password: Key_Vault_endpoint_password
   ```

   The installation proceeds and completes with the following message:

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

   A successful installation of the endpoint software creates the following directories:

   - `bin`: contains the `okvutil` program, the `root.sh` and `root.bat` scripts, and the binary files `okveps.x64` and `okveps.x86`
   - `conf`: contains the configuration file `okvclient.ora`
   - `jlib`: contains the Java library files
   - `lib`: contains the file `liborapkcs.so`
   - `log`: contains the log files
   - `ssl`: contains the TLS-related files and wallet files. The wallet files contain the endpoint credentials to connect to Oracle Key Vault.

     The `ewallet.p12` file refers to a password-protected wallet. The `cwallet.sso` file refers to an auto-login wallet.

8. On UNIX platforms, the `liborapkcs.so` file contains the library that the Oracle database uses to communicate with Oracle Key Vault. On Windows platforms, the `liborapkcs.dll` file contains the library that the Oracle database uses to communicate with Oracle Key Vault.

If you are planning to use a TDE direct connection, then run `root.sh` on Oracle Linux x86-64, Solaris, AIX, and HP-UX (IA) installations. The `liborapkcs.so` file is copied to the following directory: /opt/oracle/extapi/64/hsm/oracle/1.0.0

On Windows installations, run `root.bat`. The `liborapkcs.dll` file is copied to C:\oracle\extapi\64\hsm\oracle\1.0.0

Log in as the root user and run the `root.sh` script. On Windows installations, run `root.bat`.

```
$ sudo bin/root.sh
```

```
bin\root.bat
```

Or:

```
$ su -
# bin/root.sh
```

On Windows platforms, you are prompted for the version of the RDBMS in use when you execute `root.bat`. Switch out of user root after completing this step.

9. Run the `okvutil list` command to verify that the endpoint software installed correctly, and that the endpoint can connect to the Oracle Key Vault server.

   If the endpoint is able to connect to Key Vault, a No objects found message appears:

   ```
   $ ./okvutil list
   No objects found
   ```

   If a **Server connect failed** message appears at any time, you must troubleshoot the installation for possible issues. First check that environment variables are correctly set.

10. You can get help on the endpoint software with the `-h` option:

    ```
    java -jar okvclient.jar -h
    ```

    The following output appears:

    ```
    Oracle Key Vault Release 12.2.0.6.0 (2017-12-15 15:36:49.839 PDT)
    Production on Fri Dec 15 19:55:31 PDT 2017
    Copyright (c) 1996, 2017 Oracle. All Rights Reserved.
    Usage: java -jar okvclient.jar [-h | -help] [[-v | -verbose] [-d
    <destination directory>] [-o]]
    ```

11. After installation Oracle recommends that you securely delete the endpoint software file `okvclient.jar`.

**See Also:**

- "Glossary (page    0    )" for definitions of environment variables
- "Special Notes About Endpoint Provisioning (page 8-9)" to check environment and setup
- "Using an Online Master Key with Oracle Key Vault (page 11-10)" for TDE

## 8.4 Special Notes About Endpoint Provisioning

The default location for the `okvclient.ora` file is the `$OKV_HOME/conf` directory. When the installation completes, the `JAVA_HOME` path is added to the `okvclient.ora` configuration file for future use by `okvutil`.

When you provision endpoints you must know how the installation process determines the location of Java home and the `okvclient.ora` file.

For Oracle Database endpoints, if you are using the `srvctl` utility and setting environment variables in `sqlnet.ora` you must set them in both the operating system and the `srvctl` environment.

#### How the Location of JAVA_HOME Location is Determined

The endpoint software installation process uses two rules to determine the Java home location:

1. If a user-defined `JAVA_HOME` environment variable exists, the installation process uses this value.
2. If `JAVA_HOME` is not set, the installation process looks for it in the `java.home` system property of the Java Virtual Machine (JVM).

Once the `JAVA_HOME` path is determined, it is added to the configuration file `okvclient.ora` to be used by all `okvutil` commands.

You can force `okvutil` to use a different `JAVA_HOME` setting using one of these methods:

- Set the `JAVA_HOME` environment variable in the shell where you run `okvutil`:

  ```
  setenv JAVA_HOME path_to_Java_home
  ```

  Or:

  ```
  export JAVA_HOME = path_to_Java_home
  ```

- Set the `JAVA_HOME` property directly in the `okvclient.ora` configuration file.

  ```
  JAVA_HOME=path_to_Java_home
  ```

#### Location of the OKVCLIENT.ORA File and Environment Variables

`$OKV_HOME` is the destination directory for the endpoint software specified with the `-d` option during installation. The file `okvclient.ora` is a configuration file in the directory `$OKV_HOME/conf`.

In addition to `$OKV_HOME/conf`, a soft link to `okvclient.ora` is set up for an existing database. The location of the soft link depends on the following:

- If the `$ORACLE_BASE` variable is set, then the installation process creates a symbolic link to the `okvclient.ora` configuration file (in `$OKV_HOME/conf`) in the `$ORACLE_BASE/okv/$ORACLE_SID` location.

  If the `okvclient.ora` file already exists in the `$ORACLE_BASE/okv/$ORACLE_SID` location, then the installation process accepts the existing soft link to `okvclient.ora` as a a valid soft link.

- If the $ORACLE_BASE/okv/$ORACLE_SID directory is not set, then the installation process tries to create it.
- If the $ORACLE_HOME variable is set and the $ORACLE_BASE variable is not set, then the installation process creates a symbolic link for the $ORACLE_HOME/okv/$ORACLE_SID location to point to the configuration file in the $OKV_HOME/conf directory.

**Setting OKV_HOME for Non-database Utilities to Communicate with Key Vault**

For non-database utilities you must set the environment variable OKV_HOME to point to the destination directory for the endpoint software, because the installation process does not set this variable automatically. OKV_HOME must be set for these utilities to communicate with Key Vault. These include utilities such as Oracle Recovery Manager (RMAN) that access Oracle Key Vault for keys.

You must set OKV_HOME in all environments where you will run utilities like RMAN. For example, if you spawn a new xterm, you will need to set OKV_HOME in this environment before running RMAN.

**Environment Variables in SQLNET.ORA**

You must consider the following points while using the srvctl utility on Oracle Database endpoints:

- If you are using the srvctl utility, and if you want to include environment variables in the sqlnet.ora configuration file, then you must set these environment variables in both the operating system and the srvctl environment.
- The operating system (OS) and Server Control (srvctl) should have $ORACLE_SID, $ORACLE_HOME and $ORACLE_BASE set to the same values.

**If the Endpoint Does Not Use the Oracle Key Vault Client Software**

Third party KMIP endpoints do not use the Key Vault software okvutil and liborapkcs.so. In this case you must manually set the TLS authentication as follows:

1. Extract the ssl directory from the okvclient.jar file, as follows:

   ```
   jar xvf okvclient.jar ssl
   ```

2. Use the following files to set up the TLS authentication:
   - ssl/key.pem: endpoint private key
   - ssl/cert.pem: endpoint certificate
   - ssl/cert_req.pem: certificate request corresponding to cert.pem
   - ssl/CA.pem: trust anchor for verifying the Oracle Key Vault server certificate

## 8.5 Transparent Data Encryption Endpoint Management

TDE has supported storing TDE master encryption keys in Oracle wallets, beginning with Oracle Database 10*g* Release 2, and in Hardware Security Modules (HSMs), beginning with Oracle Database 11*g* Release 1.

Oracle Key Vault can manage TDE keys by using the same PKCS#11 interface that TDE uses to communicate with an external keystore. Therefore, you do not need to patch the database to use Key Vault for storing and retrieving TDE master keys. Oracle Key Vault supplies the PKCS#11 library to communicate with Oracle Key Vault.

Oracle Key Vault improves upon TDE key management. For example, the keys in the wallet can be uploaded directly to Key Vault for long-term retention, to be shared with other database endpoints within the same endpoint group. Therefore, you do not need to store the wallet indefinitely after migration. Migration in this context means that the database is configured to use Key Vault for wallet backup, and that the administrator intends to migrate to an Online Master Key (formerly knows as TDE direct connect).

You can continue to use the wallet, and upload wallet copies to Key Vault as part of every TDE key administration SQL operation, involving a `WITH BACKUP` SQL clause. (However, be aware that the `WITH BACKUP` clause is ignored by TDE in an Oracle Key Vault online key deployment, even if it is required for the `ADMINISTER KEY MANAGEMENT` statement.)

Example 8-1 (page 8-11) shows examples of setting an encryption key.

Oracle Database, and thus TDE are endpoints for Oracle Key Vault. Endpoint enrollment and installation ensure that the PKCS#11 library is installed in the correct location for TDE to pick up and use. When the PKCS#11 library is installed, all other configurations and operations are in effect.

> **See Also:**
>
> "Using an Online Master Key with Oracle Key Vault (page 11-10)"

**Example 8-1    Setting an Encryption Key**

```
ALTER SYSTEM SET ENCRYPTION KEY IDENTIFIED BY secret_passphrase -- For Oracle
Database 11g Release 2

ADMINISTER KEY MANAGEMENT SET ENCRYPTION KEY IDENTIFIED BY secret_passphrase
WITH BACKUP; -- For Oracle Database 12c
```

## 8.6 Endpoint okvclient.ora Configuration File

Oracle Key Vault endpoint libraries and utilities use a configuration file called `okvclient.ora`, where the configuration parameters associated with the endpoint are stored. The `okvclient.ora` file consists of key-value pairs separated by an equal sign (=). The following parameters, showing sample data, can be set in the endpoint configuration file:

- `SERVER=192.0.2.254:5696`

  This parameter specifies the IP address and port number of the Key Vault server, separated by a colon. If the port number is not specified, then it defaults to the standard KMIP port 5696.

- `STANDBY_SERVER=192.0.2.114:5696`

  This is the standby server. If high availability is configured, then this parameter shows the standby IP address. Otherwise, it shows the IP address as `127.0.0.1`.

- `SSL_WALLET_LOC=/home/oracle/okvutil/ssl/`

This parameter specifies the location of the wallet containing TLS credentials for the endpoint.

- `SERVER_POLL_TIMEOUT=300`

  The `SERVER_POLL_TIMEOUT` parameter allows you to specify a timeout for a client's attempt to connect to an Oracle Key Vault server, before trying the next server in the list. The default value is 300 (milliseconds).

  In Oracle Key Vault 12.2.0.6.0, clients first establish a non-blocking TCP connection to Oracle Key Vault to quickly detect unreachable servers. Oracle Key Vault 12.2.0.6.0 introduces the `SERVER_POLL_TIMEOUT` parameter in the `okvclient.ora` file, after which Oracle Key Vault would attempt to connect to the next server. The default value is 300 (milliseconds).

  After the first attempt, the client makes a second and final attempt to connect to the server but this time waits for twice as long as the duration specified by the `SERVER_POLL_TIMEOUT` parameter. This is done to overcome possible network congestion or delays.

## 8.7 Oracle Key Vault okvutil Endpoint Utility Reference

After installing the endpoint software, endpoint administrators can use the command-line utility `okvutil` to communicate with Key Vault to upload and download security objects.

### 8.7.1 About the okvutil Utility

the command-line utility `okvutil` enables you to locate, upload, and download security objects to and from Key Vault. You can also use `okvutil` to change the wallet password and collect system diagnostics.

The `okvutil` utility uses the TLS credentials provisioned for the endpoint to authenticate to Oracle Key Vault.

### 8.7.2 okvutil Command Syntax

The `okvutil` utility syntax provides short and long options for specifying commands.

**Syntax**

```
okvutil command arguments [-v verbosity_level]
```

### Parameters

**Table 8-2 okvutil Command Syntax**

| Parameter | Description |
|---|---|
| command | Refers to any of the following commands: `upload`, `list`, `download`, `changepwd`, `diagnostics` |
| arguments | Refers to the arguments that you pass for the accompanying command. |
| `-v, --verbose` | Refers to verbosity level. Possible values are 0,1, and 2. Verbosity level 2 provides the highest the level of detail that is printed to standard output during command execution. The meaning of verbosity values are as follows:<br>• `-v 0` disables verbose mode.<br>• `-v` 1 includes debug messages.<br>• `-v` 2 includes more detailed debug messages. |
| `-h`, `--help` | Use option to get help with any `okvutil` command. For example:<br>`okvutil` *`command`* `--help` |

### Short and Long Forms of Specifying Options

You can specify the options in either a short form or a long form.

> **Note:**
>
> Endpoint platforms AIX and HP-UX (IA) support only short form options currently

- **Short form:** Only use one hyphen and the single-letter option name. For example:

  ```
  -l /home/username
  -t wallet
  ```

- **Long form:** Provide two hyphens and the full option name. For example:

  ```
  --location /home/username
  --type wallet
  ```

The examples in this guide use the short form.

### How Password Prompts for okvutil Work

The `okvutil` commands prompt for passwords in the following situations:

- If you created a password-protected wallet during endpoint installation to access Oracle Key Vault.
- If you specify an Oracle wallet file or Java keystore file using the `-l` option, `okvutil` prompts you to provide the password for the wallet or keystore that `okvutil` is trying to upload to Oracle Key Vault.

## 8.7.3 okvutil upload Command

The `okvutil upload` command uploads security objects to Key Vault such as: Oracle wallets including auto-login wallets, Java keystores, credential files, user-defined keys, and other types of key storage files.

You can upload Oracle wallets from all currently supported releases of Oracle Database and other Oracle software products that use Oracle wallets. The `okvutil upload` command opens the wallet or Java keystore and uploads each item found as an individual security object into Oracle Key Vault. If you are uploading credential files, then Key Vault uploads them as whole files called opaque objects.

### Syntax

Short format:

```
okvutil upload [-o] -l location -t type [-g group] [-d description] [-v
verbosity_level]
```

Long format:

```
okvutil upload [--overwrite] --location location --type type [--group group]
[--description description] [--verbose verbosity_level]
```

### Parameters

**Table 8-3 okvutil upload Command Options**

| Parameter | Description |
|---|---|
| `-o`, `--overwrite` | If there are conflicts with the existing data in the Oracle Key Vault virtual wallet, then Key Vault replaces the existing data with new data that is sent by the endpoint. If there are no conflicts, then the overwrite operation is not necessary and is not performed. Use care if you plan to specify this option. |
| `-l`, `--location` | Specifies the location of an Oracle wallet file, Java keystore, or a text file containing user-defined and hex-encoded TDE master encryption identifier and key. For an Oracle wallet, the location is the directory that contains the `.p12` or `.sso` files. If you are uploading a credential file as an opaque object, then ensure that this file is no larger than 120 kilobytes (KB). |

**Table 8-3 (Cont.) okvutil upload Command Options**

| Parameter | Description |
|---|---|
| `-t, --type` | Specifies the data type of the object being uploaded to Oracle Key Vault. It must be a value from the following list:<br>• `WALLET` for an Oracle wallet<br>• `JKS` for a Java keystore<br>• `JCEKS` for a Java Cryptography Extension keystore (JCEKS)<br>• `SSH` for an SSH key file, to be uploaded as an opaque object. The maximum size is 120 KB.<br>• `KERBEROS` for a Kerberos keytab, to be uploaded as an opaque object. The maximum size is 120 KB.<br>• `TDE_KEY_BYTES` for a user-defined key to be used as a TDE master encryption key.<br>• `OTHER` for opaque objects, which are other files that store secrets. The maximum size is 120 KB.<br>The `WALLET`, `JKS`, and `JCEKS` types contain multiple objects. Oracle Key Vault uploads each of these objects individually. The `SSH`, `KERBEROS`, `TDE_KEY_BYTES`, and `OTHER` types, being opaque objects, are uploaded as single files.<br>This setting is not case-sensitive. |
| `-g, --group` | Is the name of a Key Vault virtual wallet to which the certificate store or secret store (or both) are added. This name is case-sensitive. The virtual wallet must already exist, and the user must have authorization to access it. If you omit this setting, then the default group, if there is one, is used. If there is no default group and you omit the `-g, --group` option, then the data uploaded will not be placed in a group. |
| `-d, --description` | Enables you to add a description, up to 2000 bytes. It is valid only if the `-t` *`type`*, `--`*`type`* parameter is set to `SSH`, `KERBEROS`, `TDE_KEY_BYTES`, or `OTHER`. Optional.<br>Enclose this description in double quotation marks. If there are spaces within this description, then include escape characters with the quotation marks. For example: `-d \"text with spaces\"` |
| `-v, --verbose` | Refers to the verbosity level from 0 (none), 1 (debug), 2 (detailed debug). |

### Uploading a Java Keystore Using the -v2 Option

The `okvutil upload` command enables you to upload a Java keystore.

The following example shows you how to use the `okvutil upload` command to upload a Java keystore. The `-v 2` option enables the command to list the items that are uploaded.

The `okvutil` command prompts if necessary for passwords to connect to Oracle Key Vault and to open the Oracle wallet file.

```
$ okvutil upload -l ./fin_jceks.jck -t JCEKS -g fin_wal -v 2

okvutil version 12.2.0.0.0
Configuration file: /tmp/fin_okv/conf/okvclient.ora
Server: 192.0.2.254:5696
Standby Server: 127.0.0.1:5696
Uploading from /tmp/fin_okv/keystores/jks/keystore.jks
Enter source Java keystore password:
```

```
Uploading private key
Uploading trust point
Uploading trust point
Uploading private key
Uploading private key

Uploaded 3 private keys
Uploaded 0 secret keys
Uploaded 2 trust points

Upload succeeded
```

For more information about uploading a Java Keystore, see Uploading JKS or JCEKS Keystores (page 11-5).

### Uploading a Password-Protected Wallet File

The `okvutil upload` command uploads a password-protected wallet file.

The following example shows you how to upload a password-protected wallet file when there is no password for the endpoint to connect to Oracle Key Vault.

```
$ okvutil upload -l . -t WALLET -g FinanceWallet
Enter source wallet password: password

Upload succeeded
```

For more information about uploading wallet files, see Uploading Oracle Wallets (page 11-2).

### Uploading a User-defined key to use as a TDE master encryption key

The `okvutil upload` command enables you to upload a user-defined key to use as a TDE master encryption key.

The following example shows you how to upload a user-defined key.

```
$ okvutil upload -l /tmp/tde_key_bytes.txt -t TDE_KEY_BYTES -g
"FIN_DATABASE_VIRTUAL_WALLET" -d \"This key was created for Financial database
use on 1st Jan 2018\"
```

For more information about uploading an user-defined key, see Uploading the User-Defined Key (page 11-26).

**See Also:**

- "Add Security Objects to a Virtual Wallet (page 6-3)"
- "Uploading Oracle Wallets (page 11-2)"
- *Oracle Database Security Guide* for detailed information about the `orapki` utility

## 8.7.4 okvutil list Command

The `okvutil list` command gets the available security objects that are uploaded. When used without options or with the `-g` *`group`* option, it displays the unique ID, object type, and a descriptor for each item it lists from Oracle Key Vault.

### Syntax

Short format:

```
okvutil list [-l location -t type | -g group] [-v verbosity_level]
```

Long format:

```
okvutil list [--location location --type type | --group group] [--verbose
verbosity_level]
```

### Parameters

**Table 8-4 okvutil list Command Options**

| Parameter | Description |
|---|---|
| `-l`, `--location` | Specifies the location of an Oracle wallet file or a Java keystore. For an Oracle wallet, the location is the directory that contains the `.p12` or `.sso` files. For all other types, the location is the path name of the file itself. If you omit the `-l`, `--location` option, then the default location is Oracle Key Vault. In this case, the `okvutil list` command lists all the available keys in the server. If you use this setting, then you must also include the `-t`, `--type` setting, described next. |
| `-t`, `--type` | Specifies one of the following types:<br>• `WALLET` for an Oracle wallet<br>• `JKS` for the Java keystore<br>• `JCEKS` for the Java Cryptography Extension keystore (JCEKS)<br>• `OKV_PERSISTENT_CACHE` for the persistent cache of Oracle Key Vault<br>The `WALLET`, `JKS`, and `JCEKS` types are containers for security objects which Oracle Key Vault lists individually. The `SSH`, `KERBEROS`, and `OTHER` are opaque objects, and are listed as single files.<br>This setting is not case-sensitive. |
| `-g`, `--group` | Lists the content from a single virtual wallet. This option only applies when you omit the `-l`, `--location` option to list the objects stored in Oracle Key Vault. |
| `-v`, `--verbose`: | Refers to the verbosity level from 0 (none), 1 (debug), 2 (detailed debug). |

### Example: Listing Security Objects for the Current Endpoint

The `okvutil list` command enables you to see the security objects associated with the current endpoint.

Example 8-2 (page 8-18) gets all the authorized security objects for the current endpoint. In the last three lines, the `DB Connect Password` entries refer to the password that was used to log in to the instance (for example, the password for user `psmith` on the database instance `inst01`).

### Example: Listing the Contents of an Oracle Wallet File

The `okvutil list` command enables you to see the contents of an Oracle wallet file.

Example 8-3 (page 8-18) shows the contents of an Oracle wallet file.

#### Example 8-2 Listing Security Objects for the Current Endpoint

```
$ okvutil list
Enter Oracle Key Vault endpoint password: password

Unique ID                               Type           Identifier
F63E3F4A-C8FB-5560-E043-7A6BF00AA4A6    Symmetric Key  TDE Master Key:
062C4F5BAC53E84F2DBF95B96CE577B525
F63E3F4A-C8FC-5560-E043-7A6BF00AA4A6    Symmetric Key   TDE Master Key:
069A5253CF9A384F61BFDD9CC07D8A6B07
F63E3F4A-C8FD-5560-E043-7A6BF00AA4A6    Opaque Object   -
F63E3F4A-C8FE-5560-E043-7A6BF00AA4A6    Symmetric Key   TDE Master Key:
06A66967E70DB24FE6BFD75447F518525E
F63E3F4A-C8FF-5560-E043-7A6BF00AA4A6    Symmetric Key   TDE Master Key:
0636D18F2E3FF64F7ABF80900843F37456
F63E3F4A-C900-5560-E043-7A6BF00AA4A6    Opaque Object   -
F63E3F4A-C901-5560-E043-7A6BF00AA4A6    Symmetric Key   TDE Master Key:
0611E6ABD666954F2FBF8359DE172BA787
F63E3F4A-C902-5560-E043-7A6BF00AA4A6    Symmetric Key   TDE Master Key:
0657F27D64D1C04FAEBFE00B5105B3CBAD
F63E3F4A-C91B-5560-E043-7A6BF00AA4A6    Opaque Object   Certificate Request
F63E3F4A-C91C-5560-E043-7A6BF00AA4A6    Certificate     X509 DN:OU=Class 1
Public Primary Certification Authority,O=VeriSign\, Inc.,C=US
F63E3F4A-C903-5560-E043-7A6BF00AA4A6    Secret Data     DB Connect Password:
psmith@inst01
F63E3F4A-C904-5560-E043-7A6BF00AA4A6    Secret Data     DB Connect Password:
jdaley@inst02
F63E3F4A-C905-5560-E043-7A6BF00AA4A6    Secret Data     DB Connect Password:
tjones@inst03
```

#### Example 8-3 Listing the Contents of an Oracle Wallet File

```
$ okvutil list -t WALLET -l /home/oracle/wallets
Enter target wallet password: Oracle_wallet_password

Dumping secret store of wallet:
ORACLE.SECURITY.DB.ENCRYPTION.MASTERKEY
ORACLE.SECURITY.DB.ENCRYPTION.Aa4JEUaCeE8qv0Dsmmwe5S4AAAAAAAAAAAAAAAAAAAAAAAAAAAA
A
ORACLE.SECURITY.ID.ENCRYPTION.
ORACLE.SECURITY.KB.ENCRYPTION.
ORACLE.SECURITY.TS.ENCRYPTION.BZuIPES7+k/
tv0ZwOlDeIp4CAwAAAAAAAAAAAAAAAAAAAAAAAAAA
Dumping cert store of wallet:

There are 1 Certificate Requests in the list

Certificate request:
        DN: CN=oracle
        Type: NZDST_CERT_REQ
        PUB key size: 2048

There are 0 Certificates in the list

There are 0 TPs in the list
```

## 8.7.5 okvutil download Command

The `okvutil download` command downloads security objects from Oracle Key Vault to the endpoint such as: Oracle wallets including auto-login wallets, Java keystores, credential files, and other types of key storage files.

You can only download the contents of a virtual wallet into a keystore (a container such as an Oracle wallet or a JCEKS keystore that can hold multiple security objects), and not into a credential file.

Note that some keystores only support the storage of certain types of security objects. An error occurs if you upload a DSA key from a Java keystore and later try to download it to a different type of keystore like an Oracle wallet.

### Syntax

Short format:

```
okvutil download -l location -t type [-g group | -i object_id] [-o] [-v
verbosity_level]
```

Long format:

```
okvutil download --location location --type type [--group group | --item
object_id] [--overwrite] [--verbose verbosity_level]
```

### Parameters

**Table 8-5 okvutil download Command Options**

| Parameter | Description |
|---|---|
| `-l`, `--location` | Specifies the file location to store the items that you want to download. Ensure that you have permission to create wallets in this location. Ensure that the file you download is no more than 120 KB. This setting is mandatory. |
| `-t`, `--type` | Specifies the data type of the object being downloaded to Oracle Key Vault. It must be a value from the following list:<br>• `WALLET` for an Oracle wallet<br>• `JKS` for a Java keystore<br>• `JCEKS` for a Java Cryptography Extension keystore (JCEKS)<br>• `SSH` for an SSH key file, to be downloaded as an opaque object.<br>• `KERBEROS` for a Kerberos keytab, to be downloaded as an opaque object.<br>• `OTHER` for opaque objects, which are other files that store secrets.<br>The `WALLET`, `JKS`, and `JCEKS` types contain multiple objects. Oracle Key Vault downloads each of these objects individually. The `SSH`, `KERBEROS`, and `OTHER` types, being opaque objects, are downloaded as single files.<br>This setting is not case-sensitive. This setting is mandatory. |

**Table 8-5 (Cont.) okvutil download Command Options**

| Parameter | Description |
|---|---|
| `-g, --group` | Is the name of a virtual wallet from which you download an item for the `WALLET`, `JKS`, and `JCEKS` types. The virtual wallet must already exist, and the user must have authorization to access it. The `okvutil` utility downloads the entire virtual wallet specified by the `-g` option, and stores it in a *new* wallet. There must be no existing wallet at the specified location. The `okvutil` utility will create one. `okvutil` prompts you to create and enter a password for the new wallet. Record that password for the future. Remember that the group name is case-sensitive.<br>If the type is `WALLET`, `JKS`, or `JCEKS`, then you can either include or omit the *`group`* setting. If the type is `SSH`, `KERBEROS`, or `OTHER`, then you must include the *`object_id`* option, but not include the *`group`* setting. |
| `-i`, `--item` | Refers to the unique ID of the object that you want to download, such as secrets (for example, `-i oracle.security.client.password1` for the first secure external password store (SEPS) entry inside a wallet). |
| `-o`, `--overwrite` | Downloads data into an existing `WALLET`, `JKS`, or `JCEKS` file specified by `-l`, which must exist. If a conflict arises between the data to download and the data that already exists in the container, then the new data overwrites the old data. The `-o`, `--overwrite` option does not apply to the other types (`SSH`, `KERBEROS`, and `OTHER`). Use care if you plan to specify this option.<br>If you omit the `o` or `overwrite` option when you download wallets that already exist in the current directory, then the original wallet file is renamed to either `ewallet.p12.`*`timestamp`*`.bak` or `owallet.sso.`*`timestamp`*`.bak` before the new wallet file is downloaded. For files that are not wallets (such as Java keystore files), an error appears, and you will need to rename the file or move it to a new location before performing the download. |
| `-v`, `--verbose` | Refers to the verbosity level from 0 (none), 1 (debug), 2 (detailed debug). |

### Example: Downloading a Virtual Wallet to a Java Keystore

The `okvutil download` command enables you to download a virtual wallet to a Java keystore.This is useful if you are sharing the same Java key store across multiple application servers and want to use the same wallet.

Example 8-4 (page 8-20) downloads the Key Vault virtual wallet `FinanceWallet` to a Java keystore.

**See Also:**

- You can find the available object IDs by running the `okvutil list` command, described in "okvutil list Command (page 8-17)"
- "Downloading Oracle Wallets (page 11-3)"

**Example 8-4 Downloading a Virtual Wallet to a Java Keystore**

```
$ okvutil download -l ./fin/okv/work -t JCEKS -g FinanceWallet
```

The command will prompt for a new password for the Java Keystore as below:

```
Enter new Java keystore password:
Confirm new Java keystore password:
Download succeeded
```

## 8.7.6 okvutil changepwd Command

The `okvutil changepwd` command enables you to change the password associated with the credentials used to connect to Oracle Key Vault. Use this command if you used a password-protected wallet to store the Oracle Key Vault endpoint user credentials. The new password need not be the same password for the JCKS or wallet file when it was uploaded.

### Syntax

Short format:

```
okvutil changepwd -l location -t type [-v verbosity_level]
```

Long format:

```
okvutil changepwd --location location --type type [--verbose verbosity_level]
```

### Parameters

**Table 8-6 okvutil changepwd Command Options**

| Parameter | Description |
|---|---|
| `-l`, `--location` | Specifies the directory location of the wallet whose password you want to change. |
| `-t`, `--type` | Specifies the data type. Enter `WALLET`. |
| `-v, --verbose` | Refers to the verbosity level from 0 (none), 1 (debug), 2 (detailed debug). |

### Example: Changing an Oracle Key Vault Endpoint Password

the `okvutil changepwd` enables you to change the password of an endpoint.

Example 8-5 (page 8-21) shows how to change the endpoint password. When you are prompted to create the new password, enter a password that is between 8 and 30 characters.

**Example 8-5 Changing an Oracle Key Vault Endpoint Password**

```
$ okvutil changepwd -l ./home/oracle/okvutil/ssl -t WALLET
Enter wallet password: current_endpoint_password
Enter new wallet password: new_endpoint_password
Confirm new wallet password: new_endpoint_password
```

## 8.7.7 okvutil diagnostics Command

The `okvutil diagnostics` command enables you to collect diagnostic and environmental information on an endpoint to troubleshoot deployment issues. The information is gathered in a `diagnostics.zip` file, which can be given to Oracle support for further analysis and debugging.

The information gathered includes information on:

- The Shell environment variables: `OKV_HOME`, `ORACLE_HOME`, `ORACLE_BASE`, `ORACLE_SID`, `PATH`, `CLASSPATH`
- Configuration and IP address of the Key Vault server from `okvclient.ora`
- Directory listing of `OKV_HOME` and its sub-directories
- Key Vault log files from the endpoint
- Listing of symbolic links created by the Key Vault endpoint installer
- Network settings and ping results

No sensitive user information like user credentials or security objects are collected.

### Syntax

Short format:

```
okvutil diagnostics [-v verbosity_level]
```

Long format:

```
okvutil diagnostics  [--verbose verbosity_level]
```

### Parameters

**Table 8-7 okvutil diagnostics Command Options**

| Parameter | Description |
|---|---|
| `-v, --verbose` | Refers to the verbosity level from 0 (none), 1 (debug), 2 (detailed debug) |

### Example: Collecting System Diagnostics

the `okvutil diagnostics` command enables you to collect system diagnostics in a zip file.

Example 8-6 (page 8-22) shows you how to execute the command. Wait until you see the message `Diagnostics complete` message to see the `diagnostics.zip` file in the same directory.

**Example 8-6 Collecting System Diagnostics**

```
$ okvutil diagnostics
Diagnostics collection complete.
ls
diagnostics.zip
```

## 8.8 Upgrading Endpoint Software on an Enrolled Endpoint

You would upgrade the endpoint software on an enrolled endpoint any time you upgraded to a new release of Oracle Key Vault so you have the latest software on both the Oracle Key Vault server and the endpoint. This is highly recommended for optimum performance.

Oracle Key Vault servers are capable of working with endpoint software from the prior major release, but may not work properly with endpoint software that are older.

To upgrade the software on an already enrolled endpoint you only need to download and install the software `okvclient.jar` on the endpoint. You do not need to re-enroll the endpoint.

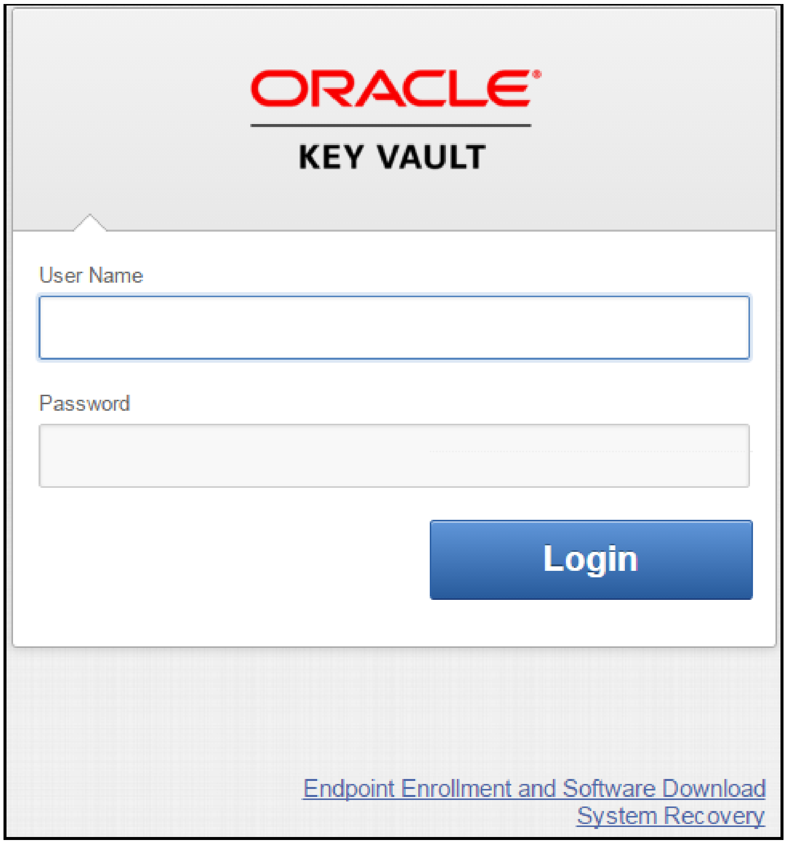
To upgrade endpoint software on an enrolled endpoint:

1. Log in to the endpoint server as the endpoint administrator.
2. Connect to the Oracle Key Vault management console.

   For example:

   ```
   https://192.0.2.254
   ```
3. The login page to the Oracle Key Vault management console appears. *Do not log in*.

**Figure 8-3 Key Vault Management Console Login Screen**



4. Click **Endpoint Enrollment and Software Download.**

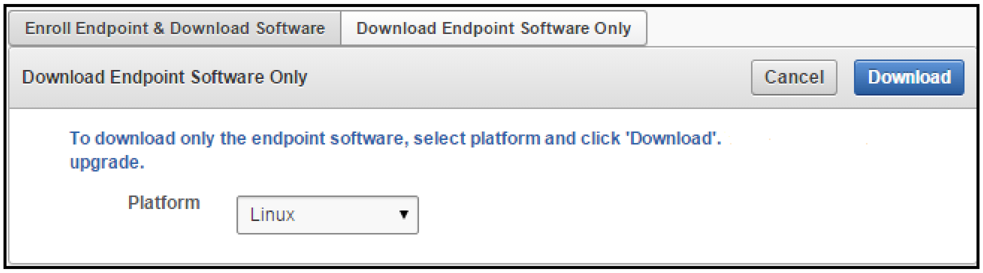
   The **Enroll Endpoint & Download Software** page appears with two tabs: **Enroll Endpoint & Download Software** and **Download Endpoint Software Only**.
5. Click **Download Endpoint Software Only**.

   The **Download Endpoint Software Only** page appears.

   Figure 8-4 has been trimmed with the result that some part of the message is truncated. The entire message follows:

**To download only the endpoint software, select platform and click 'Download'. This applies if you've already enrolled and would like to download endpoint software only in case of an upgrade.**

**Figure 8-4 Download Endpoint Software Only**



Enroll Endpoint & Download Software
Download Endpoint Software Only
Download Endpoint Software Only
Cancel
Download

To download only the endpoint software, select platform and click 'Download'.
upgrade.

Platform: Linux

6. Select the **Platform** from the drop down list and click **Download**.

   A directory window appears, and prompts you to save the endpoint software file: `okvclient.jar`. Navigate to the folder where you want to save the file.

7. Save the file.
8. Verify that the file is downloaded.
9. Now you are ready for Task 2: Install Oracle Key Vault Software on the Endpoint (page 8-6).

9

# Oracle Cloud Database as a Service Endpoints

Oracle Key Vault deployed on-premises can manage the TDE master keys for Oracle Cloud Database as a Service instances in a hybrid cloud topology.

## 9.1 About Managing Database as a Service Endpoints

An Oracle Key Vault appliance deployed on-premises can now manage TDE master keys for Database as a Service instances in the Oracle Cloud. This provides physical separation of keys from the encrypted data, and gives on-premises administrators control and visibility of how encryption keys are used to access encrypted data in the cloud. This also meets compliance requirements where encryption keys must be managed on-premises or separate from systems containing encrypted data.

## 9.2 Preparing a Database as a Service Instance to be an Oracle Key Vault Endpoint

Oracle Cloud - Database as a Service provides users with fully functional Oracle database instances that use computing and storage resources provided by Oracle Compute Cloud Service. It eliminates the need to purchase, build, and manage silos of server and storage systems. It also makes database resources and capabilities available online so users can consume them whenever and wherever they are needed.

> **See Also:**
>
> - Using Oracle Database Cloud - Database as a Service - Before You Begin with Database as a Service

## 9.2.1 Setup a Database Cloud Service Instance

Instruction for setting up a Database as a Service instance can be found in the Oracle Database Cloud Service (Database as a Service) documentation. Please refer to the links in **See Also** to setup a Database as a Service instance.

Once set up, your Oracle Database as a Service instance should have the following default values:

- A public IP address
- Two users: `oracle` and `opc` (Oracle Public Cloud)
- SSH access to both users: `oracle` and `opc`

> **See Also:**
>
> - Using Oracle Database Cloud - Database as a Service: Creating a Database Deployment
> - Using Oracle Database Cloud - Database as a Service: Connect to a Compute Node through SSH
> - Using Oracle Database Cloud - Database as a Service: Viewing Detailed Information for a Database Deployment

## 9.2.2 Create Low Privileged Operating System User on Database as a Service

By default, Database as a Service instances are provisioned with the users: `oracle` and `opc` . These users have more privileges than necessary to create the SSH tunnel, so Oracle recommends that you create another low privileged OS user named `okv` on the Database as a Service instance. Oracle Key Vault will use user `okv` to setup a SSH tunnel and communicate with the Database as a Service instances.

To create the low privileged `okv` user:

1. Run the SSH utility:

   ```
   $ ssh -i <private-key-file> opc@node-ip-address
   ```

   where:

   `private-key-file` is the path to the SSH private key file of the `opc` user.

`node-ip-address` is the public IP address of the Database as a Service compute node in x.x.x.x format.

2. If this is the first time you are connecting to the compute node, the SSH utility prompts you to confirm the public key.

   In response to the prompt, enter  yes.

3. Create the Oracle Key Vault user:

   ```
   $ sudo adduser okv
   ```

4. Append the Oracle Key Vault user `okv` to the `AllowUsers` parameter in the SSH configuration file `sshd_config` in `/etc/ssh/`:

   ```
   $ sudo vi /etc/ssh/sshd_config
     AllowUsers oracle opc okv
   ```

5. Restart the SSH daemon:

   ```
   $ sudo /sbin/service sshd restart
   ```

6. Grant the Oracle Key Vault user `okv` permission to execute `/sbin/fuser` as root.

   Edit and add the necessary entries to `/etc/sudoers` file as shown below:

   ```
   sudo chmod 740 /etc/sudoers
   sudo vi /etc/sudoers
   ```

   Add the following entry:

   ```
   okv     ALL=(root) NOPASSWD:/sbin/fuser
   sudo chmod 440 /etc/sudoers
   ```

   The `/etc/sudoers` would look something like:

   ```
   ## Allow root to run any commands anywhere
   root    ALL=(ALL)       ALL
   okv     ALL=(root) NOPASSWD:/sbin/fuser
   ```

7. Create the `authorized_keys` file for the `okv` user and set appropriate permissions for this file.

   ```
   $ sudo su -s /bin/sh okv -c "mkdir ~/.ssh"
   $ sudo su -s /bin/sh okv -c "chmod 700 ~/.ssh"
   $ sudo su -s /bin/sh okv -c "touch ~/.ssh/authorized_keys"
   $ sudo su -s /bin/sh okv -c "chmod 640 ~/.ssh/authorized_keys" su okv
   ```

8. Copy the SSH public key of Oracle Key Vault from the **ADD SSH Tunnel** page to the `authorized_keys` file in `/home/okv/.ssh/authorized_keys`.

**See Also:**

- Figure 9-2 (page 9-4)
- Using Oracle Database Cloud - Database as a Service: Connect to a Compute Node using SSH

## 9.3 Set up an SSH Tunnel between Key Vault and Database as a Service Instance

An on-premises Oracle Key Vault communicates with Oracle Cloud Database as a Service instances using a secure SSH tunnel. You can set up the SSH tunnel only after you set up the Database as a Service instance. You will need the Database as a Service instance's public IP address and name of the OS user you wish to use to establish the tunnel.

The following procedure assumes that you followed Oracle's recommendation and created a low privilege user named `okv`. It creates an SSH tunnel between Key Vault and the Database as a Service instance.

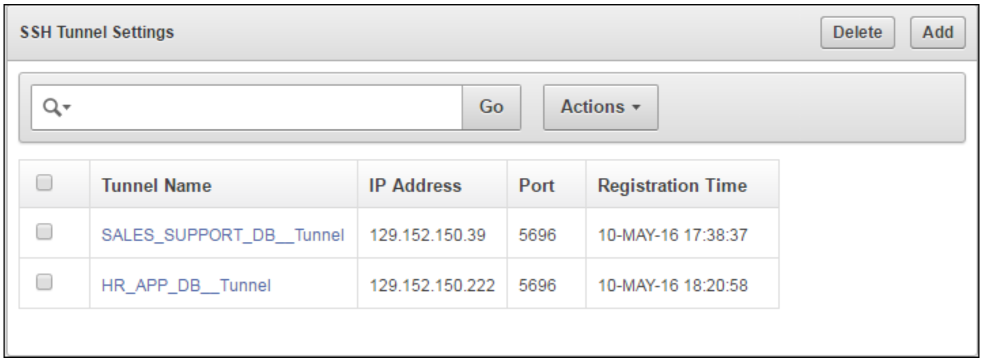
To set up the SSH connection:

1. Login to the Oracle Key Vault Server. The home page appears
2. Click **System**.

   The **Status** page appears.
3. Click **SSH Tunnel Settings** from the left side bar.

   The **SSH Tunnel Settings** page appears.

   **Figure 9-1 SSH Tunnel Settings**

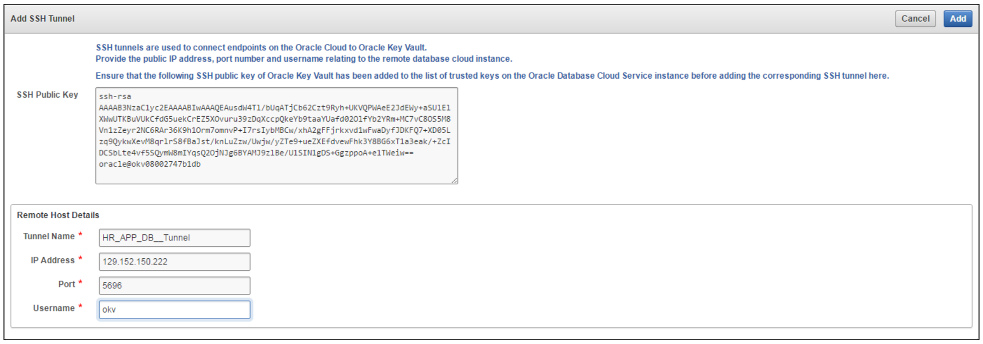
   

   | ☐ | Tunnel Name | IP Address | Port | Registration Time |
   |---|---|---|---|---|
   | ☐ | SALES_SUPPORT_DB__Tunnel | 129.152.150.39 | 5696 | 10-MAY-16 17:38:37 |
   | ☐ | HR_APP_DB__Tunnel | 129.152.150.222 | 5696 | 10-MAY-16 18:20:58 |

4. Click **Add**.

   The **Add SSH Tunnel** page appears.

   **Figure 9-2 Add SSH Tunnel**

   SSH tunnels are used to connect endpoints on the Oracle Cloud to Oracle Key Vault.
   Provide the public IP address, port number and username relating to the remote database cloud instance.

   Ensure that the following SSH public key of Oracle Key Vault has been added to the list of trusted keys on the Oracle Database Cloud Service instance before adding the corresponding SSH tunnel here.

   SSH Public Key

   ```
   ssh-rsa
   AAAAB3NzaC1yc2EAAAABIwAAAQEAusdW4Tl/bUqATjCb62Czt9Ryh+UKVQPWAeE2JdEWy+aSUlEl
   XWwUTKBuVUkCfdG5uekCrEZ5XOvuru39zDqXccpQkeYb9taaYUafd02OlfYb2YRm+MC7vC8OS5M8
   Vn1zZeyr2NC6RAr36K9h1Orm7omnvP+I7rsIybMBCw/xhA2gFFjrkxvd1wFwaDyfJDKFQ7+XD05L
   zq9QykwXevM8qrlrS8fBaJst/knLuZzw/Uwjw/yZTe9+ueZXEfdvewFhk3Y8BG6xT1a3eak/+ZcI
   DCSbLte4vf5SQymW8mIYqsQ2OjNJg6BYAMJ9zlBe/U1SIN1gDS+GgzppoA+e1TWeiw==
   oracle@okv08002747b1db
   ```

   Remote Host Details

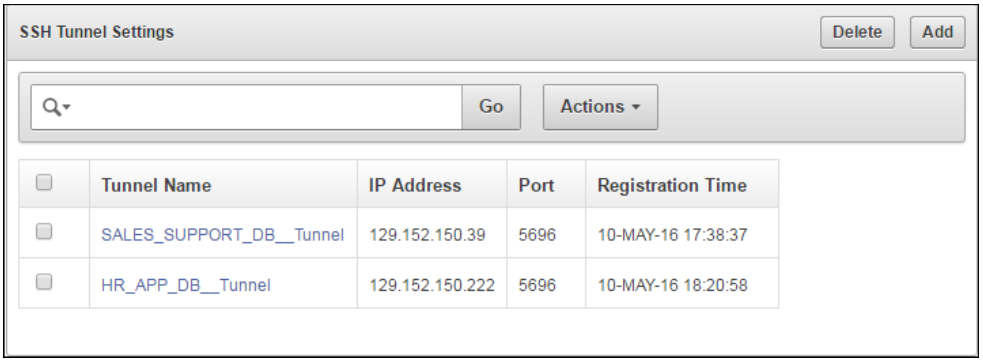
   | | |
   |---|---|
   | Tunnel Name * | HR_APP_DB__Tunnel |
   | IP Address * | 129.152.150.222 |
   | Port * | 5696 |
   | Username * | okv |

5. Copy the text in **SSH Public Key** and save it. You will need to transport it to the Database as a Service instance and add it to the `authorized_keys` file of the Database as a Service user `okv` at `/home/okv/.ssh/authorized_keys`.
6. In **Remote Host Details** enter information in the following fields:
   - **Tunnel Name -** Choose a descriptive name that identifies the tunnel, based on the Database as a Service instance to be associated with it.
   - **IP Address -** Public IP address of the Database as a Service instance
   - **Port** - enter a port number if you want to use a particular port number, or use the displayed default
   - **Username -** Enter `okv` for the username

   Note that you can complete these fields only after you set up your Database as a Service instance and obtain the public IP address and username.
7. Click **Add**.

   The **SSH Tunnel Settings** page appears. It displays the SSH Tunnel just created and pre-existing SSH tunnels.

   **Figure 9-3 SSH Tunnel Settings**

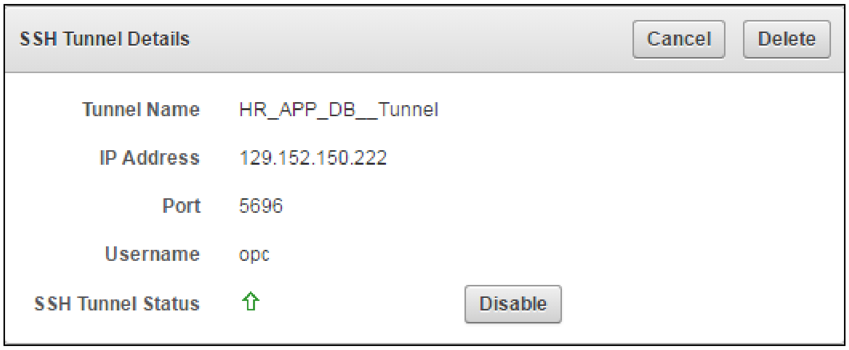
   

   | □ | Tunnel Name | IP Address | Port | Registration Time |
   |---|---|---|---|---|
   | □ | SALES_SUPPORT_DB__Tunnel | 129.152.150.39 | 5696 | 10-MAY-16 17:38:37 |
   | □ | HR_APP_DB__Tunnel | 129.152.150.222 | 5696 | 10-MAY-16 18:20:58 |

   It lists the tunnels created with the name, IP address, port, and registration time of each.

   Clicking **Add** redirects you to the **Add SSH Tunnel** page.
8. Click a tunnel name to see the **SSH Tunnel Details** page.

   **Figure 9-4 SSH Tunnel Details - Disable**

   Tunnel Name HR_APP_DB__Tunnel
   IP Address 129.152.150.222
   Port 5696
   Username opc
   SSH Tunnel Status ⇧

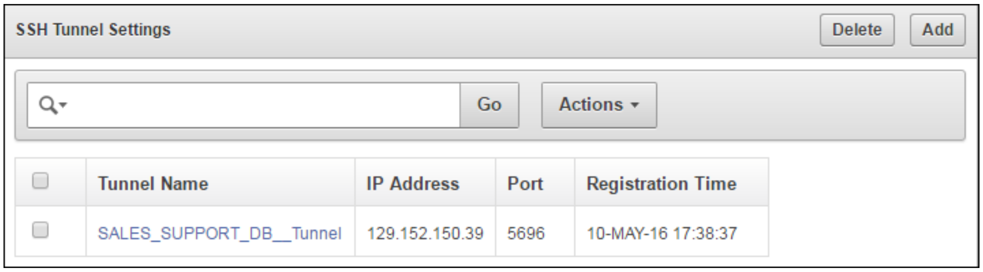
This page displays the following details for the tunnel:

- **Tunnel Name**
- **IP Address**
- **Port**
- **Username**
- **SSH Tunnel Status -** a green upward pointing arrow indicates an active tunnel and a downward pointing arrow indicates an inactive tunnel
- **Disable**, **Cancel**, and **Delete**

9. To delete a tunnel, check the box by the tunnel you want to delete and click **Delete.** You can delete more than one tunnel by selecting multiple boxes. A confirmation message appears. If you confirm, it deletes the selected tunnels.

   The **SSH Tunnel Settings** page displays the remaining tunnels.

**Figure 9-5    SSH Tunnel Settings Post Tunnel Deletion**

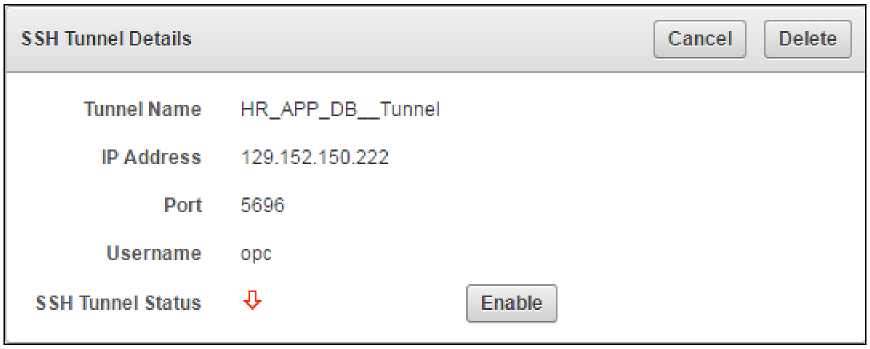


| □ | Tunnel Name | IP Address | Port | Registration Time |
|---|---|---|---|---|
| □ | SALES_SUPPORT_DB__Tunnel | 129.152.150.39 | 5696 | 10-MAY-16 17:38:37 |

10. Click **Disable** to disable the tunnel. When you disable the tunnel, the endpoints associated with this tunnel will no longer be able to communicate with Key Vault.

    A confirmation message appears. If you confirm, the status changes to disabled indicated by a downward pointing arrow. The **Disable** button is replaced by an **Enable** button.

**Figure 9-6    SSH Tunnel Details - Enable**



Tunnel Name HR_APP_DB__Tunnel
IP Address 129.152.150.222
Port 5696
Username opc
SSH Tunnel Status ⇩

## 9.4 Manageability of SSH Tunnel

The SSH tunnel is kept alive even if there is no activity between Key Vault and the Database as a Service instance. In case the tunnel drops off, it is automatically restarted.

An alert will be sent out if the tunnel is not available for any reason. An administrative user may elect to receive these alerts by email by configuring SMTP settings on Key Vault.

## 9.5 Register and Enroll Database as a Service Instance as Key Vault Endpoint

The Oracle Database as a Service instance must be enrolled before it can communicate with Oracle Key Vault server. The enrollment of Database as a Service endpoints is similar to the enrollment of on-premises endpoints with the following exceptions:

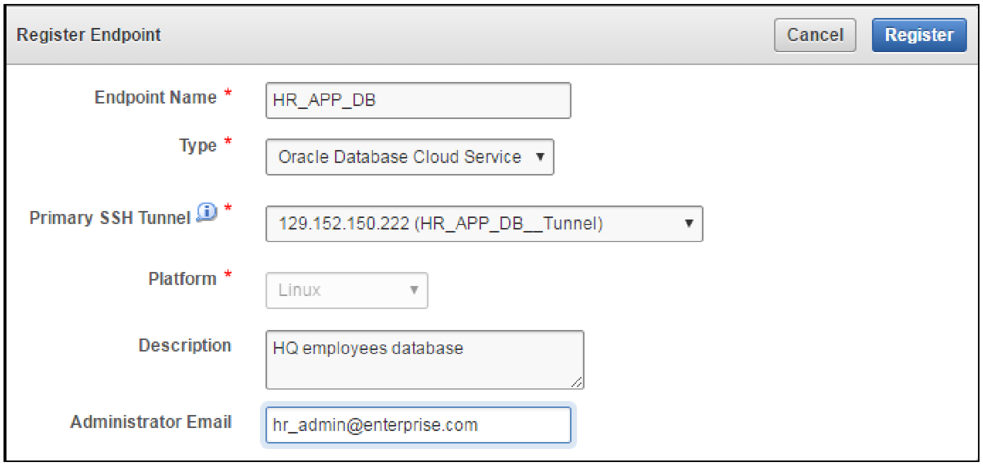
- Database as a Service endpoints should be registered with an endpoint type of “Oracle Database Cloud Service".
- Database as a Service endpoints have a primary tunnel IP associated with them. You must select the SSH tunnel with the same public IP address of the Database as a Service instance.
- The platform must be Linux. This is automatically selected and cannot be modified.
- You must download the jar file on-premises and transfer it to the Database as a Service instance using an out-of-band method like SCP or FTP.

To register and enroll a Database as a Service instance as an endpoint:

1. Click the **Endpoints** tab. The **Endpoints** page appears.
2. Click **Add**.

   The **Register Endpoint** page appears.

   **Figure 9-7 Register Endpoint**

   | | |
   |---|---|
   | Endpoint Name * | HR_APP_DB |
   | Type * | Oracle Database Cloud Service |
   | Primary SSH Tunnel * | 129.152.150.222 (HR_APP_DB__Tunnel) |
   | Platform * | Linux |
   | Description | HQ employees database |
   | Administrator Email | hr_admin@enterprise.com |

3. Enter the following endpoint details:
   - **Endpoint Name**
   - **Type** - must be Oracle Database Cloud Service
   - **Platform** - Linux is automatically selected

- **Description** - meaningful description to identify endpoint
- **Administrator Email** - Optional field to receive endpoint related alerts

4. Click **Register**.

   After a short delay the **Endpoints** page shows the new endpoint in **Registered** state with an **Enrollment Token**.

5. Click **Endpoint Name**. The **Endpoint Details** page appears.

   Associate a default wallet with the registered endpoint now before enrolling the endpoint.

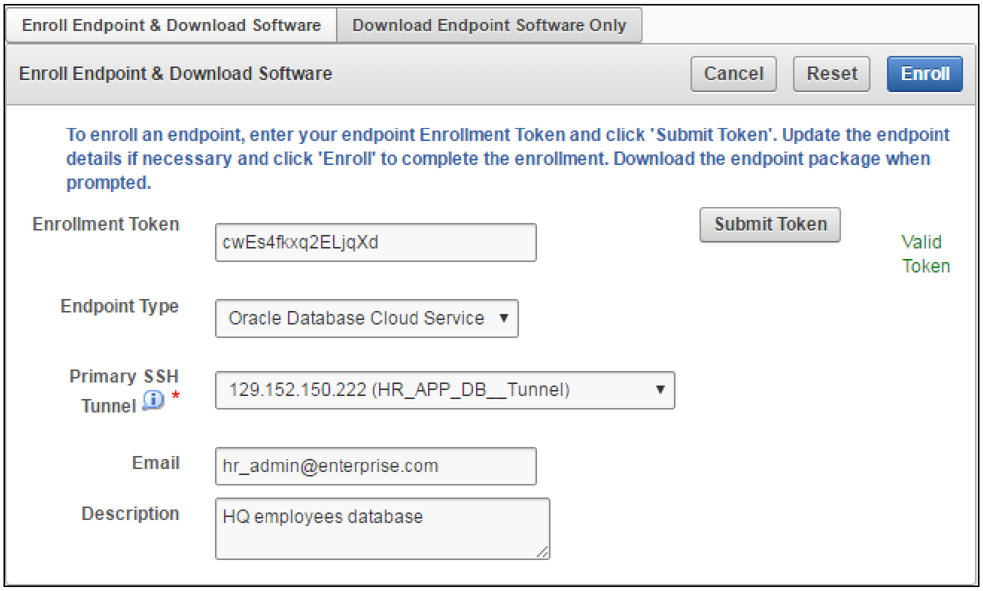
6. Copy the **Enrollment Token**. You will need it to download the endpoint software and enroll the endpoint (next step).

7. Log out of Oracle Key Vault and open a new session.

   The login page appears. *Do not log in*.

8. Click **Endpoint Enrollment and Software Download** immediately below **Login**. The **Enroll Endpoint & Download Software** page appears.

**Figure 9-8    Enroll Database as a Service Endpoint**



Enroll Endpoint & Download Software
Download Endpoint Software Only
Enroll Endpoint & Download Software
Cancel
Reset
Enroll
Submit Token
Valid Token

To enroll an endpoint, enter your endpoint Enrollment Token and click 'Submit Token'. Update the endpoint details if necessary and click 'Enroll' to complete the enrollment. Download the endpoint package when prompted.

| | |
|---|---|
| Enrollment Token | cwEs4fkxq2ELjqXd |
| Endpoint Type | Oracle Database Cloud Service |
| Primary SSH Tunnel * | 129.152.150.222 (HR_APP_DB__Tunnel) |
| Email | hr_admin@enterprise.com |
| Description | HQ employees database |

The fields are populated with the values that were chosen by the Key Vault system administrator while registering the endpoint. You can change these values while completing the enrollment of the endpoint. Note that you must select the **Primary SSH Tunnel** for Database as a Service endpoints from the drop down list. This is the only difference in the enrollment process from on-premises endpoints.

9. Enter the **Enrollment Token** and click **Submit Token** to validate the token.

10. Click **Enroll** to download the file `okvclient.jar`. to your local system. You must then move it to a secure directory on the Cloud Database as a Service instance with appropriate permissions in place so it cannot be read or copied by others.

```
$ scp -i <path-to-private-key-file> <path-to-okvclient.jar-on-local-mc>
oracle@node-ip-address:<path-to-okvclient.jar-on-cloud-db-instance>
```

Where:

`path-to-okvclient.jar-on-local-mc` refers to the location of `okvclient.jar` on an on-premises local machine.

`path-to-okvclient.jar-on-cloud-db-instance` refers to the location of `okvclient.jar` on the oracle cloud database as a service instance.

11. Ensure that you have the necessary administrative privileges to install software on the endpoint.
12. Ensure that you have JDK 1.5 or later installed, and that the `PATH` environment variable includes the `java` executable (in the `JAVA_HOME/bin` directory).

    Oracle Key Vault supports JDK versions 1.5, 1.6, 7, and 8.
13. Run the Shell utility `ORAENV` or `source ORAENV` command to set the correct environment variables on Oracle Database servers.
14. Check that the environment variables `ORACLE_BASE` and `ORACLE_HOME` are correctly set.

    If you used `ORAENV` to set these variables, you must verify that `ORACLE_BASE` points to the root directory for Oracle Databases, and that `ORACLE_HOME` points to a sub-directory under `ORACLE_BASE` where an Oracle Database is installed.
15. Navigate to the directory in which you saved the `okvclient.jar` file.
16. Run the `java` command to install the `okvclient.jar` file.

    ```
    java -jar okvclient.jar -d /home/oracle/okvutil -v
    ```

    In this command:

    - The `-d` argument specifies the directory location for the endpoint software and configuration files, in this case `/home/oracle/okvutil`.

      The environment variable `$OKV_HOME` refers to the directory where the endpoint software is installed, in this case `/home/oracle/okvutil`.
    - The `-v` argument writes the installation logs to the `$OKV_HOME/log/okvutil.deploy.log` file at the server endpoint.

    > **Note:**
    >
    > `-o` is an optional argument that allows you to overwrite the symlink reference to `okvclient.ora`, when `okvclient.jar` is deployed in a directory other than the original directory. This argument is used only when re-enrolling an endpoint.
17. The installation process prompts for a password. You can enter a password to create a **password-protected** wallet or create an **auto-login** wallet without a password as described below:

    - A password-protected wallet is an Oracle wallet file that store the endpoint's credentials to access Oracle Key Vault. This password will be required whenever the endpoint connects to Oracle Key Vault.

      Create a password-protected wallet by entering a password between 8 and 30 characters. Then press **Enter**.
    - Create an auto-login wallet by simply clicking **Enter**.

No password will be required when the endpoint connects to Oracle Key Vault. An auto-login wallet enables endpoint provisioning without human intervention.

```
Enter new Key Vault endpoint password (<enter> for auto-login): 
Key_Vault_endpoint_password
Confirm new endpoint password: Key_Vault_endpoint_password
```

The installation proceeds and completes with the following message:

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

A successful installation of the endpoint software creates the following directories:

- `bin`: contains the `okvutil` program, the `root.sh` and `root.bat` scripts, and the binary files `okveps.x64` and `okveps.x86`
- `conf`: contains the configuration file `okvclient.ora`
- `jlib`: contains the Java library files
- `lib`: contains the file `liborapkcs.so`
- `log`: contains the log files
- `ssl`: contains the TLS-related files and wallet files. The wallet files contain the endpoint credentials to connect to Oracle Key Vault.

  The `ewallet.p12` file refers to a password-protected wallet. The `cwallet.sso` file refers to an auto-login wallet.

18. On UNIX platforms, the `liborapkcs.so` file contains the library that the Oracle database uses to communicate with Oracle Key Vault. On Windows platforms, the `liborapkcs.dll` file contains the library that the Oracle database uses to communicate with Oracle Key Vault.

    If you are planning to use a TDE direct connection, then run `root.sh` on Oracle Linux x86-64, Solaris, AIX, and HP-UX (IA) installations. The `liborapkcs.so` file is copied to the following directory: /opt/oracle/extapi/64/hsm/oracle/1.0.0

    On Windows installations, run `root.bat`. The `liborapkcs.dll` file is copied to C:\oracle\extapi\64\hsm\oracle\1.0.0

    Log in as the root user and run the `root.sh` script. On Windows installations, run `root.bat`.

    ```
    $ sudo bin/root.sh
    ```

    ```
    bin\root.bat
    ```

    Or:

    ```
    $ su -
    # bin/root.sh
    ```

    On Windows platforms, you are prompted for the version of the RDBMS in use when you execute `root.bat`. Switch out of user root after completing this step.

19. Run the `okvutil list` command to verify that the endpoint software installed correctly, and that the endpoint can connect to the Oracle Key Vault server.

    If the endpoint is able to connect to Key Vault, a No objects found message appears:

```
$ ./okvutil list
No objects found
```

If a **Server connect failed** message appears at any time, you must troubleshoot the installation for possible issues. First check that environment variables are correctly set.

20. You can get help on the endpoint software with the `-h` option:

    ```
    java -jar okvclient.jar -h
    ```

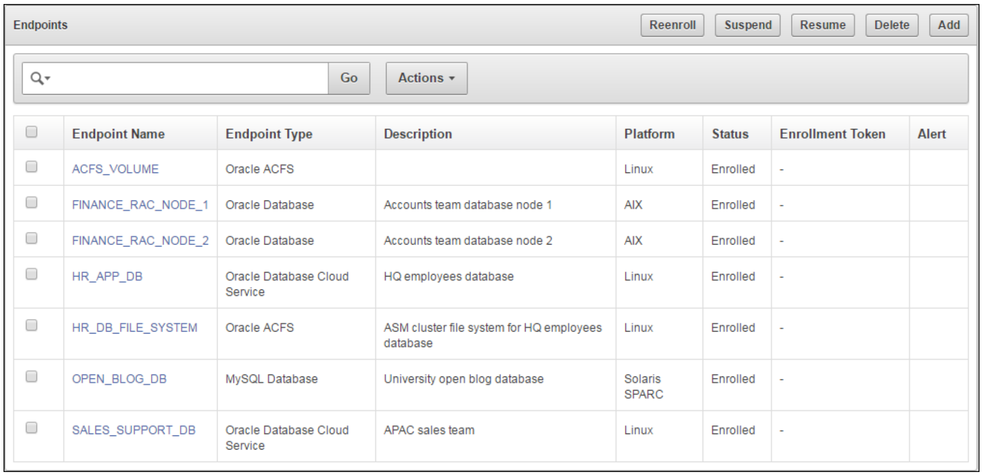
    The following output appears:

    ```
    Oracle Key Vault Release 12.2.0.6.0 (2017-12-15 15:36:49.839 PDT)
    Production on Fri Dec 15 19:55:31 PDT 2017
    Copyright (c) 1996, 2017 Oracle. All Rights Reserved.
    Usage: java -jar okvclient.jar [-h | -help] [[-v | -verbose] [-d
    <destination directory>] [-o]]
    ```

21. After installation Oracle recommends that you securely delete the endpoint software file `okvclient.jar`.

22. Click **Endpoints** to see the enrolled endpoint.

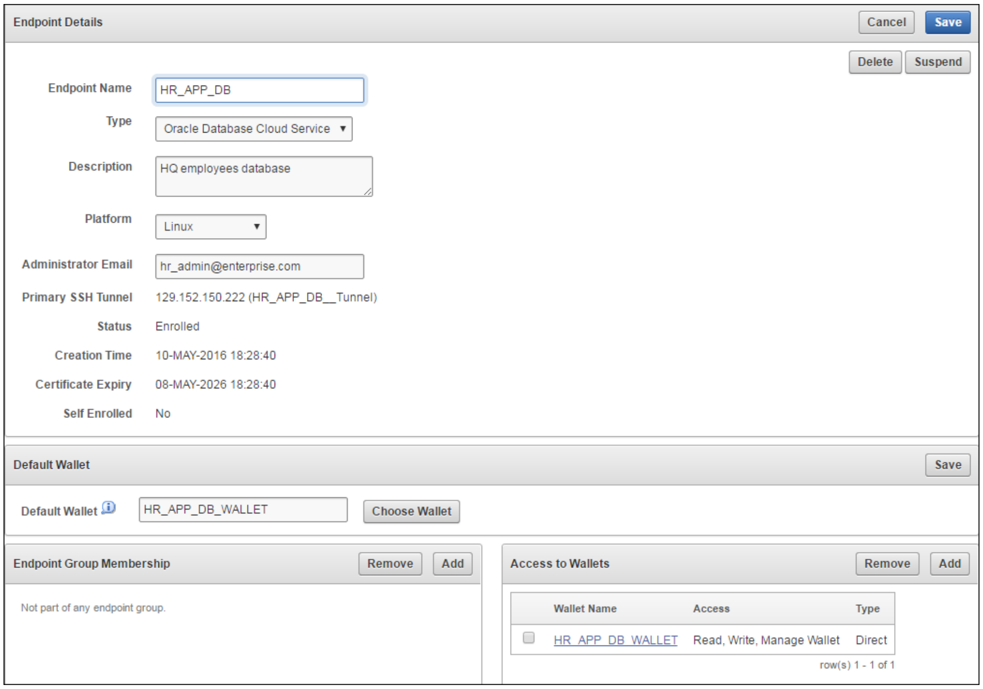
**Figure 9-9 Database as a Service Endpoints Among other Endpoints**



| □ | Endpoint Name | Endpoint Type | Description | Platform | Status | Enrollment Token | Alert |
|---|---|---|---|---|---|---|---|
| □ | ACFS_VOLUME | Oracle ACFS | | Linux | Enrolled | - | |
| □ | FINANCE_RAC_NODE_1 | Oracle Database | Accounts team database node 1 | AIX | Enrolled | - | |
| □ | FINANCE_RAC_NODE_2 | Oracle Database | Accounts team database node 2 | AIX | Enrolled | - | |
| □ | HR_APP_DB | Oracle Database Cloud Service | HQ employees database | Linux | Enrolled | - | |
| □ | HR_DB_FILE_SYSTEM | Oracle ACFS | ASM cluster file system for HQ employees database | Linux | Enrolled | - | |
| □ | OPEN_BLOG_DB | MySQL Database | University open blog database | Solaris SPARC | Enrolled | - | |
| □ | SALES_SUPPORT_DB | Oracle Database Cloud Service | APAC sales team | Linux | Enrolled | - | |

23. Click the Endpoint Name to see all the details for the endpoint on one page.

**Figure 9-10 Database as a Service Endpoint Details**



Endpoint Details

Endpoint Name: HR_APP_DB

Type: Oracle Database Cloud Service

Description: HQ employees database

Platform: Linux

Administrator Email: hr_admin@enterprise.com

Primary SSH Tunnel: 129.152.150.222 (HR_APP_DB__Tunnel)

Status: Enrolled

Creation Time: 10-MAY-2016 18:28:40

Certificate Expiry: 08-MAY-2026 18:28:40

Self Enrolled: No

Default Wallet

Default Wallet: HR_APP_DB_WALLET

Endpoint Group Membership

Not part of any endpoint group.

Access to Wallets

| | Wallet Name | Access | Type |
|---|---|---|---|
| □ | HR_APP_DB_WALLET | Read, Write, Manage Wallet | Direct |

row(s) 1 - 1 of 1

> **See Also:**
>
> - Types of Endpoint Enrollment (page 7-2)
> - Set the Default Wallet for an Endpoint (page 7-14)

## 9.6 Suspend a Database Cloud Service's Access to Oracle Key Vault

When using an on-premises Key Vault to manage the TDE master keys for Database as a Service endpoints, the master keys are never stored persistently in Oracle Cloud. This gives the on-premises Key Vault administrator the ability to control access to the encrypted data in the cloud.

The on-premises Oracle Key Vault administrator can suspend Database as a Service endpoints with a single click. This means that the Oracle Key Vault Server rejects all requests from the suspended endpoints. Since the endpoint cannot request keys from the Oracle Key Vault server, its ability to access encrypted data is lost once key cached in memory times out. For Oracle Database Cloud Service endpoints, this time out is 5 minutes by default.

The on-premises Oracle Key Vault administrator can resume a suspended endpoint. This means that the Oracle Key Vault server can start servicing requests from the reinstated endpoint. The reinstated endpoint can now retrieve keys from the Oracle Key Vault server and access sensitive data.

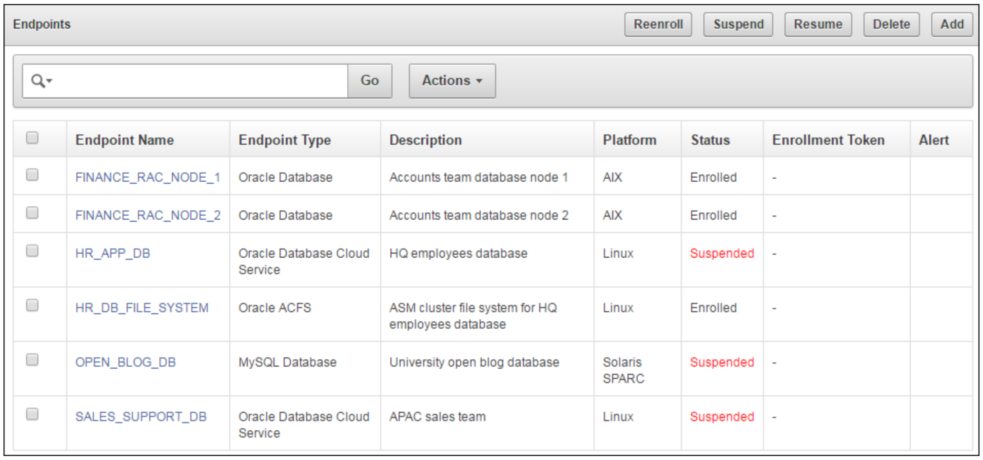
> **Caution:**
>
> The SUSPEND operation is a disruptive operation as it results in operational discontinuity and should be used with care. Usually, this option should be exercised only if there is a strong indication of abnormal activity in the Database as a Service instance.

You can only suspend enrolled endpoints. Endpoints in **Registered** state may not be suspended. Note that if you try to suspend endpoints that are already suspended, no operation will be performed. The endpoints will continue to be in suspended state.

To suspend endpoints (you can suspend multiple endpoints simultaneously):

1. Click **Endpoints**. The **Endpoints** page appears (Figure 9-9 (page 9-11)).
2. Check the boxes by the endpoints you wish to suspend.
3. Click **Suspend**. A confirmation message appears. Confirm the action by clicking **Yes**.
4. Click **Endpoints** to see the suspended endpoints. The status of suspended endpoints will be highlighted in red.

**Figure 9-11 Suspended Endpoints**



| ☐ | Endpoint Name | Endpoint Type | Description | Platform | Status | Enrollment Token | Alert |
|---|---|---|---|---|---|---|---|
| ☐ | FINANCE_RAC_NODE_1 | Oracle Database | Accounts team database node 1 | AIX | Enrolled | - | |
| ☐ | FINANCE_RAC_NODE_2 | Oracle Database | Accounts team database node 2 | AIX | Enrolled | - | |
| ☐ | HR_APP_DB | Oracle Database Cloud Service | HQ employees database | Linux | Suspended | - | |
| ☐ | HR_DB_FILE_SYSTEM | Oracle ACFS | ASM cluster file system for HQ employees database | Linux | Enrolled | - | |
| ☐ | OPEN_BLOG_DB | MySQL Database | University open blog database | Solaris SPARC | Suspended | - | |
| ☐ | SALES_SUPPORT_DB | Oracle Database Cloud Service | APAC sales team | Linux | Suspended | - | |

## 9.7 Resume a Database Cloud Service's Access to Oracle Key Vault

You can reinstate the connection between suspended Database Cloud Service endpoints and Key Vault. When you resume these endpoints their status will change to **Enrolled**. Note that resuming enrolled endpoints will not change their enrolled status.

To resume a Database Cloud Service's Access to Key Vault:

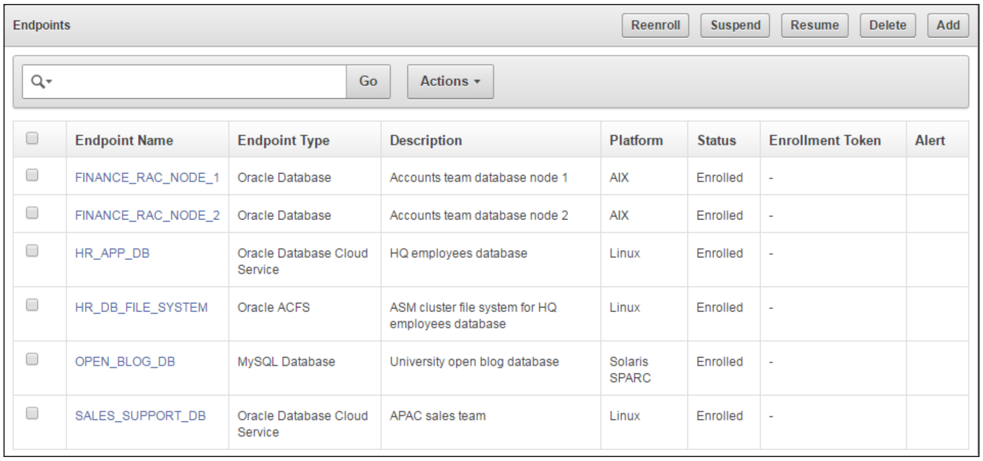
1. Click **Endpoints**. The **Endpoints** page appears.

   The suspended endpoints have status **Suspended** in red.

2. Check the boxes by the endpoints you wish to resume.
3. Click **Resume**.

   A confirmation message appears. Confirm the action by clicking **Yes**.
4. Click **Endpoints** to see the re-enrolled endpoints. Their status is **Enrolled**.

**Figure 9-12 Resumed Endpoints Showing Enrolled Status**



| □ | Endpoint Name | Endpoint Type | Description | Platform | Status | Enrollment Token | Alert |
|---|---|---|---|---|---|---|---|
| □ | FINANCE_RAC_NODE_1 | Oracle Database | Accounts team database node 1 | AIX | Enrolled | - | |
| □ | FINANCE_RAC_NODE_2 | Oracle Database | Accounts team database node 2 | AIX | Enrolled | - | |
| □ | HR_APP_DB | Oracle Database Cloud Service | HQ employees database | Linux | Enrolled | - | |
| □ | HR_DB_FILE_SYSTEM | Oracle ACFS | ASM cluster file system for HQ employees database | Linux | Enrolled | - | |
| □ | OPEN_BLOG_DB | MySQL Database | University open blog database | Solaris SPARC | Enrolled | - | |
| □ | SALES_SUPPORT_DB | Oracle Database Cloud Service | APAC sales team | Linux | Enrolled | - | |

## 9.8 Resuming a Database Endpoint Configured with a Password-based Keystore

A Database as a Service endpoint configured with auto-login keystore support will begin operations as soon as the endpoint is resumed. On the other hand, the Database as a Service endpoint configured with password keystore will not resume operations after the endpoint is resumed on the Oracle Key Vault server. The keystore on the Database as a Service instance was closed because Key Vault suspended the endpoint. You should open the password-based keystore on the Database as a Service instance to resume operations.

# 10 Endpoint Enrollment Automation with RESTful Services

The Key Vault RESTful Services utility enables you to automate the processes of endpoint enrollment, and virtual wallet management for a large distributed enterprise deployment.

## 10.1 About RESTful Services

Though the Oracle Key Vault management console user interface is efficient for managing several endpoints, the process of defining access control mappings between endpoints and virtual wallets is a manual one, with human administrators having to click through the user interface.

A large distributed enterprise deployment often requires automation through scripting to enable mass deployment. The RESTful services feature in Oracle Key Vault enables you to enroll and provision hundreds of endpoints, and define access control mappings between endpoints and their respective virtual wallets, to facilitate faster deployment with less human intervention. Additionally, you can automate the management of users, user groups, and endpoint groups with this feature.

With RESTful services, you can enroll and provision endpoints, create endpoint groups, and define access control mappings between endpoints, endpoint groups and virtual wallets. You can execute a single service command from the command line, or execute multiple service commands from a script. To run the service commands from the command line or the script, you will need a configuration file with certain properties set. In order to run the RESTful Service utility, the endpoint must have at minimum Java Runtime Environment version 1.7.0.21 installed.

You can use RESTful services in both Oracle Real Application Clusters (Oracle RAC) and multitenant environments. The configuration process in these environments is identical to the single instance environment.

After you use RESTful services to enroll and provision endpoints, you should disable the RESTful services to minimize the number of entry points to Oracle Key Vault.

You will follow these general steps to use the RESTful services execution process:

1. Enable RESTful services from the Oracle Key Vault management console.
2. Download the RESTful service utility `okvrestservices.jar`.
3. Create a configuration file, and then set the properties for the services that you want to run.
4. Execute the service using the RESTful service utility `okvrestservices.jar`, the configuration file, and service command plus options.
5. To run multiple RESTful service commands you must:
   a. Create a script, and write the RESTful commands into the script.
   b. Execute the services using the RESTful service utility `okvrestservices.jar`, the configuration file, and the script file.
6. Disable RESTful services when you are finished enrolling and provisioning endpoints.

## 10.2 Enable RESTful Services

There are three steps to enabling and using RESTful Services successfully.

- Step 1: Enable Network Services (page 10-2)
- Step 2: Enable RESTful Services (page 10-2)
- Step 3: Download the RESTful Software Utility (page 10-3)

### 10.2.1 Step 1: Enable Network Services

You must configure web access for RESTful clients by their IP addresses to access the Oracle Key Vault server. You can allow all IP addresses or restrict access to a subset of IP addresses that you designate in this step. Note, that this option will also restrict access to the Oracle Key Vault management console.

To enable network services:

1. Log in to the Oracle Key Vault management console as a user with System Administrator privileges.
2. Select **System**, then **System Settings** from the left sidebar.

   The **Settings** page appears.

   Go to the **Network Services** section
3. For **Web Access** select *one* of the IP address options for the RESTful client:
   a. **All** to allow all IP addresses.
   b. **IP address(es)** to designate a set of IP addresses. After you select this option enter the IP address(es) in the next field, separating each IP address by a space.
4. Click **Save** on the top right.

### 10.2.2 Step 2: Enable RESTful Services

To enable RESTful Services:

1. Log in to the Oracle Key Vault management console as a user with System Administrator privileges.
2. Select **System**, then **System Settings** from the left sidebar.

   The **Settings** page appears.

   Go to the **RESTful Services** section
3. Check the box to the right of **Enable**.
4. Click **Save** on the top right.

## 10.2.3 Step 3: Download the RESTful Software Utility

To download the RESTful software utility okvrestservices.jar:

1. Log in to the Oracle Key Vault management console as a user with System Administrator privileges.
2. Click **RESTful Service Utility** under **Downloads** in the left sidebar.

   The **Download RESTful Utility** page appears.
3. Click **Download** in the top right.

   A directory window appears with a prompt to save the utility file `okvrestservices.jar` in a local directory.
4. Save the file.

> **Note:**
>
> - If you install a third-party certificate you must download the RESTful software utility `okvrestservices.jar` again in order to use the new certificate.
> - You must re-download the RESTful software utility any time you change the certificate, or re-install the Key Vault appliance with new software or a backup.

# 10.3 Creating the RESTful Services Configuration File

You must set properties in the configuration file that the RESTful service utility will use to run commands.

1. Use the `okvclient.ora` file in the default location or create a file using any descriptive name. For example, for an endpoint named `hr_db`, it could be called `hr_db_endpoint.conf`.
2. Open the file and set the properties shown in the following table.

**Table 10-1 Properties to Set in the Configuration File**

| Property | Value | Option | Description |
|---|---|---|---|
| `server` | string | Required | Specifies the Oracle Key Vault server host name or IP address. The RESTful service utility. It uses the standard HTTPS port 443, which is optional. Specifying only the IP address or only host name is sufficient. |
| `script` | string | Optional (required only for multiple RESTful service commands) | Specifies the absolute path to the script file. You must set this property in order to run multiple RESTful service commands. |
| `log_level` | string | Required | Specifies one of the following log levels:<br>• `all` logs every message<br>• `severe` logs critical errors<br>• `warning` logs non-critical errors that might pose problems<br>• `info` logs general information<br>• `fine` logs detail; is useful for debugging<br>• `finest` logs the most detailed logging information |
| `log` | string | Optional | Specifies the absolute path to the log file. Set this property if you want to create a custom log file in a location of your choice. If you omit this setting, then the results are logged in the default log file `okvrestservices.log` and placed in the current directory as the default log file location. |
| `usr` | string | Optional | Specifies for the Oracle Key Vault account user name. You will be prompted to enter the user name, if you omit setting this property. Typically this user has the System Administrator or Key Administrator role with the necessary privileges to run the commands. |
| `pwd` | string | Optional | Specifies the user password. You will be prompted for the password, if you omit setting this property. For greater security, omit the password in the `configuration` file, and then enter it interactively when prompted. |
| `client_wallet` | string | Optional | Specifies the absolute path to the wallet in unattended mode. Because there is no human intervention in unattended mode, user credentials to log into the Oracle Key Vault server are placed in the wallet. If this option is used together with the user option, the command will pick up the user's credentials from the wallet to establish connection with the Key Vault server. |

3. Save the configuration file to a secure location.

## 10.4 Examples of Configuration Files

Two sample configuration files using the IP Address and Hostname for the server property are shown below:

**Example 10-1    Configuration File Using IP Address**

```
server=192.0.2.254
usr=okvadmin
log=/<absolute_path_to_your_log_file>/<your_log_file_name>
log_level=warning
```

**Example 10-2    Configuration File Using Host Name**

```
server=HR_HQ-Database
usr=okvadmin
log=/<absolute_path_to_your_log_file>/<your_log_file_name>
log_level=warning
```

## 10.5 Executing a Single RESTful Command

If you only want to run a few commands, you can run them singly from the command using the `-r` or `--service` option.

To run a single RESTful command:

1. Log in to the system, where you want to execute the command.
2. Ensure that the configuration file does not have a value for the `script` property.
3. Run the RESTful Service utility, specifying the configuration file with the `-c` option, the service with the `-r` or `--service` option, and the command specific options.

For example:

```
java -jar okvrestservices.jar -c conf_file -r create_endpoint -e hr_db_ep -d "HR
database endpoint" -q solaris64 -t oracle_db -m psmith@enterprise.com
User: Key_Vault_user_name
Password: Key_Vault_user_password
```

In this example:

- `-c` refers to the configuration file: `conf_file`.
- `-r` refers to the RESTful service: `create_endpoint`.
- `-e` refers to the endpoint name: `hr_db_ep`.
- `-d` refers to the description of the end point: `HR database endpoint`.
- `-q` refers to the endpoint platform: `solaris64`.
- `-t` refers to the endpoint type: `oracle_db`.
- `-m` refers to the endpoint email: `psmith@enterprise.com`.

> **Note:**
>
> Command line options have the priority over options specified in the configuration file or script. For example, if the property `usr` is specified in the configuration file and the command line, the command line option will override the one in the configuration file.

## 10.6 Executing Multiple RESTful Commands Using a Script

You can run a sequence of commands from the command line one at a time. However, a more efficient way to run a sequence of commands is to write them into a script. Each command in the script file is interpreted as a service command. You must invoke the script with the `-i` or `--script` option and provide the path to the script file.

> **Note:**
>
> You can define the `script` property in the configuration file to avoid entering it in the command line. The `script` parameter is entered only once: either in the configuration file or the command line.

To create the script:

1. Log in to the system where you want to execute the commands.
2. Create the script file, and write the service commands you want to run into the script file.

   For example, write the following commands into the script file to create an endpoint, an endpoint group, and add the endpoint to the endpoint group:

   ```
   create_endpoint -e hr_db_ep -d "HR database endpoint" -q solaris64 -t
   oracle_db -m psmith@enterprise.com
   create_endpoint_group -g hr_db_epg -d "HR endpoint group"
   add_epg_member -g hr_db_epg -e hr_db_ep
   ```

3. Save the script file with a descriptive name like `create_hr_endpoint_group.txt`.
4. Edit the configuration file named `conf_file` and add the `script` property. Set the property to the name of the script file and its full path. The configuration file looks like this:

   ```
   server=192.0.2.254
   usr=okvadmin
   log=/logs/okvrestservices.log
   log_level=warning
   script=/scripts/create_hr_endpoint_group.txt
   ```

5. Since the script property is defined in the configuration file you only need to specify the configuration file to run the RESTful Service utility:

   ```
   java -jar okvrestservices.jar -c conf_file
   User: Key_Vault_user
   Password: Key_Vault_user_password
   ```

   If you did *not* set the `script` property in the configuration file (**Step 4**) you must specify *both* the configuration and the script file to run the RESTful Service utility:

   ```
   java -jar okvrestservices.jar -c conf_file -i /scripts/
   create_hr_endpoint_group.txt
   User: Key_Vault_user
   Password: Key_Vault_user_password
   ```

The RESTful Services utility executes one command at a time. If a command fails the script will exit. The log file displays the results of all executed commands with their line numbers and messages reported at run time. This information appears for all log levels.

> **See Also:**
>
> Error Reporting (page 10-33) to learn more about logging in Key Vault

## 10.7 Script Creation Guidelines

Use the guidelines below to avoid script execution errors:

- The commands and syntax in the script are identical to those used on the command line.
- Each line in the script must be either a command or a line starting with the character #.
- Each command should be on its own line.
- Lines that do not have a command must start with the # character.
- Use the # character for comment and blank lines.
- The order in which command options appear do not matter.
- All required options must have valid values.
- You must specify the `-i` or `--script` option.
- Descriptions used for the `-d` or `--desc` option must be enclosed in double quotes if they contain spaces.

## 10.8 Disable RESTful Services

RESTful Services are disabled by default. We recommend that you enable RESTful Services for short periods during endpoint registration and enrollment only. After endpoints are enrolled you should disable RESTful Services.

To disable RESTful Services:

1. Log in to the Oracle Key Vault management console as a user with System Administrator privileges.
2. From the **System** tab, select **System Settings** in the left sidebar.

   The **System Settings** page appears.
3. Un-check the box to the right of **Enable** in the **RESTful Services** section.
4. In the **System Settings** page, click the **Save** button on the top right.

# 10.9 RESTful Services Command Reference

The RESTful Services command reference contains a detailed explanation of all the commands with examples, that will help you write and execute commands quickly.

## 10.9.1 RESTful Services Command Syntax

You must use the `java -jar` command to run the RESTful Services utility `okvrestservices` and provide a path to the configuration file.

The following table lists the common options used by all RESTful service commands:

**Table 10-2 Options Common to all RESTful Commands**

| Option | Required? | Description |
|---|---|---|
| `-c, --config` | Required | Refers to the absolute path to the configuration file. |
| `-i, --script <arg>` | Required for multiple RESTful service commands | Refers to the absolute path to the script file.<br>You must set this property in order to run multiple RESTful service commands. |
| `-r, --service <arg>` | Required | Refers to the RESTful service you want to execute listed in Table 2. |
| `-u, --usr <arg>` | Optional | Refers to the username of the Oracle Key Vault account user, who has the System or Key Administrator role. If you omit this option, you will be prompted to enter the username interactively. |
| `-p, --pwd <arg>` | Optional | Refers to the password for the Oracle Key Vault user account specified in the --usr option. If you omit this option, you will be prompted to enter the password interactively (recommended for greater security). |

The following table lists the RESTful service commands that you can use with the `-r` or `--service <arg>` option.

**Table 10-3 List of RESTful Service Commands**

| RESTful Service Command | Description |
|---|---|
| `create_endpoint` | Adds an endpoint to Key Vault. When added, the endpoint is in **registered** state. |
| `get_enrollment_token` | Gets the enrollment token to download the endpoint software for the registered endpoint. |
| `download` | Downloads the endpoint software `okvclient.jar` in order to install it at the endpoint. |
| `provision` | Downloads and installs the endpoint software okvclient.jar.After this the endpoint is in **enrolled** state. |
| `re_enroll` | Reenrolls an endpoint. |
| `re_enroll_all` | Reenrolls all endpoints. |
| `delete_endpoint` | Removes an endpoint from Key Vault. |
| `create_endpoint_group` | Adds a new endpoint group. |
| `add_epg_member` | Adds an endpoint to an endpoint group. The endpoint must already exist. |
| `drop_epg_member` | Removes an endpoint from an endpoint group. |
| `delete_endpoint_group` | Deletes an endpoint group. |
| `create_wallet` | Adds a virtual wallet to Oracle Key Vault. |
| `add_wallet_access_ep` | Sets access mappings on a virtual wallet for an endpoint. |
| `modify_wallet_access_ep` | Changes access mappings on a virtual wallet for an endpoint. |
| `drop_wallet_access_ep` | Removes access mappings on a virtual wallet for an endpoint. |
| `set_default_wallet` | Sets the default wallet for an endpoint. |
| `get_default_wallet` | Gets the default wallet for an endpoint. |
| `get_wallets` | Gets all virtual wallets for an endpoint. |
| `add_wallet_access_epg` | Sets access mappings on a virtual wallet for an endpoint group. |
| `modify_wallet_access_epg` | Changes access mappings on a virtual wallet for an endpoint group. |
| `drop_wallet_access_epg` | Removes access mappings on a virtual wallet for an endpoint group. |
| `delete_wallet` | Removes the virtual wallet from Key Vault. |
| `modify_wallet_desc` | Changes the virtual wallet description. |
| `modify_endpoint_name` | Changes the endpoint name. |
| `modify_endpoint_platform` | Changes the endpoint platform. |
| `modify_endpoint_type` | Changes the endpoint type. |
| `modify_endpoint_desc` | Changes the endpoint description. |
| `modify_endpoint_email` | Changes the endpoint's email. |
| `modify_endpoint_group_desc` | Changes the endpoint group's description. |

**Example 10-3    Specifying Short Form Options**

Specify short form options by using a single hyphen before the option.

```
java -jar okvrestservices.jar -c <path> [-r <RESTful_service> | -i <path>]
```

**Example 10-4    Specifying Long Form Options**

Specify long form options by using a double hyphen before the option.

```
java -jar okvrestservices.jar --config <path> [--service <RESTful_service> | --
script <path>]
```

## 10.9.2 RESTful Services Wallet Command Syntax

The following example shows RESTful service commands that pertain to Oracle wallets specified by the `--client_wallet` option. This wallet is used to store the user name and password in unattended mode to enable automated endpoint provisioning with no human intervention.

This is different from the virtual wallet specified by the `--wallet` option that are part of the virtual wallet commands.

**Table 10-4    Wallet Command Options**

| Option | Required? | Description |
|---|---|---|
| -A, --add | Optional | Adds a user to wallet. |
| -M, --modify | Optional | Modifies a user's password. |
| -L, --listuser | Optional | Lists the users who have access to a wallet. |
| -D, --delete | Optional | Deletes a user from a wallet. |
| -w, --wallet_name <arg> | Required | Stands for the wallet name. |
| -j, --client_wallet <arg> | Required | Stands for the absolute path to the wallet location. |
| -f, --force | Optional | Performs the operation without prompting for confirmation. |

**Example 10-5    Wallet Command Syntax**

Grant a user access to a wallet:

```
java -jar okvrestservices.jar -c <path_to_configuration file>/rest.init --
client_wallet <absolute path to wallet location> --add <user>
```

Modify a user password:

```
java -jar okvrestservices.jar -c <path_to_configuration file>/rest.init --
client_wallet <absolute path to wallet location> --modify <user>
```

List all the users who have access to a wallet:

```
java -jar okvrestservices.jar --config <path_to_configuration file>/rest.init --
client_wallet <absolute path to wallet location> --listuser <user>
```

Delete a user's access to a wallet:

```
java -jar okvrestservices.jar --config <path_to_configuration file>/rest.init --
client_wallet <absolute path to wallet location> --delete <user>
```

## 10.9.3 Commands to Add and Enroll Endpoints

The following group of commands: create, enroll, get_enrollment_token, download, and provision are used to add and enroll an endpoint to Key Vault. The endpoint is enrolled when the endpoint software `okvclient.jar` is downloaded and installed at the endpoint. An enrolled endpoint can upload security objects to Key Vault in order to store, share, and manage.

- create_endpoint Command (page 10-11)
- get_enrollment_token Command (page 10-12)
- download Command (page 10-13)
- provision Command (page 10-14)
- re_enroll Command (page 10-15)
- re_enroll_all Command (page 10-16)
  The `re_enroll_al` command re-enrolls all previously enrolled endpoints in order to upgrade the endpoint software.
- delete_endpoint Command (page 10-17)

### 10.9.3.1 create_endpoint Command

The `create_endpoint` command adds a new endpoint to Oracle Key Vault. After you add the endpoint, the endpoint will be in the **Registered** state.

**Syntax**

Short form:

```
create_endpoint -e endpoint_name -d "description" -q platform -m email_address
-t type
```

Long form:

```
create_endpoint --ep_name endpoint_name --desc "description" --ep_platform
platform --ep_email email_address --ep_type type
```

**Parameters**

| Parameter | Required? | Description |
|---|---|---|
| `-e, --ep_name` | Required | The name of the endpoint you want to add |
| `-d, --desc` | Optional | A user friendly description of the endpoint. If the description contains spaces, you must enclose it within double quotation marks. |
| `-q, --ep_platform` | Required | The endpoint platform. Allowed values are:<br>• `linux64`<br>• `solaris64`<br>• `solaris_sparc`<br>• `aix`<br>• `hpux`<br>• `hp-ux`<br>• `windows` |

| Parameter | Required? | Description |
|---|---|---|
| `-t`, `--ep_type` | Required | Type of the endpoint. Allowed values are:<br>• `oracle_db`<br>• `oracle_non_db`<br>• `other` |
| `-m`, `--ep_email` | Optional | Email address of the endpoint administrator |
| `-b, --type` *`arg`* | Required | Specifies the object type to check. Valid values include:<br>• `EP`<br>• `EPG`<br>• `WALLET` |
| `-c, --config` | Required | Specifies the absolute path to the configuration file |
| `-i`, `--script` *`arg`* | Required for multiple RESTful service commands | Specifies the absolute path to the script file. You must set this property in order to run multiple RESTful service commands. |
| `-p, --pwd` *`arg`* | Optional | Specifies the password for the Oracle Key Vault user account specified in the `--usr` option. If you omit this option, then you will be prompted to enter the password interactively. For greater security, omit this option. |
| `-r`, `--service` *`arg`* | Required | Specifies the RESTful service that you want to execute listed in RESTful Services Command Syntax (page 10-8) |
| `-u`, `--usr` *`arg`* | Optional | Specifies the user name of the Oracle Key Vault account user, who has the System or Key Administrator role. If you omit this option, then you will be prompted to enter the user name interactively |

### Short Form Example

In this example, an endpoint called `hr_db_ep` is added with an optional identifying description `'HR database endpoint'`, of type `oracle_db`, on platform `solaris64`, and endpoint administrator email, `psmith@example.com`.

```
java -jar okvrestservices.jar -c conf_file -r create_endpoint -e hr_db -d "HR
database endpoint" -q solaris64 -t oracle_db -m psmith@example.com -
```

### Long Form Example

```
java -jar okvrestservices.jar --config conf_file --service create_endpoint --
ep_name hr_db --desc "HR database endpoint" --ep_platform solaris64 --ep_type
oracle_db --ep_email psmith@example.com
```

## 10.9.3.2 get_enrollment_token Command

The `get_enrollment_token` command retrieves an enrollment token for a registered endpoint. This command will work only for endpoints in **registered** state. If the endpoint is already enrolled, you will get an error message.

### Syntax

Short form:

```
get_enrollment_token -e endpoint_name
```

Long form:

```
get_enrollment_token --ep_name endpoint_name
```

### Parameters

| Parameter | Required? | Description |
|---|---|---|
| `-e`, `--ep_name` | Required | Name of the endpoint. |

### Short Form Example

In this example a registered endpoint hr_db_ep gets the enrollment token, that will be used to download and install the endpoint software to the endpoint.

```
java -jar okvrestservices.jar -c conf_file -r get_enrollment_token -e hr_db_ep
```

### Long Form Example

```
java -jar okvrestservices.jar --config conf_file --service get_enrollment_token
--ep_name hr_db_ep
```

## 10.9.3.3 download Command

The `download` command downloads the endpoint software (`okvclient.jar`) to a directory that you name. The directory path is specified by the -o option. You can specify the absolute or relative path, or even set an environment variable to point to the path. If the directory exists, you will see an error message saying that the directory exists.

You can use either the download command or the provision command to enroll the endpoint. You cannot use both for a given endpoint.

### Syntax

Short form:

```
download -e endpoint_name -o <directory>
```

Long form:

```
download --ep_name endpoint_name -dir <directory>
```

### Parameters

| Parameter | Required? | Description |
|---|---|---|
| `-e`, `--ep_name` | Required | Name of the endpoint. |
| `-o`, `--dir` | Required | Absolute path to the download directory for the endpoint software. |

### Short Form Example

In this example the endpoint software, `okvclient.jar` is downloaded to `/home/oracle/downloads/` for endpoint `hr_db_ep`.

```
java -jar okvrestservices.jar -c conf_file -r download -e hr_db_ep -o /home/
oracle/downloads/
```

### Long Form Example

```
java -jar okvrestservices.jar --config conf_file --service download --ep_name
hr_db_ep --dir /home/oracle/downloads/
```

## 10.9.3.4 provision Command

You must meet the following prerequisites to run this command:

- You must be a user with system administrative privileges
- The soft link `/usr/bin/java` should point to Java 1.4 or above.
- You must know how the installation process determines the location of the `okvclient.ora` file

**See Also:**

Location of the OKVCLIENT.ORA File and Environment Variables (page 8-9)

The `provision` command downloads and installs the endpoint software in the specified directory, which should exist. This directory should have read, write and execute permissions for the owner and its group. For example, if Key Vault endpoint software is installed in an Oracle Database server, this endpoint installation directory should have read, write, and execute permissions by the user `oracle` and the group `oinstall`. This ensures that processes can access directories appropriately at runtime.

You can use either the download command or the provision command to enroll the endpoint. You cannot use both for a given endpoint

### Syntax

Short form:

```
provision [-a|-v  account_pwd ] -e endpoint_name -o <directory_path>
```

Long form:

When password is used to authenticate:

```
provision --endpoint_password account_pwd -ep_name endpoint_name --dir
<directory_path>
```

When no password is used (auto-login):

```
provision --autologin -ep_name endpoint_name --dir <directory_path>
```

### Parameters

| Parameter | Required? | Description |
|---|---|---|
| `-e, --ep_name` | Required | Name of the endpoint. |

| Parameter | Required? | Description |
|---|---|---|
| `-o, --dir` | Required | Existing directory in which to download and install the endpoint software. |
| `-v, --endpoint_password` | Optional | Endpoint password. If you omit this option (recommended), then the provision command prompts you for the password interactively. You must supply the password used for the wallet during endpoint software installation to communicate with the Key Vault server over mutually authenticated TLS. If you created an auto-login wallet without a password during endpoint software installation the endpoint credentials are stored in an Oracle wallet. |
| `-a, --autologin` | Required | It means that endpoint credentials to connect to the Key Vault server are stored in an auto-login wallet. |

### Short Form Examples

Auto-login Mode

In this example, the endpoint software is installed for endpoint `hr_db_ep` in the directory `/home/oracle/okvutil` without a password (in autologin mode).

```
java -jar okvrestservices.jar -c conf_file -r provision -a -e hr_db_ep -o /home/
oracle/okvutil/ -a
```

Password-protected Mode

In this example, the endpoint software is installed for endpoint `hr_db_ep` in the directory `/home/oracle/okvutil` with a password. Because the password option (-v --client_password) is omitted, it must be entered on the command line when prompted.

```
java -jar okvrestservices.jar -c conf_file -r provision -e hr_db_ep -o /home/
oracle/okvutil/
```

### Long Form Examples

```
java -jar okvrestservices.jar --config conf_file --service provision --autologin
--ep_name hr_db_ep --dir /home/oracle/okvutil/ -a

java -jar okvrestservices.jar --config conf_file --service provision --ep_name
hr_db_ep --dir /home/oracle/okvutil/
```

## 10.9.3.5 re_enroll Command

The `re_enroll` command re-enrolls a previously enrolled endpoint in order to upgrade the endpoint software.

### Syntax

Short form:

```
re_enroll -e endpoint_name
```

Long form:

```
re_enroll --ep_name endpoint_name
```

### Parameters

| Parameter | Required? | Description |
|---|---|---|
| `-e`, `--ep_name` | Required | Name of the endpoint. |

In this example, endpoint `hr_db_ep` will be reenrolled.

```
java -jar okvrestservices.jar -c conf_file -r re_enroll -e hr_db_ep
```

### Long Form Example

```
java -jar okvrestservices.jar --config conf_file --service re_enroll --ep_name
hr_db_ep
```

## 10.9.3.6 re_enroll_all Command

The `re_enroll_al` command re-enrolls all previously enrolled endpoints in order to upgrade the endpoint software.

### Syntax

Short and long form:

```
re_enroll_all
```

### Parameters

| Parameter | Required? | Description |
|---|---|---|
| `-c, --config` | Required | Specifies the absolute path to the configuration file |
| `-i`, `--script` *`arg`* | Required for multiple RESTful service commands | Specifies the absolute path to the script file. You must set this property in order to run multiple RESTful service commands. |
| `-p, --pwd` *`arg`* | Optional | Specifies the password for the Oracle Key Vault user account specified in the `--usr` option. Enclose this value in double quotation marks if the value contains spaces, slashes, or colons.<br>If you omit this option, then you will be prompted to enter the password interactively. For greater security, omit this option. |
| `-r`, `--service` *`arg`* | Required | Specifies the RESTful service that you want to execute listed in RESTful Services Command Syntax (page 10-8) |
| `-u`, `--usr` *`arg`* | Optional | Specifies the user name of the Oracle Key Vault account user, who has the System or Key Administrator role. If you omit this option, then you will be prompted to enter the user name interactively |

#### Short Form Example

```
java -jar okvrestservices.jar -c conf_file -r re_enroll
```

#### Long Form Example

```
java -jar okvrestservices.jar --config conf_file --service re_enroll
```

#### Related Topics

- Re-Enroll All Endpoints With a Single RESTful Command (page xxiv)

### 10.9.3.7 delete_endpoint Command

The `delete_endpoint` command removes an endpoint from Key Vault. A confirmation message appears asking if you are sure you want to delete the endpoint.

You may use the `-f` or `--force` option to remove the endpoint without a confirmation message. Use the `-f` or `--force` option carefully as it suppresses the confirmation message.

#### Syntax

Short form:

```
delete_endpoint -f -e endpoint_name
```

Long form:

```
delete_endpoint --force --ep_name endpoint_name
```

#### Parameters

| Parameter | Required? | Description |
|---|---|---|
| `-e`, `--ep_name` | Required. | Name of the endpoint. |
| `-f`, `--force` | Optional. | Forces the deletion and suppresses the confirmation message. |

#### Short Form Example

This example shows endpoint `sales_db_ep` being removed from Key Vault without confirmation.

```
java -jar okvrestservices.jar -c conf_file -r delete_endpoint -f -e sales_db_ep
```

#### Long Form Example

```
java -jar okvrestservices.jar --config conf_file --service delete_endpoint --
force --ep_name sales_db_ep
```

## 10.9.4 Modify Endpoint Details Commands

You can modify endpoint details after creating the endpoint to accommodate changes in function, name, platform, type, and email.

- modify_endpoint_name Command (page 10-18)

## 10.9.4.1 modify_endpoint_name Command

The `modify_endpoint_name` command changes the name of an endpoint.

### Syntax

Short form:

```
modify_endpoint_name -e endpoint_name -n new_endpoint_name
```

Long form:

```
modify_endpoint_name --ep_name endpoint_name --ep_new_name new_endpoint_name
```

### Parameters

| Parameter | Required? | Description |
|---|---|---|
| `-e, --ep_name` | Required | Name of the endpoint. |
| `-n, --ep_new_name` | Required | New name for this endpoint |

### Short Form Example

This example changes the name of endpoint `hr_db` to that of `hr_db_ep`.

```
java -jar okvrestservices.jar -c conf_file -r modify_endpoint_name -e hr_db -k
hr_db_ep
```

### Long Form Example

```
java -jar okvrestservices.jar --config conf_file --service modify_endpoint_name
--ep_name hr_db --ep_new_name hr_db_ep
```

## 10.9.4.2 modify_endpoint_type Command

The `modify_endpoint_type` command changes the endpoint type.

### Syntax

Short form:

```
modify_endpoint_type -e endpoint_name -t endpoint_type
```

Long form:

```
modify_endpoint_type --ep_name endpoint_name --ep_type endpoint_type
```

### Parameters

| Parameter | Required? | Description |
|---|---|---|
| `-e, --ep_name` | Required | Name of the endpoint. |
| `-t, --ep_type` | Required | Type of the endpoint. Values are as follows:<br>- `oracle_db`<br>- `oracle_non_db`<br>- `other` |

### Short Form Example

This example changes the endpoint type for endpoint `hr_db` to `oracle_db`.

```
java -jar okvrestservices.jar -c conf_file -r modify_endpoint_type -e hr_db -t
oracle_db
```

### Long Form Example

```
java -jar okvrestservices.jar --config conf_file --service modify_endpoint_type
--ep_name hr_db --ep_type oracle_db
```

## 10.9.4.3 modify_endpoint_platform Command

The `modify_endpoint_platform` command changes the platform for an endpoint.

### Syntax

Short form:

```
modify_endpoint_platform -e endpoint_name -q endpoint_platform
```

Long form:

```
modify_endpoint_platform --ep_name endpoint_name --ep_platform endpoint_platform
```

### Parameters

| Parameter | Required? | Description |
|---|---|---|
| `-e, --ep_name` | Required | Name of the endpoint |
| `-q, --ep_platform` | Required | Platform of the server for this endpoint. Values are as follows:<br>- `linux64`<br>- `solaris64`<br>- `solaris_sparc`<br>- `aix`<br>- `hpux`<br>- `hp-ux`<br>- `windows` |
| `-b, --type` *`arg`* | Required | Specifies the object type to check. Valid values include:<br>- `EP`<br>- `EPG`<br>- `WALLET` |

| Parameter | Required? | Description |
|---|---|---|
| `-c, --config` | Required | Specifies the absolute path to the configuration file |
| `-i`, `--script` *`arg`* | Required for multiple RESTful service commands | Specifies the absolute path to the script file. You must set this property in order to run multiple RESTful service commands. |
| `-p, --pwd` *`arg`* | Optional | Specifies the password for the Oracle Key Vault user account specified in the `--usr` option. If you omit this option, then you will be prompted to enter the password interactively. For greater security, omit this option. |
| `-r`, `--service` *`arg`* | Required | Specifies the RESTful service that you want to execute listed in RESTful Services Command Syntax (page 10-8) |
| `-u`, `--usr` *`arg`* | Optional | Specifies the user name of the Oracle Key Vault account user, who has the System or Key Administrator role. If you omit this option, then you will be prompted to enter the user name interactively |

### Short Form Example

This example changes the platform for endpoint `hr_db` to `aix`.

```
java -jar okvrestservices.jar -c conf_file -r modify_endpoint_platform -e hr_db
-q aix
```

### Long Form Example

```
java -jar okvrestservices.jar --config conf_file --service
modify_endpoint_platform --ep_name hr_db --ep_platform aix
```

## 10.9.4.4 modify_endpoint_desc Command

The `modify_endpoint_desc` command changes the description of an endpoint.

### Syntax

Short form:

```
modify_endpoint_desc -e endpoint_name -d "new_desc"
```

Long form:

```
modify_endpoint_desc --ep_name endpoint_name --desc "new_desc"
```

### Parameters

| Parameter | Required? | Description |
|---|---|---|
| `-e`, `--ep_name` | Required | Name of the endpoint. |
| `-d`, `--desc` | Required | New description string for this endpoint enclosed within double quotes. |

### Short Form Example

This example changes the endpoint description for endpoint hr_db to “HR database endpoint group".

```
java -jar okvrestservices.jar -c conf_file -r modify_endpoint_desc -e hr_db -d
"HR database endpoint group"
```

### Long Form Example

```
java -jar okvrestservices.jar --config conf_file --service modify_endpoint_desc
--ep_name hr_db --desc "HR database endpoint group"
```

## 10.9.4.5 modify_endpoint_email Command

The modify_endpoint_email command changes the email address for the endpoint.

### Syntax

Short form:

```
modify_endpoint_email -e endpoint_name -m endpoint_email_address
```

Long form:

```
modify_endpoint_email --ep_name endpoint_name --ep_email endpoint_email_address
```

### Parameters

| Parameter | Required? | Description |
|---|---|---|
| -e, --ep_name | Required | Name of the endpoint. |
| -m, --ep_email | Required | The new email address for this endpoint |

### Short Form Example

This example changes the email of endpoint hr_db to tjones@enterprise.com.

```
java -jar okvrestservices.jar -c conf_file -r modify_endpoint_email -e hr_db -m
tjones@enterprise.com
```

### Long Form Example

```
java -jar okvrestservices.jar --config conf_file --service modify_endpoint_email
--ep_name hr_db --ep_email tjones@enterprise.com
```

## 10.9.5 Endpoint Group Commands

The following commands describe how to create and manage endpoint groups.

- create_endpoint_group Command (page 10-22)
- add_epg_member Command (page 10-22)
- drop_epg_member Command (page 10-23)
- delete_endpoint_group Command (page 10-23)
- modify_endpoint_group_desc Command (page 10-24)

### 10.9.5.1 create_endpoint_group Command

The `create_endpoint_group` command creates a new endpoint group.

**Syntax**

Short form:

```
create_endpoint_group -g endpoint_group_name -d "endpoint group description"
```

Long form:

```
create_endpoint_group --epg_name endpoint_group_name --desc "endpoint group
description"
```

**Parameters**

| Parameter | Required? | Description |
|---|---|---|
| `-g, --epg_name` | Required | Name of the endpoint group. |
| `-d, --desc` | Optional | A user-friendly description of the endpoint group enclosed within double quotes. |

**Short Form Example**

This example shows an endpoint group called `epg_hr` being created with the description “`HR endpoint group`“.

```
java -jar okvrestservices.jar -c conf_file -r create_endpoint_group -g epg_hr -d
"HR endpoint group"
```

**Long Form Example**

```
java -jar okvrestservices.jar --config conf_file --service create_endpoint_group
--epg_name epg_hr --desc "HR endpoint group"
```

### 10.9.5.2 add_epg_member Command

The `add_epg_member` command adds an existing endpoint to an endpoint group. If the endpoint does not exist, you will get an error message.

**Syntax**

Short form:

```
add_epg_member -g endpoint_group_name -e endpoint_member
```

Long form:

```
add_epg_member --epg_name endpoint_group_name --ep_name endpoint_member
```

**Parameters**

| Parameter | Required? | Description |
|---|---|---|
| `-e, --ep_name` | Required | Name of the endpoint. |
| `-g, --epg_name` | Required | Name of the endpoint group. |

### Short Form Example

This example shows an endpoint called hr_db_ep being added to endpoint group epg_hr.

```
java -jar okvrestservices.jar -c conf_file -r add_epg_member -g epg_hr -e
hr_db_ep
```

### Long Form Example

```
java -jar okvrestservices.jar --config conf_file --service add_epg_member --
epg_name epg_hr --ep_name hr_db_ep
```

## 10.9.5.3 drop_epg_member Command

The drop_epg_member command removes an endpoint from an endpoint group.

### Syntax

Short form:

```
drop_epg_member -g endpoint_group -e endpoint_name
```

Long form:

```
drop_epg_member --epg_name endpoint_name --ep_name endpoint_group
```

### Parameters

| Parameter | Required? | Description |
|---|---|---|
| -e, --ep_name | Required | Name of the endpoint. |
| -g, --epg_name | Required | Name of the endpoint group. |

### Short Form Example

This example shows endpoint hr_db_ep being removed from endpoint group epg_hr.

```
java -jar okvrestservices.jar -c conf_file -r drop_epg_member -e hr_db_ep -g
epg_hr
```

### Long Form Example

```
java -jar okvrestservices.jar --config conf_file --service drop_epg_member --
ep_name hr_db_ep --epg_name epg_hr
```

## 10.9.5.4 delete_endpoint_group Command

The delete_endpoint_group command removes an endpoint group from Key Vault. If the endpoint group does not exist you will see an error message.

### Syntax

Short form:

```
delete_endpoint_group -f -g endpoint_group
```

Long form:

```
delete_endpoint_group --force --endpoint_group
```

**Parameters**

| Parameter | Required? | Description |
|---|---|---|
| `-g`, `--epg_name` | Required | Name of the endpoint group. |
| `-f`, `--force` | Optional | Force the deletion and suppresses the confirmation message. |

**Short Form Example**

This example deletes the endpoint group `epg_hr`.

```
java -jar okvrestservices.jar -c conf_file -r delete_endpoint_group -f -g epg_hr
```

**Long Form Example**

```
java -jar okvrestservices.jar --config conf_file --service delete_endpoint_group
--force --epg_name epg_hr
```

## 10.9.5.5 modify_endpoint_group_desc Command

The `modify_endpoint_group_desc` command changes the description of an endpoint group.

**Syntax**

Short form:

```
modify_endpoint_group_desc -g endpoint_group_name -d "endpoint_group_description"
```

Long form:

```
modify_endpoint_group_desc --epg_name endpoint_group_name --desc
"endpoint_group_description"
```

**Parameters**

| Parameter | Required? | Description |
|---|---|---|
| `-g`, `--epg_name` | Required | Name of the endpoint group. |
| `-d`, `--desc` | Required | The new description string for the endpoint group enclosed within double quotes. |

**Short Form Example**

This example shows the endpoint group `epg_hr` getting a description “`HR DB endpoint group`“.

```
java -jar okvrestservices.jar -c conf_file -r modify_endpoint_group_desc -g
epg_hr -d "HR DB endpoint group"
```

**Long Form Example**

```
java -jar okvrestservices.jar --config conf_file --service
modify_endpoint_group_desc --epg_name epg_hr --desc "HR DB endpoint group"
```

## 10.9.6 Virtual Wallet Commands

Virtual wallet commands enable you to manage the lifecycle of a virtual wallet or define access control mappings between virtual wallets and endpoints or endpoint groups.

You must be a key administrator to run virtual wallet commands.

### 10.9.6.1 create_wallet Command

The `create_wallet` command enables you to create a virtual wallet.

#### Syntax

Short form:

```
create_wallet -w virtual_wallet_name -d "wallet_description"
```

Long form:

```
create_wallet --wallet_name wallet_name --desc "wallet_description"
```

#### Parameters

| Parameter | Required? | Description |
|---|---|---|
| `-w, --wallet_name` | Required | Name of the virtual wallet. |
| `-d, --desc` | Optional | A descriptive name for the virtual wallet enclosed within double quotes. |

#### Short Form Example

This example creates a wallet named `hr_wallet` with the description “`HR DB endpoint group`“.

```
java -jar okvrestservices.jar -c conf_file -r create_wallet -w hr_wallet -d
"Virtual wallet for HR endpoint"
```

### Long Form Example

```
java -jar okvrestservices.jar --config conf_file --service create_wallet --
wallet hr_wallet --desc "Virtual wallet for HR endpoint"
```

## 10.9.6.2 modify_wallet_desc Command

The `modify_wallet_desc` command modifies the description of an existing virtual wallet.

### Syntax

Short form:

```
modify_wallet_desc -w virtual_wallet_name -d "wallet_desc"
```

Long form:

```
modify_wallet_desc --wallet_name virtual_wallet_name --desc "wallet_desc"
```

### Parameters

| Parameter | Required? | Description |
|---|---|---|
| `-w`, `--wallet_name` | Required | Name of the virtual wallet. |
| `-d`, `--desc` | Required | The new description string for the virtual wallet enclosed within double quotes. |

### Short Form Example

This example gives the wallet `hr_wallet` a new description of “`HR endpoint virtual wallet`“.

```
java -jar okvrestservices.jar -c conf_file -r modify_wallet_desc -w hr_wallet -d
"HR endpoint virtual wallet"
```

### Long Form Example

```
java -jar okvrestservices.jar --config conf_file --service modify_wallet_desc
--wallet_name hr_wallet --desc "HR endpoint virtual wallet"
```

## 10.9.6.3 add_wallet_access_ep Command

The `add_wallet_access_ep` command grant an endpoint a level of access to a virtual wallet.

### Syntax

Short form:

```
add_wallet_access_ep -e endpoint_name -w virtual_wallet_name -l
wallet_access_level
```

Long form:

```
add_wallet_access_ep --ep_name endpoint_name --wallet_name virtual_wallet_name
--access_level wallet_access_level
```

### Parameters

| Parameter | Required? | Description |
|---|---|---|
| `-e, --ep_name` | Required | Name of the endpoint. |
| `-w, --wallet_name` | Required | Name of the virtual wallet. |
| `-l, --access_level` | Required | Level of access for the virtual wallet. Values are as follows:<br>• `ro`: Read only<br>• `rm`: Read and modify<br>• `ro_mw`: Read only and manage virtual wallet<br>• `rm_mw`: Read and modify and manage virtual wallet |

### Short Form Example

This example adds the read-only access privilege on the wallet `hr_wallet` to endpoint `hr_db_ep`.

```
java -jar okvrestservices.jar -c conf_file -r add_wallet_access_ep -e hr_db_ep
-w hr_wallet -l ro
```

### Long Form Example

```
java -jar okvrestservices.jar --config conf_file --service add_wallet_access_ep
--ep_name hr_db_ep --wallet_name hr_wallet --access_level ro
```

## 10.9.6.4 modify_wallet_access_ep Command

The `modify_wallet_access_ep` command changes the virtual wallet access level to an endpoint.

### Syntax

Short form:

```
modify_wallet_access_ep -e endpoint_name -w virtual_wallet_name -l
virtual_wallet_access_level
```

Long form:

```
modify_wallet_access_ep --ep_name endpoint_name --wallet_name
virtual_wallet_name --access_level wallet_access_level
```

### Parameters

| Parameter | Required? | Description |
|---|---|---|
| `-e, --ep_name` | Required | Name of the endpoint. |
| `-w, --wallet_name` | Required | Name of the virtual wallet. |

| Parameter | Required? | Description |
|---|---|---|
| `-l, --access_level` | Required | Level of access for the virtual wallet. Values are as follows:<br>• `ro`: Read only<br>• `rm`: Read and modify<br>• `ro_mw`: Read only, and manage virtual wallet<br>• `rm_mw`: Read, modify, and manage virtual wallet |

### Short Form Example

This example modifies the access level on wallet `hr_db` to read-only plus manage wallet.

```
java -jar okvrestservices.jar -c conf_file -r modify_wallet_access_ep -e
hr_db_ep -w hr_wallet -l ro_mw
```

### Long Form Example

```
java -jar okvrestservices.jar --config conf_file --service
modify_wallet_access_ep --ep_name hr_db_ep --wallet_name hr_wallet --
access_level ro_mw
```

## 10.9.6.5 drop_wallet_access_ep Command

The `drop_wallet_access_ep` command removes an endpoint's access to a wallet.

### Syntax

Short form:

```
drop_wallet_access_ep -e endpoint_name -w virtual_wallet_name
```

Long form:

```
drop_wallet_access_ep --ep_name endpoint_name --wallet_name virtual_wallet_name
```

### Parameters

| Parameter | Required? | Description |
|---|---|---|
| `-e, --ep_name` | Required | Name of the endpoint. |
| `-w, --wallet_name` | Required | Name of the virtual wallet. |

### Short Form Example

This example removes access to wallet `hr_wallet` for the endpoint `hr_db_ep`.

```
java -jar okvrestservices.jar -c conf_file -r drop_wallet_access_ep -e hr_db_ep
-w hr_wallet
```

### Long Form Example

```
java -jar okvrestservices.jar --config conf_file --service drop_wallet_access_ep
--ep_name hr_db_ep --wallet_name hr_wallet
```

## 10.9.6.6 set_default_wallet Command

The `set_default_wallet` command sets the default wallet for an endpoint.

**Syntax**

Short form:

```
set_default_wallet -e endpoint_name -w virtual_wallet_name
```

Long form:

```
set_default_wallet --ep_name --wallet_name virtual_wallet_name
```

**Parameters**

| Parameter | Required? | Description |
|---|---|---|
| `-w, --wallet_name` | Required | Name of the virtual wallet. |
| `-e, --ep_name` | Required | Endpoint name for whom default wallet is set. |

**Short Form Example**

This example sets the default wallet `hr_wallet` for the endpoint `hr_db`.

```
java -jar okvrestservices.jar -c conf_file -r set_default_wallet -e hr_db -w
hr_wallet
```

**Long Form Example**

```
java -jar okvrestservices.jar --config conf_file --service set_default_wallet
--ep_name hr_db --wallet_name hr_wallet
```

## 10.9.6.7 get_default_wallet Command

The `get_default_wallet` command gets the default wallet associated with an endpoint.

**Syntax**

Short form:

```
get_default_wallet -e endpoint_name
```

Long form:

```
get_default_wallet --ep_name endpoint_name
```

**Parameters**

| Parameter | Required? | Description |
|---|---|---|
| `-e,--ep_name` | Required | Endpoint name, whose default wallet to get. |

### Short Form Example

To get the default wallet associated with an endpoint `hr_db` you must supply the endpoint name to the command as follows:

```
java -jar okvrestservvices.jar -c conf_file -r get_default_wallet -e hr_db
```

### Long Form Example

```
java -jar okvrestservvices.jar -c conf_file -service get_default_wallet --
ep_name hr_db
```

## 10.9.6.8 get_wallets Command

The `get_wallets` command gets all the virtual wallets associated with an endpoint.

### Syntax

Short form:

```
get_wallets -e endpoint_name
```

Long form:

```
get_wallets --ep_name endpoint_name
```

### Parameters

| Parameter | Required? | Description |
|---|---|---|
| `-e, -—ep_name` | Required | Endpoint name, whose virtual wallets to get. |

### Short Form Example

To get all the virtual wallets associated with an endpoint `hr_db` you must supply the endpoint name to the command as follows:

```
java -jar okvrestservvices.jar -c conf_file -r get_wallets -e hr_db
```

### Long Form Example

```
java -jar okvrestservvices.jar -c conf_file -service get_wallets --ep_name hr_db
```

## 10.9.6.9 add_wallet_access_epg Command

The `add_wallet_access_epg` command grants an endpoint group a level of access to a virtual wallet.

### Syntax

Short form:

```
add_wallet_access_epg -g endpoint_group_name -w virtual_wallet_name -l
virtual_wallet_access_level
```

Long form:

```
add_wallet_access_epg --epg_name endpoint_group_name --wallet_name
virtual_wallet_name --access_level wallet_access_level
```

### Parameters

| Parameter | Required? | Description |
|---|---|---|
| `-g, --epg_name` | Required | Name of the endpoint group. |
| `-w, --wallet_name` | Required | Name of the virtual wallet. |
| `-l, --access_level` | Required | Level of access for the virtual wallet. Values are as follows:<br>• `ro`: Read only<br>• `rm`: Read and modify<br>• `ro_mw`: Read only and manage virtual wallet<br>• `rm_mw`: Read and modify and manage virtual wallet |

### Short Form Example

This example shows read-only access being granted to endpoint group `epg_hr`.

```
java -jar okvrestservices.jar -c conf_file -r add_wallet_access_epg -g epg_hr -w
hr_wallet -l ro
```

### Long Form Example

```
java -jar okvrestservices.jar --config conf_file --service add_wallet_access_epg
-epg_name epg_hr --wallet_name hr_wallet --access_level ro
```

## 10.9.6.10 modify_wallet_access_epg Command

The `modify_wallet_access_epg` command modifies the virtual wallet access level to an endpoint group.

### Syntax

Short form:

```
modify_wallet_access_epg -g endpoint_group_name -w virtual_wallet_name -l
virtual_wallet_access_level
```

Long form:

```
modify_wallet_access_epg --epg_name endpoint_group_name --wallet_name
virtual_wallet_name --access_level wallet_access_level
```

### Parameters

| Parameter | Required? | Description |
|---|---|---|
| `-g, --epg_name` | Required | Name of the endpoint group. |
| `-w, --wallet_name` | Required | Name of the virtual wallet. |

| Parameter | Required? | Description |
|---|---|---|
| `-l, --access_level` | Required | Level of access for the virtual wallet. Values are as follows:<br>• `ro`: Read only<br>• `rm`: Read and modify<br>• `ro_mw`: Read only and manage virtual wallet<br>• `rm_mw`: Read and modify and manage virtual wallet |

### Short Form Example

This example shows endpoint group `epg_hr` being granted read, modify, and manage privileges on wallet `hr_wallet`.

```
java -jar okvrestservices.jar -c conf_file -r modify_wallet_access_epg -g epg_hr
-w hr_wallet -l rm_mw
```

### Long Form Example

```
java -jar okvrestservices.jar --config conf_file --service
modify_wallet_access_epg --epg_name epg_hr --wallet_name hr_wallet --
access_level rm_mw
```

## 10.9.6.11 drop_wallet_access_epg Command

The `drop_wallet_access_epg` command removes an endpoint group's access to virtual wallet.

### Syntax

Short form:

```
drop_wallet_access_epg -g endpoint_group_name -w virtual_wallet_name
```

Long form:

```
drop_wallet_access_epg --epg_name endpoint_group_name --wallet_name
virtual_wallet_name
```

### Parameters

| Parameter | Required? | Description |
|---|---|---|
| `-g, --epg_name` | Required | Name of the endpoint group. |
| `-w, --wallet_name` | Required | Name of the virtual wallet. |

### Short Form Example

This example shows endpoint group `epg_hr` being granted read, modify, and manage access to wallet `hr_wallet`.

```
java -jar okvrestservices.jar -c conf_file -r modify_wallet_access_epg -g epg_hr
-w hr_wallet -l rm_mw
```

### Long Form Example

```
java -jar okvrestservices.jar --config conf_file --service
modify_wallet_access_epg --epg_name epg_hr --wallet_name hr_wallet -l rm_mw
```

## 10.9.6.12 delete_wallet Command

The `delete_wallet` command deletes a wallet from Key Vault.

### Syntax

Short form:

```
delete_wallet -f -w virtual_wallet_name
```

Long form:

```
delete_wallet --force --wallet_name virtual_wallet_name
```

### Parameters

| Parameter | Required? | Description |
|---|---|---|
| `-w, --wallet_name` | Required | Name of the virtual wallet. |
| `-f, --force` | Optional | Forces the deletion without prompting for confirmation. |

### Short Form Example

This example shows wallet `hr_wallet` being deleted without confirmation.

```
java -jar okvrestservices.jar -c conf_file -r delete_wallet -f -w hr_wallet
```

### Long Form Example

```
java -jar okvrestservices.jar --config conf_file --service delete_wallet --force
--wallet_name hr_wallet
```

## 10.9.7 Error Reporting

The RESTful Service utility has robust error reporting, which you can use to debug in order to run RESTful Service commands quickly and successfully. The status of command execution, passed and failed, is reported promptly on the command line and written to the log file.

- About Error Reporting (page 10-34)
- Command Line Error Reporting (page 10-34)
- Error Reporting while Running Commands from a Script (page 10-34)
- See Valid Options Using -h or --help (page 10-35)
- See List of RESTful Commands Using -H or --list (page 10-35)

### 10.9.7.1 About Error Reporting

The status of command execution, passed and failed, is reported promptly on the command line and written to the log file. The specific error will be reported, with corrective actions where appropriate.

The first thing to do when a command fails is to look into the log file. If you have not created a custom log file in a location of your choice, then you can look at the default log file, `okvrestservices.log` in the current directory, where command results will be written.

To see all the messages from the Oracle Key Vault server during command execution, you can set the appropriate logging level, log file name, and the log file location in the configuration file.

The RESTful service utility reports errors such as the failure to locate a file or an environment variable like `JAVA_HOME`, incorrect command syntax, and incorrect passwords.

### 10.9.7.2 Command Line Error Reporting

Error reporting captures both faulty actions, such as incorrect passwords, and successful command executions.

**Example 10-6 Error: Running a Service Command without the -r Option**

```
java -jar okvrestservices.jar -c rest.ini modify_endpoint_desc -e ORDERS -b
ORDERS_HR
Script or service option is required.
```

**Example 10-7 Error: Incorrect Password**

```
java -jar okvrestservices.jar -c rest.ini -r modify_endpoint_desc -e ORDERS -b
ORDERS_HR
Password:
Invalid username or password. Try again after 5 seconds
```

**Example 10-8 Successful Service Command Execution**

```
java -jar okvrestservices.jar -c rest.ini -r modify_endpoint_desc -e ORDERS -b
ORDERS_HR
Password:
[Line 0 OK] [MODIFY ENDPOINT DESC] [ORDERS:ORDERS_HR]
```

**Example 10-9 Log File Entry**

In addition to the helpful error and usage messages, an entry for the action is logged in the log file with the date.

```
Mar 02, 2019 7:23:55 PM com.oracle.okv.cloud.client.OKVAutomation checkpoint
INFO: [Line 0 OK] [MODIFY ENDPOINT DESC] [ORDERS:ORDERS_HR]
```

### 10.9.7.3 Error Reporting while Running Commands from a Script

When you run multiple service commands from a script you will see the result on the command line as well as in the log file.

The following output shows the successful results of commands executed from a script.

**Example 10-10    Results of Script Execution**

```
java -jar okvrestservices.jar --config rest.ini --script initial_setup.api
Password:
[Line 1 OK] [CREATE ENDPOINT] [APP_SERVER_1:ORACLE_NON_DB:LINUX64]
[Line 2 OK] [CREATE ENDPOINT] [APP_SERVER_2:ORACLE_NON_DB:LINUX64]
[Line 11 OK] [CREATE WALLET] [ApplicationWallet]
[Line 12 OK] [CREATE WALLET] [FinanceWallet]
[Line 15 OK] [CREATE ENDPOINT GROUP] [APP_SERVER]
[Line 16 OK] [CREATE ENDPOINT GROUP] [FINANCE_RAC]
[Line 20 OK] [ADD EPG MEMBER] [APP_SERVER:APP_SERVER_2]
[Line 22 OK] [ADD EPG MEMBER] [FINANCE_RAC:FINANCE_RAC_NODE_1]
[Line 29 OK] [ADD WALLET ACCESS EPG] [APP_SERVER:ApplicationWallet:RM]
[Line 30 OK] [ADD WALLET ACCESS EPG] [FINANCE_RAC:FinanceWallet:RO]
[Line 31 OK] [ADD WALLET ACCESS EP] [HR_DATABASE_PRIMARY:HRWallet:RM_MW]
```

## 10.9.7.4 See Valid Options Using -h or --help

For a list of valid options you can use the -h or --help option with the RESTful Services utility okvrestservices.jar.

**Using the --help Option**

```
-bash-4.1$ java -jar okvrestservices.jar -help
usage: java -jar okvrestservices.jar --config <arg> [--service <arg> |--script
<arg>
-A,--add <arg>       User to add to wallet
-c,--config <arg>    System configuration file for OKV REST Services Utility
-D,--delete <arg>    User to delete from wallet
-f,--force           Confirm to delete
-h,--help            Display all available options
-L,--listuser        List all user from wallet
-M,--modify <arg>    User to modify from wallet
-p,--pwd <arg>       OKV user password
-t,--twallet <arg>   Wallet location
-u,--usr <arg>       OKV username
-x,--script <arg>    Script file
-r,--service <arg>   Service name
-z,--list            Display all service commands
```

## 10.9.7.5 See List of RESTful Commands Using -H or --list

To see the list of RESTful service commands type -H or --list at the command line.

# 11
# Oracle Key Vault Use Case Scenarios

Typical Oracle Key Vault use cases include the upload and download of security objects, the use of an Online Master Key, and TDE-configured Oracle databases. Note that the term Online Master Key replaces TDE direct connect.

## 11.1 Uploading and Downloading Oracle Wallets

In order to store and share Oracle wallets you must upload them to Key Vault. You can then create a new virtual wallet in Key Vault, and add security objects to it that you want to share. You must grant endpoints access to the virtual wallet before they can download it.

### 11.1.1 About Uploading and Downloading Oracle Wallets

The `okvutil` utility comes packaged with the endpoint software that you install at the endpoint. You can use the `okvutil upload` and `okvutil download` commands to upload and download Oracle wallets between Oracle Key Vault and its endpoints.

The Oracle Key Vault endpoint software can read an Oracle wallet at the granularity level of an individual security object. It therefore uploads the wallet contents as individual items. During download you can recreate the original wallet with the same set of security objects, or create a new wallet with different set of security objects.

You can upload and download both password-based wallets and auto-login wallets. The wallet contents can be downloaded later into a new wallet of either type. For example, an uploaded password-protected wallet can be downloaded as an auto-login wallet, or an uploaded auto-login wallet can be downloaded as a password-protected wallet.

You can use Oracle Key Vault to construct a new virtual wallet containing security objects from previously uploaded Oracle wallets. For example, given a previously uploaded Oracle wallet containing five symmetric keys and three opaque objects, you can create a new virtual wallet consisting of only three of the original five symmetric keys and one of the three original opaque objects. This virtual wallet can be downloaded like the original wallet to provide the endpoint with access to only a subset of the keys. This process does not modify the original wallet.

> **See Also:**
>
> "Oracle Key Vault okvutil Endpoint Utility Reference (page 8-12)"

## 11.1.2 Uploading Oracle Wallets

Uploading wallets to Key Vault is done with the `okvutil upload` command. Everything in the Oracle wallet, security objects and their metadata is uploaded to Key Vault, so that the wallet can be reconstructed during the download process. The Oracle wallet typically contains TDE master keys, historical TDE master keys, SSL or TLS certificates and their metadata (stored in Key Vault as opaque objects), wallet metadata, as well as keys that you have explicitly added.

To upload an Oracle wallet:

1. Ensure that the server containing the Oracle wallet has been enrolled and provisioned as a Key Vault endpoint.
2. Ensure that the endpoint has access to the virtual wallet that you want to use.

   The endpoint must have Read, Modify, and Manage Wallet access to the virtual wallet in Oracle Key Vault.
3. Run the `okvutil upload` command to upload the wallet.

   For example:

   ```
   okvutil upload -l "/etc/oracle/wallets" -t wallet -g "HRWallet"
   Enter wallet password (<enter> for auto-login): password
   Enter Oracle Key Vault endpoint password: Key_Vault_endpoint_password
   Upload succeeeded
   ```

   In this example:

   - `-l` specifies the directory location of the wallet that you are uploading.
   - `-t` indicates the type, in this case, an Oracle wallet.
   - `-g` specifies the Key Vault virtual wallet that was configured in Step 2 (page 11-2), so that this wallet can be part of that virtual wallet.

- Password prompts for the password-protected Oracle wallet, and the endpoint password.

At this point, the upload is complete. You can now share the virtual wallet with other users and endpoints.

**See Also:**

- "Managing Endpoints (page 7-2)" for more information
- "Grant Access to Endpoint Groups, Endpoints, User Groups, and Users (page 6-5)" for more information
- "How Password Prompts for okvutil Work (page 8-13)"
- "okvutil upload Command (page 8-14)" for more information about `okvutil upload`

## 11.1.3 Downloading Oracle Wallets

You can use the okvutil download command to download an Oracle wallet from the Oracle Key Vault server to an endpoint.

To download an Oracle wallet:

1. Ensure that the endpoint has Read access on the virtual wallet that you want to download.
2. Run the `okvutil download` command to download the wallet.

   For example:

   ```
   okvutil download -l "/etc/oracle/wallets/orcl/" -t WALLET -g HRWallet
   Enter new wallet password(<enter> for auto-login): Oracle_wallet_password
   Confirm new wallet password: Oracle_wallet_password
   Enter Oracle Key Vault endpoint password: Key_Vault_endpoint_password
   ```

   In this example:

   - `-l` is the location of the wallet to be created.
   - `-t` indicates the type, in this case, an Oracle wallet.
   - `-g` specifies the Oracle Key Vault virtual wallet that was configured in Step 1 (page 11-3).
   - Password prompts for the password-protected Oracle wallet, and the endpoint password.

   If the wallet already exists and you did not use the `-o` parameter to overwrite the existing wallet, then the following actions take place:

   - The existing wallet is renamed to a backup name of the format `ewallet.p12.`*`current_timestamp`* where the *`timestamp`* is number of seconds since epoch.
   - The newly downloaded wallet is given the name `ewallet.p12`.

3. If the Oracle wallet that you downloaded from Oracle Key Vault is to be used as a TDE wallet, then close the existing wallet before downloading and, if it is password-protected, then reopen it afterward. (Auto-login wallets are automatically opened the next time that they are accessed.)

   Closing and then reopening the wallet loads the wallet contents into the TDE database.

   - For Oracle Database 11*g* Release 2:

     ```
     ALTER SYSTEM SET ENCRYPTION WALLET CLOSE IDENTIFIED BY
     "Oracle_wallet_password";

     ALTER SYSTEM SET ENCRYPTION WALLET OPEN IDENTIFIED BY
     "Oracle_wallet_password";
     ```

   - For Oracle Database 12*c*:

     ```
     ADMINISTER KEY MANAGEMENT SET KEYSTORE CLOSE IDENTIFIED BY
     "Oracle_wallet_password";

     ADMINISTER KEY MANAGEMENT SET KEYSTORE OPEN IDENTIFIED BY
     "Oracle_wallet_password";
     ```

4. If you are operating in a shared server configuration such as Oracle RAC, then restart the database.

**See Also:**

- "How Password Prompts for okvutil Work (page 8-13)"
- "okvutil download Command (page 8-19)" for more information about `okvutil download`
- "Grant Access to Endpoint Groups, Endpoints, User Groups, and Users (page 6-5)"
- *Oracle Database Advanced Security Guide* for information about closing and opening keystores in Oracle Database Release 12*c*

## 11.1.4 Recommendations for Uploading and Downloading Oracle Wallets

Oracle provides recommendations for when you upload and download Oracle wallets.

- If there is a change to the content of the original wallet, such as a key rotation or a rekey operation, then upload the wallet again to Oracle Key Vault so that Key Vault has the latest copy of the wallet.
- The `okvutil upload` and `download` commands provide an overwrite (`-o`) option. Use care if you plan to specify this option because it overwrites data in the virtual wallet that conflicts with the data to be uploaded. Before you use the `-o` option, you should create a local backup of the wallet file.
- Do not try to upload the same physical Oracle wallet to more than one virtual wallet on the Oracle Key Vault server. If you want to share an Oracle wallet with multiple endpoints, then create an endpoint group.

**See Also:**

"Manage Endpoint Groups (page 7-16)"

# 11.2 Uploading and Downloading JKS and JCEKS Keystores

You can use the `okvutil upload` and `okvutil download` commands to upload and download JKS and JCEKS keystores.

- About Uploading and Downloading JKS and JCEKS Keystores (page 11-5)
- Uploading JKS or JCEKS Keystores (page 11-5)
- Downloading JKS or JCEKS Keystores (page 11-6)
- Recommendations for Uploading and Downloading JKS and JCEKS Keystores (page 11-7)

## 11.2.1 About Uploading and Downloading JKS and JCEKS Keystores

You can upload both JKS and JCEKS keystores to Oracle Key Vault for long-term retention, recovery, and sharing, and when you need them, download them to an endpoint.

Similar to wallets, when you upload a JKS or JCEKS keystore, Oracle Key Vault can read each item within the keystore. It uploads the keystore contents as individual items.

## 11.2.2 Uploading JKS or JCEKS Keystores

You can use the okvutil upload command to upload a Java keystore (JKS) or Java Cryptography Extension keystore (JCEKS) to the Oracle Key Vault server.

To upload a Java keystore:

1. Ensure that the server containing the Java keystore has been enrolled and provisioned as a Key Vault endpoint.
2. Ensure that access control has been configured for the endpoint.

   If you are uploading the keystore to a virtual wallet, then ensure that the endpoint has the Read, Modify, and Manage Wallet access to this wallet.
3. Run the `okvutil upload` command to upload the keystore.

   The following examples show how to upload the keystore to a virtual wallet:

   This example shows how to upload a JKS keystore:

   ```
   okvutil upload -l "/etc/oracle/fin_jks.jks" -t JKS -g "FinanceGrp"
   Enter source Java keystore password: Java_keystore_password
   Enter Oracle Key Vault endpoint password: Key_Vault_endpoint_password
   Upload succeeded
   ```

In this example:

- `-l` is the location of the Java keystore that is being uploaded.
- `-t` is the type of JKS or JCEKS keystore. Ensure that you upload the correct type of Java keystore when you upload and later on, when you download.
- `-g` is the virtual wallet in Oracle Key Vault where the Java keystore contents will be uploaded.
- Password prompts for the keystore and endpoint.

This example shows how to upload a JCEKS keystore:

```
okvutil upload -l "/etc/oracle/hr_jceks.jceks" -t JCEKS -g "HRGrp"
Enter source Java keystore password: password
Enter Oracle Key Vault endpoint password: password
Upload succeeded
```

At this point, the upload is complete. You are now ready to share or download the Java keystore as needed.

**See Also:**

- "How Password Prompts for okvutil Work (page 8-13)"
- "okvutil download Command (page 8-19)" for more information about `okvutil download`
- "Grant Access to Endpoint Groups, Endpoints, User Groups, and Users (page 6-5)"
- *Oracle Database Advanced Security Guide* for information about closing and opening keystores in Oracle Database Release 12*c*

## 11.2.3 Downloading JKS or JCEKS Keystores

You can use the `okvutil download` command to download an uploaded JKS or JCEKS keystore.

To download a Java keystore:

1. Ensure that the endpoint has the Read access on the virtual wallet that you want to download.
2. As an endpoint administrator, from the command line, run the `okvutil download` command to download the Java keystore.

   For example:

   ```
   okvutil download -l "/etc/oracle/new_java_files/hr_jceks.jceks" -t JCEKS
   Enter new Java keystore password: password
   Confirm new Java keystore password: password
   Enter Oracle Key Vault endpoint password: Key_Vault_endpoint_password
   ```

   In this example:

   - `-l` is the directory to which you want to download the uploaded Java keystore.

- `-t` is the type of JKS or JCEKS keystore. Ensure that you download the correct type of Java keystore.
- Password prompts for keystore and endpoint password.

> **See Also:**
>
> - "Managing Endpoints (page 7-2)" for more information
> - "How Password Prompts for okvutil Work (page 8-13)"
> - "okvutil download Command (page 8-19)" for more information about `okvutil download`
> - "Grant Access to Endpoint Groups, Endpoints, User Groups, and Users (page 6-5)"

## 11.2.4 Recommendations for Uploading and Downloading JKS and JCEKS Keystores

Oracle provides recommendations for when you upload and download JKS and JCEKS keystores.

- If there is a change to the content of the original JKS or JCEKS keystore, then upload the keystore again to Oracle Key Vault so that Key Vault has the latest copy of the keystore.
- The `okvutil upload` and `download` commands provide an overwrite (`-o`) option. Use care if you plan to specify this option because it overwrites the file. You may want to create backups of the keystores before downloading them.
- Do not try to upload the same physical JKS or JCEKS keystore to more than one virtual wallet on the Oracle Key Vault server. If you want to share a Java keystore with multiple endpoints, then create an endpoint group.

> **See Also:**
>
> "Manage Endpoint Groups (page 7-16)" for more instructions to create an endpoint group

# 11.3 Uploading and Downloading Credential Files

You can use the `okvutil upload` and `okvutil download` commands to upload and download credential files.

- About Uploading and Downloading Credential Files (page 11-8)
- Uploading a Credential File (page 11-8)
- Downloading a Credential File (page 11-9)
- Recommendations for Uploading and Downloading Credential Files (page 11-10)

## 11.3.1 About Uploading and Downloading Credential Files

Credential files are uploaded and stored as opaque objects in Key Vault, which means that Key Vault does not parse the contents of the file like an Oracle wallet or Java keystore. The upload process does not alter the credential file.

Examples of opaque objects are as follows:

- Files that contain X.509 certificates
- Kerberos keytabs
- Files containing passwords
- Files containing SSH keys

Uploading these credential files provides a central, secure location for long-term retention. After you have uploaded a credential file, you can download it in the same server location or share it with other trusted server locations. Oracle Key Vault supports credential files up to 128 KB in size.

The credential file can be located anywhere in your server infrastructure (which includes database servers and application servers) that is accessible by an Oracle Key Vault endpoint.

## 11.3.2 Uploading a Credential File

To upload a credential file with `okvutil upload`:

1. Ensure that the server that contains the credential file has been enrolled and provisioned as a Key Vault endpoint.
2. Ensure that access control has been configured for the endpoint.

   If you are uploading the credential file to a virtual wallet, then ensure that the endpoint has Read, Modify, and Manage Wallet access to the wallet.
3. Run the `okvutil upload` command.

   For example:

   ```
   okvutil upload -l "/etc/oracle/app/creds/hr.keytab" -t kerberos -g HRWallet
   -d "Kerberos keytab file for HR group, 06_11_14"
   Enter Oracle Key Vault endpoint password: Key_Vault_endpoint_password
   ```

   In this example:

   - `-l` is the directory path to the `hr.keytab` credential file that is being uploaded. Enclose the directory location in double quotation marks.
   - `-t` specifies the type of credential file, which in this example is a Kerberos keytab file. In addition to `KERBEROS`, other types that you can specify are as follows:
     - `SSH` for an SSH key file
     - `OTHER` for other files that store secrets, such as uploaded or downloaded files
   - `-g` adds the credential file to the `HRWallet` group, which already exists. This parameter enables you to upload the credential to a wallet that is specifically for the HR application users' needs, rather than to the default virtual wallet. In

this example, HRWallet is the Key Vault virtual wallet to which access control was configured in Step 2 (page 11-8).

- -d is an optional description. As a best practice, include a brief description of what the credential file is used for and the date you performed the upload. This information helps for future reference and tracking of the credential file. You can modify this description later on in the Oracle Key Vault management console if necessary.
- Password prompts: Enter the endpoint password and confirm it.

The following output should appear:

```
Upload succeeded
```

At this point, the upload is complete.

**See Also:**

- "Managing Endpoints (page 7-2)"
- "Grant Access to Endpoint Groups, Endpoints, User Groups, and Users (page 6-5)"
- "okvutil upload Command (page 8-14)"

## 11.3.3 Downloading a Credential File

To download a credential file using okvutil download:

1. Find the unique ID of the credential file that you must download, by using one of the following methods:
   - **Oracle Key Vault management console:** Log in as a user who has been granted the Key Administrator role or a user who has the necessary access to the virtual wallet. In the Oracle Key Vault management console, from the **Keys & Wallets** tab, select **All Items** to find the uploaded files. Note the unique ID of the uploaded file that you want to download. Credential files are listed as opaque objects.
   - **okvutil list command:** Run the okvutil list command from an endpoint that has access to the credential file or a virtual wallet that contains the credential file. Locate the unique ID of the credential file that you must download based on the description that you provided when you uploaded the file.
2. From the command line, run the okvutil download command to download the uploaded credential file.

   For example:

   ```
   okvutil download -l "/etc/oracle/app/newcreds/hr.keytab" -t kerberos -i
   6ba7b810-9dad-11d1-80b4-00c04fd430c8
   Enter Oracle Key Vault endpoint password: Key_Vault_endpoint_password
   ```

   In this example:

   - -l is the directory to which you want to download the uploaded credential.

- `-t` specifies the type of credential file, which in this example is a Kerberos keytab file. In addition to `KERBEROS`, other types that you can specify are as follows:
  - `SSH` for an SSH key file
  - `OTHER` for other files that store secrets, such as uploaded or downloaded files
- `-i` is the unique ID of the credential file.

Output similar to the following appears:

```
Download succeeded
```

> **See Also:**
>
> "okvutil download Command (page 8-19)"

## 11.3.4 Recommendations for Uploading and Downloading Credential Files

Oracle provides recommendations for when you upload and download credential files.

- After you complete the upload, upload credential file again the next time it is changed. Otherwise, the uploaded (and subsequent downloaded version) file will be outdated. Periodically, you should compare the last modification date of the credential file with the timestamp of the uploaded version.
- The `okvutil upload` and `download` commands provide an overwrite (`-o`) option. Use care if you plan to specify this option, because it overwrites the uploaded credential file. You may want to create backups of the credential files before beginning the upload and download processes.
- You can share one credential file among multiple server endpoints. Add the opaque object to a virtual wallet and then ensure that all of the endpoints have access to that virtual wallet. Optionally, define an endpoint group and then make all the server endpoints members of that group. Upload the credential file that you would like to share using this common wallet into Oracle Key Vault as a group, using the `-g` option of the `okvutil upload` command. Define a wallet and attach it to the endpoint group. All the members of the group will have access to that wallet.

# 11.4 Using an Online Master Key with Oracle Key Vault

You can configure Transparent Data Encryption (TDE) to perform a direct connection to an endpoint to centrally manage TDE master keys.

Note that the term Online Master Key replaces TDE direct connect.

- About Using an Online Master Key with Oracle Key Vault (page 11-11)
- Other Oracle Database Features That Oracle Key Vault Supports (page 11-11)
- Configuring a Connection Between Oracle Key Vault and a New TDE-Enabled Database (page 11-12)

- Migrating Existing TDE Wallets to Oracle Key Vault (page 11-15)
- Persistent Master Key Cache (page 11-19)
- Minimizing Downtime (page 11-24)

## 11.4.1 About Using an Online Master Key with Oracle Key Vault

For Oracle Database 11*g* Release 2 (11.2) and later, you can use an Online Master Key to centrally manage Transparent Data Encryption (TDE) master keys over a network connection as an alternative to using local Oracle wallet files.

The connection configuration entails using a PKCS#11 library to connect to Oracle Key Vault. After you perform the configuration, all future TDE master keys will be stored and managed in Oracle Key Vault. This section explains two scenarios that you can use:

- If the database does not yet have TDE wallets.
- If the database has already been configured for TDE.

TDE key management operates with Oracle Key Vault in the same way that it operates with hardware security modules (HSMs). You must open the wallet before encryption and decryption. After you close the wallet, encrypted data in tables and tablespaces is unavailable to you. You should rotate the TDE master encryption key regularly to remain in compliance with the applicable regulations.

Oracle Key Vault supports the SQL statements that were used to administer earlier TDE releases, specifically the use of the `ALTER SYSTEM` and `ADMINISTER KEY MANAGEMENT` SQL statements.

**See Also:**

- "Configuring a Connection Between Oracle Key Vault and a New TDE-Enabled Database (page 11-12)" for a non-TDE database
- "Migrating Existing TDE Wallets to Oracle Key Vault (page 11-15)" for a TDE database
- "Centralized Management of TDE Master Keys Using Online Master Key (page 2-4)"

## 11.4.2 Other Oracle Database Features That Oracle Key Vault Supports

You can deploy TDE in multiple topologies with other database features that move or use clustered deployments.

Data movement and replication are major challenges for Oracle Advanced Security TDE because it must keep the master encryption key synchronized at both endpoints. To help with these challenges, Oracle Key Vault supports common Oracle Database features.

To move data, Oracle Key Vault supports:

- Oracle Recovery Manager (RMAN) backup and recovery operations
- Oracle Data Pump
- Transportable tablespaces (Oracle Database 12*c* and later)

For clustered deployments, Oracle Key Vault supports:

- Oracle Active Data Guard
- Oracle Real Application Clusters (Oracle RAC)
- Oracle GoldenGate

## 11.4.3 Configuring a Connection Between Oracle Key Vault and a New TDE-Enabled Database

You can configure a connection between Oracle Key Vault and a database that has not yet been configured for Transparent Data Encryption.

Before you start configuring the connection, ensure that the Oracle Key Vault installation environment is the same as the Database runtime environment. The environment variables `ORACLE_HOME`, `ORACLE_BASE`, and `ORACLE_SID`, if used, must be set to the same values in `svrctl` and OS environments. This also applies if you are using the Oracle Key Vault RESTful Services utility to enroll endpoints.

Configure the connection as follows:

1. Ensure that the `ORACLE_BASE` environment variable is set before you start the `oracle` process manually. This is very important.

   If the `ORACLE_BASE` environment variable is not present, create a soft link from the:

   `$ORACLE_BASE/okv/$ORACLE_SID/okvclient.ora` file

   to the:

   *`key_vault_endpoint_installation_dir`*`/conf/okvclient.ora` file.

   In an Oracle Real Application Clusters environment, perform this step on all database instances.

   > **Note:**
   >
   > To learn more about using environment variables in `SQLNET.ORA` please see the **See Also** section at the end of this procedure.

2. Ensure that the `COMPATIBILITY` initialization parameter is set to `11.2.0.0` or later.
3. Enroll and provision the endpoint for the TDE-enabled database that contains the TDE data.

   When you initially enroll the endpoint, select Oracle Database as the endpoint type for integration with TDE.
4. Ensure that the endpoint has access to the virtual wallet that you want to use.

   The endpoint must have the Read, Modify, and Manage Wallet access.

5. Configure the `sqlnet.ora` file on this database to point to Oracle Key Vault as follows:

   ```
   ENCRYPTION_WALLET_LOCATION=(SOURCE=(METHOD=HSM))
   ```

   Transparent Data Encryption uses `HSM` as the parameter value for all external key management systems, including Oracle Key Vault. Therefore the configuration settings and administrative commands are similar to those used for an HSM.

   By default, the `sqlnet.ora` file is located in the *`ORACLE_HOME`*`/network/admin` directory or in the location set by the `TNS_ADMIN` environment variable. Endpoints use the PKCS#11 library support to manage TDE master encryption keys.

   > **Note:**
   >
   > At this stage, Oracle Key Vault can use TDE and all the TDE-related SQL statements are available. For all TDE commands and statements, use the Key Vault endpoint password that was specified during the endpoint enrollment process.

6. Reconnect to the database if you are in SQL*Plus.

   The changes will appear after you log out of the current SQL*Plus session and then reconnect again.

7. Query the `V$ENCRYPTION_WALLET` dynamic view to ensure that the `METHOD_DATA` setting in the `sqlnet.ora file` changed.

   The output of the query should now show `HSM`.

   ```
   SELECT * FROM V$ENCRYPTION_WALLET;
   ```

8. Configure TDE to integrate with Key Vault, so that Key Vault can directly manage the TDE master keys as follows:

   a. On UNIX platforms, run the `root.sh` script as the root user to copy the `liborapkcs.so` file (located in the `lib` directory) to the `/opt/oracle/extapi/64/hsm/oracle/1.0.0` directory.

      The persistent master key cache feature is implemented in the Oracle Key Vault PKCS#11 library and improves the availability of the database during intermittent network disruptions or Oracle Key Vault upgrade.

      By default, the PKCS#11 library is located in the `$OKV_HOME/bin/liborapkcs.so` file. Copy it to the following location on a UNIX system:

      ```
      /opt/oracle/extapi/64/hsm/oracle/1.0.0
      ```

      On Windows platforms, run the `root.bat` script as the root user to copy the `liborapkcs.dll` file (located in the `lib` directory) to the `C:\oracle\extapi\64\hsm\oracle\1.0.0` directory. Provide the database version when prompted.

   b. For password-protected wallets on the database, open the wallet. (Auto-login wallets are automatically opened.)

      For Oracle Database 11*g* Release 2, as a user who has been granted the `ALTER SYSTEM` system privilege:

```
ALTER SYSTEM SET ENCRYPTION WALLET OPEN IDENTIFIED BY
"Key_Vault_endpoint_password";
```

For Oracle Database 12*c*, as a user who has been granted the SYSKM administrative privilege:

```
ADMINISTER KEY MANAGEMENT SET KEYSTORE OPEN IDENTIFIED BY
"Key_Vault_endpoint_password";
```

c. Set the master encryption key.

For Oracle Database 11*g* Release 2:

```
ALTER SYSTEM SET ENCRYPTION KEY IDENTIFIED BY
"Key_Vault_endpoint_password";
```

For Oracle Database 12*c*:

```
ADMINISTER KEY MANAGEMENT SET ENCRYPTION KEY IDENTIFIED BY
"Key_Vault_endpoint_password";
```

The TDE configuration is complete at this stage. You can now encrypt tables or create encrypted tablespaces in the database.

If you have configured the `sqlnet.ora` file correctly along with the rest of the TDE configuration, a TDE master key is created in Oracle Key Vault when you set the encryption key using one of the following commands:

- `ALTER SYSTEM SET ENCRYPTION KEY`
- OR
- `ADMINISTER KEY MANAGEMENT SET ENCRYPTION KEY`

### Limitations to TDE Endpoint Integration

The limitations to TDE endpoint integration are as follows:

- All endpoints on the same machine must use the same version of the Oracle Key Vault library. There is only one location per machine for the `liborapkcs.so` file, which is:

  ```
  /opt/oracle/expapi/64/hsm/oracle/1.0.0/liborapkcs.so
  ```

- On the same machine it is best to use the same external key manager for Oracle Database, either Oracle Key Vault or HSM. Using Oracle Key Vault for one Oracle Database and HSM for another Oracle Database can cause the wrong PKCS#11 library to be loaded, because Oracle Database picks up the first PKCS#11 library while traversing the subtree:

  ```
   /opt/oracle/expapi/64/hsm/
  ```

**Caution:**

Oracle strongly recommends NEVER to remove keys from a wallet or the wallet itself after TDE is setup. Loss of keys will result in the loss of encrypted data and hamper the normal functioning of the database. This is true even:

- If there is no encrypted data in the system
- If all of the encrypted data has been decrypted
- If you have migrated your keys and wallets to a hardware security module

**See Also:**

- "Environment Variables in SQLNET.ORA (page 8-10)"
- "Endpoint Database Requirements (page 3-3)" for more information on setting the `COMPATIBLE` parameter
- "Managing Endpoints (page 7-2)"
- "Grant Access to Endpoint Groups, Endpoints, User Groups, and Users (page 6-5)" for information on granting access to virtual wallets

## 11.4.4 Migrating Existing TDE Wallets to Oracle Key Vault

You can migrate an existing TDE wallet to Oracle Key Vault, and if necessary, restore the database contents that were previously encrypted by TDE, by using an Oracle wallet.

- About Migrating Existing TDE Wallets to Oracle Key Vault (page 11-15)
- Migrating an Existing TDE Wallet to Oracle Key Vault (page 11-16)
- Restoring Database Contents Previously Encrypted by TDE Using an Oracle Wallet (page 11-18)

### 11.4.4.1 About Migrating Existing TDE Wallets to Oracle Key Vault

When the TDE wallets already exist, you must modify the `sqlnet.ora` file to recognize Oracle Key Vault before you can migrate the existing TDE wallets to Key Vault.

Along with the current TDE master key, Oracle wallets maintain historical TDE master keys that are replaced by each rekey operation that rotates the TDE master key. These historical TDE master keys help to restore Oracle Database backups that were previously made using one of the historical TDE master keys. During the TDE migration from an Oracle wallet file to Oracle Key Vault, Key Vault generates new master keys. After this master key generation, Oracle Key Vault maintains all new keys.

Oracle recommends that you upload the Oracle wallet to Key Vault, before you perform the migration. This enables you to keep a backup of the wallet with all of the historical key information, before you begin the migration. When the migration is complete, manually delete the old wallet on the client system.

If you are operating in a shared server or an Oracle RAC configuration, then you must restart the database so that the new TDE master key is updated to all the endpoint database nodes in the shared server configuration.

> **See Also:**
>
> "Restoring Database Contents Previously Encrypted by TDE Using an Oracle Wallet (page 11-18)"

### 11.4.4.2 Migrating an Existing TDE Wallet to Oracle Key Vault

You can use the `okvutil upload` command to migrate an existing TDE wallet to Oracle Key Vault. It is very important that you close the software wallet and open the HSM wallet before migrating the wallet in the same SQLPLUS session, as outlined in steps 7 and 8 below:

1. Back up the database that contains the data that you want to migrate.
2. Complete the enrollment of the endpoint.
3. If you have not done so already, then upload the existing Oracle wallet to Key Vault, by using the `okvutil upload` command.

   This step ensures that Oracle Key Vault has a copy of the wallet that contains all of the historical TDE master keys.
4. Configure the Oracle Database `sqlnet.ora` file for the HSM as follows:

   ```
   ENCRYPTION_WALLET_LOCATION=(SOURCE=(METHOD=HSM)
   (METHOD_DATA=(DIRECTORY=wallet_location)))
   ```

   By default, the `sqlnet.ora` file is located in the *`ORACLE_HOME`*`/network/admin` directory or in the location set by the `TNS_ADMIN` environment variable.
5. Reconnect to the database if you are in SQL*Plus.

   The changes do not appear until you restart the database session.
6. Query the `V$ENCRYPTION_WALLET` dynamic view to ensure that the `METHOD_DATA` setting in the `sqlnet.ora file` changed. The output of the query should now show `METHOD=HSM`.

   ```
   SELECT * FROM V$ENCRYPTION_WALLET;
   ```
7. If the endpoint is a Release 11*g*R2 Oracle database, then close the local Oracle wallet and open the HSM wallet as follows:
   1. Close the local Oracle wallet using these steps:
      1. If the auto-login wallet is open, execute the following commands:

```
oracle$ cd <wallet location>
oracle$ mv cwallet.sso cwallet.sso.bak
sqlplus> alter system set wallet close;
```

ii. If the password-protected wallet is open, execute the following command:

```
sqlplus> alter system set wallet close identified by "<wallet
password>";
```

b. Open the HSM wallet:

```
sqlplus> alter system set wallet open identified by "<HSM connect
string>";
```

8. Migrate from TDE wallets to Oracle Key Vault.

   - For Oracle Database 11*g* Release 2:

     If you entered a password for the wallet while installing the endpoint client software, then execute this command:

     ```
     sqlplus> alter system set encryption key identified by "<endpoint
     password>" migrate using "<wallet password>";
     ```

     If you chose the auto-login option while installing the endpoint client software, then execute this command:

     ```
     sqlplus> alter system set encryption key identified by "null" migrate
     using  "<wallet password>";
     ```

   - For Oracle Database 12*c*, as a user who has been granted the `SYSKM` administrative privilege:

     ```
     sqlplus> administer key management set encryption key identified by
     "<endpoint password>" MIGRATE USING "<wallet password>" with backup;
     ```

     > **Note:**
     >
     > While the `with backup` clause is required for the `administer key management` command, it is ignored by TDE in Oracle Key Vault

9. Open the wallet. If the endpoint requires a password to connect to Oracle Key Vault, then enter the password.

   - For Oracle Database 11*g* Release 2:

     ```
     ALTER SYSTEM SET ENCRYPTION WALLET OPEN IDENTIFIED BY
     "Key_Vault_endpoint_password";
     ```

   - For Oracle Database 12*c*:

     ```
     ADMINISTER KEY MANAGEMENT SET KEYSTORE OPEN IDENTIFIED BY
     "Key_Vault_endpoint_password";
     ```

10. After you complete the migration, if you are using an auto-login wallet, then re-enable it by renaming the `cwallet.sso.bak` file to `cwallet.sso`.

**See Also:**

- *Oracle Database Backup and Recovery User's Guide* for more information about backing up a database
- "Overview of Endpoint Enrollment and Provisioning (page 8-1)"
- "okvutil upload Command (page 8-14)"

### 11.4.4.3 Restoring Database Contents Previously Encrypted by TDE Using an Oracle Wallet

When an Oracle database endpoint is converted from a local Oracle wallet file to using Oracle Key Vault, there may be a subsequent need to restore backups that were encrypted using a key from this local wallet file.

In this case, you must download the necessary key from Oracle Key Vault to a local wallet file to be used when you decrypt the backup during the restore process. For example, suppose that the `Finance_DB` database had recently migrated to use an Online Master Key to Oracle Key Vault after you have uploaded the premigration wallet. If a system failure forces you to restore from a database backup taken before the migration to Oracle Key Vault, you can still restore the contents of the database by using an Oracle wallet downloaded from the Oracle virtual wallet that contains the `Finance_DB` wallet data that you had uploaded earlier.

To restore previously TDE-encrypted database contents using an Oracle wallet:

1. Download this Oracle wallet from Oracle Key Vault by using the `okvutil download` command.
2. On the endpoint where you downloaded the Oracle wallet, add the following settings to the `sqlnet.ora` file.

   ```
   METHOD_DATA=wallet_file_path

   METHOD=FILE
   ENCRYPTION_WALLET_LOCATION=(SOURCE=(METHOD=FILE)
   (METHOD_DATA=(DIRECTORY=wallet_file_path)))
   ```

   Put the `ENCRYPTION_WALLET_LOCATION` setting on one line.
3. If you created a password-protected wallet in Step 1 (page 11-18), then open the wallet using the password you specified.

   - For Oracle Database 11*g* Release 2, as a user who has been granted the `ALTER SYSTEM` system privilege:

     ```
     ALTER SYSTEM SET ENCRYPTION WALLET OPEN IDENTIFIED BY "wallet_password";
     ```

   - For Oracle Database 12*c*, as a user who has been granted the `SYSKM` administrative privilege:

     ```
     ADMINISTER KEY MANAGEMENT SET KEYSTORE OPEN IDENTIFIED BY
     "wallet_password";
     ```

   Opening the wallet enables the server to read the contents of the updated `sqlnet.ora` file. At this point, the endpoint server has been restored to a state where it now can run with the original version of the wallet.

> **See Also:**
>
> "okvutil download Command (page 8-19)"

## 11.4.5 Persistent Master Key Cache

The persistent master key cache feature enables databases to be operational when the Oracle Key Vault server is unavailable for any reason. The TDE master key is cached in the persistent master key cache in addition to the in-memory cache, to make the master key available across database processes. It eliminates the need for databases to contact the Oracle Key Vault server for every new process, redo log switch, or database startup.

### 11.4.5.1 About the Persistent Master Key Cache

The persistent master key cache ensures the availability of TDE master encryption keys by reducing dependence on the state of the Oracle Key Vault server.

The TDE master encryption key is cached in the persistent master key cache in addition to the in-memory cache, to make the master key available across database processes. It eliminates the need for databases to contact the Oracle Key Vault server for every new process, redo log switch, or database startup operation.

The following are the benefits of ensuring availability of TDE master keys:

- Continuous operation of endpoints during upgrade, primary-standby configuration, switchover, failover, and other procedures that require an Oracle Key Vault restart
- Less load on the Oracle Key Vault server when multiple sessions of a single database request the same master encryption key
- Improved scalability of Oracle Key Vault

### 11.4.5.2 About Oracle Key Vault Persistent Master Key Cache Architecture

The Oracle Key Vault persistent master key cache is implemented in Oracle Key Vault's `PKCS#11` library. When the persistent master key cache feature is configured, the Oracle Key Vault `PKCS#11` library will create the persistent master key cache when the first master key is retrieved from Oracle Key Vault.

The persistent master key cache is an auto-login wallet or a password-based wallet, depending on how Oracle Key Vault is installed:

- If Oracle Key Vault is installed with a password specified, then the persistent master key cache is a password-based wallet.
- If Oracle Key Vault is installed without a password specified, then the persistent master key cache is an auto-login wallet.

The `PKCS#11` library also implements an in-memory master key cache. When the in-memory master key cache feature is configured, the master key is cached in the process memory of the process that loaded the library into memory. The in-memory and persistent master key caches are independent of each other. They can be enabled and disabled independently.

For operations that involve encryption and decryption, `PKCS#11` attempts to look up the master encryption key in the in-memory master key cache. If it is not found, `PKCS#11` then looks up the master encryption key in the persistent master key cache. If the master key is not found in the in-memory or the persistent master key cache, then it is retrieved from the Oracle Key Vault server, if the server is online.

## 11.4.5.3 Persistent Master Key Cache Modes of Operation

In Oracle Key Vault 12.2.0.5.0 and later, the persistent master key cache operates in two modes, Oracle Key Vault first mode and persistent master encryption key cache first mode.

The difference between the two modes is the order in which the persistent master key cache and Oracle Key Vault are looked up to retrieve the master key.

- Oracle Key Vault First Mode (page 11-20)
- Persistent Master Key Cache First Mode (page 11-20)

### 11.4.5.3.1 Oracle Key Vault First Mode

In Oracle Key Vault First mode, the endpoints attempt to retrieve the master encryption key from the Oracle Key Vault server. If the Oracle Key Vault server is offline, then the endpoints attempt to retrieve the master encryption key from the persistent master key cache.

The endpoints must determine the status of the Oracle Key Vault server, and if it is offline, the endpoints then attempt to retrieve the master encryption key from the persistent master key cache. Hence, database operations that require access to master keys will experience a delay.

### 11.4.5.3.2 Persistent Master Key Cache First Mode

In Persistent Master Key Cache First mode, the endpoints attempt to retrieve the master encryption key from the persistent master key cache. If the master key is not available in the persistent master key cache, then the endpoints attempt to retrieve the master encryption key from the Oracle Key Vault server.

The modifications to the master encryption keys on the Oracle Key Vault server are not applied until the key expires in the persistent master key cache.

## 11.4.5.4 Persistent Master Key Cache - Refresh Window

The Refresh Window feature of the Persistent Master Key Cache further minimizes endpoint downtime. This feature enables the database endpoint to make multiple attempts to refresh the expired master key from the OKV server. In that sense, the endpoint waits for the OKV server to be back online for the master key refresh to complete. Meanwhile, if the master key refresh attempt fails, the keys are retrieved from the persistent cache for the duration of the refresh window.

The Refresh Window feature of the Persistent Master Key Cache thus extends the duration for which the master key is available after it is cached in the persistent master key cache. At the same time the endpoints can refresh the key during the refresh window instead of once at the end of the cache time. This addresses the possibility that persistent cache expires in the window when the Oracle Key Vault is unavailable such as when HA switchover is in progress. The refresh window terminates and the cache period begins as soon as the key is refreshed.

In the `okvclient.ora` file, the parameter `PKCS11_PERSISTENT_CACHE_REFRESH_WINDOW` is used to extend the duration for which the master key is available after it is cached in the persistent master key cache. This value reflects the amount of time it takes for the Oracle Key Vault server to recover and come back online. The value is specified in minutes. The default value for `PKCS11_PERSISTENT_CACHE_REFRESH_WINDOW` is 30 (minutes).

For more information about the Persistent Cache Refresh Window, see PKCS11_PERSISTENT_CACHE_REFRESH_WINDOW Parameter (page 11-22).

## 11.4.5.5 Persistent Master Key Cache Parameters

The following are the parameters used to configure the Persistent Master Key Cache.

- PKCS11_CACHE_TIMEOUT Parameter (page 11-21)
- PKCS11_PERSISTENT_CACHE_TIMEOUT Parameter (page 11-21)
- PKCS11_PERSISTENT_CACHE_FIRST Parameter (page 11-22)
- PKCS11_PERSISTENT_CACHE_REFRESH_WINDOW Parameter (page 11-22)

### 11.4.5.5.1 PKCS11_CACHE_TIMEOUT Parameter

The `okvclient.ora` file, the parameter `PKCS11_CACHE_TIMEOUT` specifies the duration for which the master encryption key is available after it is cached in the in-memory cache. You must specify the value minutes. When the specified duration of time elapses, the master encryption key expires. Expired master keys are not deleted from the in-memory cache.

The default value for `PKCS11_CACHE_TIMEOUT` is 60 (minutes).

### 11.4.5.5.2 PKCS11_PERSISTENT_CACHE_TIMEOUT Parameter

In the `okvclient.ora` file, the parameter `PKCS11_PERSISTENT_CACHE_TIMEOUT` specifies the duration for which the master encryption key is available after it is cached in the persistent master key cache. You must specify this value in minutes. When the specified duration of time elapses, the master encryption key expires. Expired master keys are not deleted from the persistent master key cache.

The `Cache Start Time` and `Maximum Use Time` values displayed in the `OKV Persistent Cache entries` list is updated when the master encryption key is refreshed.

The default value for `PKCS11_PERSISTENT_CACHE_TIMEOUT` is 1440 (minutes).

You can disable the persistent master key cache by setting the `PKCS11_PERSISTENT_CACHE_TIMEOUT` parameter to 0 (zero).

> **Note:**
>
> The parameter `PKCS11_PERSISTENT_CACHE_TIMEOUT` and its default value are included by default in the `okvclient.ora` file.

#### 11.4.5.5.3 PKCS11_PERSISTENT_CACHE_FIRST Parameter

In the `okvclient.ora` file, the parameter `PKCS11_PERSISTENT_CACHE_FIRST` specifies the mode of operation of the persistent master key cache. The following are the modes of operation:

- **Oracle Key Vault First Mode:** To enable Oracle Key Vault First mode, set the value of the `PKCS11_PERSISTENT_CACHE_FIRST` parameter to 0 (zero).
- **Persistent Master Key Cache First Mode:** Persistent Master Key Cache First mode is the default mode.

  To enable Persistent Master Key Cache First mode, set the value of the `PKCS11_PERSISTENT_CACHE_FIRST` parameter to 1.

**Related Topics**

- Oracle Key Vault First Mode (page 11-20)
- Persistent Master Key Cache First Mode (page 11-20)

#### 11.4.5.5.4 PKCS11_PERSISTENT_CACHE_REFRESH_WINDOW Parameter

In the `okvclient.ora` file, the `PKCS11_PERSISTENT_CACHE_REFRESH_WINDOW` parameter extends the duration for which the master encryption key is available after it is cached in the persistent master key cache. You must specify the value in minutes. The default value for `PKCS11_PERSISTENT_CACHE_REFRESH_WINDOW` is 30 (minutes).

You can disable the persistent master key cache by setting the `PKCS11_PERSISTENT_CACHE_REFRESH_WINDOW` value to 0 (zero).

### 11.4.5.6 Listing the Contents of the Persistent Master Key Cache

After installing the endpoint software, endpoint administrators can use the command-line utility `okvutil` to communicate with Oracle Key Vault to view, upload, and download security objects.

The `okvutil list` command enables you to list the master encryption keys that are cached in the persistent master key cache.

The following example shows how to list the master encryption keys cached in the persistent master key cache:

```
$ ./okvutil list -t okv_persistent_cache -l $ORACLE_HOME/okv/$ORACLE_SID
Enter Oracle Key Vault endpoint password:

OKV Persistent Cache entries:
Current Persistent Cache Timeout is 600 seconds
Version Unique ID                            TDE Master Key
Identifier          Cache Start Time    Maximum Use Time  Maximum
Refresh Window  Status
02      55D745B1-2F30-667F-E053-0100007FAFDB
0636846AAF88F74FC6BF1DB68538797B69 22:38:12 2019-08-03 600
seconds       0 seconds                Expired
02      55D745B1-2F2E-667F-E053-0100007FAFDB
063AC48E9433734F7EBF97180276E719C4 22:37:10 2019-08-03 600
seconds       180 seconds              Available
02      55D745B1-2F2D-667F-E053-0100007FAFDB
0604425983989C4F6ABF7BD9E1D55459C4 22:37:00 2019-08-03 600
seconds       180 seconds              Available
02      55D70FA4-81D1-5C8A-E053-0100007F8217
06172EACB79F4C4F32BFB7D50B0ACA7101 03:44:22 2019-08-03 300
seconds       0 seconds                Expired
02      55D745B1-2F2B-667F-E053-0100007FAFDB
06983C4664FFC04F6ABF72F961A15AD943 22:36:49 2019-08-03 600
seconds       300 seconds              Available
02      55D745B1-2F29-667F-E053-0100007FAFDB
0639E05D58B27B4FFDBFAEC5EAA08DB301 03:26:40 2019-08-03 300
seconds       0 seconds                Expired
02      55D745B1-2F28-667F-E053-0100007FAFDB
06A29F4039E1B74FDCBFA687E0608EEEBA 03:19:17 2019-08-03 300
seconds       0 seconds                Expired
02      55D745B1-2F27-667F-E053-0100007FAFDB
0678287C2877B74FF3BF0BA33A17A59F94 03:19:21 2019-08-03 300
seconds       0 seconds                Expired
```

The following table lists the columns in the `OKV Persistent Cache entries` list:

| Column Name | Description |
|---|---|
| `Version` | Persistent master key cache version |
| `Unique ID` | KMIP identifier assigned to the master key |
| `TDE Master Key Identifier` | Database ID assigned to the master key |
| `Cache Start Time` | Time at which the master key was cached |
| `Maximum Use Time` | Duration for which the master key was/is cached |
| `Maximum Refresh Window` | Extended duration for which the master key is available after it is cached in the persistent master key cache |

| Column Name | Description |
|---|---|
| `Status` | Indicates whether the master key is available or expired |

### 11.4.5.7 Oracle Database Deployments and Persistent Master Key Cache

You should be aware of how the persistent master key cache affects the integration of other Oracle features with Oracle Key Vault.

- **Database restart when the Oracle Key Vault Server is offline:** When you configure Oracle Key Vault to use an auto-login wallet, the database connects to the Oracle Key Vault server when the database is restarted. If the Oracle Key Vault server is offline when the database restarts, then the database retrieves master encryption keys from the persistent master key cache. Database operations resume normally if the master encryption keys are active and have not expired. Ensure that the passwords of the persistent master key cache and the Oracle Key Vault endpoint wallet are synchronized.

  > **Note:**
  >
  > The persistent master key cache must be deleted when the endpoint wallet credentials are modified.

- **Using the persistent master key cache in an Oracle Real Application Cluster (Oracle RAC) environment:** In an Oracle RAC environment, each Oracle RAC node is an unique database endpoint, and uses an unique persistent master key cache.
  In an Oracle RAC Environment, you must query the database from each RAC node to cache the most recent version of the master encryption key in the persistent master key cache of each RAC node.
- **Using persistent master key cache in an Active Data Guard Environment:** Rotation of the master encryption key in the primary server's database caches the master encryption key in the persistent master key cache of the primary server's database.
  The standby server retrieves and caches the new master key in the persistent master key cache of the standby server's database after the new `REDO` logs from the primary server are applied on the standby server. To avoid disruptions, you should synchronize the primary and standby servers immediately after the rotation of the master encryption key in the primary server's database.

## 11.4.6 Minimizing Downtime

Business-critical operations require data to be accessible and recoverable with minimum downtime. You can configure Oracle Key Vault to ensure minimum downtime in the following ways:

- **Configuring High Availability:** High Availability is configured by adding redundancy in the form of a standby server. The standby server takes over from the primary server in the event of a failure, thus eliminating single points of failure, and minimizing downtime. For more information about configuring High Availability, see Configuring High Availability (page 4-3).

- **Enabling Read-Only Restricted Mode:** High Availability Read-Only Restricted mode ensures endpoint operational continuity when primary or standby Oracle Key Vault servers are affected by server, hardware, or network failures. When an unplanned shutdown causes the standby server to become unreachable, the primary server is still available to the endpoints.

  If High Availability Read-Only Restricted mode is disabled, the primary server will become unavailable and stop accepting requests in the event of a standby failure. Endpoints connected to Oracle Key Vault are unable to retrieve keys until connectivity is restored between primary and standby servers.

  To ensure endpoint operational continuity in the event of a primary or standby server failure, enable Read-Only Restricted mode.
- **Enabling Persistent Master Key Cache:** The Persistent Master Key Cache ensures that the endpoints can access keys in the event of a primary or standby server failure. While the surviving server is taking over from the failed peer, the endpoints can retrieve keys from the persistent cache and continue operations normally. For more information about Persistent Master Key Cache, see Persistent Master Key Cache (page 11-19).
- **Apply the TDE Heartbeat database patch on endpoints:** Apply the database patch for Bug 22734547 to tune the Oracle Key Vault heartbeat.

It is highly recommended that you back up Key Vault data regularly on a schedule. This practice ensures that backups are current and hold the most recent data. The backup can be used to restore a new or existing Key Vault server and be fully operational with minimum downtime and data loss.

If the OKV installation uses an online master key (formerly known as TDE direct connect), then during an upgrade, ensure that you upgrade database endpoints in parallel to reduce total downtime.

## 11.5 Using an User-Defined Key as the TDE Master Encryption Key

You can now import your generated key to be used as the Transparent Data Encryption (TDE) master encryption key in Oracle Key Vault.

- How Using a User-Defined Key as the TDE Master Encryption Key Works (page 11-25)
- Uploading the User-Defined Key (page 11-26)
- Activating the User-Defined Key as a TDE Master Encryption Key (page 11-28)

### 11.5.1 How Using a User-Defined Key as the TDE Master Encryption Key Works

Key Administrators can upload this user-defined key to the groups that they have write access to. This feature provides Key Administrators with more control on creation of the master key used to encrypt TDE data encryption keys.

The type parameter of the okvutil upload command features a new option TDE_KEY_BYTES that allows you to upload user-defined key bytes to Oracle Key Vault to be used as the TDE master key. The key is then activated as a TDE master encryption key by running the ADMINISTER KEY MANAGEMENT command on the database. For more information about activating a TDE master encryption key, see Activating TDE Master Encryption Keys.

You must first upload the user-defined key, and then activate this key as a TDE master encryption key.

## 11.5.2 Uploading the User-Defined Key

User-defined keys are uploaded to Oracle Key Vault using the okvutil upload command. The raw bytes data of the user-defined key is stored in a text file and uploaded to Oracle Key Vault.

The raw bytes data uploaded to Oracle Key Vault forms part of the TDE Master Key and the TDE Master Key Identifier. Additional metadata is added to the raw bytes data to enable the database to identify and activate the data as the TDE Master Key and the TDE Master Key Identifier.

In the text file, the raw bytes data that forms the TDE Master Key is prefixed by:

```
TDE Master Key
```

The raw bytes data that forms the TDE Master Key Identifier is prefixed by:

```
TDE Master Key Identifier
```

The TDE Master Key Identifier represents the master key in the database. Once the key is activated you should see the user-defined raw bytes that form the TDE Master Key Identifier as the subset of the KEY_ID column of the v$encryption_keys view. In Oracle Key Vault, the TDE Master Key and TDE Master Key Identifier are stored as managed KMIP objects with a symmetric key as a KMIP object type.

To upload the user-defined key:

1. Create a text file containing the raw bytes data of the user-defined key.

   Use the following format:

   ```
   TDE Master Key Identifier: <contiguous TDE Master Key Identifier
   bytes encoded in hex (32 bytes long)>
   TDE Master Key: <contiguous TDE Master Key bytes encoded in hex (64
   bytes long)>
   ```

   Example:

   ```
   TDE Master Key Identifier: 1F1E1D1C1B1A10191817161514131210
   TDE Master Key:
   000102030405060708090A0B0C0D0E0F101112131415161718191A1B1C1D1E1F
   ```

2. Save the file as `tde_key_bytes.txt`.
3. Use the `okvutil upload` command to upload `tde_key_bytes.txt`.

   The format of the `okvutil upload` command is:

   ```
   okvutil upload [--overwrite] --location location --type type [--
   group group] [--
   description description] [--verbose verbosity_level]
   ```

   Example:

   ```
   $OKV_HOME/bin/okvutil upload -l /home/oracle/tde_key_bytes.txt -t
   TDE_KEY_BYTES -g "FIN_DATABASE_VIRTUAL_WALLET" -d "This key was
   created for Financial database use on 1st Jan 2018"
   ```

   In this example:

   - `-l` specifies the path to the `tde_key_bytes.txt` file

     .
   - `-t` specifies the type of the object to upload. To upload a user-defined key, specify the type as `TDE_KEY_BYTES`.
   - `-g` specifies the name of an Oracle Key Vault virtual wallet to which the key is added.
   - `-d` specifies a description for the key.

   > **Note:**
   >
   > When `-t` is `TDE_KEY_BYTES`, the description specified for `–d` is displayed as the tag in the `v$encryption_keys` view of Oracle Database 12.1 and higher.

4. Specify the required parameters and press **Enter**.
5. Enter the Oracle Key Vault endpoint password and press **Enter**.

   The message `Upload succeeded` is displayed.

   ```
   $OKV_HOME/bin/okvutil upload -l /home/oracle/tde_key_bytes.txt -t
   TDE_KEY_BYTES -g "FIN_DATABASE_VIRTUAL_WALLET" -d "This key was
   created for Financial database use on 1st Jan 2018"
   Enter Oracle Key Vault endpoint password:
   Upload succeeded
   ```

The raw bytes data of the user-defined key is uploaded. The next step is to activate the user-defined key as a TDE master encryption key.

## 11.5.3 Activating the User-Defined Key as a TDE Master Encryption Key

After uploading the user-defined key, activate the key as a TDE master encryption key.

> **Note:**
>
> The raw bytes data uploaded to Oracle Key Vault for the `TDE Master Key Identifier` is displayed as the `NAME` attribute of the KMIP object that is created as the corresponding TDE master encryption key in Oracle Key Vault.

To activate the user-defined key:

1. Get the TDE master key identifiers of the user-defined key from the **Item Details** page.
   a. Log in to the Oracle Key Vault management console as a user with the Key Administrator role or as a user with access to the virtual wallet.
   b. Click the tab **Keys & Wallets**. The **Wallets** page appears.
   c. Click **All Items** in the left sidebar. The **All Items** page appears displaying all the security objects in Key Vault.
   d. Click the pencil icon in the **Details** column corresponding to the user-defined key. The **Item Details** page appears displaying the attributes of the key.
   e. Click **Advanced** to view the cryptographic attributes of the key.
   f. The required Key ID is displayed. The Key ID is prefixed with `ORACLE.TDE.HSM.MK`.

      Example:

      ```
      ORACLE.TDE.HSM.MK.061F1E1D1C1B1A10191817161514131210
      ```

      > **Note:**
      >
      > The TDE master key identifiers contain the user defined raw bytes data prefixed by additional metadata.

   g. Copy and store the Key ID displayed after the prefix `ORACLE.TDE.HSM.MK.`.

      Example:

      ```
      061F1E1D1C1B1A10191817161514131210
      ```

2. Connect to the database and activate the key as a TDE master encryption key using the `ADMINISTER KEY MANAGEMENT` command.

The format of the ADMINISTER KEY MANAGEMENT command is:

```
 ADMINISTER KEY MANAGEMENT USE [ENCRYPTION] KEY <Key ID for
activation>
       [USING TAG <tag>]
       IDENTIFIED BY <keystore_password>
       [WITH BACKUP [USING <backup_id>]];
```

Example:

```
$ORACLE_HOME/bin/sqlplus / as sysdba
SQL> Administer key management use key
'061F1E1D1C1B1A10191817161514131210' identified by "Welcome_1";
```

The user-defined key is activated as a TDE master encryption key.

**Note:**

In Oracle Database 12.1 and higher, you can query the TAG column of the V$ENCRYPTION_KEYS view for the identifier of the newly created key.

## 11.6 Using a TDE-Configured Oracle Database in an Oracle RAC Environment

In a TDE-configured Oracle database, if you upload Oracle wallets in an Oracle Real Application Cluster (Oracle RAC) environment, then the nodes within the cluster must share a virtual wallet.

You can enable the cluster to share the virtual wallet by using either of the following approaches:

- Define a virtual wallet as a shared default wallet for all of the nodes in the cluster.
- If each node in the cluster has a separate default wallet, then grant each node access to every other node's default wallet to have the same effect. An endpoint group may be used to simplify the process of granting each node access to the default wallets of the other nodes.

  The advantage of this approach over the first is that if you had several nodes that already had default wallets and you wanted them to share wallets, then you would not have to reassign their default wallets. This is particularly important because in the first approach, in order to reassign an endpoint's default wallet, you must reenroll the endpoint. An endpoint group can be used to simplify the process of granting each node access to the default virtual wallets of the other nodes.

As with single-instance database environments, after you download a password-protected wallet, you must manually open it. If you have one wallet on the primary node and then download the wallet to the other nodes, you must explicitly open the wallets on each of these nodes.

Each RAC node is a different end point of the database and has its own individual persistent cache. For RAC databases, a query should be initiated from each RAC node to cache the latest master key in the RAC node for uninterrupted operations

**See Also:**

- About the Persistent Master Key Cache (page 11-19)

# 11.7 Using a TDE-Configured Oracle Database in an Oracle GoldenGate Environment

You can migrate Oracle wallets that contain Oracle GoldenGate shared secrets and TDE master keys to the Oracle Key Vault server.

- About Uploading Oracle Wallets in an Oracle GoldenGate Environment (page 11-30)
- Using an Online Master Key in an Oracle GoldenGate Deployment (page 11-30)
- Migrating TDE Wallets on an Oracle GoldenGate Deployment to Oracle Key Vault (page 11-31)

## 11.7.1 About Uploading Oracle Wallets in an Oracle GoldenGate Environment

In an environment where Oracle Key Vault is not used and an Oracle TDE-enabled database is configured with an Oracle wallet with Oracle GoldenGate, this database (called the source database) stores an Oracle GoldenGate shared secret in the same Oracle wallet where master keys are stored.

This means that when you configure the source database as an Oracle Key Vault endpoint, the Oracle GoldenGate shared secret is stored in Oracle Key Vault in the same virtual wallet where the master keys are stored for the TDE-enabled source database.

When you migrate an Oracle wallet that contains an Oracle GoldenGate shared secret and TDE master keys to Oracle Key Vault using the `okvutil` command-line utility, the default wallet for the TDE-enabled source database now stores the entire Oracle wallet migrated with shared secret and master keys.

In addition, if the configured target database is an Oracle database, then you must ensure that this target database is TDE-enabled so that all the TDE commands can be replicated. Note that the two Oracle TDE-enabled databases, source and target, do not need to have the same master key in the Oracle wallet. If you configure this target database as a new Key Vault endpoint, then you can upload and download wallets to and from Oracle Key Vault as you normally would with any independent Key Vault endpoint. No additional configuration is necessary.

## 11.7.2 Using an Online Master Key in an Oracle GoldenGate Deployment

Note, that the term Online Master Key replaces the term TDE direct connect.

There are two configuration steps to using the Online Master Key in an Oracle GoldenGate deployment:

1. Configure a connection between the source database in the GoldenGate deployment and Oracle Key Vault.
2. Configure the storage of Oracle GoldenGate secrets in the Oracle wallet on the source database.

At this stage, the configuration is complete. If you have configured the `sqlnet.ora` file correctly and completed the other configuration required for TDE on the source database, then when you set the encryption key (using either `ALTER SYSTEM SET ENCRYPTION KEY` or `ADMINISTER KEY MANAGEMENT SET ENCRYPTION KEY`), a TDE master key is created in Oracle Key Vault. You can encrypt tables or create encrypted tablespaces in the database. The encrypted data created in the source database continues to be replicated on the target database after this procedure is performed. The other Oracle GoldenGate shared secrets are stored in Oracle Key Vault.

**See Also:**

- "Configuring a Connection Between Oracle Key Vault and a New TDE-Enabled Database (page 11-12)" for instructions to connect a source database in GoldenGate to Key Vault.
- *Oracle Database Advanced Security Guide* for more information on configuring the storage of Oracle GoldenGate secrets in the source database.

## 11.7.3 Migrating TDE Wallets on an Oracle GoldenGate Deployment to Oracle Key Vault

In an Oracle GoldenGate environment with a TDE-configured database, an Oracle wallet contains both the TDE master keys and the Oracle GoldenGate shared secret.

Note that target databases (if they are Oracle TDE-enabled databases) that are used in this Oracle GoldenGate environment can also be configured to use Oracle Key Vault or continue to use Oracle wallet. You should treat these databases as you would any standalone TDE database endpoint.

After you complete this migration, the configuration is complete. If you have configured the `sqlnet.ora` file correctly and completed the other configuration required for TDE, then when you set the encryption key (using either `ALTER SYSTEM SET ENCRYPTION KEY` or `ADMINISTER KEY MANAGEMENT SET ENCRYPTION KEY`), a TDE master key is created in Oracle Key Vault. You can continue to create and use encrypted tables or tablespaces in the database. The encrypted data created in the source database continues to be replicated on the target database after this procedure is performed.

**See Also:**

"Migrating Existing TDE Wallets to Oracle Key Vault (page 11-15)" to migrate the Oracle wallet to Key Vault

# 11.8 Using a TDE-Configured Oracle Database in an Oracle Active Data Guard Environment

You can perform the following activities in an Oracle Active Data Guard environment: upload Oracle wallets to an Oracle Key Vault server, use Online Master Keys, migrate wallets between endpoints, and migrate and check TDE wallets for a logical standby database.

- About Uploading Oracle Wallets in an Oracle Active Data Guard Environment (page 11-32)
- Uploading Oracle Wallets in an Oracle Active Data Guard Environment (page 11-33)
- Performing an Online Master Key Connection in an Oracle Active Data Guard Environment (page 11-33)
- Migrating Oracle Wallets in an Oracle Active Data Guard Environment (page 11-34)
- Reverse Migrating Oracle Wallets in an Active Data Guard Environment (page 11-35)
- Migrating an Oracle TDE Wallet to Oracle Key Vault for a Logical Standby Database (page 11-36)
- Checking the Oracle TDE Wallet Migration for a Logical Standby Database (page 11-36)

## 11.8.1 About Uploading Oracle Wallets in an Oracle Active Data Guard Environment

In an Oracle Active Data Guard environment with a TDE-enabled primary and a standby database using an Oracle wallet, you must physically copy the Oracle wallet file from the primary database to the standby after the initial TDE configuration and later, whenever you rekey the master key on the primary database.

Whereas, when using Oracle Key Vault with a TDE-enabled Active Data Guard database, the primary and standby databases must be registered in Oracle Key Vault as endpoints. You must ensure that the endpoints that are registered for both the primary and standby databases have same default virtual wallet in Oracle Key Vault.

This way, the two databases can achieve centralized key and wallet management without the need of a manual copy of the wallet file from the primary database to the standby database.

**Persistent Cache in an Active Data Guard Environment**

A REKEY on the primary database will cache the Master Key in its own persistent cache. When the new REDO logs from the primary are applied on the standby, only then will the standby fetch the new key from the OKV and cache it in the persistent cache of the standby. There is a time lag between the caching of the key in primary and the caching of the key in standby. It is recommended that the primary and standby be synchronized as soon as possible after the REKEY.

**See Also:**

- About the Persistent Master Key Cache (page 11-19)

## 11.8.2 Uploading Oracle Wallets in an Oracle Active Data Guard Environment

You can upload an Oracle wallet to an Active Data Guard environment as follows:

1. Register one endpoint each for the primary and standby databases.
2. Download the `okvclient.jar` file for each endpoint on the respective databases.
3. Ensure that the endpoint password is the same as the TDE wallet password if you must perform a migration or a reverse migration.
4. Ensure that both the primary and standby database endpoints use the same default virtual wallet.

**See Also:**

- "Managing Endpoints (page 7-2)"
- "Task 2: Install Oracle Key Vault Software on the Endpoint (page 8-6)"

## 11.8.3 Performing an Online Master Key Connection in an Oracle Active Data Guard Environment

The procedure for performing a TDE direct connection in an Oracle Active Data Guard environment is the same as in a standard Oracle Database environment.

**See Also:**

"About Using an Online Master Key with Oracle Key Vault (page 11-11)" for more information

## 11.8.4 Migrating Oracle Wallets in an Oracle Active Data Guard Environment

You can migrate an Oracle wallet in an Oracle Active Data Guard environment as follows:

1. Use the `okvutil upload` command to upload the contents of the local Oracle wallet that is on the primary database to Oracle Key Vault.
2. Perform the steps to migrate the wallet.
3. Close the existing Oracle wallet on the standby database.
   - For Oracle Database 11*g* Release 2, as a user who has been granted the `ALTER SYSTEM` system privilege:

     ```
     ALTER SYSTEM SET ENCRYPTION WALLET CLOSE IDENTIFIED BY
     "Key_Vault_endpoint_password";
     ```

   - For Oracle Database 12*c*, as a user who has been granted the `SYSKM` administrative privilege:

     ```
     ADMINISTER KEY MANAGEMENT SET KEYSTORE CLOSE IDENTIFIED BY
     "Key_Vault_endpoint_password";
     ```

4. Shut down the standby database.

   For example:

   ```
   SHUTDOWN IMMEDIATE
   ```

5. Restart the standby database.

   For example:

   ```
   STARTUP
   ```

6. Open the Oracle wallet.
   - For Oracle Database 11*g* Release 2, as a user who has been granted the `ALTER SYSTEM` system privilege:

     ```
     ALTER SYSTEM SET ENCRYPTION WALLET OPEN IDENTIFIED BY
     "Key_Vault_endpoint_password";
     ```

   - For Oracle Database 12*c*, as a user who has been granted the `SYSKM` administrative privilege:

     ```
     ADMINISTER KEY MANAGEMENT SET KEYSTORE OPEN IDENTIFIED BY
     "Key_Vault_endpoint_password";
     ```

7. Start the apply process on the standby database.

**See Also:**

- "okvutil upload Command (page 8-14)"
- "Migrating Existing TDE Wallets to Oracle Key Vault (page 11-15)"
- See *Oracle Data Guard Concepts and Administration* for information about starting the apply process on the standby database

## 11.8.5 Reverse Migrating Oracle Wallets in an Active Data Guard Environment

To reverse migrate an Oracle wallet in an Active Data Guard environment:

1. Use the `okvutil download` command to download the Oracle wallet keys onto the primary database from Oracle Key Vault. Download these keys to a local keystore.

   See "okvutil download Command (page 8-19)".

2. Perform a reverse migration.

   See *Oracle Database Advanced Security Guide* for more information about performing reverse migrations.

3. Close the existing Oracle wallet on the standby database.

   - For Oracle Database 11*g*R2:

     ```
     ALTER SYSTEM SET ENCRYPTION WALLET CLOSE IDENTIFIED BY
     "Key_Vault_endpoint_password";
     ```

   - For Oracle Database 12*c*:

     ```
     ADMINISTER KEY MANAGEMENT SET KEYSTORE CLOSE IDENTIFIED BY
     "Key_Vault_endpoint_password";
     ```

4. Copy the Oracle wallet from the primary database to the standby database.

   See *Oracle Database Advanced Security Guide* for more information.

5. Open the Oracle wallet on the standby database.

   - For Oracle Database 11*g*R2:

     ```
     ALTER SYSTEM SET ENCRYPTION WALLET OPEN IDENTIFIED BY
     "Key_Vault_endpoint_password";
     ```

   - For Oracle Database 12*c*:

     ```
     ADMINISTER KEY MANAGEMENT SET KEYSTORE OPEN IDENTIFIED BY
     "Key_Vault_endpoint_password";
     ```

6. Start the apply process on the standby database.

   See *Oracle Data Guard Concepts and Administration* for information about starting the apply process on the standby database.

**Note:**

If the endpoint password and the local TDE wallet password are different, then use the auto-login HSM feature. See *Oracle Database Advanced Security Guide* for more information about configuring auto-login wallets.

## 11.8.6 Migrating an Oracle TDE Wallet to Oracle Key Vault for a Logical Standby Database

If you have a logical standby database configured and are using Oracle Database Release 12*c*, then you can migrate a TDE wallet to Oracle Key Vault as follows:

1. Register the primary and standby endpoints to have the same default virtual wallet.
2. If necessary, download and install the `okvclient.jar` file to each endpoint.
3. Perform the migration on the primary database.
4. Complete the SQL Apply process on the logical standby and then restart the standby database.
5. To check that the status the that migration was successful, query the `V$ENCRYPTION_WALLET` dynamic view.

**See Also:**

- "Manage Endpoint Groups (page 7-16)"
- "Finalizing Enrollment and Provisioning (page 8-3)"
- "Migrating an Existing TDE Wallet to Oracle Key Vault (page 11-16)"
- *Oracle Data Guard Concepts and Administration* for more information about the SQL Apply process

## 11.8.7 Checking the Oracle TDE Wallet Migration for a Logical Standby Database

In an Oracle Database Release 12*c* environment, after you have migrated an Oracle TDE wallet in a logical standby configuration, you can check the configuration by querying the `WRL_TYPE` and `WALLET_ORDER` columns of the `V$ENCRYPTION_WALLET` dynamic view.

By querying the `V$ENCRYPTION_WALLET` view, you can track the primary keystore. If you have only a single wallet configured, then the `WALLET_ORDER` column is set to `SINGLE`. In a two-wallet or mixed configuration, the column is set to `PRIMARY` or `SECONDARY`, depending on where the active master key is located. Consider the following queries.

In the following query, only a single wallet is configured:

```
SELECT WRL_TYPE, WALLET_ORDER FROM V$ENCRYPTION_WALLET;

WRL_TYPE             WALLET_OR
-------------------- ---------
FILE                 SINGLE
```

In this query in a logical standby configuration, the active master key has been migrated to an Oracle Key Vault virtual wallet:

```
SELECT WRL_TYPE, WALLET_ORDER FROM V$ENCRYPTION_WALLET;

WRL_TYPE             WALLET_OR
-------------------- ---------
FILE                 SECONDARY
HSM                  PRIMARY
```

This query should show the `HSM` as the `PRIMARY` wallet in both the primary and standby database for the logical configuration.

## 11.9 MySQL Integration with Oracle Key Vault

Oracle Key Vault supports integration with MySQL from Release 12.2 and later.

> **Note:**
>
> MySQL Windows databases are not supported.

Oracle Key Vault can manage MySQL TDE encryption keys.

## 11.10 Upload a Keystore from Automatic Storage Management to Oracle Key Vault

You can copy a keystore from Automatic Storage Management (ASM) to Oracle Key Vault and vice versa in a two-step process, which is described here.

Uploading a keystore from ASM to Oracle Key Vault is a two step process:

1. Copy the keystore from ASM to the file sytem.
2. Upload the keystore from the file system to Oracle Key Vault.

Copying a keystore from ASM to the file system or vice versa requires the keystore merge operation that merges one software keystore to an existing key store. Therefore, in order to copy a keystore from a source path to a target path, a keystore must exist at the target path.

You can use the `ADMINISTER KEY MANAGEMENT` statement to move a software keystore out of Automatic Storage Management (ASM) as follows:

1. Initialize a target keystore on the file system with the following command:

   ```
   ADMINISTER KEY MANAGEMENT CREATE KEYSTORE targetKeystorePath IDENTIFIED BY
   targetKeystorePassword;
   ```

In this specification:

- *targetKeystorePath* is the directory path to the target keystore on the file system.
- *targetKeystorePassword* is a password that you create for the keystore.

For example:

```
ADMINISTER KEY MANAGEMENT CREATE KEYSTORE '/etc/ORACLE/KEYSTORE/DB1/'
IDENTIFIED BY "destination_password";
```

Where '/etc/ORACLE/KEYSTORE/DB1/' is the path to the target keystore in the file system and “destination_password” is the keystore password.

The keystore from ASM can now be copied to the target keystore.

2. Copy the keystore from ASM to the target keystore that you just created.

   This step requires that you merge the keystore from ASM to the file system as follows:

   ```
   ADMINISTER KEY MANAGEMENT MERGE KEYSTORE srcKeystorePath IDENTIFIED BY
   srcKeystorePassword INTO EXISTING KEYSTORE targetKeystorePath IDENTIFIED BY
   targetKeystorePassword WITH BACKUP USING backupIdentifier;
   ```

   In this specification:

   - *srcKeystorePath* is the directory path to the source keystore.
   - *srcKeystorePassword* is the source keystore password.
   - *targetKeystorePath* is the path to the target keystore.
   - *targetKeystorePassword* is the target keystore password.
   - *backupIdentifier* is the backup identifier to be added to the backup file name.

   For example:

   ```
   ADMINISTER KEY MANAGEMENT MERGE KEYSTORE '+DATAFILE' IDENTIFIED BY
   "srcPassword" INTO EXISTING KEYSTORE '/etc/ORACLE/KEYSTORE/DB1/' IDENTIFIED
   BY "destination_password" WITH BACKUP USING "bkup";
   ```

   The keystore is copied to the file system and can now be uploaded to Oracle Key Vault.

3. Upload keystore from file system to Oracle Key Vault using the `okvutil upload` command.

   ```
   $  okvutil upload -l location -t type
   ```

   Where:

   - `location` is the path to the target keystore in the file system
   - `type` is wallet

   For example:

   ```
   $ okvutil upload -l etc/ORACLE/KEYSTORE/DB1 -t wallet
   ```

The keystore is now copied from ASM to Oracle Key Vault.
To copy a keystore from Oracle Key Vault to ASM you would reverse the steps:

1. Initialize a target keystore on the file system, *if it does not exist*. If it exists, skip this step.
2. Copy the keystore from Key Vault to the target keystore on the file system using the `okvutil download` command.

   ```
   $  okvutil download -l location -t type
   ```

   Where:

   - `location` is the path to the target keystore in the file system
   - `type` is wallet

   For example:

   ```
   $ okvutil download -l etc/ORACLE/KEYSTORE/DB1 -t wallet
   ```

3. Copy the keystore from the target keystore to ASM.

12

# General Oracle Key Vault Management

General management consists of system and audit management tasks. You must be an administrative user with the System Administrator and Audit Manager role to perform these tasks.

- About General Oracle Key Vault Management (page 12-1)
- Remote Monitoring Using SNMP (page 12-1)
- Email Notification (page 12-8)
- Oracle Key Vault System Administration (page 12-13)
- Oracle Key Vault Alert Configuration (page 12-23)
- Oracle Key Vault Auditing (page 12-27)
- Oracle Key Vault Reports (page 12-31)
- Upgrade Oracle Key Vault Server Software (page 12-36)

## 12.1 About General Oracle Key Vault Management

System administrators configure system settings, remote monitoring, email notification, backup, and recovery. Audit managers configure alerts, and download system diagnostics for further debugging and analysis.

> **See Also:**
>
> Managing Oracle Key Vault Users (page 5-1)
>
> Managing Oracle Key Vault Endpoints (page 7-1)
>
> Managing Oracle Key Vault Virtual Wallets and Security Objects (page 6-1)

## 12.2 Remote Monitoring Using SNMP

With SNMP enabled, system administrators can remotely monitor the Key Vault appliance and its services. The collected data can be further processed and presented for the needs of the enterprise.

- About Using SNMP for Oracle Key Vault (page 12-2)
- Granting SNMP Access to Users (page 12-3)
- Changing the SNMP User Name and Password (page 12-3)
- Changing SNMP Settings on the Standby Server (page 12-3)
- Remotely Monitoring Oracle Key Vault Using SNMP (page 12-4)

- SNMP Management Information Base Variables for Oracle Key Vault (page 12-5)
- Example: Simplified Remote Monitoring of Oracle Key Vault Using SNMP (page 12-6)

## 12.2.1 About Using SNMP for Oracle Key Vault

You can use the Simple Network Management Protocol (SNMP) to monitor devices on a network for resource usage.

Monitoring Oracle Key Vault is an important aspect how critical Oracle Key Vault's availability is when hundreds or thousands of Oracle and MySQL databases store their TDE master encryption keys in Oracle Key Vault. The types of resource usage that you should monitor include memory, CPU utilization, and processes.

You can use Simple Network Management Protocol (SNMP) third-party tool to monitor remote systems that access Oracle Key Vault. The benefits of using SNMP to monitor Oracle Key Vault are as follows:

- There is no need to allow SSH access to Oracle Key Vault. (SSH access should only be enabled for the window of time in which it is being used.)
- You do not need to install additional tools to perform an SNMP monitoring operation.

Oracle Key Vault uses SNMP version 3 for user authentication and data encryption features. Unlike SNMP versions 1 and 2 that communicate in readable, insecure plaintext, SNMP 3 authenticates users and encrypts data on the communication channel between the monitoring server and the target. The information from Oracle Key Vault is unreadable to an intruder, even if the communication channel is intercepted.

In addition, with SNMP enabled on Oracle Key Vault, you can determine whether the key management server (KMIP daemon) is running. To track this information, you must use a third-party SNMP client to poll the Oracle Key Vault instance, because Oracle Key Vault does not provide SNMP client software.

Oracle Key Vault audits the creation and modification of SNMP credentials.

You must be a user with the System Administrator role to configure the SNMP account with a user name and password. These SNMP credentials are needed to access SNMP data.

**Note:**

You must ensure that the SNMP username and password is *not* the same username and password as any of the Oracle Key Vault administrative user accounts with the System Administrator, Key Administrator, or Audit Manager role.

## 12.2.2 Granting SNMP Access to Users

You can grant any user, including users who are not Oracle Key Vault administrators, access to SNMP data.

1. Log in to the Oracle Key Vault management console as a user with the System Administrator role.
2. Select the **System** tab, and then select **Monitoring Settings** from the left side bar.

   The Monitoring page appears.
3. In the Monitoring page, enter the following information:
   - **SNMP Access:** Select **All** to enable a client at any IP address to poll Oracle Key Vault for information, **Disabled** to prevent any client, regardless of the client IP address, to poll Oracle Key Vault for information, or **IP Address(es)** if you want to restrict polling to clients with specific IP addresses. If you select **IP Address(es)**, then enter the IP addresses of the users you want to grant access to in the IP Address field. Separate multiple IP addresses by a space. You cannot enter a range of IP addresses. You must list each IP address individually.
   - **Username:** Enter a name to associate with the SNMP configuration that will perform the monitoring.
   - **Password and Confirm Password:** Enter a secure password for this user that is at least 8 or more characters and contains at least one of each of the following: an uppercase letter, a lowercase letter, a number, and a special character from the set: period (`.`), comma (`,`), underscore (_), plus sign (+), colon (`:`), space. The SNMP password must ***not*** be the same as the password used to login into the Oracle Key Vault management console in any of the administrative roles.
4. Click **Save**.

## 12.2.3 Changing the SNMP User Name and Password

You can change the SNMP user name and password for a node at any time.

1. Log in to the Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **System** tab, and then select **Monitoring Settings**.
3. In the **Username**, **Password**, and **Reenter Password** fields, enter the user name and password information.
4. Click **Save**.

## 12.2.4 Changing SNMP Settings on the Standby Server

You change the SNMP settings from the command line on the standby server. To add SNMP support in a primary-standby environment, you should configure SNMP on both the primary and standby servers before pairing them. This is because the standby server is no longer accessible from the Oracle Key Vault management console because all requests are forwarded to the primary server. However, you can change SNMP settings on the standby server in a primary-standby environment.

1. Log in to the standby server as the `support` user.
2. Switch to the root user.

   ```
   su -
   ```

3. Go to the Oracle Key Vault bin directory.

   ```
   cd /usr/local/okv/bin/
   ```

4. Run the `stdby_snmp_enable` script.

   ```
   ./stdby_snmp_enable parameter "options"
   ```

   In this specification:

   - *`parameter`* can be the following:
     - `-a`, which sets the SNMP access. It accepts the following *`options`*:
       - `all` grants SNMP access.
       - `disabled` disables SNMP access.
       - *`IP_addresses`* specifies one or more IP addresses to be granted SNMP access. Separate each IP address with a space.
     - `-u` sets the user's SNMP name.
     - `-p` sets the user's SNMP password.
   - *`options`* is only used with the `-a` parameter.

The following examples show how to change SNMP settings on a standby server:

To grant SNMP access to all IP addresses and assign a user name `snmpuser` and password *`password`*:

```
./stdby_snmp_enable -a "all" -u "snmpuser" -p "password"
```

To disable SNMP access from all IP addresses:

```
./stdby_snmp_enable -a "disabled"
```

To grant SNMP access to certain IP addresses and assign user name `snmpuser` and password *`password`*:

```
./stdby_snmp_enable -a "192.0.2.1 192.0.2.3 192.0.2.3" -u "snmpuser" -p
"password"
```

## 12.2.5 Remotely Monitoring Oracle Key Vault Using SNMP

SNMP enables you to monitor the vital components of Oracle Key Vault remotely without having to install new software in Oracle Key Vault. Though there are third-party tools that graphically display the information that SNMP extracts from Oracle Key Vault, the examples shown here are given with `snmpwalk` and `snmpget` from the command line on a remote computer that has a network connection into the SNMP account in Oracle Key Vault.

1. Log in to the remote host that will monitor Oracle Key Vault.
2. Confirm that the `UCD-SNMP-MIB` is installed on the remote host from which Oracle Key Vault is monitored.
3. Query the object ID for an Oracle Key Vault-supported SNMP Management Information Base (MIB) variable.

   For example, suppose you wanted to track the number of processes running for the SNMP host. You can use a third-party SNMP client utility to query the status of the KMIP MIB whose object ID is `1.3.6.1.4.1.2021.2`, as follows:

   ```
   third_party_snmp_client_command -v 3 OKV_IP_address -u SNMP_user -
   a SHA -A SNMP_password -x AES -X SNMP_password -l authPriv
   iso.3.6.1.4.1.2021.2.1.2.1
   ```

   The output is similar to the following:

   ```
   iso.3.6.1.4.1.2021.2.1.2.1 = STRING: "mwecsvc"              <== Event
   collector
   iso.3.6.1.4.1.2021.2.1.2.2 = STRING: "httpd"                <== httpd
   iso.3.6.1.4.1.2021.2.1.2.3 = STRING: "kmipd"                <== KMIP daemon
   iso.3.6.1.4.1.2021.2.1.2.4 = STRING: "ora_pmon_dbfwdb"      <== embedded DB
   ```

**Related Topics**

- SNMP Management Information Base Variables for Oracle Key Vault (page 12-5)

## 12.2.6 SNMP Management Information Base Variables for Oracle Key Vault

Oracle Key Vault provides a set of SNMP Management Information Base (MIB) variables that you can track.

The following table lists the MIB variables that are supported.

**Table 12-1 MIBs That SNMP Tracks for Oracle Key Vault**

| MIB Variable | Object ID | Description |
|---|---|---|
| `hrSystemUptime` | 1.3.6.1.2.1.25.1.1 | Tracks the amount of time that an Oracle Key Vault instance has been running |
| `ifAdminStatus.`*`x`* | 1.3.6.1.2.1.2.2.1.7 | Tracks if the Oracle Key Vault network interface (*x*) are running, not running, or being tested. Values are as follows:<br>• 1: Instance is running<br>• 2: Instance is down<br>• 3: Instance is being tested |
| `memAvailReal` | 1.3.6.1.4.1.2021.4.6 | Tracks the available RAM |
| `memTotalReal` | 1.3.6.1.4.1.2021.4.5 | Tracks the total amount of RAM being used |

Table 12-1 (Cont.) MIBs That SNMP Tracks for Oracle Key Vault

| MIB Variable | Object ID | Description |
|---|---|---|
| `ssCpuRawIdle` | 1.3.6.1.4.1.2021.11.53 | For CPU monitoring; tracks the number of ticks (typically 1/100s) spent idle |
| `ssCpuRawInterrupt` | 1.3.6.1.4.1.2021.11.56 | For CPU monitoring; tracks the number of ticks (typically 1/100s) spent processing hardware interrupts |
| `ssCpuRawKernel` | 1.3.6.1.4.1.2021.11.55 | For CPU monitoring; tracks the number of ticks (typically 1/100s) spent processing kernel-level code |
| `ssCpuRawNice` | 1.3.6.1.4.1.2021.11.51 | For CPU monitoring; tracks the number of ticks (typically 1/100s) spent processing reduced-priority code |
| `ssCpuRawSystem` | 1.3.6.1.4.1.2021.11.52 | For CPU monitoring; tracks the number of ticks (typically 1/100s) spent processing system-level code |
| `ssCpuRawUser` | 1.3.6.1.4.1.2021.11.50 | For CPU monitoring; tracks the number of ticks (typically 1/100s) spent processing user-level code |
| `ssCpuRawWait` | 1.3.6.1.4.1.2021.11.54 | For CPU monitoring; tracks the number of ticks (typically 1/100s) spent waiting for input-output (IO) |
| `UCD-SNMP-MIB.prTable` | 1.3.6.1.4.1.2021.2 | Tracks the number of processes running under a certain name. Names we monitor are `httpd` (the http server), `kmipd` (the kmip daemon), and `ora_pmon_dbfwdb (an indicator if the DB is down)` |

**See Also:**

For more information refer to the Net-SNMP documentation at `http://www.net-snmp.org`

## 12.2.7 Example: Simplified Remote Monitoring of Oracle Key Vault Using SNMP

In Linux, you can simplify the SNMP commands you manually enter to find Oracle Key Vault information, yet still have useful and detailed output.

The configuration in this section assumes that you have granted SNMP access to a trusted user. It also assumes that the you have installed the SNMP Management Information Base (MIB) variables on the remote host that will monitor Oracle Key Vault.

For example, a lengthy version of the `snmpwalk` command for an SNMP user named `snmp_admin` is as follows:

```
snmpwalk -v3 OKV_IP_address -n "" -l authPriv -u snmp_admin -a SHA -A
snmp_user_password -x AES -X snmp_user_password
```

This command lists the vital services that are running on Oracle Key Vault. However, you can modify the command (and other SNMP commands) to be not only shorter, but to show additional information, such as whether the services are running or not running.

To simplify this type of command, you can edit the `/etc/snmp/snmp.conf` configuration file so that the SNMP commands you enter will automatically include commonly used settings, such as the default user or the default security level. The example in this topic omits password parameters so that users can enter the password at the command line interactively.

1. Log in to the remote host that will monitor Oracle Key Vault.
2. Edit the `/etc/snmp/snmp.conf`, which appears as follows:

   ```
   # As the snmp packages come without MIB files due to license reasons,
   # loading MIBs is disabled by default. If you added the MIBs you
   # can reenable loading them by commenting out the following line.
     mibs :
   ```

3. Comment out the `# mibs :` line and then add the following lines, as follows:

   ```
   # loading MIBs is disabled by default. If you added the MIBs you
   # can reenable loading them by commenting out the following line.
   # mibs :
   defSecurityName snmp_admin
   defSecurityLevel authPriv
   defAuthType SHA
   defPrivType AES
   ```

   In this example:

   - `defSecurityName`: Enter the name of the user to whom you granted SNMP access. This example uses `snmp_admin`.
   - `defSecurityLevel`: Enter the default security level to use. This example uses `authPriv`, which enables communication with authentication and privacy.
   - `defAuthType`: Enter the default authorization type. This example uses `SHA`.
   - `defPrivType`: Enter the default privilege type. This example uses `AES`.

4. Restart `snmpd` to load the configuration file.
   For example, for Linux 7:

   ```
   systemctl restart snmpd
   ```

   For Linux 6:

   ```
   service snmpd restart
   ```

5. To run the simplified version of the `snmpwalk` command that was shown earlier, enter the following command:

```
snmpwalk okv_ip_address prNames -A snmp_user_pwd -X snmp_user_pwd
```

In this command, `prNames` refers to "process names", which displays the names of processes instead of numbers. For example:

```
$ snmpwalk 192.0.2.254 prNames -A snmp_user_pwd -X snmp_user_pwd
UCD-SNMP-MIB::prNames.1 = STRING: mwecsvc
UCD-SNMP-MIB::prNames.2 = STRING: httpd
UCD-SNMP-MIB::prNames.3 = STRING: kmipd
UCD-SNMP-MIB::prNames.4 = STRING: ora_pmon_dbfwdb
```

An example of running the `snmptable` command now becomes the following.

```
snmptable okv_ip_address prTable -A snmp_user_pwd -X snmp_user_pwd
```

Output similar to the following appears.

```
SNMP table: UCD-SNMP-MIB::prTable
prIndex         prNames prMin prMax prCount prErrorFlag prErrMessage prErrFix
prErrFixCmd
      1         mwecsvc     1     1       1     noError      noError
      2           httpd     1    20       9     noError
noError
      3           kmipd     1     2       2     noError
noError
      4 ora_pmon_dbfwdb     1     1       1     noError      noError
```

The next example shows how you would now run the `snmpdf` command:

```
snmpdf okv_ip_address -A snmp_user_pwd -X snmp_user_pwd
```

Output similar to the following appears.

```
Description                Size (kB)  Used    Available   Used%
/                              20027260   7249732 12777528      36%
/dev/shm                        8174120         0  8174120       0% < -- not
used by Oracle Key Vault
/usr/local/dbfw                  999320    251180   748140      25% < -- not
used by Oracle Key Vault
/usr/local/dbfw/tmp             6932408     15764  6916644       0%
/var/tmp                        5932616     15848  5916768       0% < -- not
used by Oracle Key Vault
/opt/dbfw                        999320      1544   997776       0% < -- not
used by Oracle Key Vault
/home                            999320      6416   992904       0% < -- not
used by Oracle Key Vault
/var/log                        5932616     22992  5909624       0%
/tmp                            1999184      3072  1996112       0%
/var/dbfw                       2966224      4524  2961700       0% < -- not
used by Oracle Key Vault
/usr/local/dbfw/volatile        1048576         0  1048576       0% < -- not
used by Oracle Key Vault
/var/lib/oracle               143592160  45620964  97971196      31%
```

## 12.3 Email Notification

Email notifications can be used to communicate Key Vault status changes directly to administrators without logging into the management console.

To enable email notification you must set your email preferences in Key Vault. You can choose the events that you want updates to. The events include Key Vault system status like disk utilization, backup, and high availability, or user and endpoint status like expiration of user passwords, endpoint certificates, and keys.

## 12.3.1 About Email Notification

In addition to alerting users on status changes in Oracle Key Vault, email notifications enable administrators to complete the processes of endpoint enrollment and user password reset.

For example:

- The enrollment token generated during endpoint enrollment can be mailed directly to the endpoint administrator from Oracle Key Vault.
- An Oracle Key Vault system administrator can send the random temporary password directly to the user when the user password is reset.

To enable email notifications successfully, there must be a direct connection between Oracle Key Vault and the SMTP server.

You can disable email notifications at any time.

## 12.3.2 Configure Email Settings

You can enable email notification by configuring the Simple Mail Transfer Protocol (SMTP) server properties of the user's email account. Oracle Key Vault supports anonymous and insecure connections to the SMTP server.

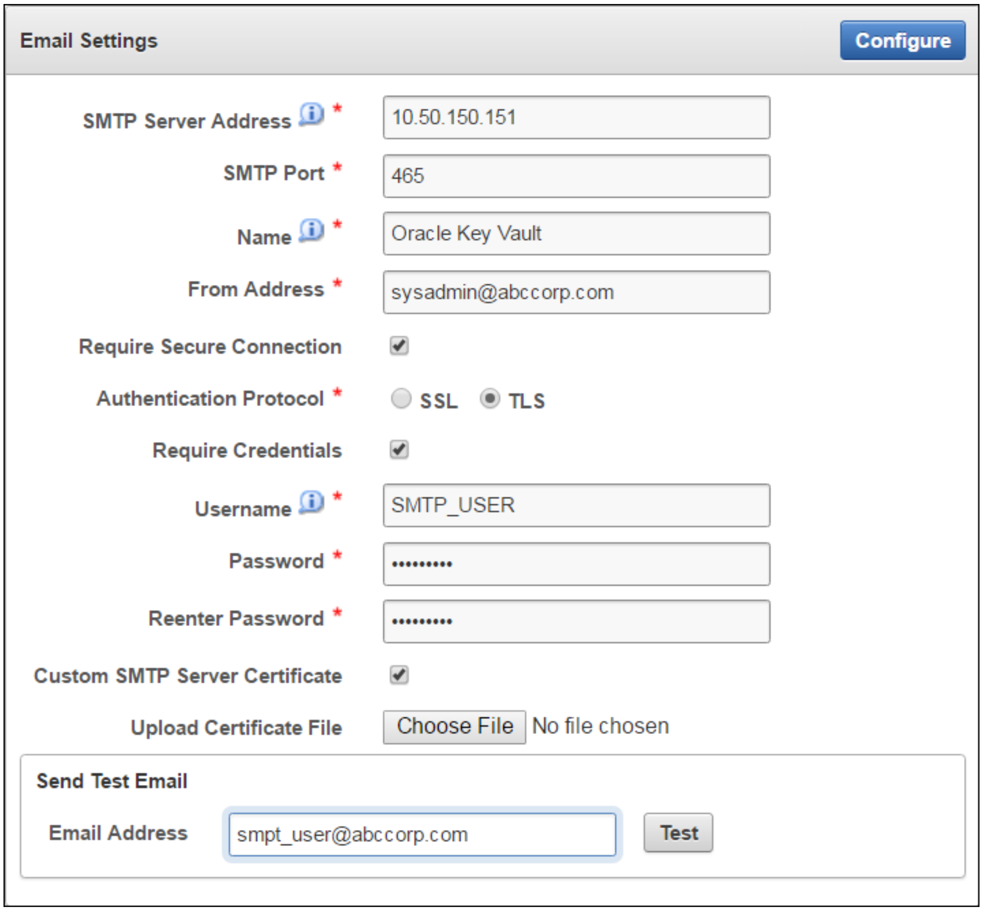
By default, the default Java truststore packaged with Key Vault's Java library is used to validate the server certificate. Optionally, you can upload a custom truststore in order to use a specific certificate or certificate chain at the same time you configure SMTP settings.

The SMTP server configuration can be modified at any time. If a custom SMTP certificate was used initially, and the user later decides to use the default, you simply have to modify the trust store setting to default, instead of custom.

To configure email settings follow these steps:

1. Log in to the Oracle Key Vault management console as a user with the System Administrator role.
2. Click the **System** tab, and then click **Email Settings**. The **Email Settings** page appears.

Figure 12-1    Email Settings



Email Settings

|  |  |
|---|---|
| SMTP Server Address * | 10.50.150.151 |
| SMTP Port * | 465 |
| Name * | Oracle Key Vault |
| From Address * | sysadmin@abccorp.com |
| Require Secure Connection | ☑ |
| Authentication Protocol * | ○ SSL ● TLS |
| Require Credentials | ☑ |
| Username * | SMTP_USER |
| Password * | ••••••••• |
| Reenter Password * | ••••••••• |
| Custom SMTP Server Certificate | ☑ |
| Upload Certificate File | No file chosen |

Send Test Email

Email Address: smpt_user@abccorp.com

3. In the **Email Settings** page, enter values for the following:
   - **SMTP Server Address:** Enter a valid SMTP server address or hostname for the user account. This setting should match the SMTP server setting of the user's email account. Ensure that the SMTP server or hostname is reachable from Key Vault. If you enter the SMTP hostname, you must configure DNS from the **System Settings** menu, so the hostname can be resolved.
   - **SMTP Port:** Enter the SMTP port number of the outgoing SMTP server, usually 465. This port number can be another number, if expressly configured that way in your organization.
   - **Name:** Enter an alias for the SMTP user that will appear in the From field of the email.
   - **From Address:** Enter the email address you want to provide as a sender.
   - If the SMTP server requires a secure connection, select **Require Secure Connection**.

> **Note:**
>
> If you are using anonymous relay on Microsoft Exchange Server, or an external SMTP server such as Gmail or Office 365, do not select **Require Secure Connection**. Ensure that your firewall rules allow forwarding of SMTP requests to an external SMTP server.

If **Require Secure Connection** is selected, the **Authentication Protocol** field is displayed with two options, **SSL** and **TLS**:

  – Select the authentication protocol for the email server, either **SSL** or **TLS**. The default is **TLS**.

- If you have an SMTP user account, check the box **Require Credentials**. When checked, the input fields **Username**, **Password**, and **Reenter Password** appear:
  – Enter the username of the SMTP user account.

  – Enter the password for the SMTP user account.

  – Reenter the password for the SMTP user account.

> **Caution:**
>
> It is recommended to have a secure connection to the SMTP server, as auto-generated tokens are sent over email for operations like the creation of administrative users and Key Vault system alerts.

Do not check **Require Credentials** for non-secure connections.

- If **Custom SMTP Server Certificate** is checked, the field **Upload Certificate File** appears with a **Choose File** button to its right. Select this option, if you want to upload a custom SMTP server's certificate to establish a TLS session between SMTP and Oracle Key Vault. This is how an administrator can add a custom truststore, in cases where the default Java truststore does not contain a necessary certificate.
  – **Upload Certificate File:** Click **Choose File** to upload a custom certificate file.

4. Click **Configure**.

   On successful configuration, a `SMTP successfully configured` message is displayed.

   If the configuration fails, you should check the SMTP server settings of the user email account and verify that they are correct. Error messages highlight the field where the error occurs to help isolate the problem.

### 12.3.3 Test the Email Configuration

You can test the email configuration of the SMTP user account any time *after* saving the configuration. If you change an existing SMTP configuration, you must save the configuration in order to test it.

To test the email configuration:

1. Log in to the Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **System** tab, and then select **Email Settings.**

   The **Email Settings** page appears.
3. Configure the user's SMTP settings.
4. Save the configuration. You must save the configuration in order to test it.
5. In the **Send Test Email** section, enter the user email address in the **Email Address** field. Then click **Test**.

   An email is sent to the user with `Oracle Key Vault: Test Message` in the subject line.

   Depending on the Oracle Key Vault server timestamp, the email notification may not show up as the latest email.

   The email notification may also not show up in your inbox, in which case you must check the spam folder.

   If the email notification is not received, click the **Reports** tab and select **System Reports** from the left sidebar. On the **System Reports** page, click **Notification Report**. Check the list to determine the issue encountered while sending the email notification.

**See Also:**

"Configure Email Settings (page 12-9)"

## 12.3.4 Disable Email Notifications for a User

An Oracle Key Vault user may elect not to receive email alerts. Only a user with the System Administrator role, or a user managing his own account can disable email notifications.

To disable email notifications:

1. Log in to the Oracle Key Vault management console as a user with the System Administrator role.
2. Select the **Users** tab.

   The **Manage Users** page appears.
3. Click **User Name** of the user.

   The **User Details** page appears.
4. Check the box to the left of text **Do not receive email alerts**.
5. Click the **Save** button on the top right.

   A '**Successfully updated user attributes**' confirmation message appears.

# 12.4 Oracle Key Vault System Administration

The System Administrator is the only administrative user who can access the **System** tab and menus. This user configures system settings, recovers the system when no other administrative users are present, and downloads the system diagnostics file for further analysis.

## 12.4.1 Configure System Settings

You can configure the system time, syslog, DNS, network services, RESTful services and Oracle Audit Vault Integration. In addition you can reboot and power off Oracle Key Vault.

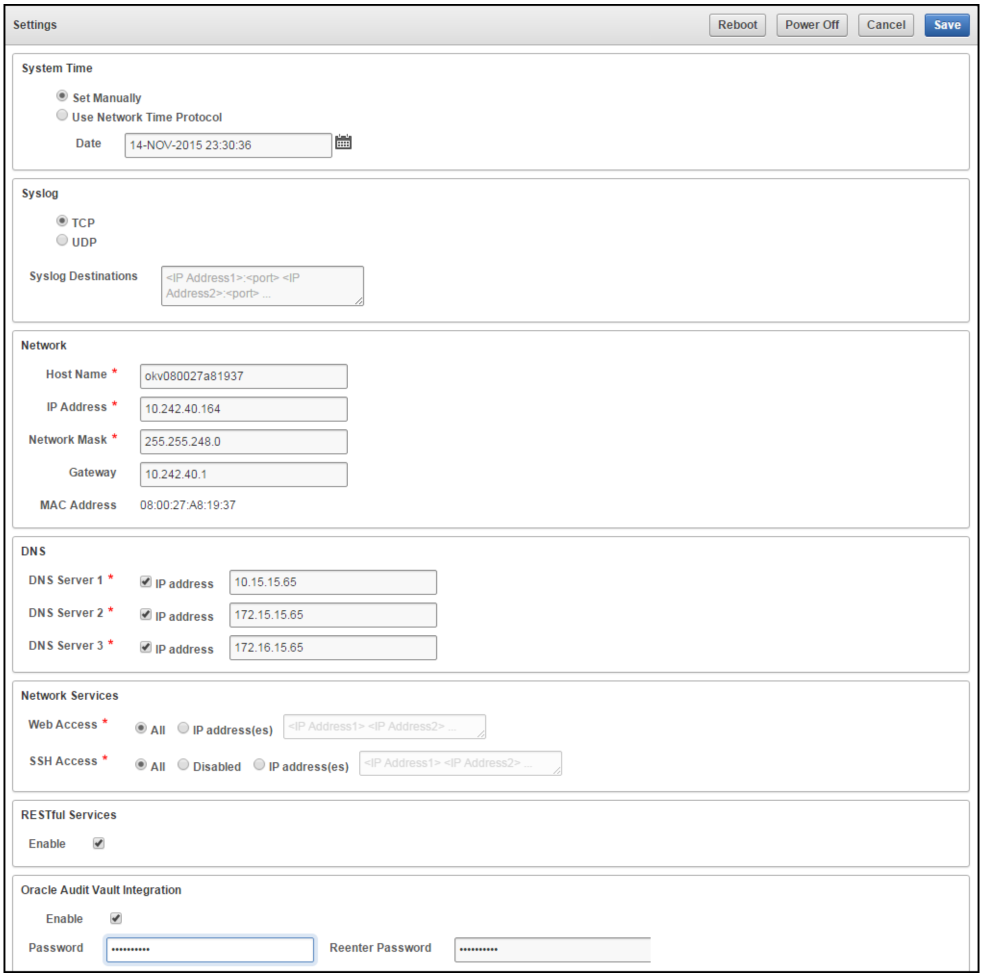
On the system **Settings** page, you can configure the system time, syslog, DNS, network services, RESTful services and Oracle Audit Vault Integration. Click **System**, then **System Settings** in the left side bar to arrive at this page.

To configure system settings:

1. Log into the Key Vault management console as a user with the System Administrator role.
2. Select **System**, then **System Settings** from the left sidebar.

   The **Settings** page appears.

**Figure 12-2 Oracle Key Vault System Settings**



System Time

● Set Manually
○ Use Network Time Protocol

Date: 14-NOV-2015 23:30:36

Syslog

● TCP
○ UDP

Syslog Destinations: <IP Address1>:<port> <IP Address2>:<port> ...

Network

| | |
|---|---|
| Host Name * | okv080027a81937 |
| IP Address * | 10.242.40.164 |
| Network Mask * | 255.255.248.0 |
| Gateway | 10.242.40.1 |
| MAC Address | 08:00:27:A8:19:37 |

DNS

| | | |
|---|---|---|
| DNS Server 1 * | ☑ IP address | 10.15.15.65 |
| DNS Server 2 * | ☑ IP address | 172.15.15.65 |
| DNS Server 3 * | ☑ IP address | 172.16.15.65 |

Network Services

Web Access * ● All ○ IP address(es) <IP Address1> <IP Address2> ...

SSH Access * ● All ○ Disabled ○ IP address(es) <IP Address1> <IP Address2> ...

RESTful Services

Enable ☑

Oracle Audit Vault Integration

Enable ☑

Password •••••••••• Reenter Password ••••••••••

The system **Settings** page has the following panes:

- **System Time**

  You can configure Oracle Key Vault to use an NTP server to remain synchronized with the current time. If an NTP server is not available, then you can set the current time manually. You should use the calendar icon to set the date and time so that these values are stored in the correct format. In a high availability deployment, you must set the primary and standby servers to the same time.

- **Syslog**

  All system related alerts are sent to syslog. These include the following: Disk Utilization, System Backup, Failed System Backup, High Availability Role Change, High Availability Destination Failure, SSH Tunnel Failure

  Select the protocol to transfer syslog files: **TCP** for Transmission Control Protocol, a connection-oriented, reliable protocol, or **UDP** for User Datagram Protocol, a connection-less, best effort protocol.

  You can set the destination computer for syslog files by entering the IP address (and port number for TCP) in the format shown in the **Syslog**

**Destinations** field. For more than one destination computer add the IP address (and port number for TCP) of each destination computer separated by a space.

> **Note:**
>
> For TCP, specify the IP address and the port number. For UDP, specify only the IP address.

You can elect to send Key Vault alerts to syslog to allow external monitoring.

- **Network**

  Fields in this pane are automatically populated with the IP address and hostname of your Oracle Key Vault server. But if anything changes, you have the ability to update the **Host Name**, **IP Address**, **Network Mask** and the **Gateway** for your Key Vault installation. You cannot change the **MAC Address**, as this is the hard-wired address of the network interface.

  If you have a high availability configuration, then you must unpair the primary and standby Oracle Audit Vault Servers before changing the IP address. After you have changed the IP address of the primary or standby Oracle Audit Vault Server, pair the two servers again. After you complete the pairing process, redeploy the Oracle Audit Vault agents to ensure that they are updated with the new IP addresses for both the primary and the standby Oracle Audit Vault servers.

- **DNS**

  You can configure Domain Name Service (DNS) to translate host names to IP addresses. This is useful, if you only know the hostname and not the IP address of a server you need access to. For example, while configuring the SMTP server for email notifications, you can optionally enter the host name instead of the IP Address, after you set up DNS.

- **Network Services**

  You can enable services for **Web** Access and **SSH Access** (Secure Shell Access) for all, none, or a subset of clients, determined by their IP addresses by selecting one of the following options:

  – **All**, to select all IP addresses

  – **IP address(es)** to select a set of IP address(es) that you specify in the next field, separating each IP address by a space.

  The **IP address(es)** web access option allows you to restrict access to the Oracle Key Vault management console to a limited set of users that you specify to meet your organizational needs.

  Enabling **SSH Access** gives you access to Key Vault from the command line. This helps you diagnose problems not immediately apparent from the management console. You must log in as the user `'support'`, with the support password you created during installation.

  If you are using the Bash shell, you may need to download patch sets or security fixes that work with SSH Access. Instructions on downloading and enabling patch sets or security fixes come with the patch set release notes.

**Best Practice:** Enable SSH access for short durations, solely for diagnostics and troubleshooting purposes, and disable it as soon as you are done.

> **Note:**
>
> Enabling or disabling SSH access will enable or disable the **inbound** SSH connection to the Oracle Key Vault server. Enabling or disabling SSH access in this manner has no bearing on the SSH Tunnel settings or any other outbound SSH connections that the Oracle Key Vault server itself establishes. SSH connections can still be established by the Oracle Key Vault to other servers as in the case of SSH Tunnel settings.

- **RESTful Services**
  - First, ensure that the **Web Access** options in **Network Services** are set.
  - Next, check the box after **Enable** to enable **RESTful Services**. RESTful services allow you to automate endpoint enrollment and provisioning.
- **Oracle Audit Vault Integration**

  Check the box after **Enable** to enable audit report consolidation between Key Vault and Audit Vault. It will prompt you to enter and confirm the password.

3. Modify any of the system settings and click **Save**.
4. You can reboot or power off the Key Vault server by clicking **Reboot** and **Power Off** in the top right.

## 12.4.2 System Recovery

In an emergency when no administrative users are available, or you need to change the password of administrative users, you can recover the system with the recovery passphrase that was created during Key Vault installation. In addition, you can change the recovery passphrase to keep up with security best practices.

- Recovering Credentials for Administrators (page 12-16)
- Change the Recovery Passphrase (page 12-17)
- Change the Installation Passphrase (page 12-17)

### 12.4.2.1 Recovering Credentials for Administrators

You can recover the system by adding credentials for administrative users.

1. From a web browser using HTTPS, enter the IP address of the Oracle Key Vault installation.
2. In the Oracle Key Vault login page, *do not log in*.
3. Click the **System Recovery** link at the lower right corner of the page.
4. In the **Recovery Passphrase** field, enter the recovery passphrase and then click **Login**.

   The **Administrator Recovery** page appears with two tabs above it: **Administrator Recovery** and **Recovery Passphrase**.

5. In the **Administrator Recovery** page, fill out the fields in the Key Administrator, System Administrator, and Audit Manager panes to assign these roles to new or existing user accounts.
6. Click **Save**.

**Related Topics**

- Performing Post-Installation Tasks (page 3-10)

## 12.4.2.2 Change the Recovery Passphrase

Oracle highly recommends that a user with the System Administrator role perform a new backup whenever the recovery passphrase changes, so that there is always a backup protected with the current recovery passphrase. This ensures that you will have at least one backup with the latest data.

To change the recovery passphrase:

1. From a web browser, enter the IP address of your Key Vault installation. The Key Vault login page appears. *Do not log in*.
2. Click the **System Recovery** link.

   A new login page appears with a single field: **Recovery Passphrase**.
3. Enter the recovery passphrase and click **Login**.

   The **Administrator Recovery** page appears with two tabs above it: Administrator Recovery and Recovery Passphrase.
4. Click **Recovery Passphrase**.

   The **Recovery Passphrase** page appears with two fields to enter and re-enter the new passphrase.
5. Enter the new recovery passphrase in the two fields.
6. Click **Submit**.

## 12.4.2.3 Change the Installation Passphrase

The Installation Passphrase is specified during installation. The Installation Passphrase is used to log in to Oracle Key Vault and complete the post-installation tasks.

The installation passphrase must have 8 or more characters and contain at least one of each of the following: an uppercase letter, a lowercase letter, number, and special character from the set: period (.), comma (,), underscore (_), plus sign (+), colon (:), space.

If you forget the installation passphrase, you can specify a new installation passphrase.

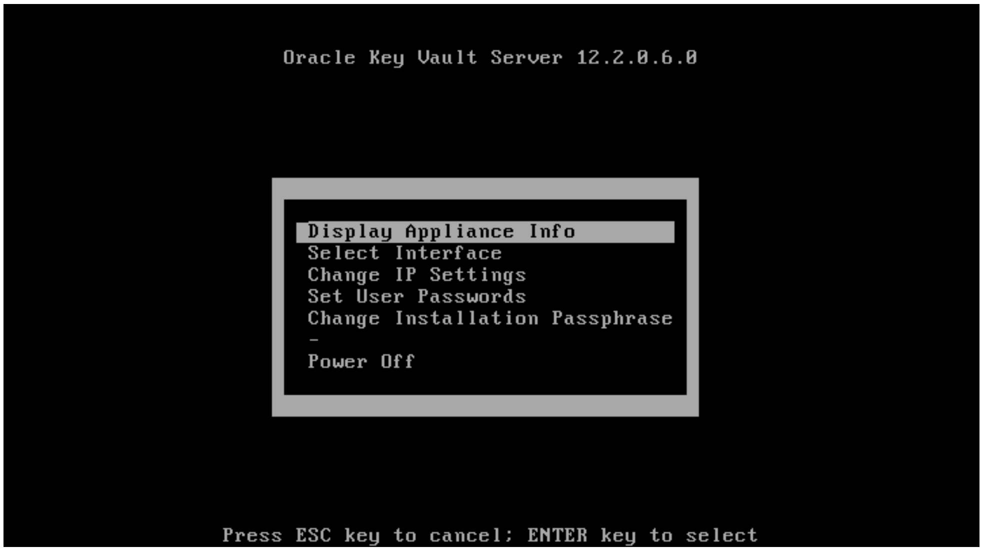
It is important to store the installation passphrase securely.

To change the Installation Passphrase:

1. Using SSH, log in to the Oracle Key Vault server terminal as the System Administrator.

   The **Oracle Key Vault Server <Release Number>** screen appears.

**Figure 12-3    Oracle Key Vault Server <Release Number> Screen**

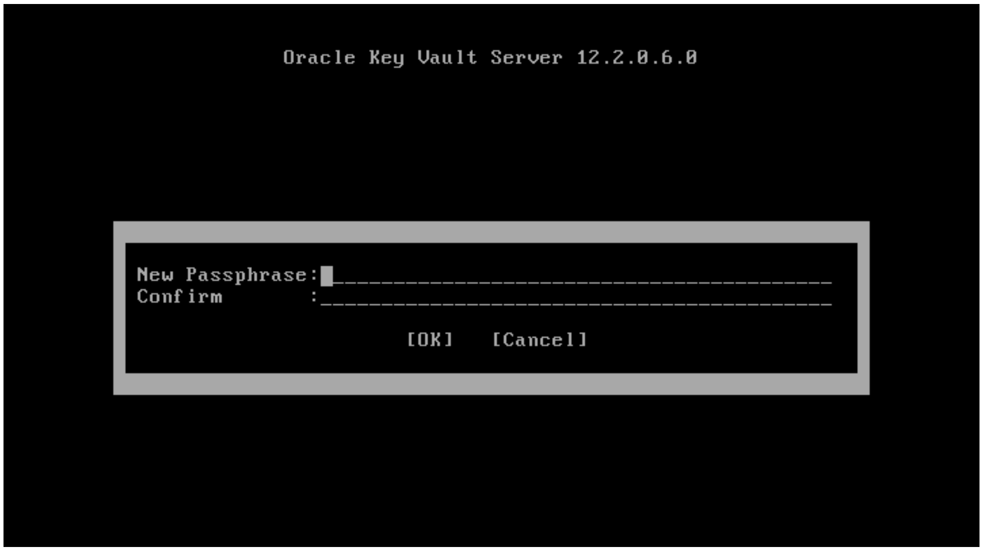


```
Oracle Key Vault Server 12.2.0.6.0

Display Appliance Info
Select Interface
Change IP Settings
Set User Passwords
Change Installation Passphrase
-
Power Off

Press ESC key to cancel; ENTER key to select
```

2. Select **Change Installation Passphrase** and press **Enter**.

   The **New Passphrase** screen appears.

**Figure 12-4    New Passphrase Screen**

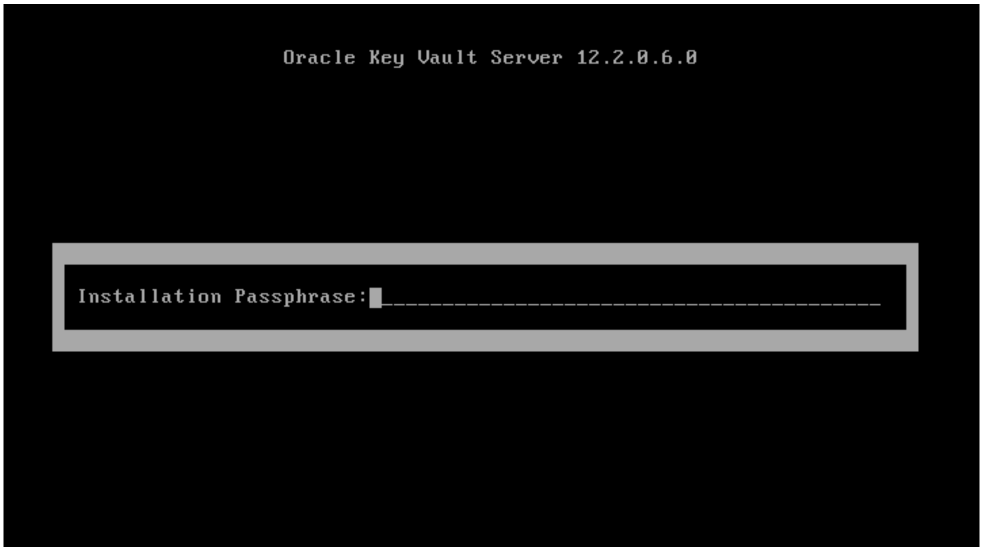


```
Oracle Key Vault Server 12.2.0.6.0

New Passphrase:______________________________________________
Confirm       :______________________________________________

                    [OK]   [Cancel]
```

3. Type the new installation passphrase in the **New Passphrase** and **Confirm** fields. Select **OK** and press **Enter**.

   The **Installation Passphrase** screen appears.

**Figure 12-5 Installation Passphrase Screen**



```
Oracle Key Vault Server 12.2.0.6.0

Installation Passphrase:_____________________________________________
```

4. Enter the old Installation Passphrase and press **Enter**.

The Installation Passphrase is changed.

## 12.4.3 Download System Diagnostics

You can view status information about disk usage, server uptime, version, high availability, and backup on the **Status** page. Further, you can download the diagnostics file and provide it to Oracle support for further analysis and debugging. This feature provides advanced debug and troubleshooting capability for problems you may encounter.

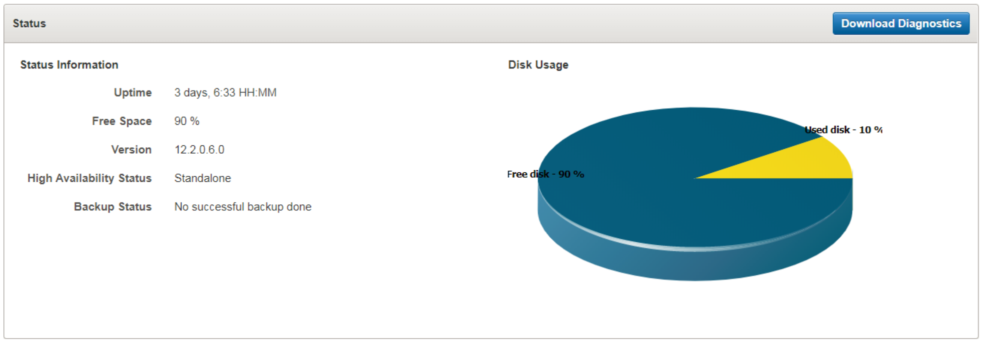
In Oracle Key Vault 12.2.0.6.0 and later, diagnostics reporting is not enabled by default. The user must enable the feature to generate diagnostics reports. Once enabled, the user can configure the necessary information to be captured in diagnostics reports. The user can customize and package diagnostics reports with flexibility.

If you plan to upgrade Oracle Key Vault, then you must remove the diagnostics generation utility before performing the upgrade.

To download the diagnostics file:

1. Log in to the Oracle Key Vault management console as a root user.
2. Select **System**. The **Status** page appears.

**Figure 12-6    System Status Page**


Status

Download Diagnostics

Status Information

| | |
|---|---|
| Uptime | 3 days, 6:33 HH:MM |
| Free Space | 90 % |
| Version | 12.2.0.6.0 |
| High Availability Status | Standalone |
| Backup Status | No successful backup done |

Disk Usage

The **Status** page displays the following information:

- **Uptime**
- **Free Space**
- **Version**
- **High Availability Status**
- **Backup Status**
- **Disk Usage**

3. Click **Download Diagnostics**.

   If the diagnostics generation utility is not installed:

   a. You are prompted to save the `diagnostics-not-enabled.readme` file.

   b. Save and open `diagnostics-not-enabled.readme`. Follow the instructions to install, enable, and run the diagnostics generation utility.

   c. Install the diagnostics generation utility:

   ```
   /usr/local/dbfw/bin/priv/dbfw-diagnostics-package.rb --install
   ```

   d. Enable the collection of diagnostics:

   ```
   /usr/local/dbfw/bin/priv/dbfw-diagnostics-package.rb --enable ALL
   ```

4. If diagnostics collection is enabled, you are prompted to download a `.zip` file containing the diagnostics reports.

   Save the `.zip` file containing the diagnostics reports.

You can customize the `dbfw-diagnostics-package.yml` file in the `/usr/local/dbfw/etc/` directory to include and exclude a combination of files in multiple categories. Each section of `dbfw-diagnostics-package.yml` contains options to enable and disable a specific category by setting the value to `true` or `false`.

For more information about installing, enabling, and running the diagnostics generation utility, refer to the help:

```
/usr/local/dbfw/bin/priv/dbfw-diagnostics-package.rb --help
```

To free up disk space, you can remove `dbfw-diagnostics-package.rb` after installing the diagnostics generation utility. You must also remove the diagnostics generation utility before you upgrade Oracle Key Vault.

```
/usr/local/dbfw/bin/priv/dbfw-diagnostics-package.rb --remove
```

**Related Topics**

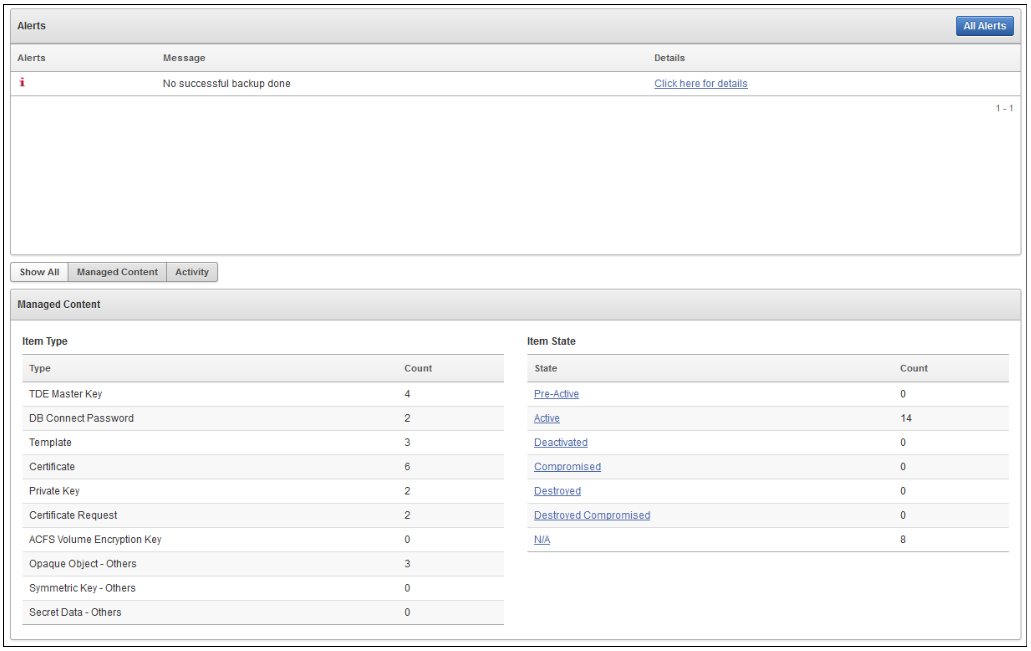
- Transaction Check Error: Diagnostics Generation Utility (page B-5)

## 12.4.4 View the Oracle Key Vault Dashboard

The **Home** tab of the management console displays the dashboard when you log into the management console. The dashboard presents the current status of the Oracle Key Vault at a high level and is visible to all users.

**Alerts** and **Managed Content** are the first sections you will see on logging in.

**Figure 12-7 Alerts and Managed Content Panes**



Alerts

| Alerts | Message | Details |
|---|---|---|
| i | No successful backup done | Click here for details |

1 - 1

Managed Content

Item Type

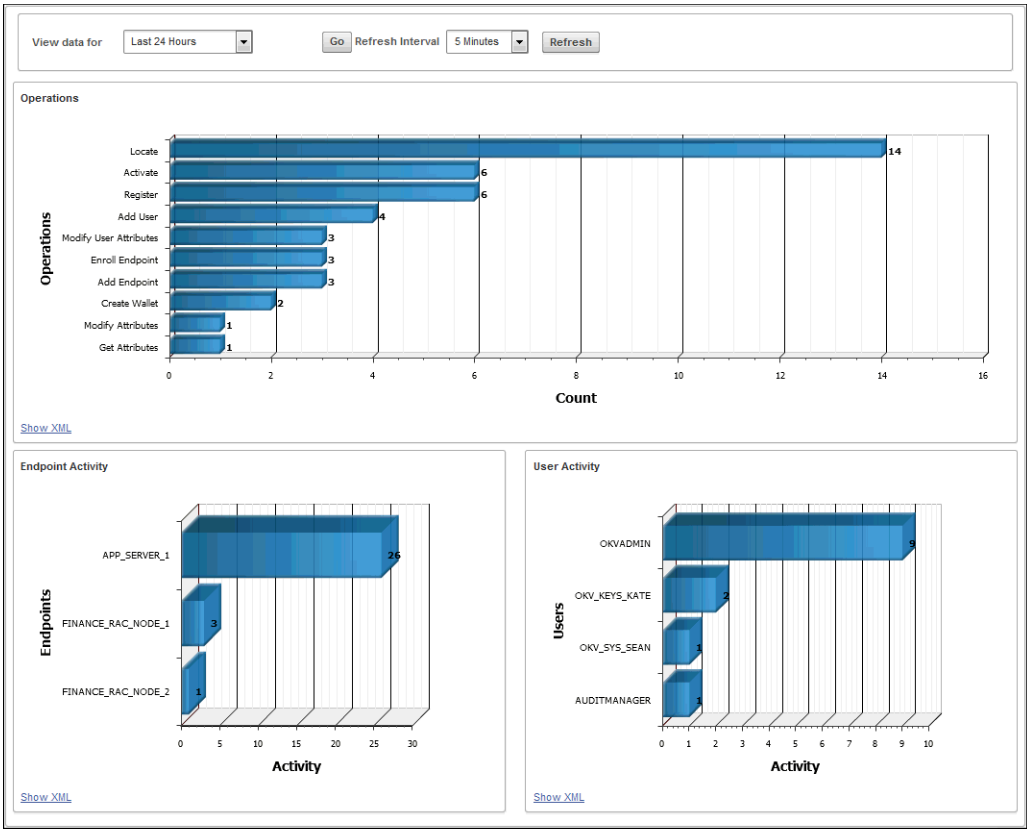
| Type | Count |
|---|---|
| TDE Master Key | 4 |
| DB Connect Password | 2 |
| Template | 3 |
| Certificate | 6 |
| Private Key | 2 |
| Certificate Request | 2 |
| ACFS Volume Encryption Key | 0 |
| Opaque Object - Others | 3 |
| Symmetric Key - Others | 0 |
| Secret Data - Others | 0 |

Item State

| State | Count |
|---|---|
| Pre-Active | 0 |
| Active | 14 |
| Deactivated | 0 |
| Compromised | 0 |
| Destroyed | 0 |
| Destroyed Compromised | 0 |
| N/A | 8 |

The **Data Interval**, **Operations**, **Endpoint Activity**, and **User Activity** panes of the **Home** page follow **Alerts** and **Managed Content**.

**Figure 12-8 Data Interval, Operations, Endpoint Activity, and User Activity Panes**



View data for Last 24 Hours Go Refresh Interval 5 Minutes Refresh

## 12.4.5 Status Panes in the Dashboard

The status panes on the dashboard provide useful high level information as follows:

- **Alerts**

  To take corrective action on a particular alert:

  1. Click the link in the **Details** column corresponding to the alert. The appropriate page appears.
  2. Take corrective action appropriate to the alert.

  To configure the alerts you want to see on the dashboard:

  1. Click the **Reports** tab, then **Alerts** from the left side bar.

     The **Alerts** page appears.
  2. Click **Configure** from the top right, or **Configure Alerts** from the left sidebar under **ALERTS**.

     The **Configure Alerts** page appears.
  3. Select the **Alert Type** and click **Save**.

- **Managed Content**

The **Managed Content** pane of the dashboard displays aggregated information about security objects currently stored and managed in Oracle Key Vault.

This status pane categorizes the aggregate information based on the item type such as keys, certificates, opaque objects, private keys, and TDE master keys, as well as the item state such as pre-active, active, and deactivated.

In the **Managed Content** pane, the item type and item state are displayed at the last time refreshed, which is set by the refresh interval described in the Data Interval status pane.

- **Data Interval**

  This pane shows the length of the time period.

  This time period can be **Last 24 hours**, **Last week**, or **Last Month**, or a user-defined date range. It also shows the refresh interval for the Operations, Endpoint Activity, and User Activity sections described later.

- **Operations**

  The **Operations** pane contains a bar graph with bars for key-related operations such as locate, activate, add endpoint, and assign default wallet.

- **Endpoint Activity**

  The **Endpoint Activity** pane contains a bar graph for tracking the number of operations performed by each endpoint.

- **User Activity**

  The **User Activity** pane contains a three-dimensional bar graph for tracking the number of operations performed by each user.

> **See Also:**
>
> "Search for Security Object Items (page 6-12)" for details of Item Types and Item States

## 12.5 Oracle Key Vault Alert Configuration

You can select the type of alerts that you want to see in the Oracle Key Vault dashboard. The dashboard is the first page you see on logging into to the management console. You can navigate to this page by clicking the **Home** tab. All users can see the alerts on security objects they have access to, but only users with the System Administrator role can configure alerts.

- About Configuring Alerts (page 12-23)
- Configuring Alerts (page 12-24)
- Viewing Open Alerts (page 12-25)

### 12.5.1 About Configuring Alerts

Oracle Key Vault has a variety of alerts that you can configure with appropriate thresholds according to your requirements. You can configure the following alerts:

- Disk Utilization
- Endpoint Certificate Expiration
- Failed System Backup
- High Availability Data Guard Broker Status
- High Availability Data Guard Fast-Start Failover Status
- High Availability Destination Failure
- High Availability Restricted Mode
- High Availability Role Change
- Key Rotations
- SSH Tunnel Failure
- System Backup
- User Password Expiration
- Invalid HSM Configuration

> **See Also:**
>
> Create an Oracle Key Vault User (page 5-2)
>
> *Oracle Key Vault Integration with Hardware Security Module*

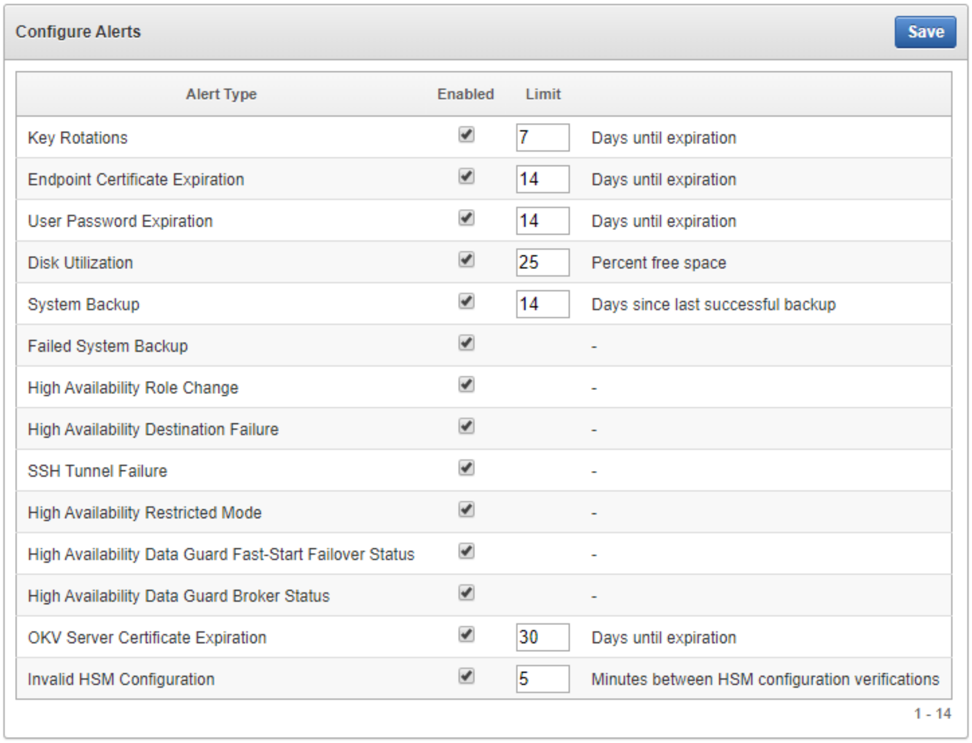
## 12.5.2 Configuring Alerts

To configure alerts:

1. Log in to the Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **Reports** tab.
3. Select **Configure Alerts** from the left sidebar.

   The **Configure Alerts** page appears.

**Figure 12-9    Configure Alerts Page**



| Alert Type | Enabled | Limit | |
|---|---|---|---|
| Key Rotations | ☑ | 7 | Days until expiration |
| Endpoint Certificate Expiration | ☑ | 14 | Days until expiration |
| User Password Expiration | ☑ | 14 | Days until expiration |
| Disk Utilization | ☑ | 25 | Percent free space |
| System Backup | ☑ | 14 | Days since last successful backup |
| Failed System Backup | ☑ | | - |
| High Availability Role Change | ☑ | | - |
| High Availability Destination Failure | ☑ | | - |
| SSH Tunnel Failure | ☑ | | - |
| High Availability Restricted Mode | ☑ | | - |
| High Availability Data Guard Fast-Start Failover Status | ☑ | | - |
| High Availability Data Guard Broker Status | ☑ | | - |
| OKV Server Certificate Expiration | ☑ | 30 | Days until expiration |
| Invalid HSM Configuration | ☑ | 5 | Minutes between HSM configuration verifications |

1 - 14

4. Check the box(es) in the **Enabled** column to the right of the alert type(s) to enable it. Then set the threshold value in the box under **Limit**. This value determines when the alert will be sent. You can un-check the boxes by alerts that you do not want to appear in the dashboard.
5. Click **Save**.

> **See Also:**
>
> " View the Oracle Key Vault Dashboard (page 12-21)"

## 12.5.3 Viewing Open Alerts

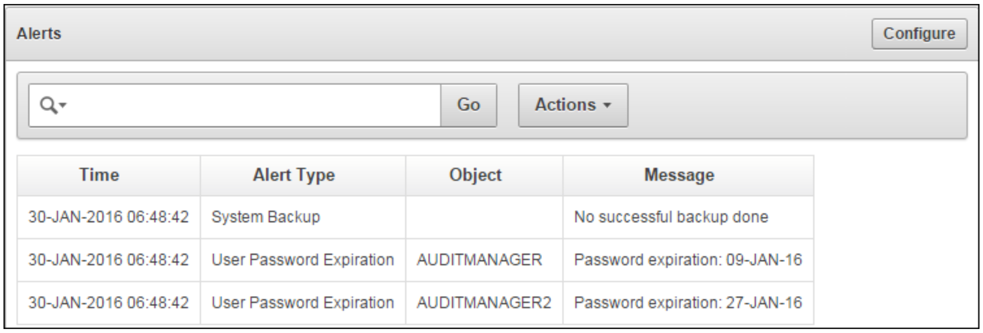
Users with the System Administrator role can view all alerts. Users without system administrator privileges can only view alerts related to objects they can access.

To view open alerts follow these steps:

1. Log in to the Oracle Key Vault management console as a user who has the System Administrator role.
2. Click the **Reports** tab. The **Audit Trail** appears.
3. Click **Alerts** from the left sidebar.

The **Alerts** page appears displaying all the alerts that have not been resolved. When you resolve the issue stated in the alert message, the alerts are automatically removed. They cannot be explicitly deleted

**Figure 12-10 Alerts Page**



| Time | Alert Type | Object | Message |
|---|---|---|---|
| 30-JAN-2016 06:48:42 | System Backup | | No successful backup done |
| 30-JAN-2016 06:48:42 | User Password Expiration | AUDITMANAGER | Password expiration: 09-JAN-16 |
| 30-JAN-2016 06:48:42 | User Password Expiration | AUDITMANAGER2 | Password expiration: 27-JAN-16 |

Oracle Key Vault sends all system alerts to the `syslog`. The following is an example of a system alert in `syslog`:

```
July 29 18:36:29 okv080027361e7e logger[13171]: No successful backup done
for 4 day(s)
```

The following table lists the conditions that trigger alerts, and the accompanying system alert message:

| Condition | System Alert Message |
|---|---|
| Key Rotations | Key expiration: *<date>* |
| Endpoint Certificate Expiration | Endpoint certificate expiration: *<date>* |
| User Password Expiration | Password expiration: *<date>* |
| Disk Utilization | Free disk space is below *<threshold value>* (currently *<current value>*) |
| System Backup | No successful backup for *<number>* day(s) |
| Failed System Backup | Most recent backup failed! |
| High Availability Role Change | HA role changed. Primary IP Address: *<IP address>* |
| High Availability Destination Failure | HA destination failure |
| SSH Tunnel Failure | SSH tunnel is not available |
| High Availability Restricted Mode | HA running in read-only restricted mode |
| High Availability Data Guard Fast-Start Failover Status | HA FSFO is not synchronized. FSFO status is <HA status> |
| High Availability Data Guard Broker Status | Dataguard Broker is disabled |

| Condition | System Alert Message |
|---|---|
| OKV Server Certificate Expiration | `The Oracle Key Vault Server certificate is expiring within 30 days. Please refer to the Oracle Key Vault Administrator's Guide.` |
| Invalid HSM Configuration | `HSM configuration error. Please refer to the "HSM Alert" section in the Oracle Key Vault Integration with Hardware Security Module` |

## 12.6 Oracle Key Vault Auditing

Oracle Key Vault records and time-stamps all endpoint and user activity, with details on who initiated which action, with what keys and tokens, and the result of the action.

- About Auditing in Oracle Key Vault (page 12-27)
- Configuring Syslog to Store Audit Records (page 12-28)
- View Audit Records (page 12-29)
- Export or Delete Audit Records (page 12-29)
- Audit Consolidation with Audit Vault and Database Firewall (page 12-30)

### 12.6.1 About Auditing in Oracle Key Vault

Oracle Key Vault records all endpoint and user activity including endpoint groups and user groups from endpoint enrollment and user password reset, to the management of keys and wallets, and changes to system settings and SNMP credentials. In addition, it records the success or failure of each action.

This recording of comprehensive system activity is presented in an audit trail, which, while visible to all users, can only be managed by a user with the Audit Manager role. This user alone has the privilege to export or delete audit records up to a given date.

Auditing in Oracle KeyVault is enabled by default.

A user with the Audit Manager role can see all the audit records and manage them. Other users can see only those audit records, which pertain to security objects that they have created, or have been granted access to.

You can export audit records to view system activity off line. After exporting the records, you can delete them from the system to free up resources.

**See Also:**

"Overview of Administrative Roles (page 2-9)" for more on Audit Manager privileges

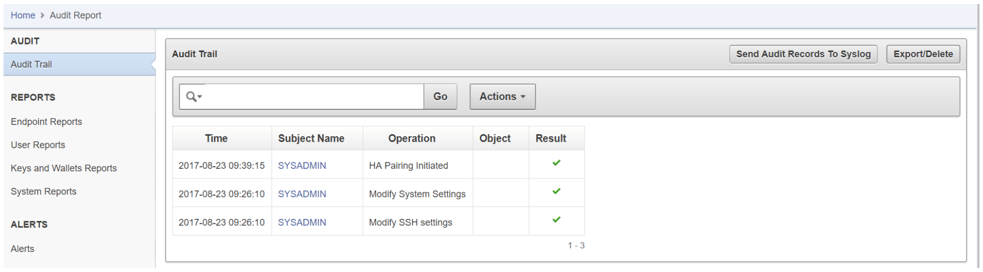
## 12.6.2 Configuring Syslog to Store Audit Records

You can configure the Oracle Key Vault system log to store audit records if the System Administrator has enabled this functionality.

To configure the Oracle Key Vault system log to store audit records:

1. Log in to the Oracle Key Vault management console as the Audit Manager.
2. Click the **Reports** tab. The **Audit Trail** page is displayed.

**Figure 12-11 Audit Trail Page**



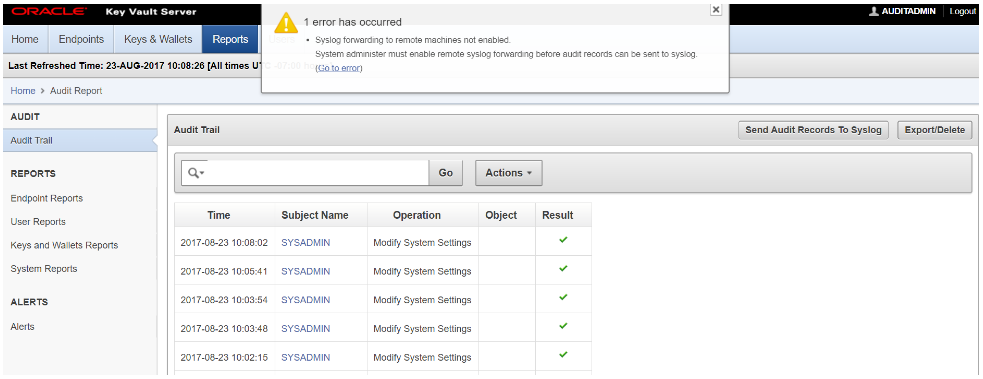
| Time | Subject Name | Operation | Object | Result |
|---|---|---|---|---|
| 2017-08-23 09:39:15 | SYSADMIN | HA Pairing Initiated | | ✓ |
| 2017-08-23 09:26:10 | SYSADMIN | Modify System Settings | | ✓ |
| 2017-08-23 09:26:10 | SYSADMIN | Modify SSH settings | | ✓ |

1 - 3

3. Click **Send Audit Records To Syslog**.

   If Syslog is not configured, an error message “**Syslog forwarding to remote machines not enabled.**” is displayed.

**Figure 12-12 Error Message-Remote Syslog is Not Enabled**



Last Refreshed Time: 23-AUG-2017 10:08:26 [All times U

1 error has occurred

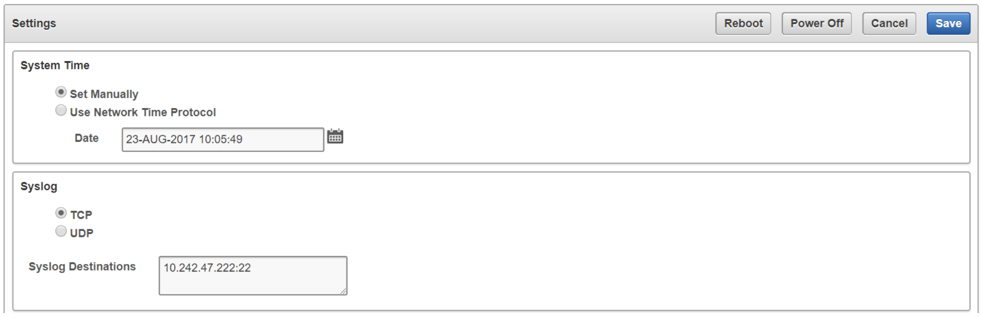
- Syslog forwarding to remote machines not enabled.
  System administer must enable remote syslog forwarding before audit records can be sent to syslog.
  (Go to error)

| Time | Subject Name | Operation | Object | Result |
|---|---|---|---|---|
| 2017-08-23 10:08:02 | SYSADMIN | Modify System Settings | | ✓ |
| 2017-08-23 10:05:41 | SYSADMIN | Modify System Settings | | ✓ |
| 2017-08-23 10:03:54 | SYSADMIN | Modify System Settings | | ✓ |
| 2017-08-23 10:03:48 | SYSADMIN | Modify System Settings | | ✓ |
| 2017-08-23 10:02:15 | SYSADMIN | Modify System Settings | | ✓ |

For more information about configuring Syslog, see Configure System Settings (page 12-13).

4. If Syslog is configured, the **Settings** page is displayed. In the **Syslog** section, do the following:
   a. Select the protocol to use to transfer Syslog files: **TCP** or **UDP**.
   b. Enter the IP address of the remote system where Syslog files are stored.

**Figure 12-13    Settings Page**



System Time

● Set Manually
○ Use Network Time Protocol

Date 23-AUG-2017 10:05:49

Syslog

● TCP
○ UDP

Syslog Destinations 10.242.47.222:22

5. Click **Save**.

The remote Syslog stores audit records.

## 12.6.3 View Audit Records

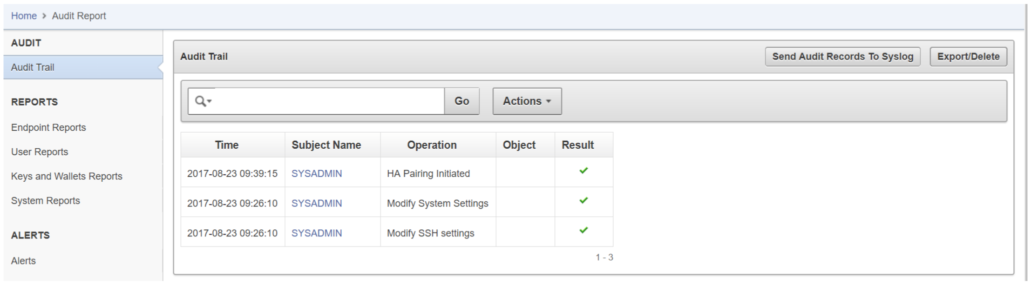
The reports page shows the **Audit Trail** by default. The **Audit Trail** page lists all system activity with details on who (**Subject Name**) performed what (**Operation**), when (**Time**), using what (**Object**), and the result of the action (**Result**).

To view the audit trail follow these steps:

1. Log in to the Oracle Key Vault management console as a user who has the Audit Manager role.
2. Click the **Reports** tab.

   The **Audit Trail** page appears.

**Figure 12-14    Audit Trail Page**



| Time | Subject Name | Operation | Object | Result |
|---|---|---|---|---|
| 2017-08-23 09:39:15 | SYSADMIN | HA Pairing Initiated | | ✓ |
| 2017-08-23 09:26:10 | SYSADMIN | Modify System Settings | | ✓ |
| 2017-08-23 09:26:10 | SYSADMIN | Modify SSH settings | | ✓ |

1 - 3

## 12.6.4 Export or Delete Audit Records

A user with the Audit Manger role may export or delete the audit trail as needed. Audit records are exported in a `.csv` file that can be downloaded to the user's local system. The `.csv` file contains the same details found in the **Audit Trail** on the **Reports** page.

Note, that the timestamp in the `.csv file` will reflect the time zone of the particular Key Vault server, whose records were exported.

To export or delete the audit trail:

1. Log in to the Oracle Key Vault management console as a user who has the Audit Manager role.

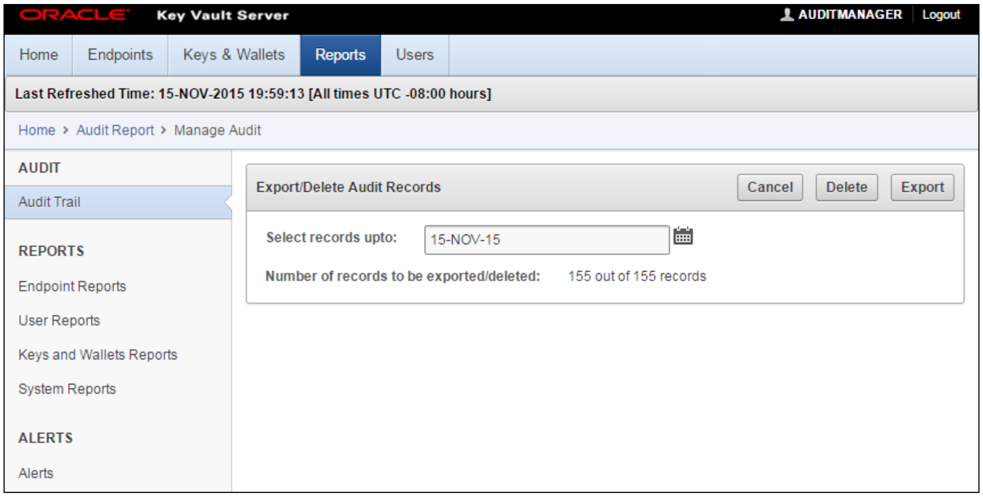
2. Click the **Reports** tab.

   The **Audit Trail** is displayed.

3. Click **Export/Delete** on the top right.

   The **Export/Delete Audit Records** page appears.

**Figure 12-15 Export/Delete Audit Records Page**



Last Refreshed Time: 15-NOV-2015 19:59:13 [All times UTC -08:00 hours]

Select records upto: 15-NOV-15

Number of records to be exported/deleted: 155 out of 155 records

4. Select the date by clicking the calendar icon.

   The number of records appears.

5. Click **Export** to download the audit records in `.csv` file format to a local folder.

   After you export the records you can delete them from Key Vault to free up resources.

6. Click **Delete** to remove audit records.

   A confirmation message appears, warning you about permanent loss of audit records.

7. Click **OK** to delete and **Cancel** to stop.

## 12.6.5 Audit Consolidation with Audit Vault and Database Firewall

Oracle Key Vault audit data can be forwarded to Audit Vault and Database Firewall (AVDF) for audit consolidation.

To enable audit consolidation with AVDF:

1. Log in to the Oracle Key Vault management console as a user who has the System Administrator role.

2. Click the **System** tab, then **System Settings**.

   The **Settings** page appears.

3. Click the box to the right of **Enable** in the **Oracle Audit Vault Integration** pane. Two password fields appear: **Enter Password** and **Reenter Password**.

4. Enter the password and confirm it.

You must keep this password in a safe place. You will need it when you create a secured target on the AVDF side.

**See Also:**

- Figure 12-2 (page 12-14) for Key Vault System Settings
- "Audit Vault Database Firewall Integration Instructions (page B-2)"

# 12.7 Oracle Key Vault Reports

Oracle Key Vault collects statistical information on system activity, the expiration of certificates, keys, and passwords, entitlement status, and metadata in four report categories: endpoints, users, keys, and system.

- About Oracle Key Vault Reports (page 12-31)
- View Reports for Endpoint, User, Keys and Wallets, and System (page 12-32)

## 12.7.1 About Oracle Key Vault Reports

Oracle Key Vault provides four types of reports for endpoints, users, keys and wallets, and system.

The four report types and their description are:

- Endpoint reports contain details of all endpoint and endpoint group activity, certificate and password expiration, and access privileges.
- User reports contain details of all user and user group activity, their certificate and password expiration, and access privileges.
- Keys and wallets reports list the access privileges granted to all keys and wallets, and the details of TDE master keys managed by Key Vault.
- System reports contain a history of system backups taken and scheduled, details of remote restoration points, and RESTful API usage.

The Audit Manager can view all reports. The Key Administrator can view User reports and Keys and Wallets reports. Users with System Administrator privileges can view Endpoint, User, and System reports.

**See Also:**

"Oracle Key Vault Auditing (page 12-27)" for more on audit reports

## 12.7.2 View Reports for Endpoint, User, Keys and Wallets, and System

The **Reports** page list the four report types in the left side bar under the heading **REPORTS**.

To view the reports for endpoints, users, keys and wallets, and system:

1. Log in to the Oracle Key Vault management console as a user who has the Audit Manager role.
2. Click the **Reports** tab to get to the **Reports** page.
3. The **REPORTS** heading displays four reports: **Endpoint Reports**, **User Reports**, **Keys and Wallets Reports**, **System Reports**.

- View Endpoint Reports (page 12-32)
- View User Reports (page 12-34)
- View Keys and Wallets Reports (page 12-35)
- View System Reports (page 12-35)

> **See Also:**
>
> Figure 12-14 (page 12-29) shows the Key Vault Audit Trail

### 12.7.2.1 View Endpoint Reports

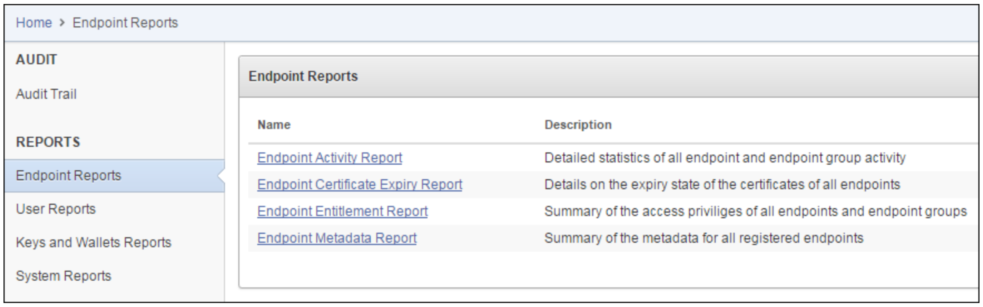
Key Vault offers four endpoint reports for **Endpoint Activity**, **Endpoint Certificate Expiry**, **Endpoint Entitlement**, and **Endpoint Metadata**.

To view endpoint reports:

1. Log in to the Oracle Key Vault management console as a user who has the Audit Manager role.
2. Click the **Reports** tab to get to the **Reports** page.
3. Click **Endpoint Reports** under **Reports** in the left sidebar.

   The **Endpoint Reports** page appears displaying four endpoint report types.

### Figure 12-16 Endpoint Reports



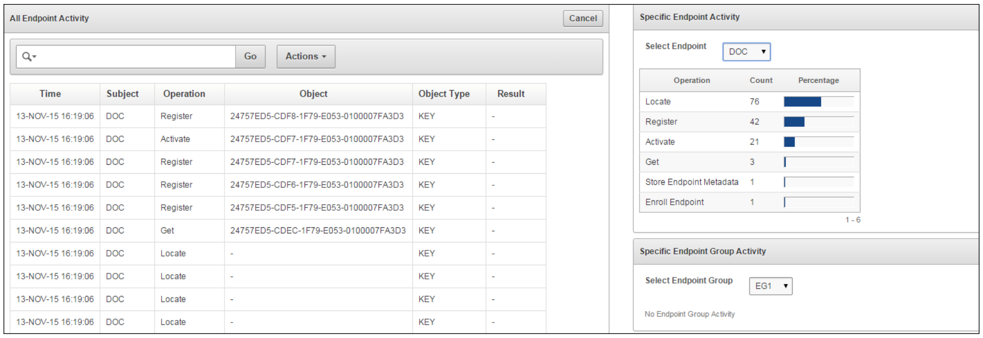
Endpoint Reports

| Name | Description |
|---|---|
| Endpoint Activity Report | Detailed statistics of all endpoint and endpoint group activity |
| Endpoint Certificate Expiry Report | Details on the expiry state of the certificates of all endpoints |
| Endpoint Entitlement Report | Summary of the access privileges of all endpoints and endpoint groups |
| Endpoint Metadata Report | Summary of the metadata for all registered endpoints |

Click the link under **Name** to view the report you want.

4. Click **Endpoint Activity Report** to view the corresponding report.

### Figure 12-17 Endpoint Activity Report



| Time | Subject | Operation | Object | Object Type | Result |
|---|---|---|---|---|---|
| 13-NOV-15 16:19:06 | DOC | Register | 24757ED5-CDF8-1F79-E053-0100007FA3D3 | KEY | - |
| 13-NOV-15 16:19:06 | DOC | Activate | 24757ED5-CDF7-1F79-E053-0100007FA3D3 | KEY | - |
| 13-NOV-15 16:19:06 | DOC | Register | 24757ED5-CDF7-1F79-E053-0100007FA3D3 | KEY | - |
| 13-NOV-15 16:19:06 | DOC | Register | 24757ED5-CDF6-1F79-E053-0100007FA3D3 | KEY | - |
| 13-NOV-15 16:19:06 | DOC | Register | 24757ED5-CDF5-1F79-E053-0100007FA3D3 | KEY | - |
| 13-NOV-15 16:19:06 | DOC | Get | 24757ED5-CDEC-1F79-E053-0100007FA3D3 | KEY | - |
| 13-NOV-15 16:19:06 | DOC | Locate | - | KEY | - |
| 13-NOV-15 16:19:06 | DOC | Locate | - | KEY | - |
| 13-NOV-15 16:19:06 | DOC | Locate | - | KEY | - |
| 13-NOV-15 16:19:06 | DOC | Locate | - | KEY | - |

Specific Endpoint Activity

Select Endpoint: DOC

| Operation | Count | Percentage |
|---|---|---|
| Locate | 76 | |
| Register | 42 | |
| Activate | 21 | |
| Get | 3 | |
| Store Endpoint Metadata | 1 | |
| Enroll Endpoint | 1 | |

1 - 6

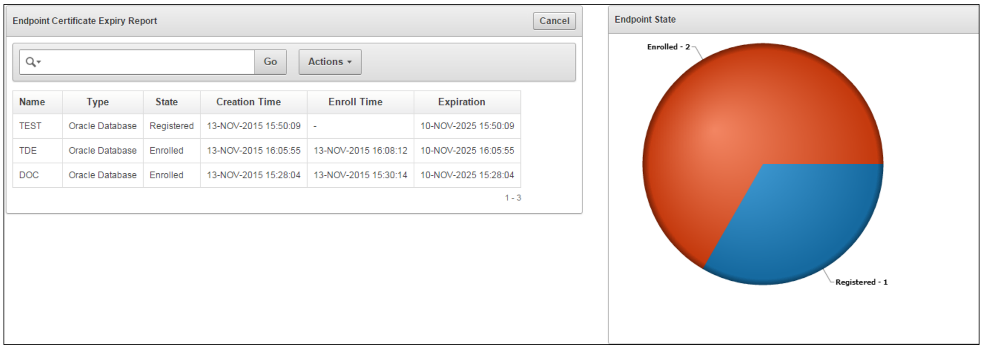
Specific Endpoint Group Activity

Select Endpoint Group: EG1

No Endpoint Group Activity

5. Click **Endpoint Certificate Expiry Report** to view the corresponding report.

### Figure 12-18 Endpoint Certificate Expiry Report

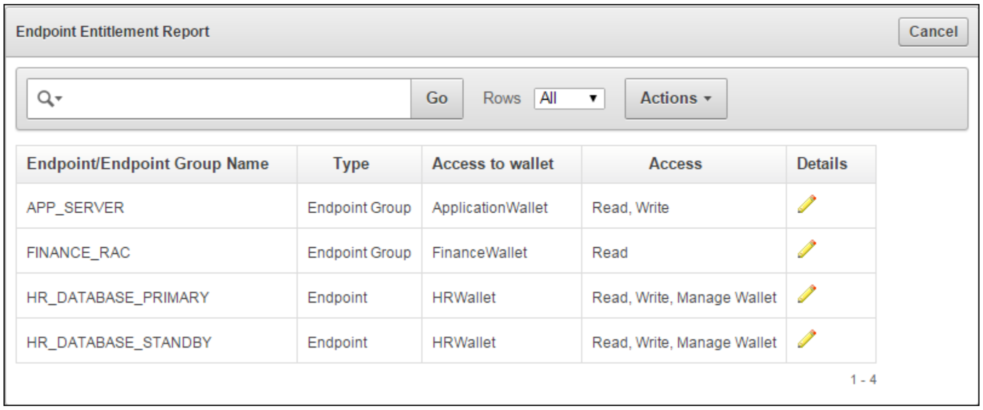


| Name | Type | State | Creation Time | Enroll Time | Expiration |
|---|---|---|---|---|---|
| TEST | Oracle Database | Registered | 13-NOV-2015 15:50:09 | - | 10-NOV-2025 15:50:09 |
| TDE | Oracle Database | Enrolled | 13-NOV-2015 16:05:55 | 13-NOV-2015 16:08:12 | 10-NOV-2025 16:05:55 |
| DOC | Oracle Database | Enrolled | 13-NOV-2015 15:28:04 | 13-NOV-2015 15:30:14 | 10-NOV-2025 15:28:04 |

1 - 3

6. Click **Endpoint Entitlement Report** to view the corresponding report.

**Figure 12-19    Endpoint Entitlement Report**

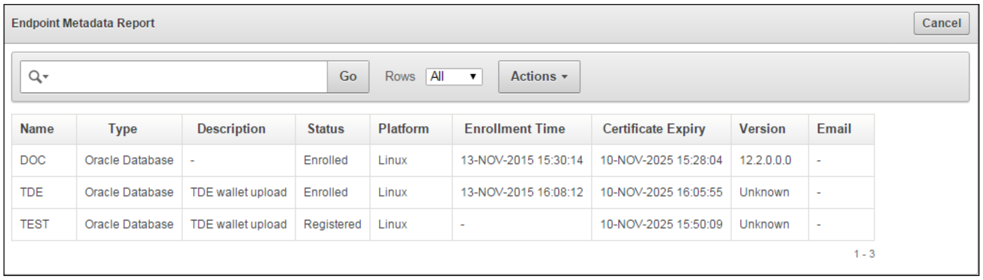


| Endpoint/Endpoint Group Name | Type | Access to wallet | Access | Details |
|---|---|---|---|---|
| APP_SERVER | Endpoint Group | ApplicationWallet | Read, Write | ✎ |
| FINANCE_RAC | Endpoint Group | FinanceWallet | Read | ✎ |
| HR_DATABASE_PRIMARY | Endpoint | HRWallet | Read, Write, Manage Wallet | ✎ |
| HR_DATABASE_STANDBY | Endpoint | HRWallet | Read, Write, Manage Wallet | ✎ |

1 - 4

7. Click **Endpoint Metadata Report** to view the corresponding report.

**Figure 12-20    Endpoint Metadata Report**



| Name | Type | Description | Status | Platform | Enrollment Time | Certificate Expiry | Version | Email |
|---|---|---|---|---|---|---|---|---|
| DOC | Oracle Database | - | Enrolled | Linux | 13-NOV-2015 15:30:14 | 10-NOV-2025 15:28:04 | 12.2.0.0.0 | - |
| TDE | Oracle Database | TDE wallet upload | Enrolled | Linux | 13-NOV-2015 16:08:12 | 10-NOV-2025 16:05:55 | Unknown | - |
| TEST | Oracle Database | TDE wallet upload | Registered | Linux | - | 10-NOV-2025 15:50:09 | Unknown | - |

1 - 3

## 12.7.2.2 View User Reports

Key Vault offers four user reports for **User Activity**, **User Entitlement**, **User Expiry**, and **User Failed Login**.

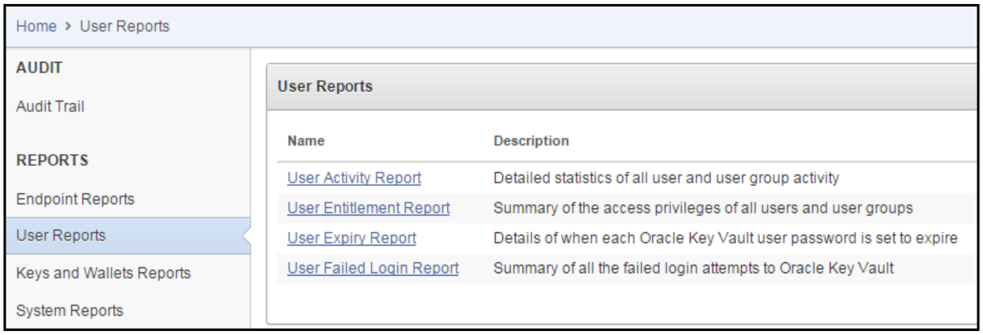
To view user reports:

1. Log in to the Oracle Key Vault management console as a user who has the Audit Manager role.
2. Click the **Reports** tab.
3. Click **User Reports** to see user specific reports.

   The **User Reports** page appears displaying the four types of user reports.

   Click the report name to see the corresponding user report.

**Figure 12-21 User Reports**



| Name | Description |
|---|---|
| User Activity Report | Detailed statistics of all user and user group activity |
| User Entitlement Report | Summary of the access privileges of all users and user groups |
| User Expiry Report | Details of when each Oracle Key Vault user password is set to expire |
| User Failed Login Report | Summary of all the failed login attempts to Oracle Key Vault |

## 12.7.2.3 View Keys and Wallets Reports

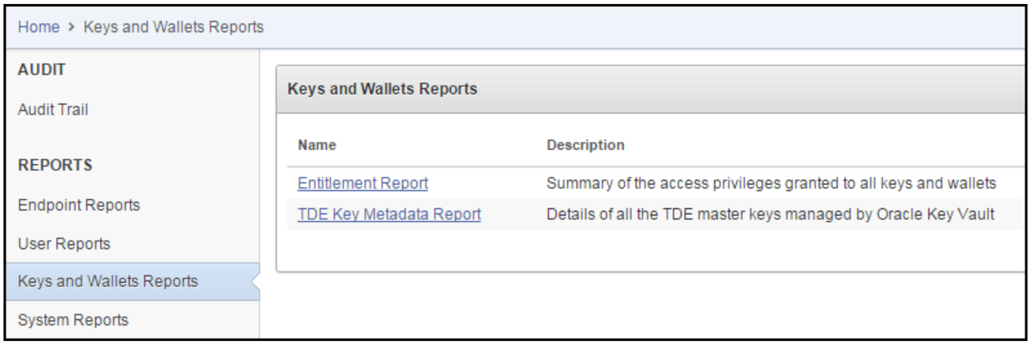
Key Vault offers two reports for keys and wallets: **Entitlement** and **TDE Key Metadata**.

To view reports for keys and wallets:

1. Log in to the Oracle Key Vault management console as a user who has the Audit Manager role.
2. Click the **Reports** tab.
3. Click **Keys and Wallets Reports** under the **REPORTS** heading.

   The **Keys and Wallets Reports** page appears displaying the reports available. Click the report name to see the corresponding report.

**Figure 12-22 Keys and Wallets Reports**



| Name | Description |
|---|---|
| Entitlement Report | Summary of the access privileges granted to all keys and wallets |
| TDE Key Metadata Report | Details of all the TDE master keys managed by Oracle Key Vault |

## 12.7.2.4 View System Reports

Key Vault offers three system reports for **Backup History**, **Backup Restoration Catalog**, and **RESTful API Usage**.

To view system reports:

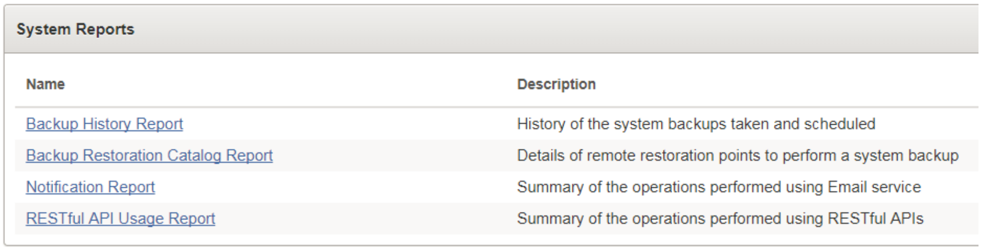
1. Log in to the Oracle Key Vault management console as a user who has the Audit Manager role.
2. Click the **Reports** tab.

3. Click **System Reports** under the **REPORTS** heading.

   The **System Reports** page appears displaying the system reports available.

   Click the report type to see the corresponding system report.

**Figure 12-23 System Reports**



System Reports

| Name | Description |
|---|---|
| Backup History Report | History of the system backups taken and scheduled |
| Backup Restoration Catalog Report | Details of remote restoration points to perform a system backup |
| Notification Report | Summary of the operations performed using Email service |
| RESTful API Usage Report | Summary of the operations performed using RESTful APIs |

# 12.8 Upgrade Oracle Key Vault Server Software

When you upgrade the Key Vault server software appliance, it is recommended that you also upgrade the endpoint software to get the latest enhancements. However, the previous version of endpoint software will continue to function with the upgraded Oracle Key Vault Server.

- How an Oracle Key Vault Server Software Upgrade Works (page 12-36)
- Step 1: Backup the Server before Upgrade (page 12-36)
- Step 2: Pre-Upgrade Tasks for Release 12.2.0.0.0 (page 12-37)
- Step 3: Upgrade the Oracle Key Vault Server or Server Pair (page 12-37)
- Step 4: Upgrade Endpoint Software (page 12-40)
- Step 5: Backup Just Upgraded Server (page 12-41)

## 12.8.1 How an Oracle Key Vault Server Software Upgrade Works

You must upgrade in the step order shown: first perform a full backup of Key Vault, upgrade the Key Vault server or server pair in the case of a high availability deployment, the endpoint software, and last, perform another full backup of the upgraded server. Note that upgrading requires a reboot of the Oracle Key Vault appliance.

The Oracle Key Vault server is not available to endpoints for a limited duration during the upgrade. You can enable the persistent cache feature to enable endpoints to continue operation during the upgrade process.

## 12.8.2 Step 1: Backup the Server before Upgrade

Before you upgrade the Key Vault server we recommend that you backup the server you are upgrading. This step ensures that you can recover in case the upgrade fails unexpectedly.

**Caution:**

Do not skip this step. Back up the server before you perform the upgrade so your data is safe and recoverable.

## 12.8.3 Step 2: Pre-Upgrade Tasks for Release 12.2.0.0.0

To ensure a smooth upgrade to Oracle Key Vault 12.2.0.0.0, the following steps are recommended:

- Ensure that the minimum disk space requirement for an upgrade is met.
- Ensure that no full or incremental backup jobs are running. Delete all scheduled full or incremental backup jobs before the upgrade.
- Plan for downtime according to the following specifications:

| Oracle Key Vault Usage | Downtime required |
|---|---|
| Wallet upload or download | NO |
| Java Keystore upload or download | NO |
| Transparent Data Encryption (TDE) direct connect | YES |
| Primary Server Upgrade in a high availability deployment | YES |

  If an online master key (formerly known as TDE direct connect) is used with Oracle Key Vault, then plan for a downtime of 15 minutes during the Oracle Database endpoint software upgrades. Database endpoints can be upgraded in parallel to reduce total downtime.

  For a primary server upgrade in a high availability deployment plan for a downtime of 1 hour.
- If the Oracle Key Vault system has a syslog destination configured, ensure that the remote syslog destination is reachable from the Oracle Key Vault system, and that logs are being correctly forwarded. If the remote syslog destination is not reachable from the Oracle Key Vault system, then the upgrade process can become much slower than normal.

**Related Topics**

- Configuring Syslog to Store Audit Records (page 12-28)

## 12.8.4 Step 3: Upgrade the Oracle Key Vault Server or Server Pair

How you upgrade the Oracle Key Vault server depends on whether you are using a standalone environment or a high availability deployment.

- About the Upgrade of an Oracle Key Vault Server or Server Pair (page 12-38)
- Upgrade a Standalone Key Vault Server (page 12-38)
- Upgrade a Pair of Key Vault Servers in a High Availability Deployment (page 12-39)

### 12.8.4.1 About the Upgrade of an Oracle Key Vault Server or Server Pair

Oracle Key Vault may be deployed as a standalone appliance in test and development environments or in a high availability configuration in production environments. In a standalone deployment you must upgrade a single Key Vault server, but in a high availability deployment you must upgrade both primary and standby Key Vault servers. Note that persistent caching enables endpoints to continue to be operational during the upgrade process.

> **Note:**
>
> - Ensure that the system you are upgrading has 8 GB memory. From release 12.2.0.2.0 and onwards you must have 8 GB memory. In a high availability deployment both primary and standby servers must have 8 GB system memory.
> - If you are upgrading from a system with 4 GB memory, first add an additional 4 GB memory to the system before upgrading.

### 12.8.4.2 Upgrade a Standalone Key Vault Server

This procedure is for upgrading a single Key Vault server in a standalone deployment, the most typical deployment in test and development environments.

To upgrade to Oracle Key Vault 12.2.0.0.0:

1. Ensure that you have backed up the server you are upgrading so your data is safe and recoverable. Do not proceed without completing this step.
2. Ensure that SSH access is enabled by logging into the management console and checking **System Settings ->Network Services ->SSH Access**.
3. Ensure you have enough space in the destination directory for the upgrade ISO.
4. Log in to the Oracle Key Vault Server through SSH as user `support`, then switch user (`su`) to `root`.
5. Copy the upgrade ISO file to the destination directory using **Secure Copy Protocol** or other secure transmission method:

   ```
   scp remote_host:remote_path/okv-upgrade-disc-12.2.0.0.0.iso /var/lib/oracle/
   <destination_directory where you are copying the iso file to>
   ```

   Where:

   `remote_host` is the IP address of the computer containing the ISO upgrade file

   `remote_path` is the directory of the ISO upgrade file
6. Make the upgrade accessible using the `mount` command:

   ```
   root# /bin/mount -oloop,ro /var/lib/oracle/okv-upgrade-disc-12.2.0.0.0.iso /
   images
   ```
7. Clear the cache using this command:

   ```
   root# yum -c/images/upgrade.repo clean all
   ```

8. Apply the upgrade with this command:

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

   If the system is successfully upgraded, then the command will display the following message:

   ```
   Remove media and reboot now to fully apply changes.
   ```

   If you see an error message, then check the log file `/var/log/messages` for additional information.

9. Reboot the Key Vault server by running the command:

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

   On first reboot after upgrade, the system will apply changes. This can take up to 45 minutes. Do not shut down the system during this time.

   The upgrade is completed, when the screen with heading: Oracle Key Vault Server 12.2.0.0.0 appears. The revision should reflect the upgraded release. Below the heading appears the menu item **Display Appliance Info**. Select **Display Appliance Info** and press the **Enter** key to see the IP address settings for the appliance.

10. Log in to the Oracle Key Vault management console UI as System Administrator. Select the **System** tab, and then **Status**. Verify that the version displayed is 12.2.0.0.0.

### 12.8.4.3 Upgrade a Pair of Key Vault Servers in a High Availability Deployment

> **Note:**
>
> - Allocate 1 hour to upgrade the primary server after upgrading the standby. You must upgrade standby and primary servers in one session with as little time between the standby and primary upgrade. Note that the upgrade time is approximate and a function of the volume of data stored and managed by Oracle Key Vault. For large volumes of data the upgrade time may be greater than an hour.
> - While the upgrade is in progress, do not change any settings or perform any other operations that are not part of the upgrade instructions below.
> - Upgrade the Oracle Key Vault Server during a planned maintenance window because the upgrade process requires the endpoints to be shutdown during the upgrade.
> - Ensure that both the primary and standby systems have 8 GB memory.
> - With persistent cache enabled endpoints will continue to be operational during the upgrade process.

To upgrade a pair of Oracle Key Vault Servers configured for high availability:

1. Ensure that you have backed up the server you are upgrading so your data is safe and recoverable. Do not proceed without completing this step.
2. First upgrade the standby server while the primary server is running.

Follow **Step 2** through to **Step 10** of the standalone mode upgrade process.

3. Ensure that the upgraded standby Oracle Key Vault Server is restarted and running.
4. Upgrade the primary Oracle Key Vault Server following **Steps 1-10** of the standalone mode upgrade.

   After both the standby and primary Key Vault servers are upgraded, the two servers will automatically sync up.
5. Log in to the Oracle Key Vault management console as System Administrator. Select the **System** tab, and then **Status**. Verify that the **Version** field displays the new software version 12.2.0.0.0.

## 12.8.5 Step 4: Upgrade Endpoint Software

To upgrade the endpoint software:

1. Ensure that you have upgraded the Key Vault server(s) outlined in **Step 1**. If you are upgrading the endpoint software for an Oracle database configured for direct-connect, then shutdown the database.
2. Download the endpoint software (`okvclient.jar`) for your platform from the Oracle Key Vault Server as follows:
   a. Go to the Oracle Key Vault management console login screen.
   b. Click the **Endpoint Enrollment and Software Download** link.
   c. Go to the **Download Endpoint Software** section, and select the appropriate platform from the drop down list.
   d. Click the **Download** button.
3. Identify the path to your existing endpoint installation which is about to be upgraded. For example, `/home/oracle/okvutil`
4. Install the endpoint software by executing the following command:

   `java -jar okvclient.jar -d` *`<path to the existing endpoint directory>`*

   For example, `java -jar okvclient.jar -d /home/oracle/okvutil`
5. On UNIX platforms, run `root.sh` from the `bin` directory of endpoint installation directory to copy the latest `liborapkcs.so` file for Oracle Database endpoints. On Windows platforms, run `root.bat` from the `bin` directory of endpoint installation directory to copy the latest `liborapkcs.dll` file for Oracle Database endpoints. This step is needed only for online TDE master key management by Oracle Key Vault. For example,

   ```
   $ sudo ./$OKV_HOME/bin/root.sh
   ```

   ```
   bin\root.bat
   ```

   Or,

   ```
   $ su -
   # bin/root.sh
   ```

   On Windows platforms, you are prompted for the version of the RDBMS in use when you execute root.bat.
6. Restart the endpoint if it was shutdown in **Step 1**.

## 12.8.6 Step 5: Backup Just Upgraded Server

You must perform the following tasks after completing a successful upgrade:

- Take a full backup of the upgraded Oracle Key Vault Server Database to a new remote destination. Avoid using the old backup destination for the new backups.
- Schedule a new periodic incremental backup to the new destination defined in the step above.
- Password hashing has been upgraded to a more secure standard starting at version 12.1.0.2. This change affects the operating system passwords, support and root. You must change Oracle Key Vault administrative passwords after the upgrade to take advantage of the more secure hash if you have not done so already.

  Password hashing is applicable only when upgrading from Oracle Key Vault 12.1.0.0.0 to Oracle Key Vault 12.2.0.0.0 directly or from Oracle Key Vault 12.1.0.1.0 to Oracle Key Vault 12.2.0.0.0 directly. This fix was included in Oracle Key Vault 12.1.0.2.0 and onwards.

# 13 Managing Certificates

In addition to Oracle Key Vault-generated certificates, you can manage third-party certificates.

## 13.1 Rotating Certificates

You can rotate Oracle Key Vault-generated certificates by using the Oracle Key Vault management console.

**Related Topics**

- Oracle Key Vault Server Certificate Rotation (page xxiv)

### 13.1.1 About Rotating Certificates

The certificate rotation process captures all certificates in the Oracle Key Vault server. It does not capture third-party certificates.

A certificate in Oracle Key Vault lasts 730 days. If you do not rotate the certificate (both server and endpoint certificates), then the endpoints that use the certificate cannot connect to the Oracle Key Vault server. When this happens, you must re-enroll the endpoint. To avoid this scenario, you can configure an alert to remind you to rotate the certificate before the 730-day limit is up. The rotation process handles the rotation for all certificates in one operation. You can find how much time the Oracle Key Vault server certificate has before it expires by checking the **OKV Server Certificate Expiration** setting on the Configure Alerts page in the Oracle Key Vault management console. To find the expiry time of the endpoints' certificates, you must to navigate to the Endpoints page and check the **Certificate Expires** field.

If you have a high availability configuration, then Oracle Key Vault automatically synchronizes the certificates in both systems. You do not have to perform any extra configuration.

**Related Topics**

- Configuring Alerts (page 12-24)

## 13.1.2 Advice for Managing Certificate Rotations

Oracle Key Vault provides advice on the best ways to rotate certificates.

- In a primary-standby configuration, do not perform certificate rotation if the primary database is in read-only restricted mode. Only initiate a certificate rotation when both servers in the configuration are active and synchronized with each other.
- If you are performing certificate rotation on a system that was upgraded from a previous release, ensure that you upgrade the endpoints as well. Endpoints whose software has not been upgraded will not receive updated credentials.
- You cannot perform a certificate rotation while a backup operation or a restore operation is in progress.
- Before performing a certificate rotation, back up the Oracle Key Vault system.
- In order for the certificate rotation process to fully complete, you must delete and re-enroll all endpoints that are *not* in the Enrolled state. If you no longer need the endpoint, then you only need to delete it.
- If a given endpoint does not receive its rotated certificates due to network or other issues, or is in the "Suspended" state, re-enenroll the endpoint, or delete it if you no longer need it. This will allow the certificate rotation process to continue on to completion. You can find the current certificate rotation status by going to the Endpoints page and looking for Common Name of Certificate Issuer.

## 13.1.3 Rotating All Certificates

You can use the Oracle Key Vault management console to rotate certificates.

Before you begin the rotation process, check the endpoint software version and ensure that it uses the current version of the Oracle Key Vault release 12.2 software. You must use the current version of Oracle Key Vault release 12.2 if you want to use the automatic rotation feature. If you are using an earlier Oracle Key Vault release 12.2 patch, then the endpoint cannot receive the updated credentials.

1. Log in to Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **System** tab.
3. Select **Manage System Certificate**.
4. In the Manage System Certificate page, click **Generate System Certificate**.
5. In the confirmation dialog box, select **OK**.

   This creates a new CA certificate, but does not enable it. At this stage, endpoints can still use their old credentials to connect using the previous certificate. The Old Certificate area shows the details of the currently active CA. The New Certificate area shows that the certificate has been rotated and displays its common name. If you want to cancel the rotation process, click **Abort** to cancel the process and clean up the new CA directory that was generated.
6. Click **Activate Certificate**.

   Clicking **Activate Certificate** begins the process of putting the new Oracle Key Vault CA into use. When it completes, the endpoints should be able to connect to the Oracle Key Vault server using either the new or the old Oracle Key Vault CA.

This process may take a few minutes to complete. You cannot cancel the rotation process after you click **Activate Certificate**.

7. In the confirmation dialog box, click **OK**.

   A message appears saying that the automatic certificate update of the endpoints is in progress. In the background, Oracle Key Vault starts regenerating certificates for its endpoints, for a few endpoints at a time (so that not all endpoints are updated at once). To check if the credentials for an endpoint have been updated, click the **Check Endpoint Progress** button. The Endpoints page appears. If, for a given endpoint, the **Common Name of Certificate Issuer** field shows the common name of the old CA, the new credentials have not yet been generated. However, if, for existing endpoints, the field shows **Updating to Current Certificate Issuer**, the process has begun. Endpoints should be able to retrieve updated credentials a few minutes after this status has changed.

   After the new credentials have been generated for a given endpoint, when the endpoint next makes a connection to the Oracle Key Vault server, the new credentials for the certificate are sent to the endpoint. After an endpoint has received its updated credentials from the Oracle Key Vault server, it must try to connect to the Oracle Key Vault server to let the server know that it has successfully received the credentials. When the endpoint succeeds in this, the value in the **Common Name of Certificate Issuer** field for that endpoint on the Endpoints page should reflect the common name of the new Oracle Key Vault CA certificate.

After all the endpoints have been updated to using the new CA, the Oracle Key Vault server begins the process of fully rotating its own server certificates in the background. The process can be deemed to be complete when the Manage Server Certificate page no longer shows two certificates listed, but only a single one reflecting the new CA certificate. The **OKV Server Expiration Date** field in the System Settings page should reflect the expiration time of the new CA certificate as well.
After you complete the rotation, you should configure an alert for the next time the new certificate should be rotated. To configure the alert, in the Configure Alerts page, select the check box after **OKV Server Certificate Expiration**.

**Related Topics**

- Configuring Alerts (page 12-24)

## 13.1.4 Checking the Certificate Rotation Status

You can use the Oracle Key Vault management console to check the status of a certificate rotation.

1. Log in to Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **Endpoints** tab.
3. Select **Endpoints**.

   On the Endpoints page, you can see a status of the rotation process (**Updating to current certificate issuer**) in the Endpoints page. When it is complete, it will show the name of the common name of the new Oracle Key Vault CA.

   If there are errors with the certificate rotation of an endpoint, then Oracle recommends that you re-enroll the endpoint.

# 13.2 Third Party Certificates

Oracle Key Vault enables you to install a certificate signed by a third-party CA for more secure connections. Users can upload certificates signed by a third-party Certificate Authority (CA) to Key Vault to prove their identity, encrypt the communication channel, and protect the data that is exchanged

To install a third-party certificate you must generate a certificate request, get it signed by a Certificate Authority (CA), and upload the signed certificate back to Oracle Key Vault.

- Download Certificate Request (page 13-4)
- Getting Certificate Signed (page 13-5)
- Upload Signed Certificate to Oracle Key Vault (page 13-5)
- Notes on Using Third-Party Certificates (page 13-5)

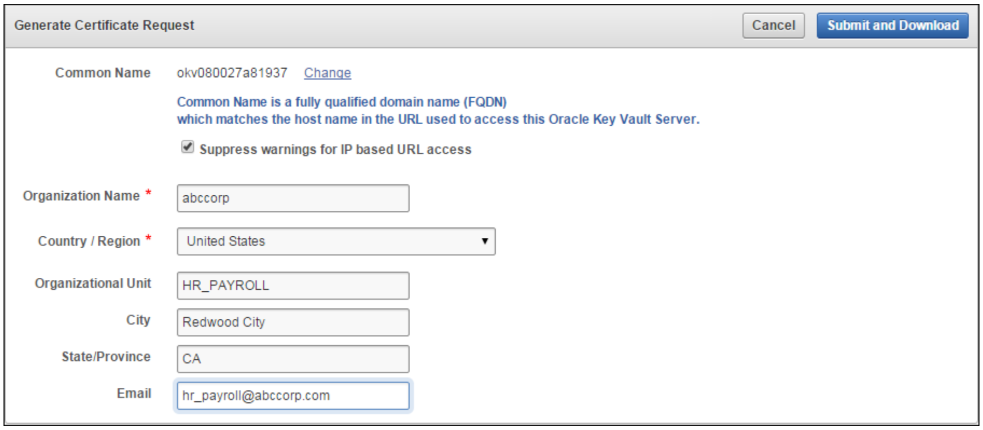
## 13.2.1 Download Certificate Request

When you request the certificate, you have the option to suppress warning messages from the browser, that appear when the browser detects a mismatch between the attributes of the server certificate and the attributes of the login session to the Oracle Key Vault management console. See **Step 4** of the download certificate request to do this.

To generate an Oracle Key Vault certificate request:

1. Click the **System** tab, then **Console Certificate** from the **System** menu to bring up the **Console Certificate** page.
2. Click **Generate Certificate Request** on the top right to bring up the **Generate Certificate Request** page.

**Figure 13-1 Generate Certificate Request Page**



Common Name okv080027a81937 Change

Common Name is a fully qualified domain name (FQDN)
which matches the host name in the URL used to access this Oracle Key Vault Server.

☑ Suppress warnings for IP based URL access

| | |
|---|---|
| Organization Name * | abccorp |
| Country / Region * | United States ▾ |
| Organizational Unit | HR_PAYROLL |
| City | Redwood City |
| State/Province | CA |
| Email | hr_payroll@abccorp.com |

3. The first field on this page, **Common Name**, is automatically populated with the host name of the Oracle Key Vault server. If you want to change this, click **Change**. This will bring you to the **System Settings** page where you can change the host name in the **Network** pane.

4. Check the box to the left of text **Suppress warnings for IP based URL access** if you want to suppress browser warnings for server IP address changes.
5. Enter the required fields marked with an asterisk, **Organization Name** and **Country/Region**. You must enter values for these fields in order to proceed without errors. You may enter values in the reset of the optional fields as needed.
6. Then click **Submit and Download** to the top right. This will bring up a directory window, where you can save the `certificate.csr` file. Select a directory and save the file.

## 13.2.2 Getting Certificate Signed

After you download the Oracle Key Vault `certificate.csr` file, you may use any out-of-band method to get it signed by a CA of your choice.

You may then upload the signed certificate back to Oracle Key Vault using the management console.

## 13.2.3 Upload Signed Certificate to Oracle Key Vault

To upload the signed certificate back to Oracle Key Vault:

1. Click the **System** tab and up click **Console Certificate** in the left **System** menu to bring up the **Console Certificate** page.
2. Click **Upload Certificate** to the top right to bring up the **Upload Certificate** page.
3. Click **Choose File** which will bring up a directory window on your local system. Navigate to the directory where you stored the signed certificate and select it. When you are done, you will see the filename to the right of text **Choose File**.
4. Finally click **Upload** to the top right. If the certificate is installed with no errors, you will see its details appear in a new **Uploaded Certificate Details** panel just below **Console Certificate**.
5. You can deactivate the certificate by clicking **Deactivate** to the top right of the **Uploaded Certificate Details** section.
6. When you deactivate the certificate the **Deactivate** button will be replaced by an **Apply Certificate** button. Click this button to re-activate the certificate.

## 13.2.4 Notes on Using Third-Party Certificates

You must perform additional steps when you use third-party certificates in the following situations:

- High Availability

  If you want to use a third-party certificate in a high availability configuration, you must install it on the primary and standby servers first, and then pair them.
- RESTful Services

  Whenever you install a third-party certificate you must re-download the RESTful software utility in order to use the new certificate.
- Restore data from a backup

  If you install a third-party certificate, perform a backup, and then restore another Key Vault appliance from that backup, you will have to re-install the third-party certificate on the new appliance in order to use it. The restore process does not copy the third-party certificate.

# A

# Installing and Configuring Oracle Key Vault 12.2.0.4.0 and Earlier

Before installing Oracle Key Vault 12.2.0.4.0 and earlier, ensure that the server meets the recommended requirements. For more information about the Oracle Key Vault installation requirements, see Oracle Key Vault Installation Requirements (page 3-1).

## A.1 Downloading the Oracle Key Vault Appliance Software

For a fresh installation, the Oracle Key Vault appliance software can be downloaded from Software Delivery Cloud. Note that this package cannot be used to upgrade Oracle Key Vault.

For an upgrade, the Oracle Key Vault appliance software can be downloaded from the Oracle Automated Release Updates (ARU) website.

To download the Oracle Key Vault Appliance Software:

1. Use a web browser to access the Oracle Software Delivery Cloud portal:

   https://edelivery.oracle.com

2. Click **Sign In**. Enter your **User ID** and **Password**, if required.
3. In the **Search By** field, type **Key Vault**.
4. From the list that is displayed, select one of the following:
   - Oracle Key Vault 12.2.0.4.0
   - Oracle Key Vault 12.2.0.3.0
   - Oracle Key Vault 12.2.0.2.0
   - Oracle Key Vault 12.2.0.1.0
   - Oracle Key Vault 12.2.0.0.0
5. Click **Continue**.
6. On the **Download Queue** page, verify the details of the installation package, and click **Continue**.
7. The **Oracle Standard Terms and Restrictions** dialog box is displayed.
8. Select **I have reviewed and accept the terms of the Commercial License, Special Programs License, and/or Trial License**, and click **Continue**.
9. The **File Download** dialog box is displayed. Click **View Digest Details**.

Oracle Key Vault 12.2.0.4.0 and earlier consists of a single ISO file `<file_name>.iso`.

10. Copy the checksum displayed beside **MD5** and store it for later reference.
11. Click **Download** and select a location to save the ISO file.
12. Click **Save**.

    The size of the ISO file exceeds 4 GB, and will take time to download, depending on the network speed. The estimated download time and speed are displayed in the **File Download** dialog box.
13. The ISO file is downloaded to the specified location. Verify the MD5 checksum of the downloaded file:

    ```
    md5sum <file_name>.iso
    ```

    Ensure that the checksum matches the value that you copied from the **File Download** dialog box in *Step 10*.
14. Burn `<file_name>.iso` to a DVD-ROM disc.

You can now install Oracle Key Vault on the server.

## A.2 Installing the Oracle Key Vault Appliance Software

The installation process installs all required software components onto a dedicated server. The installation process may take from 30 minutes to an hour to complete, depending on the server resources where you are installing Oracle Key Vault.

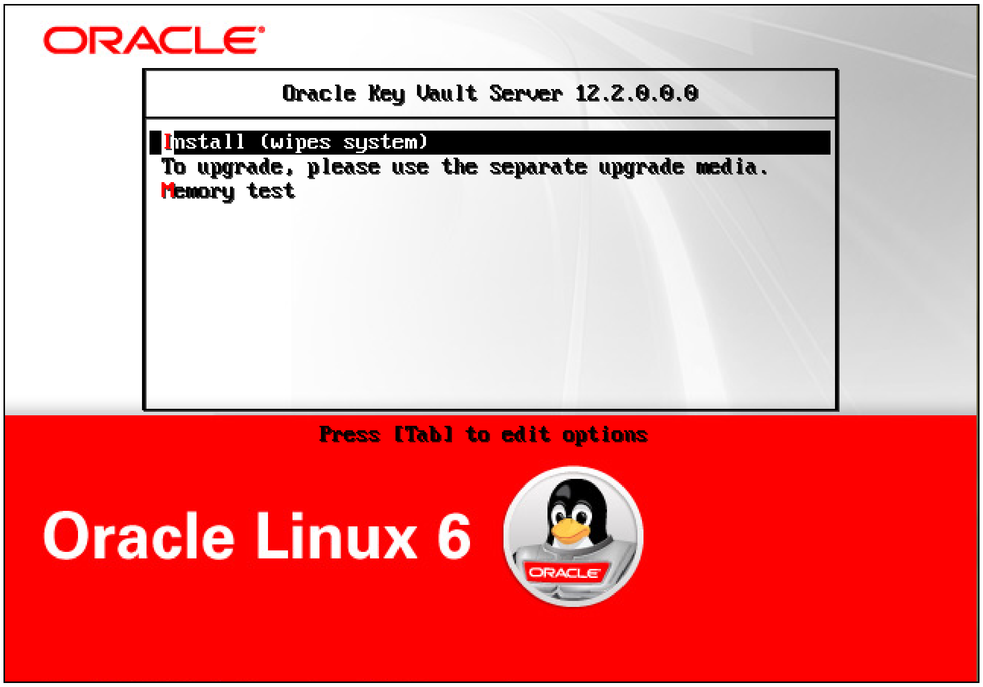
> **Caution:**
>
> The Oracle Key Vault installation wipes the server and installs a stripped-down version of Oracle Linux, thus erasing existing software and data on the server.

- Ensure that the server meets the recommended requirements.
- Request a fixed IP address, network mask, and gateway address from your network administrator for the dedicated server. You will need this information to configure the network in *Step 10*.

To install the Oracle Key Vault appliance:

1. Insert the DVD-ROM disc containing `okv-installer-12.2.0.x.0.iso` into the CD/DVD drive and restart the computer.
2. The installation starts, and the initial splash screen is displayed.

**Figure A-1    Oracle Key Vault Install Screen**



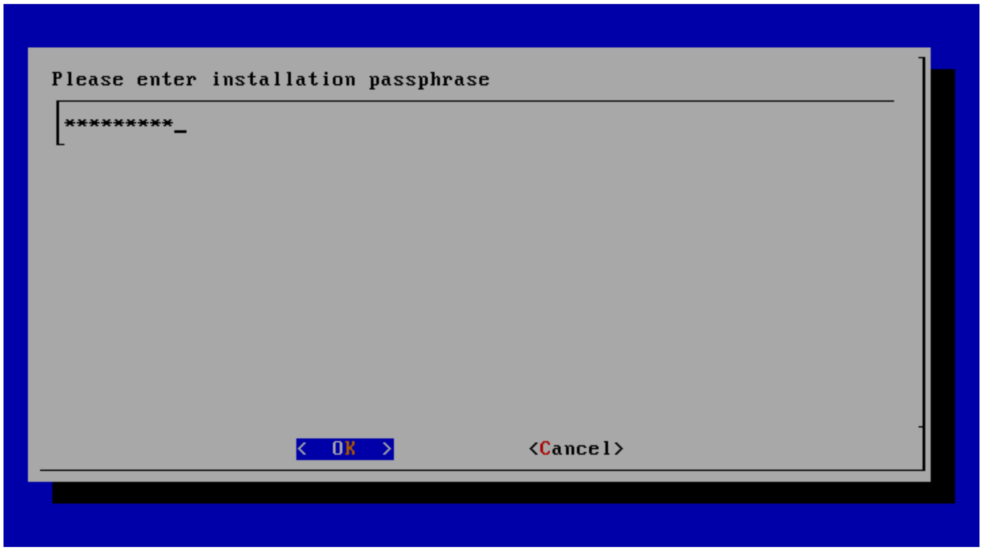
Oracle Key Vault Server 12.2.0.0.0

Install (wipes system)
To upgrade, please use the separate upgrade media.
Memory test

Press [Tab] to edit options

3. The installation proceeds and after several minutes, the message **Please enter installation passphrase** is displayed.

**Figure A-2    Installation Passphrase Screen**



Please enter installation passphrase

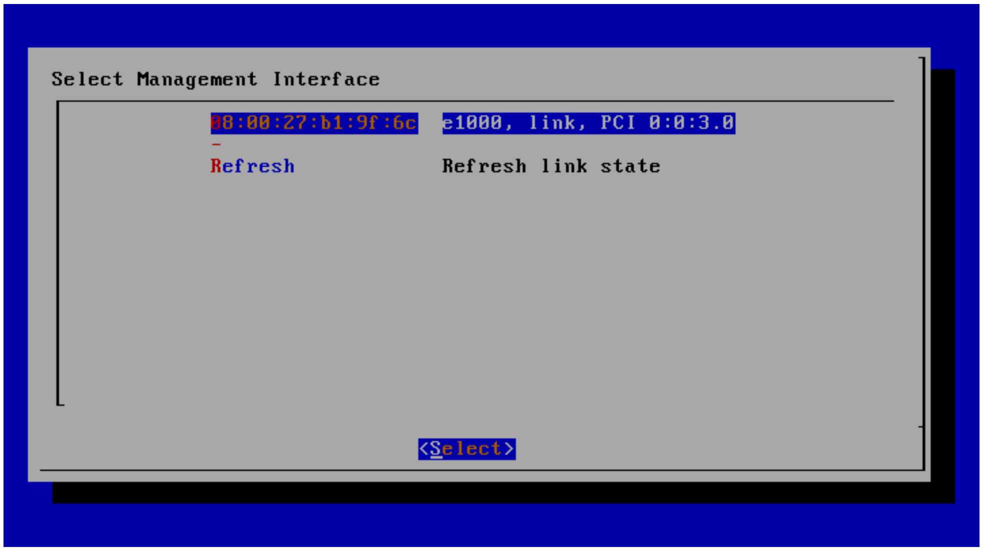
```
**********_
```

The installation passphrase must have 8 or more characters and contain at least one of each of the following: an uppercase letter, a lowercase letter, number, and special character from the set: period (.), comma (,), underscore (_), plus sign (+), colon (:), space.

It is important to store the installation passphrase securely. You will need it later to authenticate yourself at the Key Vault management console and complete the post-installation tasks.

4. Enter the installation passphrase, and press **Enter**.
5. Confirm the installation passphrase, and press **Enter**.
6. The message **Installation passphrase was successfully configured** is displayed. Press **Enter**. The **Select Management Interface** screen is displayed.

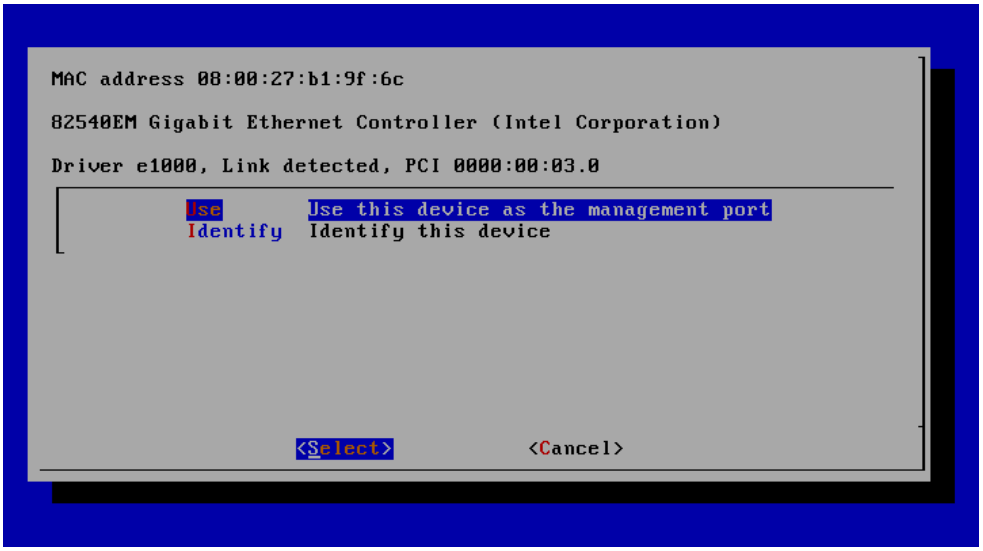
**Figure A-3 Select Management Interface Screen**



```
Select Management Interface

          08:00:27:b1:9f:6c    e1000, link, PCI 0:0:3.0
          -
          Refresh              Refresh link state

                        <Select>
```

7. Select the interface and press **Enter**. If more than one network interface is available, select the interface that you want to serve as the management interface, and to communicate with endpoints.
8. The **Identify Management Interface** screen is displayed.

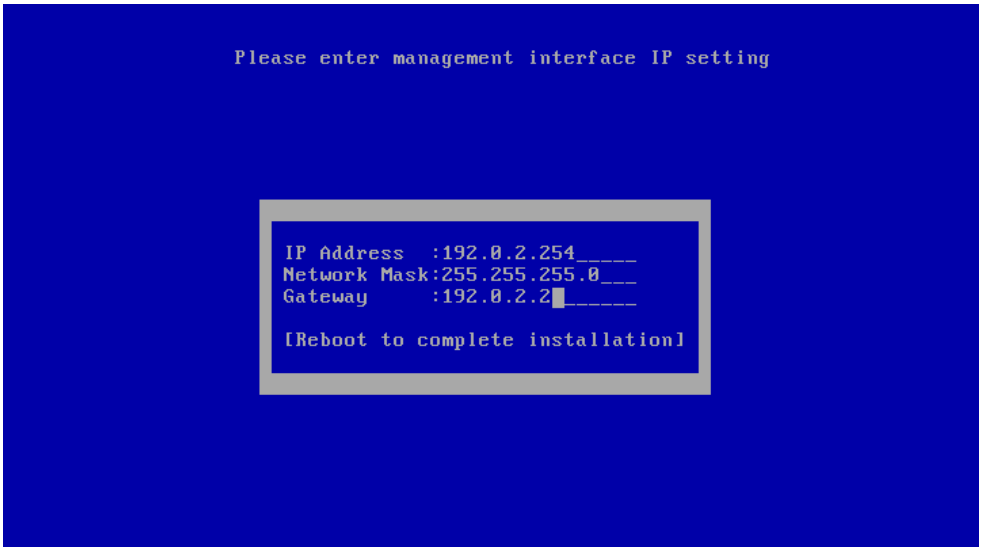
**Figure A-4 Identify Management Interface Screen**



```
MAC address 08:00:27:b1:9f:6c

82540EM Gigabit Ethernet Controller (Intel Corporation)

Driver e1000, Link detected, PCI 0000:00:03.0

          Use         Use this device as the management port
          Identify    Identify this device

                <Select>            <Cancel>
```

9. Press **Enter**. The **IP Address Setting for Management Interface Screen** is displayed.

**Figure A-5 IP Address Setting for Management Interface Screen**



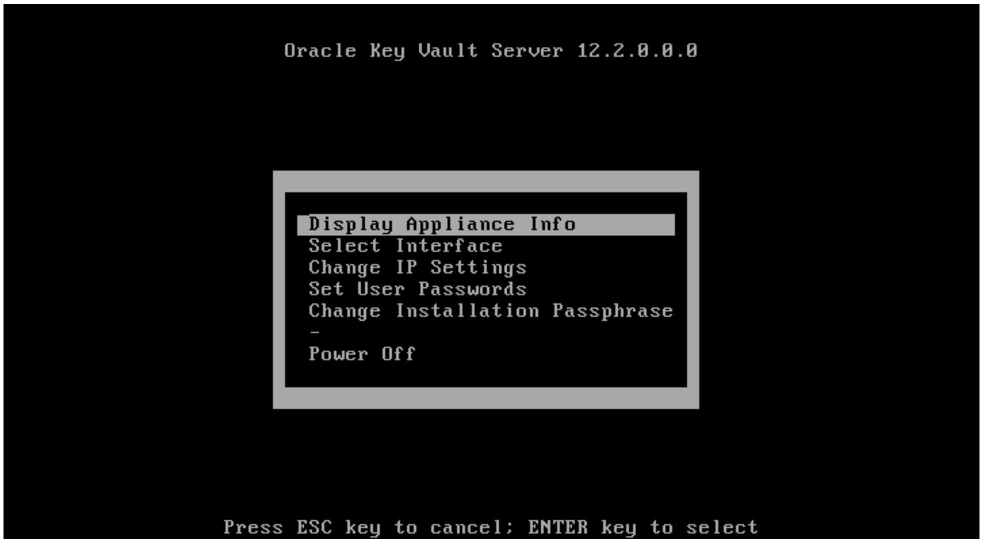
```
Please enter management interface IP setting

IP Address   :192.0.2.254_____
Network Mask:255.255.255.0___
Gateway      :192.0.2.2█______

[Reboot to complete installation]
```

10. Enter the fixed IP address, network mask, and gateway address you received from your network administrator. Select **Reboot to complete installation** and press **Enter**.

    The installer installs and configures the operating system, database, and Oracle Key Vault on the server to make it a self-contained hardened appliance. The installation and configuration process can take between 30 minutes to an hour. Press the **Shift** key to check installation status.

11. If the installation completed successfully, the **Oracle Key Vault Server <Release Number>** screen appears.

**Figure A-6 Oracle Key Vault Server <Release Number> Screen**



```
Oracle Key Vault Server 12.2.0.0.0

Display Appliance Info
Select Interface
Change IP Settings
Set User Passwords
Change Installation Passphrase
-
Power Off

Press ESC key to cancel; ENTER key to select
```

Select **Display Appliance Info** and press **Enter** to see the IP address settings for the appliance. Make a note of the IP address of the appliance. You will need it to log into the browser-based management console of Oracle Key Vault.

If you need to correct the IP Address, network mask, or the IP gateway for any reason, you can select **Change IP Settings** and enter the new IP settings.

Select **Set User Passwords** to set the Root and Support User passwords. You can also set the Root and Support User passwords when performing Post-Installation Tasks (page 3-10).

You have the option to change the installation passphrase by selecting **Change Installation Passphrase**. For more information about changing the installation passphrase, see Change the Installation Passphrase (page 12-17).

> **Note:**
>
> You will need to enter the old installation passphrase in order to update the installation passphrase.

Make a note of the installation passphrase. You will need it to log into the management console for the first time, in order to complete the post-installation tasks.

## A.3 Performing Post-Installation Tasks

After you install Oracle Key Vault, you must complete the following post-installation tasks: setting up the administrative user accounts, and passwords for recovery, root, and support.

To perform the post-installation tasks:

1. Use a web browser to connect to the Oracle Key Vault server.

   To connect in to an Oracle Key Vault server whose IP address is 192.0.2.254, enter the following in the Address Bar:

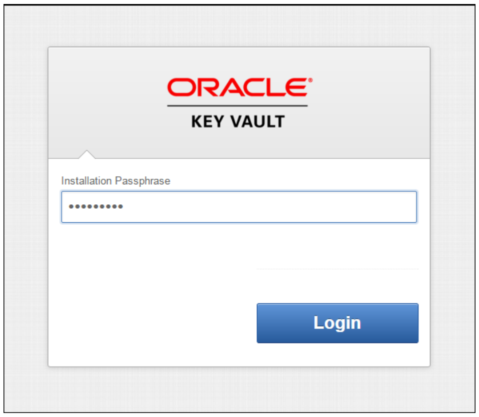
   ```
   https://192.0.2.254
   ```

2. If the web browser displays a security warning message stating that you are connecting to a website with an untrusted or self-signed security certificate, accept the security warning message and proceed to connect to the Oracle Key Vault server.

   > **Note:**
   >
   > After completing the post-installation tasks, you can upload a custom certificate or certificate chain that is trusted by the browser, so that you can connect to the Oracle Key Vault server without encountering the security warning message. For more information about uploading a custom certificate, see Third Party Certificates (page 13-4).

3. The **Installation Passphrase** screen is displayed.

**Figure A-7 Installation Passphrase Screen**

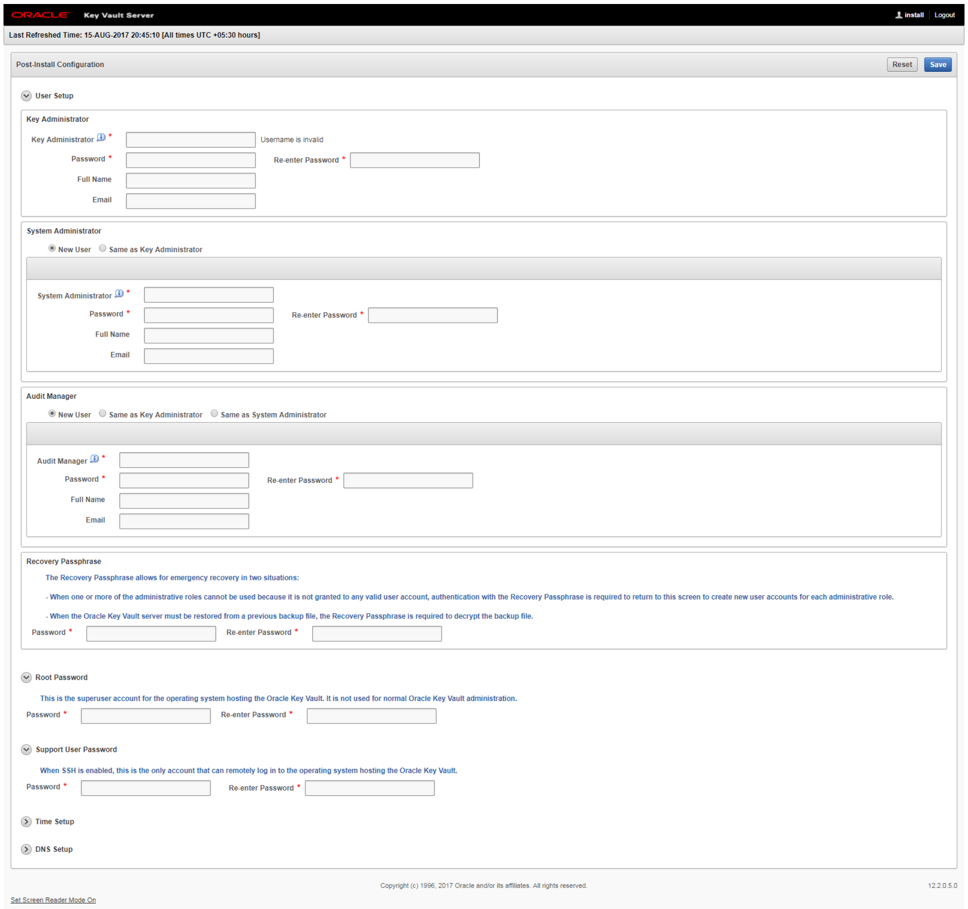


**Note:**

The **Installation Passphrase** screen is displayed when you connect to the Oracle Key Vault server for the first time, in order to complete the post-installation tasks. After you complete the post-installation tasks, the Oracle Key Vault login screen is displayed when you access the Oracle Key Vault management console through the web browser.

4. Enter the installation passphrase. The **Post-Install Configuration** screen is displayed.

### Figure A-8 Post-Install Configuration Screen



ORACLE Key Vault Server — install | Logout

Last Refreshed Time: 15-AUG-2017 20:45:10 [All times UTC +05:30 hours]

Post-Install Configuration — Reset | Save

User Setup

Key Administrator

Key Administrator * Username is invalid
Password * Re-enter Password *
Full Name
Email

System Administrator

● New User ○ Same as Key Administrator

System Administrator *
Password * Re-enter Password *
Full Name
Email

Audit Manager

● New User ○ Same as Key Administrator ○ Same as System Administrator

Audit Manager *
Password * Re-enter Password *
Full Name
Email

Recovery Passphrase

The Recovery Passphrase allows for emergency recovery in two situations:

- When one or more of the administrative roles cannot be used because it is not granted to any valid user account, authentication with the Recovery Passphrase is required to return to this screen to create new user accounts for each administrative role.

- When the Oracle Key Vault server must be restored from a previous backup file, the Recovery Passphrase is required to decrypt the backup file.

Password * Re-enter Password *

Root Password

This is the superuser account for the operating system hosting the Oracle Key Vault. It is not used for normal Oracle Key Vault administration.

Password * Re-enter Password *

Support User Password

When SSH is enabled, this is the only account that can remotely log in to the operating system hosting the Oracle Key Vault.

Password * Re-enter Password *

Time Setup

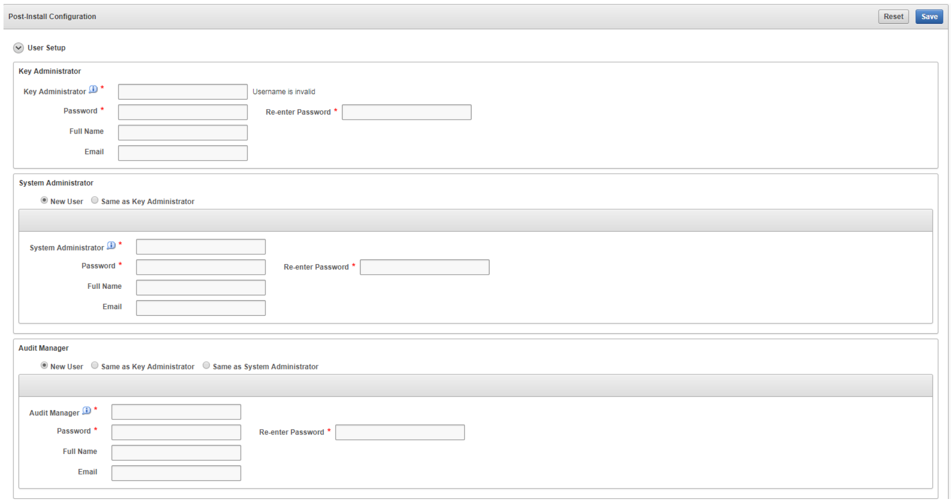
DNS Setup

Copyright (c) 1996, 2017 Oracle and/or its affiliates. All rights reserved. 12.2.0.5.0

Set Screen Reader Mode On

5. In the **User Setup** section, create three administrative user accounts for the Key Administrator, System Administrator, and Audit Manager.

### Figure A-9 Post-Install Configuration — User Setup



Post-Install Configuration — Reset | Save

User Setup

Key Administrator

Key Administrator * Username is invalid
Password * Re-enter Password *
Full Name
Email

System Administrator

● New User ○ Same as Key Administrator

System Administrator *
Password * Re-enter Password *
Full Name
Email

Audit Manager

● New User ○ Same as Key Administrator ○ Same as System Administrator

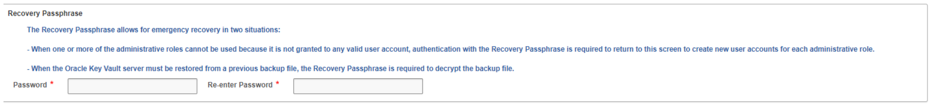
Audit Manager *
Password * Re-enter Password *
Full Name
Email

In the **User Setup** section:

- Enter the user name and password, the full name (optional), and email (optional) for each administrative user account.
- You can create a different user account for each of these administrative roles for a strict separation of duties, or combine roles as needed.
- Passwords must have 8 or more characters and contain at least one of each of the following: an uppercase letter, a lowercase letter, number, and one special character from the set: period (`.`), comma (`,`), underscore (_), plus sign (+), colon (`:`), space.

6. In the **Recovery Passphrase** section, set the recovery passphrase.

**Figure A-10 Post-Install Configuration — Recovery Passphrase**



Recovery Passphrase

The Recovery Passphrase allows for emergency recovery in two situations:

- When one or more of the administrative roles cannot be used because it is not granted to any valid user account, authentication with the Recovery Passphrase is required to return to this screen to create new user accounts for each administrative role.

- When the Oracle Key Vault server must be restored from a previous backup file, the Recovery Passphrase is required to decrypt the backup file.

Password * Re-enter Password *

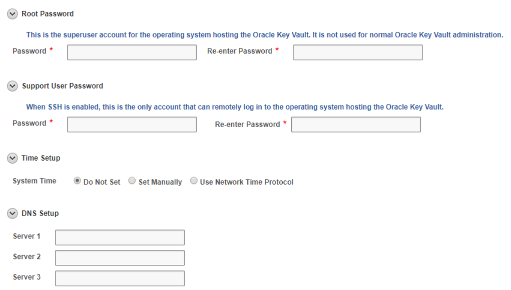
The recovery passphrase has the same minimum requirements as user passwords. For greater security, it is recommended that you make the recovery passphrase longer and more complex. You must keep the recovery passphrase safe and retrievable because it is required in the following situations:

- In an emergency, when there are no administrative users available to access Key Vault.
- To restore Key Vault data from a backup.
- To reset the recovery passphrase.

> **Caution:**
>
> **It is important to establish a secure process for the storage and retrieval of the recovery passphrase, including older recovery passphrases. The only way to recover from a lost recovery passphrase is to re-install Key Vault.**

7. In the next section, set the Root and Support User passwords, if you did not set the passwords using the **Set User Passwords** option on the **Oracle Key Vault Server <Release Number>** screen in the previous procedure, Installing the Oracle Key Vault Appliance Software (page 3-5).

**Figure A-11 Post-Install Configuration — Root and Support User Passwords**



Root Password

This is the superuser account for the operating system hosting the Oracle Key Vault. It is not used for normal Oracle Key Vault administration.

Password * Re-enter Password *

Support User Password

When SSH is enabled, this is the only account that can remotely log in to the operating system hosting the Oracle Key Vault.

Password * Re-enter Password *

Time Setup

System Time ● Do Not Set ○ Set Manually ○ Use Network Time Protocol

DNS Setup

Server 1

Server 2

Server 3

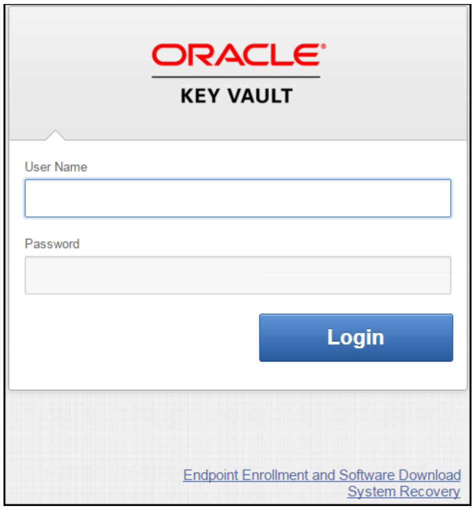
The root password is the super user account for the operating system hosting Key Vault. You will need the support password to log into Key Vault remotely using the SSH protocol.

> **Caution:**
>
> **Keep the root and support user passwords safe because these passwords are set during post-installation only. After post-installation you cannot change them from the Oracle Key Vault management console.**

The **Time Setup** and **DNS Setup** settings are optional at this stage, and can be set up later by a System Administrator.

8. Click **Save** in the upper right corner of the **Post-Install Configuration** screen. The Oracle Key Vault Management Console login screen is displayed.

**Figure A-12 Oracle Key Vault Management Console Login Screen**



You can now login to the Oracle Key Vault management console with the credentials of any of the user accounts created during the post-installation process. For more information about the Oracle Key Vault management console, see Logging In to the Oracle Key Vault Management Console (page 3-14).

B

# Troubleshooting Oracle Key Vault

This section contains checklists and tips for commonly encountered errors that will help you install and deploy Key Vault quickly.

**See Also:**

- "Recommendations for Uploading and Downloading Oracle Wallets (page 11-4)"
- "Recommendations for Uploading and Downloading JKS and JCEKS Keystores (page 11-7)"
- "Recommendations for Uploading and Downloading Credential Files (page 11-10)"

## B.1 Key Vault Pre-Installation Checklist

The pre-installation checklist covers all the requirements to successfully install Key Vault.

**Table B-1 Oracle Key Vault Pre-Installation Checklist**

| Item# | Check | Task |
|---|---|---|
| 1. [ x ] | System Requirements | Confirm that you have enough CPU, Memory, and Disk as described here, in "System Requirements (page 3-1)" |
| 2. [ x ] | Open all the required network ports in your firewall | For details on network ports, see "Network Ports (page 3-2)" |
| 3. [ x ] | Supported endpoint platforms | We have expanded platform support as listed here, "Supported Endpoint Platforms (page 3-3)" |
| 4. [ x ] | Set the COMPATIBLE initialization parameter for Online Master Key (previously TDE direct connect) | How to set this parameter for Oracle Database 11.2.0.0 and higher is described here, "Supported Endpoint Platforms (page 3-3)" |
| 5. [ x ] | Get a fixed IP address, Network Mask, and Gateway Address from your network administrator | You will need this for Step 9 of Task 1 of the installation process described here, Installing the Oracle Key Vault Appliance Software (page 3-5) |

## B.2 Audit Vault Database Firewall Integration Instructions

To consolidate audits between Audit Vault and Database Firewall (AVDF) with Oracle Key Vault, you must integrate AVDF with Key Vault as follows:

1. Install an AVDF 12.2.server and set up the AV administrator and auditor users.
2. Install an OKV 12.2 server. Then, log on as the system administrator and enable `ssh` access.
3. Register the OKV server as a secured target on the AVDF server. Log in as AV administrator (`avadmin`), and go to **Secured Targets** > **Register**. Enter the OKV server name as the name. Set type to the `Oracle Key Vault` plugin and use the connect string: jdbc:oracle:thin:@//127.0.0.1:1521/dbfwdb. For username and password, use `avcollector/<integration_password>`. Set collection attributes on the secured target:

   ```
   av.collector.securedTargetVersion 12.2.0.0.0
   av.collector.TimeZoneOffset +5:30
   ```

   Note: You can use any offset that applies to your situation.
4. Register the OKV server as a host on the AVDF machine. Go to **Hosts > Register** and add it. You will need the OKV server IP. Note that after the host is added, a new entry will appear with an **Agent Activation Key**. Copy this value and save it somewhere.
5. Download the AVDF agent. Go to **Hosts > Agent > Agent Release** and get the agent (first item in the list). Save as `agent.jar` and upload to the OKV server using `scp`.
6. Installing `agent.jar` on the OKV server:

   a. Log on to the OKV command line as support and su to root to create the directory, get the agent, and set permissions:

```
cd /usr/local/okv
mkdir avdf
cp /home/support/agent.jar /usr/local/okv/avdf
chown oracle:oinstall /usr/local/okv/avdf /usr/local/okv/avdf/*
```

b. su to oracle and extract the agent:

```
su - oracle
cd /usr/local/okv/avdf
java -jar agent.jar -d /usr/local/okv/avdf
```

c. As oracle, start the agent and enter the ADVF **Agent Activation Key**:

```
cd /usr/local/okv/avdf/bin
./agentctl start -k
```

d. Using the OKV browser-based console, enable the database user. To enable the database user, go to **System** > **System Settings**. In the **Oracle Audit Vault Integration** section, check **Enable**. Enter the integration password.

7. Add audit trail to AVDF. Log on as the AV administrator (`avadmin`) user. Go to **Secured Targets** > **Audit Trails** > **Add** to add a new audit trail. Select type as TABLE, and choose the secured target and host that you created. Set the table name to `keyvault.audit_trail`. After saving, select the audit trail and start collection.

8. View data collected by AVDF: Log into AVDF as `avauditor` and go to **Reports** > **All Activity** > **All Activity Report**.

## B.3 RESTful Services Troubleshooting Help

The Key Vault log files capture all the error messages sent by the server. The error messages are written to the `/var/log/messages` file. The first debugging step is to read the `messages` file.

To check for log file errors, do the following as a root user:

```
root# vi /var/log/messages
```

## B.4 Error: Cannot Open Keystore Message

The `Cannot Open Keystore` error can appear when you try to upload a Java keystore to the Oracle Key Vault server.

You can try the following solutions:

- Ensure that the `PATH` environment variable has been correctly set.
- Check where the keytool and Java are pointing to, by entering the following commands in a shell:

  ```
  which keytool
  which java
  ```

  Ensure that you are using Oracle Java.

## B.5 KMIP Error: Invalid Field

The `Invalid Field` KMIP error can occur when you are trying to upload Oracle wallets to virtual wallets on multiple endpoints as follows:

1. You configure two or more endpoints (for example, Endpoint A and Endpoint B) to share a wallet (Oracle Wallet C), and hence also share the wallet keys.
2. You register Endpoints A and B with Oracle Key Vault.
3. You create a default wallet (Virtual Wallet A) for Endpoint A and then a default wallet (Virtual Wallet B) for Endpoint B. Each virtual wallet is accessible only to the corresponding endpoint. For example, Endpoint B has no access to Virtual Wallet A.
4. You upload Oracle Wallet C into Virtual Wallet A on Endpoint A.
5. You attempt to upload Oracle Wallet C from Endpoint B into Virtual Wallet B Endpoint B.

The KMIP error occurs because there are two copies of the same key being created and Endpoint B does not have visibility for both. If Endpoint A tries to upload the first key again, Oracle Key Vault detects this action and accounts for it. But because in Step 5, Endpoint B is not allowed to see the first key, Oracle Key Vault is unable to perform the necessary harmonization for the two Oracle wallets.

This is expected behavior. Instead, create an endpoint group so that you can share the wallet with multiple endpoints. See "Manage Endpoint Groups (page 7-16)" for more information.

> **Note:**
>
> The KMIP error can occur for other scenarios, but this scenario is the most common.

## B.6 WARNING: Could Not Store Private Key Errors

If you upload two keystores with the same file name but with different contents, a `WARNING: Could not store private key` error is generated.

This occurs if you use the same alias (`-alias slserver`) in each `keytool` command. When you download two such keystores that have the same alias, the `okvutil download` process ignores the second one because the JKS aliases must be unique. Download the second keystore using a unique alias.

> **See Also:**
>
> "Downloading JKS or JCEKS Keystores (page 11-6)"

## B.7 Errors After Upgrading Oracle Key Vault

After you perform an upgrade of Oracle Key Vault on an standalone server, `ORA-1109`, `ORA-00313`, and `ORA-00312` error messages may appear in the `/var/log/messages` log file.

You can safely ignore these messages. Error messages also appear in the `/var/log/debug` file.

## B.8 Error: Failed to Open Wallet

If you are attempting Online Master Key (previously TDE direct connect) and encounter the error **Failed to Open Wallet**, you must first check your environment variable ORACLE_BASE. See **Step 1** of "Configuring a Connection Between Oracle Key Vault and a New TDE-Enabled Database (page 11-12)"

You must also set the following environment variables: ORACLE_SID, ORACLE_HOME, and OKV_HOME needed by the PKCS #11 library as follows:

1. Set system ENV variables for the USER.
2. Shutdown DB.
3. Restart Service.
4. Restart DB.

## B.9 Error: Provision Command Fails if /usr/bin/java does not Exist

The RESTful Service command to provision an endpoint fails if the soft link `/usr/bin/java` does not exist or point to the correct Java directory. The Java version should be 1.4 or greater.

To create the soft link to your Java home directory:

```
ln -s <your_Java_home_directory>/bin/java /usr/bin/java
```

## B.10 Transaction Check Error: Diagnostics Generation Utility

If you are trying to perform an upgrade of Oracle Key Vault, a transaction check error may appear.

For example:

```
file /usr/local/dbfw/etc/dbfw-diagnostics-package.yml from install of
appliance-18.1.0.0.0-52_190425.2253.d.x86_64 conflicts with file from
package okv-diagnostic-12.2.0.8.0-40_181013.1730.x86_64
```

The problem is that the diagnostic generation utility interferes with the upgrade process. You must remove the diagnostic generation utility before you can perform the upgrade.

**Related Topics**

- Download System Diagnostics (page 12-19)

# B.11 Fast-Start Failover (FSFO) Suspended (ORA-16818)

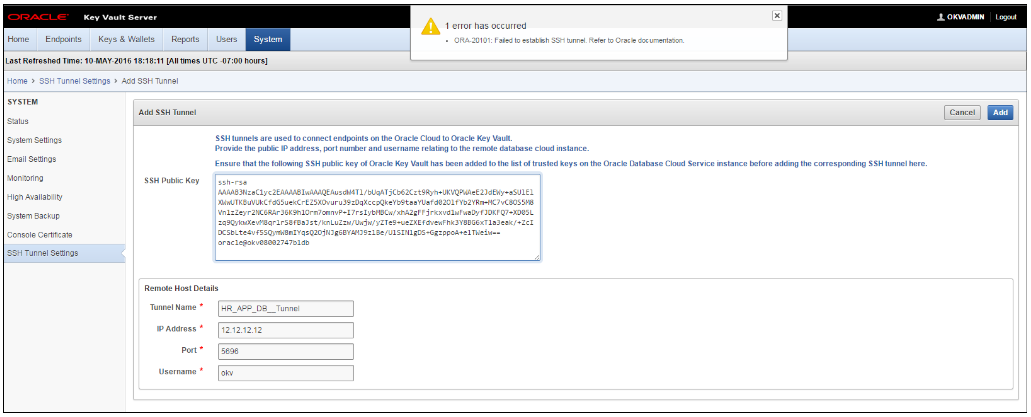
If the primary was shutdown gracefully in a controlled way like clicking on Power Off instead of 'pulling the plug', a fast start failover is not going to be performed. In a graceful shutdown the primary server's failover status goes into a suspended state with the standby waiting indefinitely for the primary to come back up. This is the expected behavior for FSFO as defined by Oracle DataGuard avoid a split brain scenario. By design, an FSFO will occur only when the primary goes down unexpectedly. If you do a shutdown immediate (or shutdown normal), FSFO will not occur.

# B.12 SSH Tunnel Add Failure

You might get the following error message while trying to set up the SSH tunnel:

**Failed to establish SSH tunnel. Refer to Oracle documentation.**

**Figure B-1 SSH Tunnel Add Failure**



1 error has occurred

- ORA-20101: Failed to establish SSH tunnel. Refer to Oracle documentation.

Last Refreshed Time: 10-MAY-2016 18:18:11 [All times UTC -07:00 hours]

Home › SSH Tunnel Settings › Add SSH Tunnel

Add SSH Tunnel

SSH tunnels are used to connect endpoints on the Oracle Cloud to Oracle Key Vault.
Provide the public IP address, port number and username relating to the remote database cloud instance.

Ensure that the following SSH public key of Oracle Key Vault has been added to the list of trusted keys on the Oracle Database Cloud Service instance before adding the corresponding SSH tunnel here.

SSH Public Key

```
ssh-rsa
AAAAB3NzaC1yc2EAAAABIwAAAQEAusdW4T1/bUqATjCb62Czt9Ryh+UKVQPWAeE2JdEWy+aSU1E1
XWwUTKBuVUkCfdG5uekCrEZ5XOvuru39zDqXccpQkeYb9taaYUafd02O1fYb2YRm+MC7vC8OS5M8
Vn1zZeyr2NC6RAr36K9h1Orm7omnvP+I7rsIybMBCw/xhA2gFFjrkxvd1wFwaDyfJDKFQ7+XD05L
zq9QykwXevM8qrlrS8fBaJst/knLuZzw/Uwjw/yZTe9+ueZXEfdvewFhk3Y8BG6xT1a3eak/+ZcI
DCSbLte4vf5SQymW8mIYqsQ2OjNJg6BYAMJ9z1Be/U1SIN1gDS+GgzppoA+e1TWeiw==
oracle@okv08002747b1db
```

Remote Host Details

Tunnel Name * HR_APP_DB__Tunnel

IP Address * 12.12.12.12

Port * 5696

Username * okv

The failure may be due to one or more of the following:

- Invalid Tunnel Name
- Invalid IP Address
- Invalid Port
- Invalid Username
- The public SSH Key Vault key is not copied to the `authorized_keys` file of the `okv` user on the Database as a Service instance
- The Database as a Service instance is not reachable because of network overload

Check your input values and connection and retry.

## B.13 TDE Endpoint Integration Issues

This section addressed the common problems with TDE endpoint integration and how to overcome them.

**Notes on Installing Oracle Key Vault Library**

- You must run root.sh to install the Oracle Key Vault library only ONCE on a machine with multiple Oracle Databases
- During an upgrade you must upgrade the library only after all the end points are shut down. Oracle Key Vault servers are backwards compatible with endpoint libraries at present.

**If Using SVRCTL Tool to Manage Database**

If you use the svrctl tool to manage the database, note that the tool may set `ORACLE_BASE` to `NULL`. A good practice is to set `ORACLE_BASE` to `ORACLE_HOME`, if `ORACLE_BASE` is not used in your environment.

**Primary and Standby Should Manage Security Objects in the Same Way**

In a high availability configuration both the primary and standby servers should use the same mechanism to manage security objects. They should either both use a wallet or both use Oracle Key Vault.

## B.14 Failover Situations in High Availability Mode

The following sections list common failover scenarios that are encountered in High Availability mode deployments, for various versions of Oracle Key Vault, and for Oracle Key Vault 12.2.0.5.0 with Read-Only Restricted mode disabled, and with Read-Only Restricted mode enabled.

- Types of Failover Situations (page B-7)
- Failover Situations without Read-Only Restricted Mode (page B-8)
- Failover Situations with Read-Only Restricted Mode (page B-10)
- Performing a Planned Shutdown (page B-13)

### B.14.1 Types of Failover Situations

The types of failover situations are:

- **Planned shutdown of the primary server:** The primary server is shut down by a system administrator during an upgrade or maintenance window.
- **Planned shutdown of the standby server:** The standby server is shut down by a system administrator during an upgrade or maintenance window.
- **Unplanned shutdown of the primary server:** The primary server is offline due to unforeseen circumstances such as power loss or network failure.

- **Unplanned shutdown of the standby server:** The standby server is offline due to unforeseen circumstances such as power loss or network failure.

## B.14.2 Failover Situations without Read-Only Restricted Mode

The following table describes the various failover situations that may be encountered in the following versions of Oracle Key Vault:

- 12.2.0.0.0
- 12.2.0.1.0
- 12.2.0.2.0
- 12.2.0.3.0
- 12.2.0.4.0
- 12.2.0.5.0 (without Read-Only Restricted Mode)

**Table B-2 Failover Situations without Read-Only Restricted Mode**

| Number | Failover Situation | Primary Server State | Standby Server State | Does failover occur? | Details | Recovery Steps |
|---|---|---|---|---|---|---|
| 1 | **Primary Server: Planned Shutdown during upgrade** | Down | Up | No | When the primary server goes offline during an upgrade, the standby server waits in read-only mode for the primary server to come back online. During the upgrade, you cannot access the Oracle Key Vault management console.<br>A failover does not occur.<br>For more information about performing a primary server planned shutdown during upgrade, see Performing a Primary Server Planned Shutdown during Upgrade (page B-13). | When the primary server is back online post-upgrade, the standby server will automatically synchronize with the primary server. The primary and standby servers retain their earlier roles, and both servers will continue to operate in High Availability mode. |

**Table B-2 (Cont.) Failover Situations without Read-Only Restricted Mode**

| Number | Failover Situation | Primary Server State | Standby Server State | Does failover occur? | Details | Recovery Steps |
|---|---|---|---|---|---|---|
| **2** | **Primary Server: Planned Shutdown during maintenance** | Down | Up | Yes | When the primary server is powered off or rebooted during maintenance, the standby server takes over from the primary server.<br>Note:<br>For more information about performing a primary server planned shutdown during maintenance, see Performing a Primary Server Planned Shutdown during Maintenance (page B-14). | The standby server is now the new primary server. When the old primary server is back online post-maintenance, it will automatically synchronize with the new primary server, and takes over the role of standby server. Both servers will continue to operate in High Availability mode.<br>**Note:** When the primary server is offline, data replication is disabled. If the new primary server goes offline before synchronizing with the new standby server, it may cause a loss of critical data. |
| **3** | **Standby Server: Planned Shutdown** | Up and functional | Down | No | When the standby server is powered off during upgrade or maintenance, the primary server continues operating as the primary server. Read/Write operations are allowed.<br>For more information about performing a standby server planned shutdown during upgrade, see Performing a Standby Server Planned Shutdown during Upgrade (page B-14).<br>For more information about performing a standby server planned shutdown during maintenance, see Performing a Standby Server Planned Shutdown during Maintenance (page B-14). | When the standby server is back online post-upgrade or post-maintenance, the primary server will automatically synchronize with the standby server. The primary and standby servers retain their earlier roles, and both servers will continue to operate in High Availability mode.<br>**Note:** When the standby server is offline, data replication is disabled. If the primary server goes offline before synchronizing with the standby server, it may cause a loss of critical data. |
| **4** | **Primary Server: Unplanned Shutdown** | Down | Up | Yes | When the primary server goes offline because of power loss, network failure, or hardware failure, the standby server waits for the duration specified in the **Fast Start Failover Threshold** field on the **Configure High Availability** page. If the primary server is not reachable after the specified duration has elapsed, the standby server takes over from the primary server. | The standby server is now the new primary server. Rectify the failure that affected the primary server by rebooting the server or restoring network connectivity. When the primary server is back online, it will automatically synchronize with the new primary server, and takes over the role of standby server. |

**Table B-2 (Cont.) Failover Situations without Read-Only Restricted Mode**

| Number | Failover Situation | Primary Server State | Standby Server State | Does failover occur? | Details | Recovery Steps |
|---|---|---|---|---|---|---|
| **5** | **Standby Server: Unplanned Shutdown** | Up | Down | No | When the standby server goes offline because of power loss, network failure, or hardware failure, the primary server becomes unavailable. All operations are disabled. | Rectify the failure that affected the standby server by rebooting the server or restoring network connectivity. When the standby server is back online, it will automatically synchronize with the primary server. If it is not possible to re-establish synchronization or network connectivity between the primary and standby servers, contact Oracle Support. |

## B.14.3 Failover Situations with Read-Only Restricted Mode

The following table describes the various failover situations that may be encountered in Oracle Key Vault 12.2.0.5.0 and later (with Read-Only Restricted mode enabled):

**Table B-3 Failover Situations in 12.2.0.5.0 and later (with Read-Only Restricted Mode enabled)**

| Number | Failover Situation | Primary Server State | Standby Server State | Does failover occur? | Details | Recovery Steps |
|---|---|---|---|---|---|---|

**Table B-3 (Cont.) Failover Situations in 12.2.0.5.0 and later (with Read-Only Restricted Mode enabled)**

| | | | | | | |
|---|---|---|---|---|---|---|
| **1** | **Primary Server: Planned Shutdown during upgrade** | Down | Up | No | When the primary server goes offline during an upgrade, the standby server enters Read-Only Restricted mode and waits for the primary server to come back online. During the upgrade, you cannot access the Oracle Key Vault management console.<br>A failover does not occur.<br>For more information about performing a primary server planned shutdown during upgrade, see Performing a Primary Server Planned Shutdown during Upgrade (page B-13). | When the primary server is back online post-upgrade, the standby server will automatically synchronize with the primary server. The primary and standby servers retain their earlier roles, and both servers will continue to operate in High Availability mode. |
| **2** | **Primary Server: Planned Shutdown during maintenance** | Down | Up | Yes | When the primary server is powered off or rebooted during maintenance, the standby server enters Read-Only Restricted mode, and takes over from the primary server. The Oracle Key Vault management console displays a warning.<br>For more information about performing a primary server planned shutdown during maintenance, see Performing a Primary Server Planned Shutdown during Maintenance (page B-14). | The standby server is now the new primary server. When the old primary server is back online post-maintenance, it will automatically synchronize with the new primary server, and takes over the role of standby server. Both servers will continue to operate in High Availability mode. |

**Table B-3 (Cont.) Failover Situations in 12.2.0.5.0 and later (with Read-Only Restricted Mode enabled)**

| | | | | | | |
|---|---|---|---|---|---|---|
| 3 | **Standby Server: Planned Shutdown** | Up and functional | Down | No | When the standby server is powered off during upgrade or maintenance, the primary server enters Read-Only Restricted mode, and continues operating as the primary server. The Oracle Key Vault management console displays a warning.<br>For more information about performing a standby server planned shutdown during upgrade, see Performing a Standby Server Planned Shutdown during Upgrade (page B-14).<br>For more information about performing a standby server planned shutdown during maintenance, see Performing a Standby Server Planned Shutdown during Maintenance (page B-14). | When the standby server is back online post-upgrade or post-maintenance, the primary server will automatically synchronize with the standby server. The primary and standby servers retain their earlier roles, and both servers will continue to operate in High Availability mode. |
| 4 | **Primary Server: Unplanned Shutdown** | Down | Up | Yes | When the primary server goes offline because of power loss, network failure, or hardware failure, the standby server waits for the duration specified in the **Fast Start Failover Threshold** field on the **Configure High Availability** page. If the primary server is not reachable after the specified duration has elapsed, the standby server enters Read-Only Restricted mode, and takes over from the primary server. | The standby server is now the new primary server. Rectify the failure that affected the primary server by rebooting the server or restoring network connectivity. When the primary server is back online, it will automatically synchronize with the new primary server, and takes over the role of standby server. |

**Table B-3 (Cont.) Failover Situations in 12.2.0.5.0 and later (with Read-Only Restricted Mode enabled)**

| | | | | | | |
|---|---|---|---|---|---|---|
| **5** | **Standby Server: Unplanned Shutdown** | Up | Down | No | When the standby server goes offline because of power loss, network failure, or hardware failure, the primary server waits for the duration specified in the **Fast Start Failover Threshold** field on the **Configure High Availability** page. If the standby server is not reachable after the specified duration has elapsed, the primary server enters Read-Only Restricted mode, and continues operating as the primary server. | Rectify the failure that affected the standby server by rebooting the server or restoring network connectivity. When the standby server is back online, it will automatically synchronize with the primary server. If it is not possible to re-establish synchronization or network connectivity between the primary and standby servers, contact Oracle Support. |

## B.14.4 Performing a Planned Shutdown

A planned shutdown is performed by a system administrator during an upgrade or maintenance window.

- Primary Server Planned Shutdown (page B-13)
- Standby Server Planned Shutdown (page B-14)

### B.14.4.1 Primary Server Planned Shutdown

A planned shutdown of the primary server is performed by a system administrator during an upgrade or a maintenance window.

> **Note:**
>
> After an upgrade, the primary and standby server retain their old roles. After a maintenance window, the primary and standby server switch roles.

- Performing a Primary Server Planned Shutdown during Upgrade (page B-13)
- Performing a Primary Server Planned Shutdown during Maintenance (page B-14)

#### B.14.4.1.1 Performing a Primary Server Planned Shutdown during Upgrade

To perform a primary server planned shutdown during upgrade, follow the procedure Upgrade a Pair of Key Vault Servers in a High Availability Deployment (page 12-39).

During an upgrade, failover does not occur. The primary and standby servers retain their earlier roles after the primary server is back online post-upgrade.

#### B.14.4.1.2 Performing a Primary Server Planned Shutdown during Maintenance

To perform a primary server planned shutdown during maintenance, power off or reboot Oracle Key Vault.

Power off Oracle Key Vault by clicking the **Power Off** button on the **Settings** page. You can also power off Oracle Key Vault by using the terminal.

Reboot Oracle Key Vault by clicking the **Reboot** button on the **Settings** page. You can also reboot Oracle Key Vault by using the terminal.

When the primary server is shut down, the standby server waits for the duration specified in the **Fast Start Failover Threshold** field on the **Configure High Availability** page. When the duration has elapsed, the standby server will failover and take over from the primary server. The standby server is now the new primary server.

When the old primary server is back online after maintenance, it will take over the role of standby server.

### B.14.4.2 Standby Server Planned Shutdown

A planned shutdown of the standby server is performed by a system administrator during a maintenance window. During upgrade, the standby server shuts down automatically, and no manual steps are necessary.

- Performing a Standby Server Planned Shutdown during Upgrade (page B-14)
- Performing a Standby Server Planned Shutdown during Maintenance (page B-14)

#### B.14.4.2.1 Performing a Standby Server Planned Shutdown during Upgrade

To perform a standby server planned shutdown during upgrade, follow the procedure Upgrade a Pair of Key Vault Servers in a High Availability Deployment (page 12-39).

> **Note:**
>
> When you reboot the standby server during upgrade, the upgrade script initiates an automatic shutdown. There are no manual steps to be performed after the standby server is rebooted.

#### B.14.4.2.2 Performing a Standby Server Planned Shutdown during Maintenance

To perform a standby server planned shutdown during maintenance:

1. Log in to the standby server terminal using SSH as user `support`, then switch user (`su`) to `root`.
2. Switch user (`su`) to `oracle`.

3. Open SQL *Plus and run the following commands on the database:

   ```
   alter database recover managed standby database cancel
   shutdown immediate
   ```

4. Power off the standby server.

# C

# Security Technical Implementation Guides Compliance Standards

Oracle Key Vault follows the Security Technical Implementation Guides (STIG)-based compliance standards.

## C.1 About Security Technical Implementation Guides

A Security Technical Implementation Guide (STIG) is a methodology followed by the U.S. Department of Defense (DOD) to reduce the attack surface of computer systems and networks, thereby ensuring a lockdown of highly confidential information stored within the DOD network. STIGs provide secure configuration standards for the DOD's Information Assurance (IA) and IA-enabled devices and systems. STIGs are created by the Defense Information Systems Agency (DISA).

For over a decade, Oracle has worked closely with the DOD to develop, publish, and maintain a growing list of STIGs for a variety of core Oracle products and technologies including:

- Oracle Database
- Oracle Solaris
- Oracle Linux
- Oracle WebLogic

When STIGs are updated, Oracle analyzes the latest recommendations in order to identify new ways to improve the security of its products by:

- Implementing new and innovative security capabilities that are then added to future STIG updates
- Delivering functionality to automate the assessment and implementation of STIG recommendations
- Improving “out of the box" security configuration settings based upon STIG recommendations

**Related Topics**

- STIG Standards for Oracle
- STIG Home

# C.2 Enabling and Disabling STIG Rules on Oracle Key Vault

You can enable STIG rules on Oracle Key Vault by enabling Strict mode.

- Enabling STIG Rules on Oracle Key Vault (page C-2)
- Disabling STIG Rules on Oracle Key Vault (page C-2)

## C.2.1 Enabling STIG Rules on Oracle Key Vault

To enable strict mode:

1. Log in to the operating system of the Key Vault server as the root user.
2. Run the following command as `root`:

```
/usr/local/dbfw/bin/stig --enable
```

## C.2.2 Disabling STIG Rules on Oracle Key Vault

To disable strict mode:

1. Log in to the operating system of the Key Vault server as the root user.
2. Run the following command as `root`:

```
/usr/local/dbfw/bin/stig --disable
```

# C.3 Current Implementation of STIG Rules on Oracle Key Vault

Oracle has developed a security-hardened configuration of Oracle Key Vault that supports U.S. Department of Defense Security Technical Implementation Guide (STIG) recommendations.

Table C-1 (page C-2) lists the three vulnerability categories that STIG recommendations.

**Table C-1 Vulnerability Categories**

| Category | Description |
|---|---|
| CAT I | Any vulnerability, the exploitation of which will, directly and immediately result in loss of Confidentiality, Availability, or Integrity. |
| CAT II | Any vulnerability, the exploitation of which has a potential to result in loss of Confidentiality, Availability, or Integrity. |
| CAT III | Any vulnerability, the existence of which degrades measures to protect against loss of Confidentiality, Availability, or Integrity. |

# C.4 Current Implementation of Database STIG Rules

Table C-2 (page C-3) shows the current implementation of Database STIG rules on Oracle Key Vault.

**Table C-2    Current Implementation of Database STIG Rules**

| STIG ID | Title | Severity | Addressed by Script | Addressed by Documentation | Action required | Implemented | Notes |
|---|---|---|---|---|---|---|---|
| DG0004-ORACLE 11 | DBMS application object owner accounts | CAT II | No | No | None | No | Application object owner accounts `KEYVAULT`, `APEX_040200`, `MANAGEMENT`, and `AVSYS` are locked after the installation of Oracle Key Vault. |
| DG0008-ORACLE 11 | DBMS application object ownership | No | No | Yes | No | No | No |
| DG0014-ORACLE 11 | DBMS demonstration and sample databases | CAT II | No | No | None | No | All default demonstration and sample database objects have been removed. |
| DG0071-ORACLE 11 | DBMS password change variance | CAT II | No | No | No | No | Currently not supported |
| DG0073-ORACLE 11 | DBMS failed login account lock | CAT II | Yes | No | No | No | For profiles `FAILED_LOGIN_ATTEMPTS` is set to the required limit in the script. |
| DG0075-ORACLE 11 | DBMS links to external databases | CAT II | No | Yes | No | No | No |
| DG0077-ORACLE 11 | Production data protection on a shared system | CAT II | No | No | None | No | No |
| DG0116-ORACLE 11 | DBMS privileged role assignments | CAT II | Yes | Yes | No | No | No |
| DG0117-ORACLE 11 | DBMS administrative privilege assignment | CAT II | No | No | No | No | Currently not supported |
| DG0121-ORACLE 11 | DBMS application user privilege assignment | CAT II | No | No | No | No | Currently not supported |

**Table C-2 (Cont.) Current Implementation of Database STIG Rules**

| STIG ID | Title | Severity | Addressed by Script | Addressed by Documentation | Action required | Implemented | Notes |
|---|---|---|---|---|---|---|---|
| DG0123-ORACLE 11 | DBMS Administrative data access | CAT II | No | No | No | No | Currently not supported |
| DG0125-ORACLE 11 | DBMS account password expiration | CAT II | Yes | No | No | No | For profiles `PASSWORD_LIFE_TIME` is set to the required limit in the script. |
| DG0126-ORACLE 11 | DBMS account password reuse | CAT II | No | No | None | No | No. |
| DG0128-ORACLE 11 | DBMS default passwords | CAT I | Yes | No | No | No | Account CTXSYS is assigned a random password in the script. |
| DG0133-ORACLE 11 | DBMS Account lock time | CAT II | Yes | No | No | No | No |
| DG0141-ORACLE 11 | DBMS access control bypass | CAT II | Yes | No | No | No | Users can use a script to audit the following events:<br>`DROP ANY SYNONYM`<br>`DROP ANY INDEXTYPE` |
| DG0142-ORACLE 11 | DBMS Privileged action audit | CAT II | No | No | None | No | No |
| DG0192-ORACLE 11 | DBMS fully-qualified name for remote access | CAT II | Yes | No | No | No | Currently not supported |
| DO0231-ORACLE 11 | Oracle application object owner tablespaces | CAT II | No | No | No | No | Currently not supported |
| DO0250-ORACLE 11 | Oracle database link usage | CAT II | No | Yes | No | No | No |
| DO0270-ORACLE 11 | Oracle redo log file availability | CAT II | No | No | No | No | Currently not supported |
| DO0350-ORACLE 11 | Oracle system privilege assignment | CAT II | No | No | No | No | Currently not supported |

**Table C-2 (Cont.) Current Implementation of Database STIG Rules**

| STIG ID | Title | Severity | Addressed by Script | Addressed by Documentation | Action required | Implemented | Notes |
|---|---|---|---|---|---|---|---|
| DO3475-ORACLE 11 | Oracle PUBLIC access to restricted packages | CAT II | No | No | No | No | Currently not supported |
| DO3536-ORACLE 11 | Oracle IDLE_TIME profile parameter | CAT II | Yes | No | No | No | No |
| DO3540-ORACLE 11 | Oracle SQL92_SECURITY parameter | CAT II | No | No | None | No | Parameter SQL92_SECURITY is already set to TRUE. |
| DO3609-ORACLE 11 | System privileges granted WITH ADMIN OPTION | CAT II | No | No | No | No | Currently not supported |
| DO3610-ORACLE 11 | Oracle minimum object auditing | CAT II | No | No | No | No | Currently not supported |
| DO3689-ORACLE 11 | Oracle object permission assignment to PUBLIC | CAT II | No | No | No | No | Currently not supported |
| DO3696-ORACLE 11 | Oracle RESOURCE _LIMIT parameter | CAT II | No | No | No | No | Currently not supported |

# C.5 Additional Notes

Additional notes regarding STIG IDs are in Table C-2 (page C-3).

- DG0008-ORACLE11 STIG Rule (page C-5)

## C.5.1 DG0008-ORACLE11 STIG Rule

Object owner accounts in Oracle Key Vault Server:

- KEYVAULT
- APEX_040200
- AVSYS
- MANAGEMENT

## C.6 Current Implementation of Operating System STIG Rules

Table C-3 (page C-6) shows the current implementation of Operating System STIG Rules on Oracle Key Vault.

**Table C-3 Current Implementation of Operating System STIG Rules**

| STIG ID | Title | Severity | Key Vault Server - Default | Key Vault Server - STIG |
|---|---|---|---|---|
| SV-50237r1_rule | Automated file system mounting tools must not be enabled unless needed. | CAT III | No action required | Addressed by script |
| SV-50238r2_rule | Auditing must be enabled at boot by setting a kernel parameter. | CAT III | No action required | Addressed by script |
| SV-50243r1_rule | The `/etc/gshadow` file must be owned by root. | CAT II | No action required | Addressed by script |
| SV-50248r1_rule | The `/etc/gshadow` file must be group-owned by root. | CAT II | No action required | Addressed by script |
| SV-50249r1_rule | The `/etc/gshadow` file must have mode 0000. | CAT II | No action required | Addressed by script |
| SV-50250r1_rule | The `/etc/passwd` file must be owned by root. | CAT II | No action required | Addressed by script |
| SV-50251r1_rule | The `/etc/passwd` file must be group-owned by root. | CAT II | No action required | Addressed by script |
| SV-50255r1_rule | The system must use a separate file system for `/tmp`. | CAT III | No action required | Addressed by script |
| SV-50256r1_rule | The system must use a separate file system for `/var`. | CAT III | Addressed by script | Implemented differently |
| SV-50257r1_rule | The `/etc/passwd` file must have mode 0644 or less permissive. | CAT II | No action required | Addressed by script |
| SV-50258r1_rule | The `/etc/group` file must be owned by root. | CAT II | No action required | Addressed by script |
| SV-50259r1_rule | The `/etc/group` file must be group-owned by root. | CAT II | No action required | Addressed by script |
| SV-50261r1_rule | The `/etc/group` file must have mode 0644 or less permissive. | CAT II | No action required | Addressed by script |
| SV-50263r1_rule | The system must use a separate file system for `/var/log`. | CAT III | No action required | Addressed by script |
| SV-50266r1_rule | Library files must be owned by `root`. | CAT II | No action required | Addressed by script |
| SV-50267r1_rule | The system must use a separate file system for the system audit data path. | CAT III | Addressed by script | Not implemented |
| SV-50269r2_rule | All system command files must have mode 0755 or less permissive. | CAT II | No action required | Addressed by script |

**Table C-3 (Cont.) Current Implementation of Operating System STIG Rules**

| STIG ID | Title | Severity | Key Vault Server - Default | Key Vault Server - STIG |
|---|---|---|---|---|
| SV-50270r2_rule | The audit system must alert designated staff members when the audit storage volume approaches capacity. | CAT II | Addressed by script | Not implemented |
| SV-50272r1_rule | All system command files must be owned by root. | CAT II | No action required | Addressed by script |
| SV-50273r1_rule | The system must use a separate file system for user home directories. | CAT III | No action required | Addressed by script |
| SV-50275r1_rule | The system must require passwords to contain a minimum of 14 characters. | CAT II | Addressed by script | Addressed by script |
| SV-50277r1_rule | Users must not be able to change passwords more than once every 24 hours. | CAT II | No action required | Addressed by script |
| SV-50278r2_rule | The Red Hat Network Service (`rhnsd`) service must not be running, unless using RHN or an RHN Satellite. | CAT III | No action required | Addressed by script |
| SV-50279r1_rule | User passwords must be changed at least every 60 days. | CAT II | Addressed by script | Addressed by script |
| SV-50280r1_rule | Users must be warned 7 days in advance of password expiration. | CAT III | No action required | Addressed by script |
| SV-50282r1_rule | The system must require passwords to contain at least one numeric character. | CAT III | No action required | Addressed by script |
| SV-50283r1_rule | The system package management tool must cryptographically verify the authenticity of system software packages during installation. | CAT II | No action required | Addressed by script |
| SV-50288r1_rule | The system package management tool must cryptographically verify the authenticity of all software packages during installation. | CAT III | No action required | Addressed by script |
| SV-50290r1_rule | A file integrity tool must be installed. | CAT II | No action required | Addressed by script |
| SV-50291r2_rule | The operating system must enforce requirements for the connection of mobile devices to operating systems. | CAT II | No action required | Addressed by script |
| SV-50292r1_rule | There must be no `.rhosts` or hosts.equiv files on the system. | CAT I | No action required | Addressed by script |
| SV-50293r1_rule | The system must prevent the root account from logging in from virtual consoles. | CAT II | No action required | Addressed by script |

Table C-3 (Cont.) Current Implementation of Operating System STIG Rules

| STIG ID | Title | Severity | Key Vault Server - Default | Key Vault Server - STIG |
|---|---|---|---|---|
| SV-50295r1_rule | The system must prevent the root account from logging in from serial consoles. | CAT III | No action required | Addressed by script |
| SV-50296r1_rule | Audit log files must be owned by `root`. | CAT II | No action required | Addressed by script |
| SV-50298r2_rule | The system must not have accounts configured with blank or null passwords. | CAT I | No action required | Addressed by script |
| SV-50299r1_rule | Audit log files must have mode 0640 or less permissive. | CAT II | No action required | Addressed by script |
| SV-50300r1_rule | The `/etc/passwd` file must not contain password hashes. | CAT II | No action required | Addressed by script |
| SV-50301r2_rule | The root account must be the only account having a UID of 0. | CAT II | No action required | Addressed by script |
| SV-50302r3_rule | The system must disable accounts after excessive login failures within a 15-minute interval. | CAT II | No action required | Addressed by script |
| SV-50303r1_rule | The `/etc/shadow` file must be owned by root. | CAT II | No action required | Addressed by script |
| SV-50304r1_rule | The `/etc/shadow` file must be group-owned by root. | CAT II | No action required | Addressed by script |
| SV-50305r1_rule | The `/etc/shadow` file must have mode 0000. | CAT II | No action required | Addressed by script |
| SV-50312r1_rule | IP forwarding for IPv4 must not be enabled, unless the system is a router. | CAT II | No action required | Addressed by script |
| SV-50313r2_rule | The operating system must prevent public IPv4 access into an organizations internal networks, except as appropriately mediated by managed interfaces employing boundary protection devices. | CAT II | No action required | Addressed by script |
| SV-50314r1_rule | The systems local IPv4 firewall must implement a deny-all, allow-by-exception policy for inbound packets. | CAT II | No action required | Addressed by script |
| SV-50315r2_rule | The Datagram Congestion Control Protocol (DCCP) must be disabled unless required. | CAT II | No action required | Addressed by script |
| SV-50316r2_rule | The Stream Control Transmission Protocol (SCTP) must be disabled unless required. | CAT II | No action required | Addressed by script |

**Table C-3    (Cont.) Current Implementation of Operating System STIG Rules**

| STIG ID | Title | Severity | Key Vault Server - Default | Key Vault Server - STIG |
|---|---|---|---|---|
| SV-50317r2_rule | The Reliable Datagram Sockets (RDS) protocol must be disabled unless required. | CAT III | No action required | Addressed by script |
| SV-50318r2_rule | The Transparent Inter-Process Communication (TIPC) protocol must be disabled unless required. | CAT II | No action required | Addressed by script |
| SV-50319r2_rule | All rsyslog-generated log files must be owned by root. | CAT II | No action required | Addressed by script |
| SV-50321r1_rule | The operating system must back up audit records on an organization defined frequency onto a different system or media than the system being audited. | CAT II | Addressed by script | Not implemented |
| SV-50322r1_rule | The operating system must support the requirement to centrally manage the content of audit records generated by organization defined information system components. | CAT II | Addressed by script | Not implemented |
| SV-50323r2_rule | The audit system must be configured to audit all attempts to alter system time through `settimeofday`. | CAT III | No action required | Addressed by script |
| SV-50324r2_rule | The system must not accept IPv4 source-routed packets on any interface. | CAT II | No action required | Addressed by script |
| SV-50325r1_rule | The system must not accept ICMPv4 redirect packets on any interface. | CAT II | No action required | Addressed by script |
| SV-50326r3_rule | The audit system must be configured to audit all attempts to alter system time through `stime`. | CAT III | No action required | Addressed by script |
| SV-50327r1_rule | The system must not accept ICMPv4 secure redirect packets on any interface. | CAT II | No action required | Addressed by script |
| SV-50328r2_rule | The audit system must be configured to audit all attempts to alter system time through `clock_settime`. | CAT III | No action required | Addressed by script |
| SV-50329r1_rule | The system must log Martian packets. | CAT III | Addressed by script | Not implemented |
| SV-50330r1_rule | The system must not accept IPv4 source-routed packets by default. | CAT II | No action required | Addressed by script |
| SV-50331r1_rule | The audit system must be configured to audit all attempts to alter system time through `/etc/localtime`. | CAT III | No action required | Addressed by script |

Table C-3 (Cont.) Current Implementation of Operating System STIG Rules

| STIG ID | Title | Severity | Key Vault Server - Default | Key Vault Server - STIG |
|---|---|---|---|---|
| SV-50332r1_rule | The operating system must automatically audit account creation. | CAT III | No action required | Addressed by script |
| SV-50333r1_rule | The system must not accept ICMPv4 secure redirect packets by default. | CAT II | No action required | Addressed by script |
| SV-50334r2_rule | The system must ignore ICMPv4 redirect messages by default. | CAT III | No action required | Addressed by script |
| SV-50335r1_rule | The operating system must automatically audit account modification. | CAT III | No action required | Addressed by script |
| SV-50336r2_rule | The system must not respond to ICMPv4 sent to a broadcast address. | CAT III | No action required | Addressed by script |
| SV-50337r1_rule | The operating system must automatically audit account disabling actions. | CAT III | No action required | Addressed by script |
| SV-50338r2_rule | The system must ignore ICMPv4 bogus error responses. | CAT III | No action required | Addressed by script |
| SV-50339r1_rule | The operating system must automatically audit account termination. | CAT III | No action required | Addressed by script |
| SV-50340r1_rule | The system must be configured to use TCP `syncookies`. | CAT II | No action required | Addressed by script |
| SV-50342r1_rule | The audit system must be configured to audit modifications to the systems Mandatory Access Control (MAC) configuration (`SELinux`). | CAT III | No action required | Addressed by script |
| SV-50343r1_rule | The system must use a reverse-path filter for IPv4 network traffic when possible on all interfaces. | CAT II | No action required | Addressed by script |
| SV-50344r2_rule | The audit system must be configured to audit all discretionary access control permission modifications using `chmod`. | CAT III | No action required | Addressed by script |
| SV-50345r1_rule | The system must use a reverse-path filter for IPv4 network traffic when possible by default. | CAT II | No action required | Addressed by script |
| SV-50346r2_rule | The audit system must be configured to audit all discretionary access control permission modifications using `chown`. | CAT III | No action required | Addressed by script |

Table C-3 (Cont.) Current Implementation of Operating System STIG Rules

| STIG ID | Title | Severity | Key Vault Server - Default | Key Vault Server - STIG |
|---|---|---|---|---|
| SV-50347r2_rule | The IPv6 protocol handler must not be bound to the network stack unless needed. | CAT II | No action required | Addressed by script |
| SV-50348r2_rule | The audit system must be configured to audit all discretionary access control permission modifications using `fchmod`. | CAT III | No action required | Addressed by script |
| SV-50349r2_rule | The system must ignore ICMPv6 redirects by default. | CAT II | No action required | Addressed by script |
| SV-50351r2_rule | The audit system must be configured to audit all discretionary access control permission modifications using `fchmodat`. | CAT III | No action required | Addressed by script |
| SV-50353r2_rule | The audit system must be configured to audit all discretionary access control permission modifications using `fchown`. | CAT III | No action required | Addressed by script |
| SV-50355r2_rule | The audit system must be configured to audit all discretionary access control permission modifications using `fchownat`. | CAT III | No action required | Addressed by script |
| SV-50356r2_rule | The system must employ a local IPv4 firewall. | CAT II | No action required | Addressed by script |
| SV-50357r2_rule | The audit system must be configured to audit all discretionary access control permission modifications using `fremovexattr`. | CAT III | No action required | Addressed by script |
| SV-50358r2_rule | The audit system must be configured to audit all discretionary access control permission modifications using `fsetxattr`. | CAT III | No action required | Addressed by script |
| SV-50359r2_rule | The audit system must be configured to audit all discretionary access control permission modifications using `lchown`. | CAT III | No action required | Addressed by script |
| SV-50360r2_rule | The audit system must be configured to audit all discretionary access control permission modifications using `lremovexattr`. | CAT III | No action required | Addressed by script |

**Table C-3 (Cont.) Current Implementation of Operating System STIG Rules**

| STIG ID | Title | Severity | Key Vault Server - Default | Key Vault Server - STIG |
|---|---|---|---|---|
| SV-50362r2_rule | The audit system must be configured to audit all discretionary access control permission modifications using `lsetxattr`. | CAT III | No action required | Addressed by script |
| SV-50364r2_rule | The audit system must be configured to audit all discretionary access control permission modifications using `removexattr`. | CAT III | No action required | Addressed by script |
| SV-50366r2_rule | The audit system must be configured to audit all discretionary access control permission modifications using `setxattr`. | CAT III | No action required | Addressed by script |
| SV-50369r2_rule | The audit system must be configured to audit successful file system mounts. | CAT III | No action required | Addressed by script |
| SV-50370r1_rule | The system must require passwords to contain at least one uppercase alphabetic character. | CAT III | No action required | Addressed by script |
| SV-50371r1_rule | The system must require passwords to contain at least one special character. | CAT III | No action required | Addressed by script |
| SV-50372r1_rule | The system must require passwords to contain at least one lowercase alphabetic character. | CAT III | No action required | Addressed by script |
| SV-50373r1_rule | The system must require at least four characters be changed between the old and new passwords during a password change. | CAT III | No action required | Addressed by script |
| SV-50374r3_rule | The system must disable accounts after three consecutive unsuccessful logon attempts. | CAT II | No action required | Addressed by script |
| SV-50375r1_rule | The system must use a FIPS 140-2 approved cryptographic hashing algorithm for generating account password hashes (system-auth). | CAT II | No action required | Addressed by script |
| SV-50376r4_rule | The audit system must be configured to audit user deletions of files and programs. | CAT III | No action required | Addressed by script |
| SV-50377r1_rule | The system must use a FIPS 140-2 approved cryptographic hashing algorithm for generating account password hashes (`login.defs`). | CAT II | No action required | Addressed by script |

**Table C-3 (Cont.) Current Implementation of Operating System STIG Rules**

| STIG ID | Title | Severity | Key Vault Server - Default | Key Vault Server - STIG |
|---|---|---|---|---|
| SV-50378r1_rule | The system must use a FIPS 140-2 approved cryptographic hashing algorithm for generating account password hashes (`libuser.conf`). | CAT II | No action required | Addressed by script |
| SV-50379r1_rule | The audit system must be configured to audit changes to the `/etc/sudoers` file. | CAT III | No action required | Addressed by script |
| SV-50380r1_rule | The system boot loader configuration file(s) must be owned by `root`. | CAT II | No action required | Addressed by script |
| SV-50381r1_rule | The audit system must be configured to audit the loading and unloading of dynamic kernel modules. | CAT II | No action required | Addressed by script |
| SV-50382r1_rule | The system boot loader configuration file(s) must be group-owned by root. | CAT II | No action required | Addressed by script |
| SV-50383r2_rule | The `xinetd` service must be disabled if no network services utilizing it are enabled. | CAT II | No action required | Addressed by script |
| SV-50384r2_rule | The system boot loader configuration file(s) must have mode 0600 or less permissive. | CAT II | No action required | Addressed by script |
| SV-50385r1_rule | The `xinetd` service must be uninstalled if no network services utilizing it are enabled. | CAT III | No action required | Addressed by script |
| SV-50386r1_rule | The system boot loader must require authentication. | CAT II | Addressed by script | Not implemented |
| SV-50387r1_rule | The system must require authentication upon booting into single-user and maintenance modes. | CAT II | Addressed by script | Not implemented |
| SV-50388r1_rule | The `telnet-server` package must not be installed. | CAT I | No action required | Addressed by script |
| SV-50389r1_rule | The system must not permit interactive boot. | CAT II | No action required | Addressed by script |
| SV-50390r2_rule | The `telnet` daemon must not be running. | CAT I | No action required | Addressed by script |
| SV-50391r1_rule | The system must allow locking of the console screen in text mode. | CAT III | Addressed by script | Not implemented |
| SV-50392r1_rule | The `rsh-server` package must not be installed. | CAT I | No action required | Addressed by script |

### Table C-3 (Cont.) Current Implementation of Operating System STIG Rules

| STIG ID | Title | Severity | Key Vault Server - Default | Key Vault Server - STIG |
|---|---|---|---|---|
| SV-50393r3_rule | The system must require administrator action to unlock an account locked by excessive failed login attempts. | CAT II | Addressed by script | Addressed by script |
| SV-50395r2_rule | The `rshd` service must not be running. | CAT I | No action required | Addressed by script |
| SV-50399r2_rule | The `rexecd` service must not be running. | CAT I | No action required | Addressed by script |
| SV-50401r1_rule | The system must not send ICMPv4 redirects by default. | CAT II | No action required | Addressed by script |
| SV-50402r1_rule | The system must not send ICMPv4 redirects from any interface. | CAT II | No action required | Addressed by script |
| SV-50403r2_rule | The `rlogind` service must not be running. | CAT I | No action required | Addressed by script |
| SV-50404r1_rule | The `ypserv` package must not be installed. | CAT II | No action required | Addressed by script |
| SV-50405r2_rule | The `ypbind` service must not be running. | CAT II | No action required | Addressed by script |
| SV-50406r2_rule | The `cron` service must be running. | CAT II | No action required | Addressed by script |
| SV-50407r1_rule | The `tftp-server` package must not be installed. | CAT II | No action required | Addressed by script |
| SV-50408r1_rule | The SSH daemon must be configured to use only the SSHv2 protocol. | CAT I | No action required | Addressed by script |
| SV-50409r1_rule | The SSH daemon must set a timeout interval on idle sessions. | CAT III | No action required | Addressed by script |
| SV-50411r1_rule | The SSH daemon must set a timeout count on idle sessions. | CAT III | No action required | Addressed by script |
| SV-50412r1_rule | The SSH daemon must ignore `rhosts` files. | CAT II | No action required | Addressed by script |
| SV-50413r1_rule | The SSH daemon must not allow host-based authentication. | CAT II | No action required | Addressed by script |
| SV-50414r1_rule | The system must not permit root logins using remote access programs such as ssh. | CAT II | No action required | Addressed by script |
| SV-50415r1_rule | The SSH daemon must not allow authentication using an empty password. | CAT I | No action required | Addressed by script |
| SV-50416r1_rule | The SSH daemon must be configured with the Department of Defense (DoD) login banner. | CAT II | Addressed by script | Not implemented |
| SV-50417r1_rule | The SSH daemon must not permit user environment settings. | CAT III | No action required | Addressed by script |

**Table C-3 (Cont.) Current Implementation of Operating System STIG Rules**

| STIG ID | Title | Severity | Key Vault Server - Default | Key Vault Server - STIG |
|---|---|---|---|---|
| SV-50419r2_rule | The `avahi` service must be disabled. | CAT III | No action required | Addressed by script |
| SV-50421r1_rule | The system clock must be synchronized continuously, or at least daily. | CAT II | Addressed by script | Addressed by documentation |
| SV-50422r1_rule | The system clock must be synchronized to an authoritative DoD time source. | CAT II | No action required | Addressed by script |
| SV-50423r2_rule | Mail relaying must be restricted. | CAT II | Addressed by script | Not applicable |
| SV-50428r1_rule | The `openldap-servers` package must not be installed unless required. | CAT III | No action required | Addressed by script |
| SV-50430r3_rule | The graphical desktop environment must set the idle timeout to no more than 15 minutes. | CAT II | No action required | Addressed by script |
| SV-50431r3_rule | The graphical desktop environment must automatically lock after 15 minutes of inactivity and the system must require user reauthentication to unlock the environment. | CAT II | No action required | Addressed by script |
| SV-50434r1_rule | The system must set a maximum audit log file size. | CAT II | No action required | Addressed by script |
| SV-50435r1_rule | The system must rotate audit log files that reach the maximum file size. | CAT II | No action required | Addressed by script |
| SV-50436r2_rule | The audit system must be configured to audit all attempts to alter system time through `adjtimex`. | CAT III | No action required | Addressed by script |
| SV-50437r1_rule | The system must retain enough rotated audit logs to cover the required log retention period. | CAT II | No action required | Addressed by script |
| SV-50439r3_rule | The graphical desktop environment must have automatic lock enabled. | CAT II | No action required | Addressed by script |
| SV-50440r3_rule | The system must display a publicly-viewable pattern during a graphical desktop environment session lock. | CAT III | No action required | Addressed by script |
| SV-50441r2_rule | The Automatic Bug Reporting Tool (`abrtd`) service must not be running. | CAT III | No action required | Addressed by script |
| SV-50442r2_rule | The `atd` service must be disabled. | CAT III | No action required | Addressed by script |

**Table C-3 (Cont.) Current Implementation of Operating System STIG Rules**

| STIG ID | Title | Severity | Key Vault Server - Default | Key Vault Server - STIG |
|---|---|---|---|---|
| SV-50443r1_rule | The system default umask for daemons must be 027 or 022. | CAT III | No action required | Addressed by script |
| SV-50445r2_rule | The ntpdate service must not be running. | CAT III | No action required | Addressed by script |
| SV-50446r1_rule | The system default umask in /etc/login.defs must be 077. | CAT III | No action required | Addressed by script |
| SV-50447r2_rule | The oddjobd service must not be running. | CAT III | No action required | Addressed by script |
| SV-50448r1_rule | The system default umask in /etc/profile must be 077. | CAT III | Addressed by script | Not implemented |
| SV-50449r2_rule | The qpidd service must not be running. | CAT III | No action required | Addressed by script |
| SV-50450r1_rule | The system default umask for the csh shell must be 077. | CAT III | Addressed by script | Not implemented |
| SV-50451r2_rule | The rdisc service must not be running. | CAT III | No action required | Addressed by script |
| SV-50452r1_rule | The system default umask for the bash shell must be 077. | CAT III | Addressed by script | Not implemented |
| SV-50457r1_rule | The system must use SMB client signing for connecting to samba servers using smbclient. | CAT III | Addressed by script | Not implemented |
| SV-50470r1_rule | The postfix service must be enabled for mail delivery. | CAT III | Addressed by script | Not implemented |
| SV-50472r1_rule | The sendmail package must be removed. | CAT II | No action required | Addressed by script |
| SV-50473r2_rule | The netconsole service must be disabled unless required. | CAT III | No action required | Addressed by script |
| SV-50475r1_rule | X Windows must not be enabled unless required. | CAT II | No action required | Addressed by script |
| SV-50476r2_rule | Process core dumps must be disabled unless needed. | CAT III | Addressed by script | Not implemented |
| SV-50477r1_rule | The xorg-x11-server-common (X Windows) package must not be installed, unless required. | CAT III | No action required | Addressed by script |
| SV-50480r2_rule | The DHCP client must be disabled if not needed. | CAT II | Addressed by script | Not implemented |
| SV-50481r1_rule | The audit system must identify staff members to receive notifications of audit log storage volume capacity issues. | CAT II | No action required | Addressed by script |

Table C-3 (Cont.) Current Implementation of Operating System STIG Rules

| STIG ID | Title | Severity | Key Vault Server - Default | Key Vault Server - STIG |
|---|---|---|---|---|
| SV-50485r2_rule | The system must limit users to 10 simultaneous system logins, or a site-defined number, in accordance with operational requirements. | CAT III | Addressed by script | Not implemented |
| SV-50488r2_rule | The system must provide VPN connectivity for communications over untrusted networks. | CAT III | Addressed by script | Not implemented |
| SV-50489r2_rule | A login banner must be displayed immediately prior to, or as part of, graphical desktop environment login prompts. | CAT II | No action required | Addressed by script |
| SV-50492r2_rule | The Bluetooth service must be disabled. | CAT II | No action required | Addressed by script |
| SV-50493r1_rule | Accounts must be locked upon 35 days of inactivity. | CAT III | Addressed by script | Not implemented |
| SV-50495r1_rule | The operating system must manage information system identifiers for users and devices by disabling the user identifier after an organization defined time period of inactivity. | CAT III | Addressed by script | Not implemented |
| SV-50498r2_rule | The sticky bit must be set on all public directories. | CAT III | Addressed by script | No action required |
| SV-50500r2_rule | All public directories must be owned by a system account. | CAT III | No action required | Addressed by script |
| SV-50502r1_rule | The TFTP daemon must operate in secure mode which provides access only to a single directory on the host file system. | CAT I | No action required | Addressed by script |
| SV-65547r1_rule | The system must use a Linux Security Module at boot time. | CAT II | No action required | Addressed by script |
| SV-65573r1_rule | The system must use a Linux Security Module configured to enforce limits on system services. | CAT II | Addressed by script | Not implemented |
| SV-65579r1_rule | The system must use a Linux Security Module configured to limit the privileges of system services. | CAT III | No action required | Addressed by script |
| SV-66089r1_rule | The operating system, upon successful logoNoccess, must display to the user the number of unsuccessful logoNoccess attempts since the last successful logoNoccess. | CAT II | No action required | Addressed by script |

**Table C-3 (Cont.) Current Implementation of Operating System STIG Rules**

| STIG ID | Title | Severity | Key Vault Server - Default | Key Vault Server - STIG |
|---|---|---|---|---|
| SV-68627r1_rule | The audit system must switch the system to single-user mode when available audit storage volume becomes dangerously low. | CAT II | Addressed by script | Not implemented |

# D

# Prerequisites for Endpoint Installation on AIX 5.3

OKV 12.2.0.7.0 and higher releases also support AIX 5.3 as an endpoint platform. Before enrolling and using an endpoint on AIX 5.3, the steps below must be performed on the OKV server:

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

   ```
   su - root
   ```

2. Add the following to `/usr/local/okv/bin/kmippparam.ora`:

   ```
   echo "SERVER_SSL_VERSION = TLS1,TLS1.2" >> /usr/local/okv/bin/
   kmipparam.ora
   ```

3. Restart the kmip service.

   ```
   service kmip restart
   ```

You can now enroll and use an endpoint on AIX 5.3.

# Glossary

**Audit Manager**

An administrator role that enables a user to manage audit lifecycle and policies and to separate the role of auditing from managing the appliance.

**auto-login wallet**

An Oracle wallet file that can be accessed without a password. An auto-login wallet is stored in a `cwallet.sso` file.

**credential file**

A file containing sensitive information like user ids, passwords, and keys. The file is stored as an opaque object, which means that its individual contents are not interpreted by Key Vault. The entire file is uploaded and downloaded as an object.

See also security objects.

**default wallet**

A special virtual wallet that is associated with an endpoint, where all the endpoint's security objects can be automatically uploaded.

**endpoint**

Computer systems like database servers, application servers, and other information systems, where keys and credentials are used to access encrypted data and other systems.

**endpoint administrator**

Owner of an endpoint. They are typically system, security, or database administrators, but they can be any personnel charged with deploying, managing and maintaining security within an enterprise. They are responsible for enrolling endpoints and controlling endpoint access to security objects.

### endpoint group

A group of endpoints created to share a set of security objects.

### JAVA_HOME

JAVA_HOME corresponds the location of Java files (JDK/JRE) in the system. This allows Java applications to look up the JAVA_HOME variable in order to run.

### Java keystore file

A file that can hold multiple security objects such as keys and certificates.

### Key Administrator

An administrator role that enables a user to manage the key lifecycle and control access to all security objects within Key Vault.

### keystore

A generalized term for a container that stores encryption keys including but not limited to TDE encryption keys.

### Management Information Base (MIB)

See MIB.

### master encryption key

See TDE master encryption key.

### MIB

In an SNMP configuration, a text file that describes the variables that contain the information that SNMP can access. The variables described in a MIB, which are also called MIB objects, are the items that can be monitored using SNMP. There is one MIB for each element being monitored.

### OKV_HOME

Corresponds to the environment in which Oracle Key Vault endpoint software will reside. It contains subdirectories for endpoint software like the configuration files, log files, libraries, binaries, and other files needed by the endpoint software utility.

### opaque object

A security object that cannot be interpreted by Oracle Key Vault.

### Oracle wallet file

An Oracle wallet file is a container that can hold multiple security objects such as keys and certificates. It uses the PKCS#12 cryptographic standard.

Oracle wallets can be managed by Key Vault just like other security objects. They can be can be encrypted and protected with a password or not. An Oracle wallet that can be accessed without a password is called an auto-login wallet.

See also auto-login wallet, password-protected wallet.

### ORACLE_BASE

`ORACLE_BASE` is the root of the Oracle Database directory tree. The Oracle Base directory is the top level directory that you can use to install the various Oracle software products. You can use the same Oracle base directory for multiple installations. For example, `/u01/app/oracle` is an Oracle base directory created by the oracle user.

### ORACLE_HOME

The directory path to install Oracle components (for example, `/u01/app/oracle/product/12.1.0/db_n`). You are prompted to enter an Oracle home in the Path field of the Specify File Locations window.

Corresponds to the environment in which Oracle Database products run. If you install an OFA-compliant database, using Oracle Universal Installer defaults, Oracle home (known as `$ORACLE_HOME` in this guide) is located beneath `$ORACLE_BASE`. The default Oracle home is `db_n` where `n` is the Oracle home number. It contains subdirectories for Oracle Database software executables and network files.

### ORACLE_SID

The Oracle System ID (SID) is used to uniquely identify a particular database on a system. For this reason, one cannot have more than one database with the same SID on a computer system.

When using RAC, all instances belonging to the same database must have a unique SID.

### oraenv

`oraenv` and `coraenv` are Unix/ Linux command line utilities that set the required environment variables (`ORACLE_SID`, `ORACLE_HOME` and `PATH`) to allow a user to connect to a given database instance. If these environment variables are not set, commands such as `SQL*Plus`, `imp`, `exp` will not work (or not be found).

Use `coraenv` when using the C Shell and `oraenv` when using a Bourne, Korn or Bash shell.

### password-protected wallet

An encrypted Oracle wallet that has a user-defined password stored in an `ewallet.p12` file.

### PKCS#11 library

A library that allows an Oracle TDE database to connect to Oracle Key Vault to manage the master keys.

### security objects

Security objects can be public and private encryption keys, Oracle wallets, Java keystores, Java Cryptography Extension keystores, certificates, and credential files.

### software appliance

A self-contained preconfigured product that can be installed on supported hardware dedicated for a specific purpose.

### sqlnet.ora file

The sqlnet.ora file resides in $ORACLE_HOME/network/admin. It is a configuration file for the client or server that specifies the:

- Client domain to append to unqualified service names or net service names
- Order of naming methods for the client to use when resolving a name
- Logging and tracing features to use
- Route of connections
- External naming parameters
- Oracle Advanced Security parameters

### System Administrator

An administrator role that enables a user to create users, endpoints and their respective groups, configure system settings and alerts, and generally administer Oracle Key Vault.

### TDE master encryption key

A key that encrypts the data encryption keys for tables and tablespaces.

**template**

A collection of attributes for security objects. When a security object is created using a template, the attributes in the template are automatically assigned to the new object.

**users**

Users can be administrators, auditors, or ordinary users with no administrative roles.

**virtual wallet**

A container for security objects like public and private encryption keys, TDE master encryption keys, passwords, credentials, and certificates in Oracle Key Vault. The main purpose of a virtual wallet is to enable sharing of keys among endpoints.

# Index